Spotify Prediction

Group Members: Natalie Gomas, Davis Ulrich, Alex Popescu, Tyler Nunez, Graham Kutchek

Research Problem Statement

Research Problem

- We set out to develop a DIY predictionary algorithm for our personal playlists
- Based on our own spotify playlists, could we develop similar song suggestions?
- We wanted to build a model which provided personalized predictions, based on input songs

Ain't No Sunshine	Bill Withers	Just As I Am	2019-12-01
Breakfast In America - Remastered	Supertramp	Breakfast In Ame...	2016-10-24
Moving in Stereo	The Cars	The Cars	2019-05-09

Research Problem, cont.

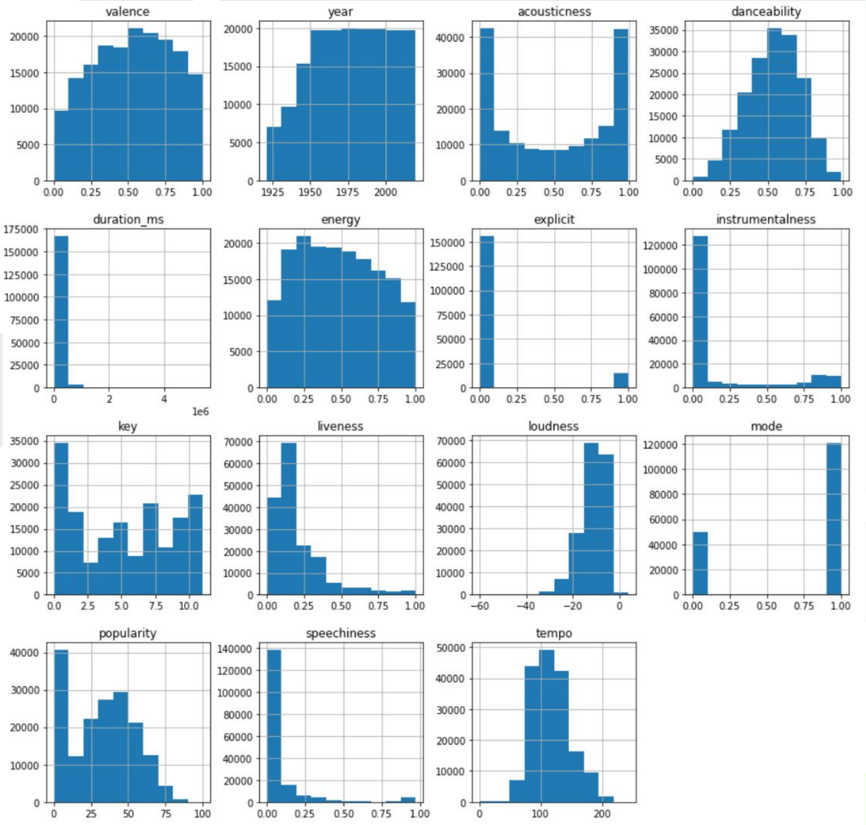
- Stakeholders for this work include anyone who uses Spotify and has interest in finding new music!
- Spotify's current prediction models are hidden and therefore our research enables users to have more hands on control of song suggestions
- We all love music, which motivated us to pursue this music centric data topic

The Dataset

The Dataset

- The data we utilized was sourced from a **kaggle** project which attempted to provide open-source, large-scale spotify data.
- Including over 170,000 track entries, the dataset was more than comprehensive for the scope of our goals

Here are the features of our dataset:



Dataset Cont.

- The Dataset included multiple csv's which sorted individual tracks by song, artist, genre, and year
- Our main dataframe was sorted by song, and had 16 features that described each song

Features:

Primary:

- **id** (Id of track generated by Spotify)

Numerical:

- **acousticness** (Ranges from 0 to 1)
- **danceability** (Ranges from 0 to 1)
- **energy** (Ranges from 0 to 1)
- **duration_ms** (Integer typically ranging from 200k to 300k)
- **instrumentalness** (Ranges from 0 to 1)
- **valence** (Ranges from 0 to 1)
- **popularity** (Ranges from 0 to 100)
- **tempo** (Float typically ranging from 50 to 150)
- **liveness** (Ranges from 0 to 1)
- **loudness** (Float typically ranging from -60 to 0)
- **speechiness** (Ranges from 0 to 1)
- **year** (Ranges from 1921 to 2020)

Dummy:

- **mode** (0 = Minor, 1 = Major)
- **explicit** (0 = No explicit content, 1 = Explicit content)

Categorical:

