Jordan Yiu
Prof. Michael Lesk
Problem Solving with Data
14 December 2021

Final Project - Spotify Song Analysis

## I. Introduction

Spotify is the largest music streaming service in the world, with over 70 million songs in its catalogue. Since its main goal as a streaming service is to draw people to the platform and retain them, the application would need robust features in combination with its catalogue to keep users from migrating to other platforms. Features such as auto-generated playlists and recommendations are possible due to how they process the vast amounts of data they have on their catalogue, but are camouflaged to the user. This project will analyze my own Spotify data to try and find correlations between the music and how that might explain my music tastes as well as approximate how Spotify might use this data for their own implementations in their service.

## II. Getting the Data & Definitions

The data is accessible through the Spotify Web API [7] and was retrieved through the Spotify library for Python. The data consists of 2,494 songs from my own Spotify account data and consists of audio analysis from The Echo Nest [1], an audio intelligence and data processing platform acquired by Spotify in 2014. In terms of audio analysis, Spotify catalogues different features across all its songs, each of which is intended to measure properties of the song in a binary classification manner. For example, "acousticness" measures the probability of the song using acoustic instruments and not having any electronic or synthesized tracks, while "liveness" measures the probability of being a live recording. The data is measured on a scale from 0.000 to 1.000, where the lower the number, the lower the possibility of the song matching that described feature. Here is the JSON that is returned by the API for the track "Piano Black" by the SEATBELTS. The song is from the anime *Cowboy Bebop*, which uses a jazz soundtrack. This would explain the high values in danceability, energy, and instrumentalness. Though Spotify does allow users accessing the API to retrieve genre data for music, this information was not gathered by the code as written. In the future, this can be added as a feature to the code for more in-depth analysis .

```
test = sp.audio_features("3u2TWIOpWwFEndbTyDLWu2")
test
[21]  ✓ 0.1s
```


```
[{'danceability': 0.701,
  'energy': 0.725,
  'key': 10,
  'loudness': -10.296,
  'mode': 0,
  'speechiness': 0.0461,
  'acousticness': 0.166,
  'instrumentalness': 0.898,
  'liveness': 0.109,
  'valence': 0.65,
  'tempo': 120.866,
  'type': 'audio_features',
  'id': '3u2TWIOpWwFEndbTyDLWu2',
  'uri': 'spotify:track:3u2TWIOpWwFEndbTyDLWu2',
```


In order to analyze the data, the terms at hand must be defined. The following table contains Spotify's in-house definitions for each of the features:

*Features definitions from Spotify Web API [5]*

| Feature | Definition |
|---|---|
| Acousticness | A measurement of whether a song has acoustic instruments or not |
| 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 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. |
| Key | The key the track is in. 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. |
| 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 typically 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. |
| 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. |
| Time Signature | An estimated time signature. The time signature (meter) is a notational convention to specify how many beats are in each bar (or measure). The time signature ranges from 3 to 7 indicating time signatures of "3/4", to "7/4". >= 3<= 7 |
| 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).>= 0<= 1 |