- **key** (All keys on octave encoded as values ranging from 0 to 11, starting on C as 0, C# as 1 and so on…)
- **artists** (List of artists mentioned)
- **release_date** (Date of release mostly in yyyy-mm-dd format, however precision of date may vary)
- **name** (Name of the song)

Data Pre-processing

Data Pre-Processing

- For initial data pre-processing, we created unique identifiers for songs and artists that allowed us to represent them as integers
- We also removed low-impact features such as release date, and normalized columns to increase performance of the model

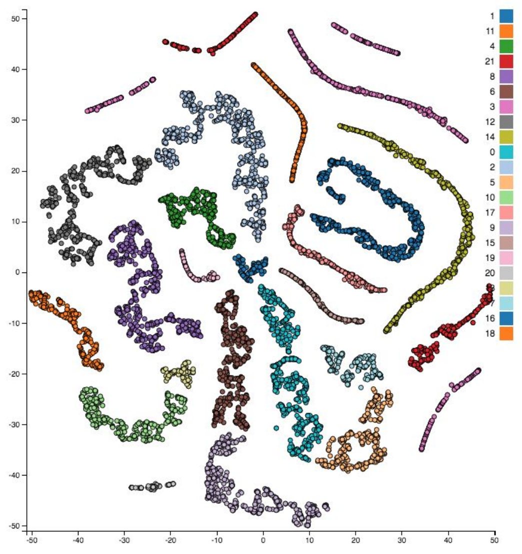
Dimensionality Reduction

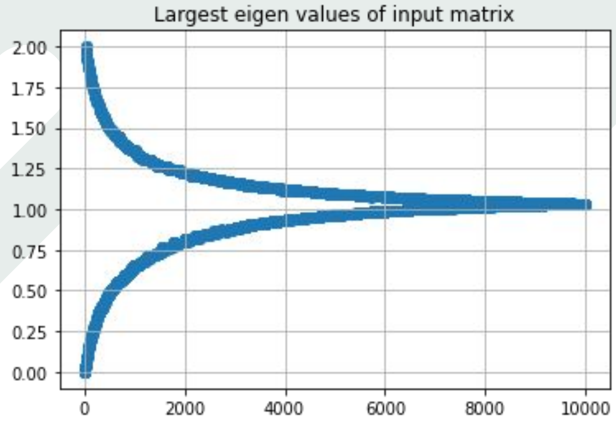
- Our last step before constructing the model was to reduce our 16 song features down to 2 features
- We used a combination of principal component analysis and t-SNE to do this
- This dimensionality reduction was crucial to do before fitting the clustering model or else we ended with all datapoints in the same cluster

Model Approaches

Spectral Clustering

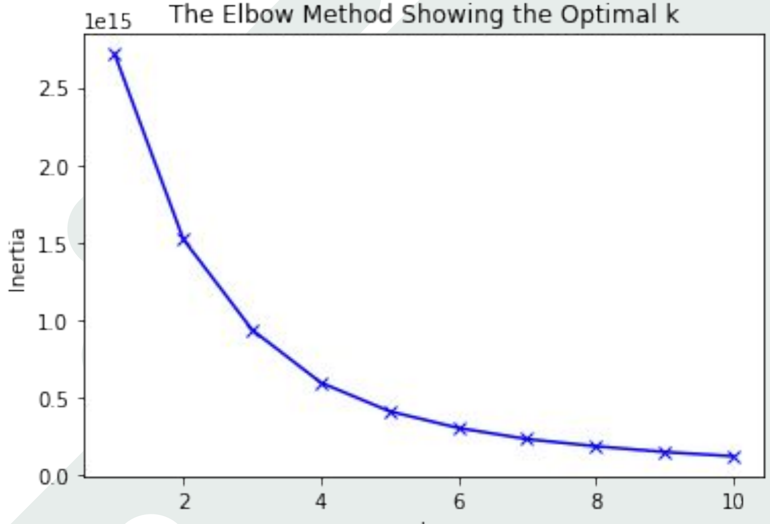
- We utilized spectral clustering to produce genre-based cluster mappings
- This approach allowed us to associate specific tracks with other tracks of the same genre
- Optimal k was found through eigen decomposition, but corrected by visualizing it

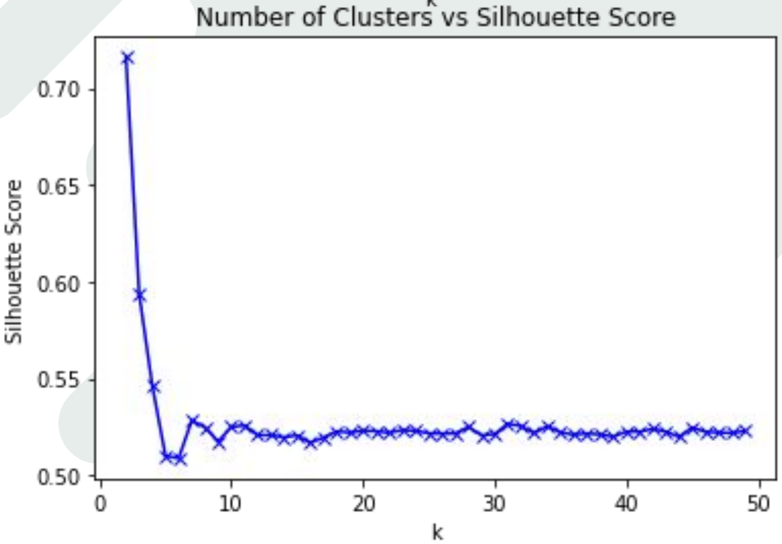


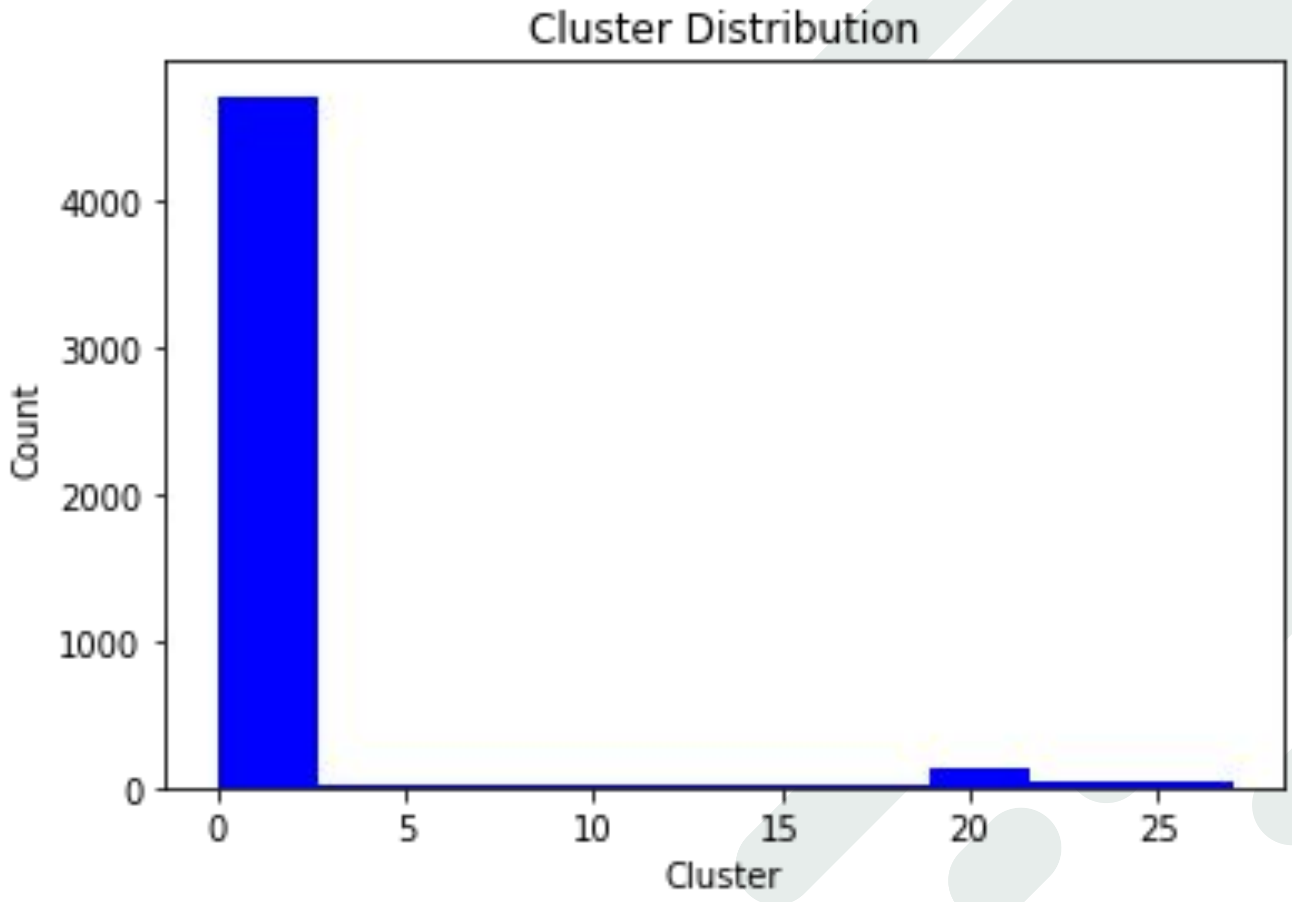


Other approaches

- Explored basic clustering with a k-means modeling process
- K-means process yielded inconclusive results compared to spectral model
- Dimensionality reduction was key to model success