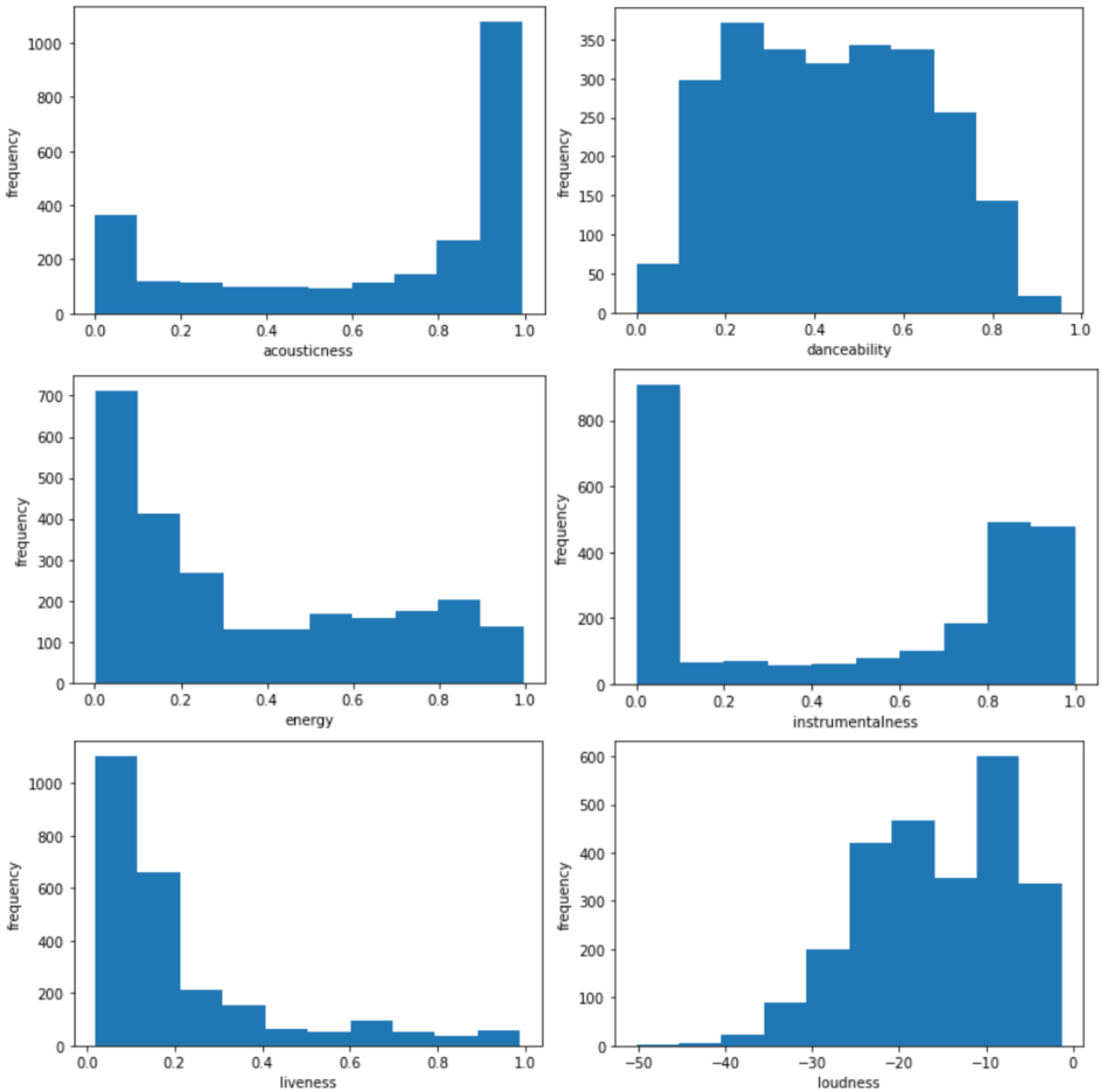
And here is an example of what the collected data looks like:

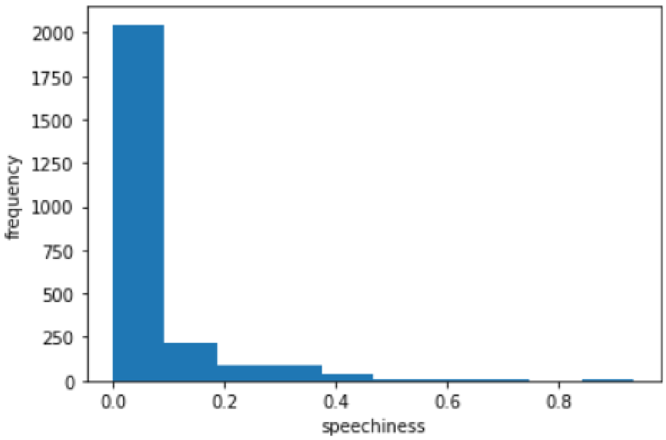
| | Unnamed: 0 | artist | name | acousticness | danceability | energy | instrumentalness | liveness | loudness | speechiness | tempo | valence | time_signature | key |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | spotify:track:1mmTLkwVGn2OoqMEWGoEsj | 'Vulfpeck' | 'The Sweet Science (Live at Madison Square Gar... | 0.8680 | 0.302 | 0.4990 | 0.857 | 0.4690 | -11.764 | 0.0494 | 122.508 | 0.1920 | 3.0 | 2.0 |
| 1 | spotify:track:4Z9fL58gHxgBLPQRaevsdJ | 'Ginger Root' | 'For Once in My Life' | 0.2190 | 0.434 | 0.8170 | 0.306 | 0.2870 | -5.589 | 0.0695 | 105.877 | 0.6180 | 4.0 | 4.0 |
| 2 | spotify:track:6zukGpcjPkCJDOYUhlGWiE | 'Celestial Aeon Project' | 'Epilogue (From "The Legend of Zelda Breath of... | 0.9870 | 0.692 | 0.0881 | 0.948 | 0.0930 | -19.591 | 0.1130 | 136.071 | 0.2480 | 4.0 | 7.0 |
| 3 | spotify:track:3fSenClo4FAV1Gkmc4wVyz | 'Taeko Onuki' | 'Tsuki no Kizahashi' | 0.9910 | 0.445 | 0.0570 | 0.120 | 0.0817 | -19.163 | 0.0400 | 84.516 | 0.1640 | 4.0 | 0.0 |
| 4 | spotify:track:49DMmKK4yBRmXyCDpJI2fr | 'Gordon Crosse' | 'Concerto for Chamber Orchestra, Op. 8: II. Le... | 0.9480 | 0.125 | 0.1540 | 0.893 | 0.1010 | -19.090 | 0.0444 | 84.289 | 0.0350 | 3.0 | 8.0 |

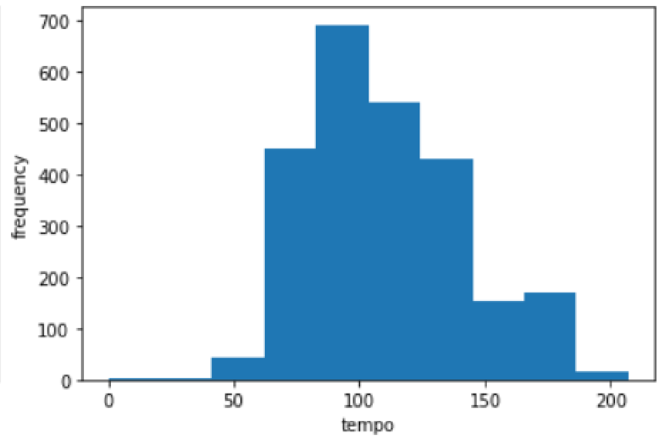
## III. Analysis

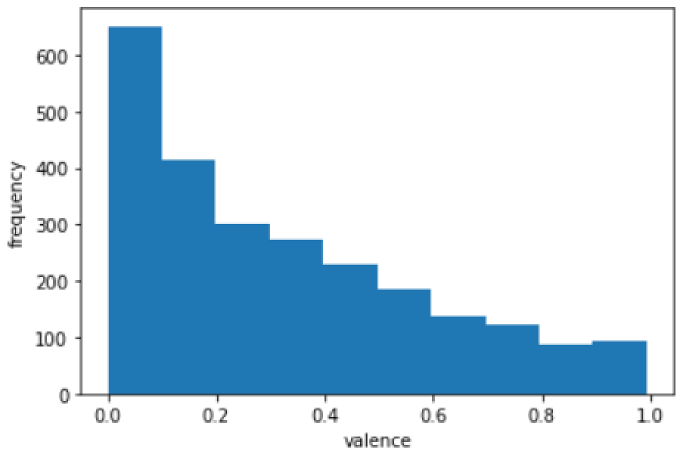
With this data now in tabular format, it is now possible to learn a user's music tastes. We should first look at features that most people notice when listening to a song, such as energy, valence, speechiness, acousticness, tempo, and then danceability. It should be noted that Spotify has scaled and standardized their data so that values are between -1 and 1, with exceptions to "loudness" and tempo, which are measured in decibels (db) and positive integers, respectively.