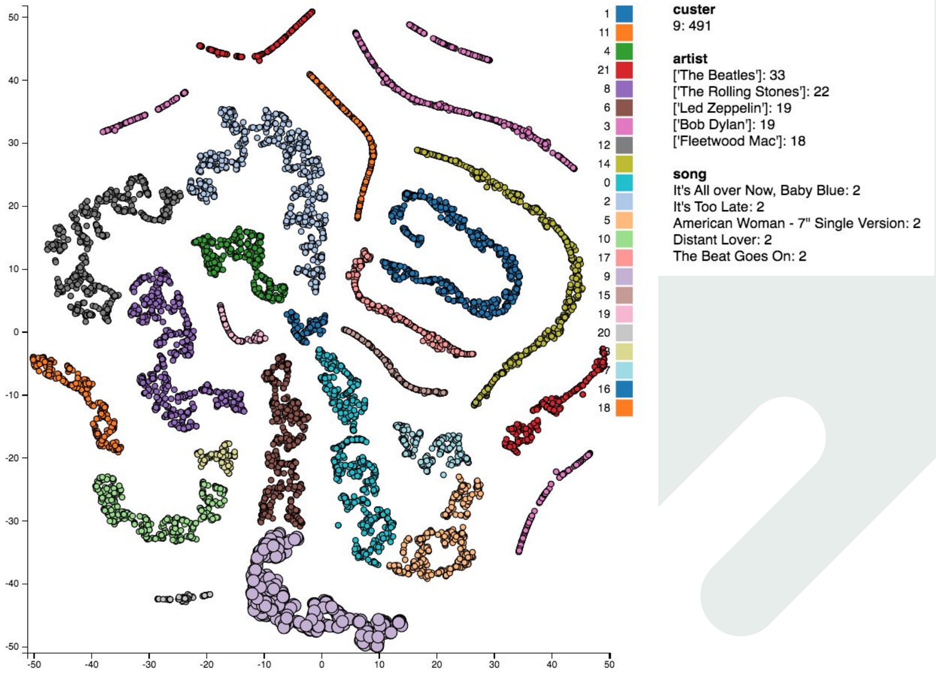




Cluster Distribution
4000
3000
2000
1000
0
Count
0
5
10
15
20
25
Cluster

Final Model

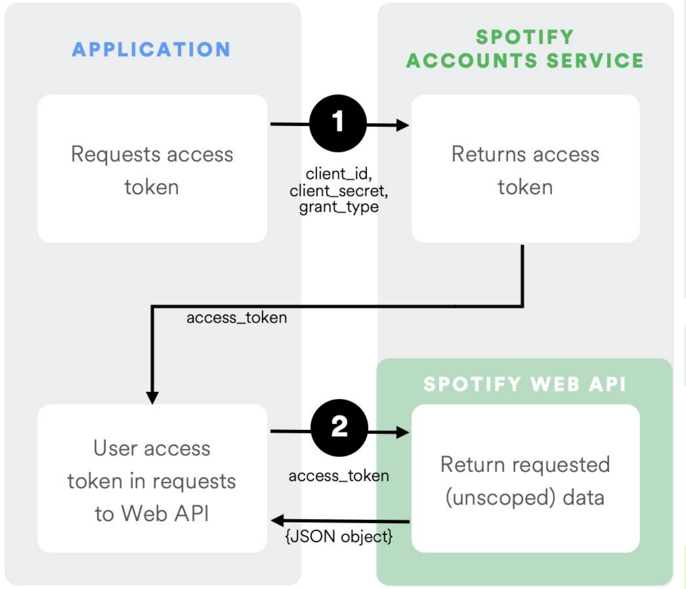
- Our final model had 22 clusters, each of which created their own distinct genre of music
- For example, cluster 9 at the bottom was comprised of rock & roll songs
- These clusters were made without using genre information in the model



Song Prediction

Generating Personal Playlist Data

- Used Spotify Developer API to get same metrics for personal playlists
- Created individual DataFrames for each person



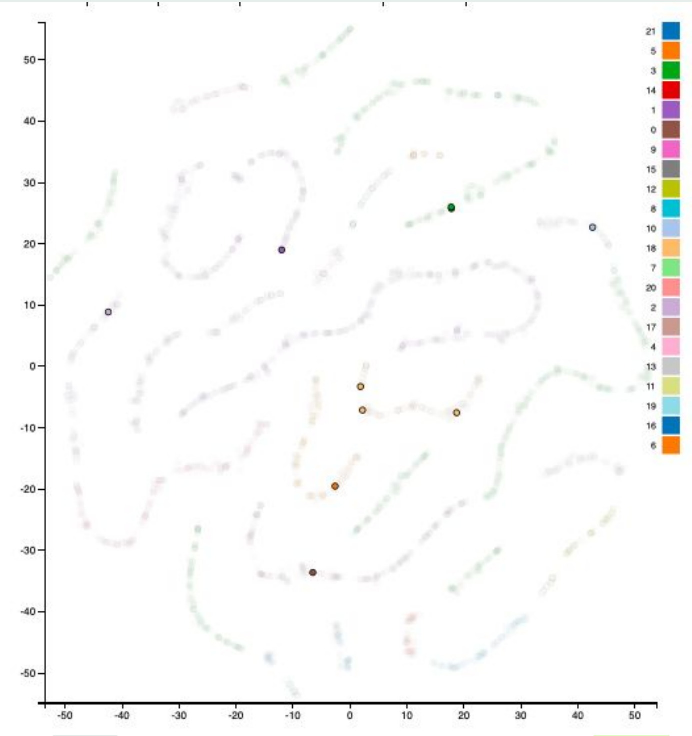
Adding Our Music

- Removed our songs from the main DataFrame
- Formatted our individual playlists to be visualized with main data set
- Assigned one clustering to find “home” cluster
- Assigned second clustering with our names

Recommendations
Alex: Caught Up -- Usher
Davis: I Remember You -- Skid Row
Graham: Roundabout -- Yes
Natalie: Haunting -- Halsey

Playlist based predictions

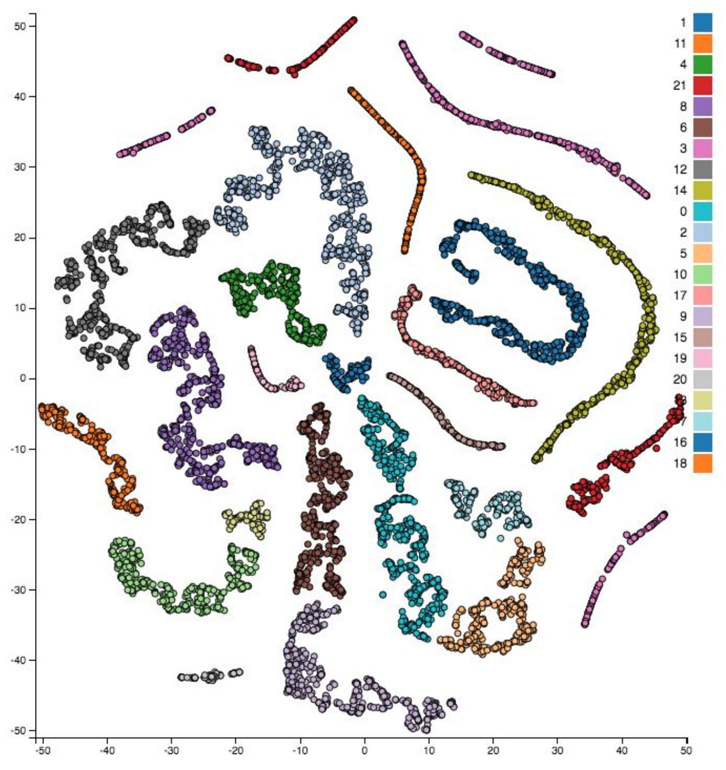
- Personalized suggestions started with formatting personal playlists
- Each song on the playlist was then compared with the clusterings
- Songs are then suggested based on the cluster with most songs from the playlist



Results and Conclusions

Overall Findings

- The model proved to ultimately produce strong suggestions based on our playlist inputs
- Predictions came at a relatively high computational cost



User takeaways

- Users can feed the model specific input data for training and subsequent playlists for fine tuned suggestions
- Overall the model allows user to diversify their music taste

Reflections

- Preprocessing, dimensionality reduction, and model selection were all essential for yielding useful predictions
- Elevated user customization and model control would be a future goal
- Expansion beyond Spotify?

Links / sources

- "Spotify Dataset 1921-2020, 160k+ Tracks ” Yamaç Eren Ay
  - https://www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks
- “Interactive 2D Visualization Tool for Cluster Analysis” Zach Pardos
  - https://github.com/CAHLR/d3-scatterplot