First, a look at the frequency of each song feature to get a feel for what I prefer in the music I listen to.



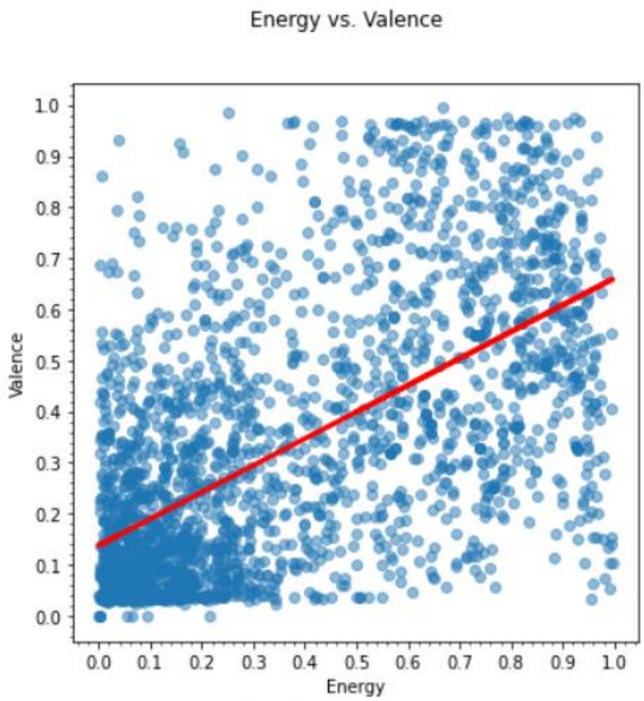


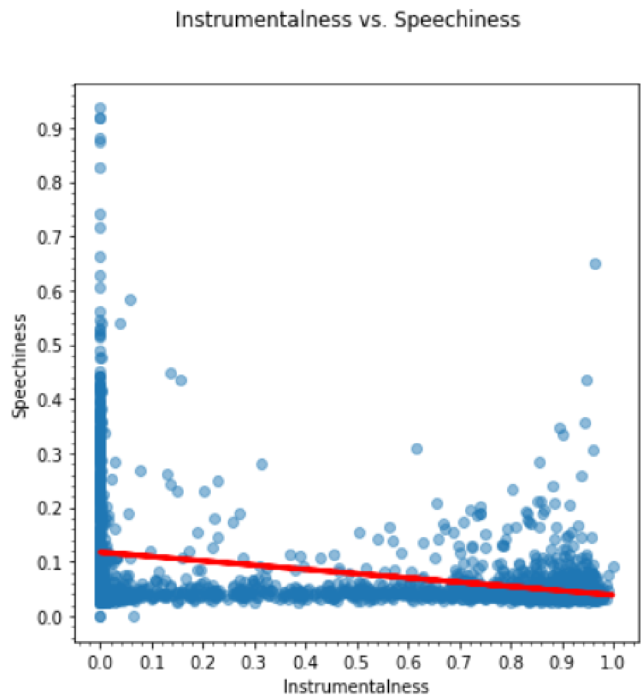



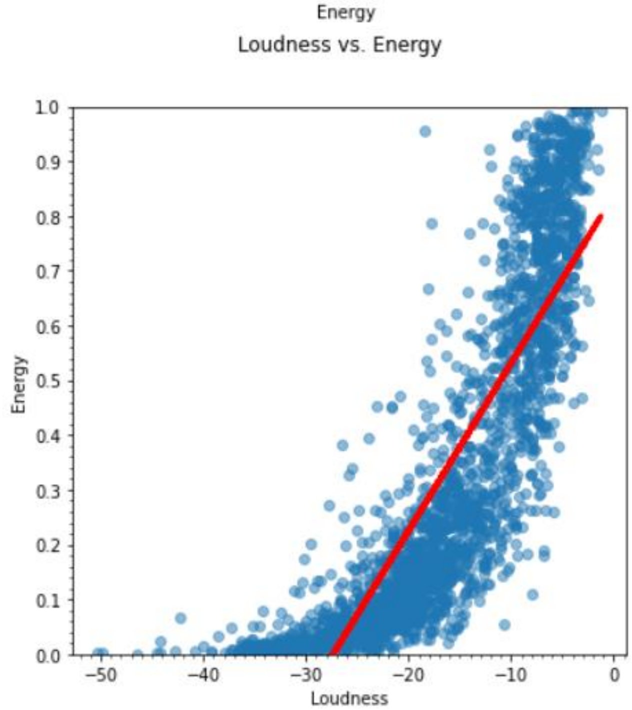



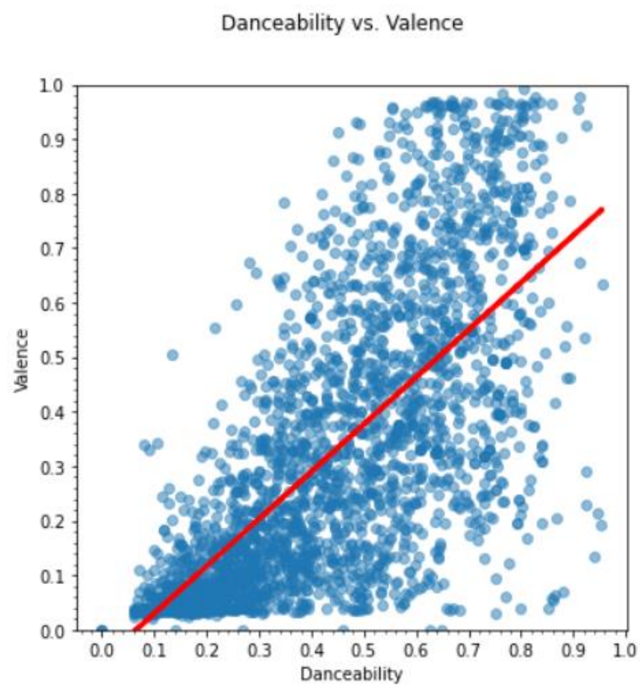

You can see that there is a bias towards music with very little speech. My preferred genres are classical (choral, orchestral, and chamber), jazz, fun, or a fusion of R&B/fun/soul, all of which feature little to no vocal arrangements. This would also explain the frequency of acousticness and energy, as classical music is typically not as energetic nor as frequent as other popular genres. An explanation for the distribution of instrumentalness would be that classical choral music typically does not use any sort of instruments. Next, a look at how these features correlate with each other.

The graphs represent, in clockwise order: a scatterplot and linear regression of the energy vs. valence, instrumentalness vs. speechiness, loudness vs. energy, and danceability vs. valence. There is a very clear positive correlation between loudness and energy, energy and valence, and danceability and valence. The correlation between these factors is appropriate within the context of the genres that I listen to the most, as they correspond with what the genres are primarily known for. These relationships are not universal, as a more diverse music-listener will have a different spread with very different linear regressions. For example, someone that listens to a lot of pop music will have a more balanced spread or nearly no correlation between speechiness and instrumentalness, as the genre typically has an equal balance between the components of lyricism and band presence.



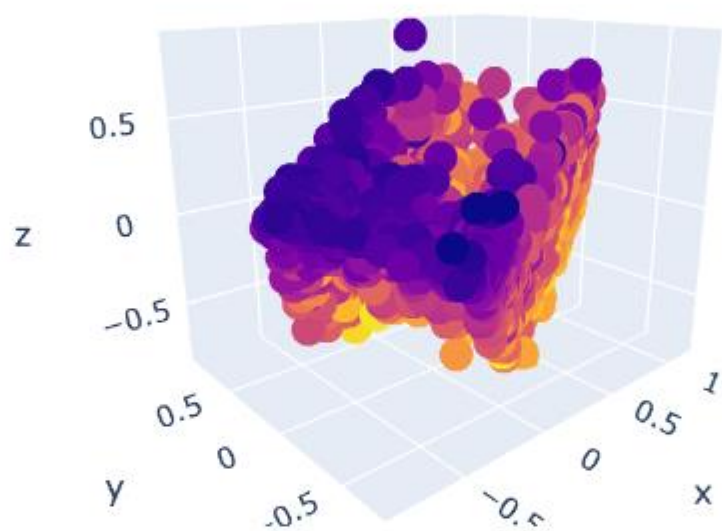




I will use principal component analysis (PCA) and TSNE plots to illustrate how closely related the songs are to each other based on chosen song features. The closer in proximity points are to each other, the more similar they are. Note that one of the plots is in 3D, but a Python file will allow you to navigate its dimensions and view each marker to see artist and track name.

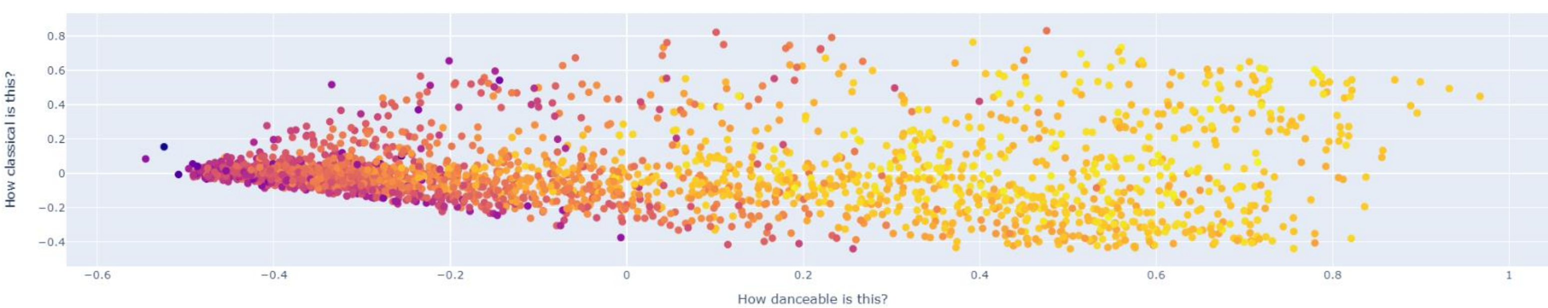
3D PCA Plot.

Components: acousticness, danceability, energy,

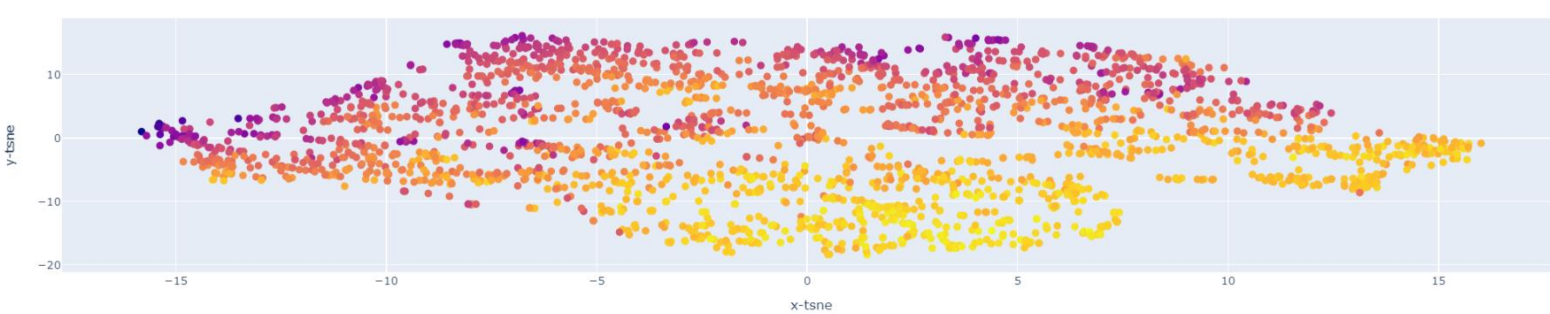
instrumentalness, liveness, loudness, speechiness, tempo, valence




2D PCA Plot. Components: energy, liveness, tempo, valence.




TSNE Plot. Components: acousticness, danceability, energy, instrumentalness, liveness, loudness, speechiness, tempo, valence.



Both PCA plots place two genres relatively close together: classical and 80's Japanese city pop. They are very different genres, but their affective and musical content are similar. This is due to the use of chord progressions, harmony, song structure, and use of melody. If genre were recorded in the data, it would be possible to label each point by genre to get a better visualization of how different or similar each song is from each other. This would also highlight a problem with classification by genre, as there are many labels for songs that are similar in style, but the artists might consider their own work to be different from their contemporaries.

## IV. Discussion and Conclusions

With the data that Spotify records about their song catalogue, it is possible to find correlations between song features. However, these correlations would likely only appear when examining songs from a narrow selection of genres. It can also be surmised that Spotify uses this data and relationships to power their own recommendation services, though they most likely restrict the songs to be within the same genre to avoid this issue. One possible way to circumvent this would be to have more data on the tracks and instruments used in the song, which could be collected automatically through music information retrieval and a model trained to identify instruments or by having artists input it manually when submitting a song to be hosted by the company. A service could also be developed that would allow Digital Audio Workstation (DAW) files to be uploaded and analyzed by a trained model to record the information within these files. Some additional features that could be included into their API could be tonalness, rhythmic complexity, harmonic complexity, and general complexity. Tonalness would measure how tonal a song is (a tendency for a song to stay in one key), while rhythmic and harmonic complexity would measure the rhythmic or harmonic variation throughout the entire song. Complexity could be a combination of these metrics, similar to how energy is a measure of tempo, loudness, and valence.

The idea of music recommendation based on acoustic features has been around for quite some time, as Shao proposed using dynamic music similarity measurements to improve similarity measurements that would be used by an algorithm or model in 2009 [2]. It has been noted that extraction of audio features is difficult the more polyphonic (multiple voices/instruments) an audio sample is, but with open-source libraries like Spleeter [3], it is possible to separate a compressed audio file into its individual tracks. Spotify's use of simplified features is a step in the right direction for recommendation models, but recommendations based purely on extracted acoustic features is becoming more attainable with progress in machine learning.

# References

[1] *The Echo Nest*. (2017, April 20). https://web.archive.org/web/20170420095740/http://the.echonest.com/

[2] Shao, B., Wang, D., Li, T., & Ogihara, M. (2009). Music recommendation based on acoustic features and user access patterns. IEEE Transactions on Audio, Speech, and Language Processing, 17(8), 1602–1611. https://doi.org/10.1109/TASL.2009.2020893

[3] Moussallam, M. (2020, February 3). Releasing spleeter: Deezer r&d source separation engine. Medium. https://deezer.io/releasing-spleeter-deezer-r-d-source-separation-engine-2b88985e797e

[4] *Web API reference | Spotify for developers. (n.d.). Retrieved December 14, 2021, from https://developer.spotify.com/documentation/web-api/reference/#/operations/get-several-audio-features*

[5] *Welcome to Spotipy! —Spotipy 2.0 documentation*. (n.d.). Retrieved December 14, 2021, from https://spotipy.readthedocs.io/en/2.19.0/