Determining Popularity of Pop Music Songs and Lyrics with Spotify Data

James Pecore, Data Scientist

Agenda

I. Introductions
II. Explanatory Data Analysis
III. Lyrical Sentiment Analysis
IV. Modeling
V. Conclusions

Problem Statement

1. Spotify uses its popularity parameter to rank songs, albums, and artists.
   a. This "stream-popularity" metric is based on how often users stream songs from Spotify.
2. But how does this stream-popularity metric compare with other metrics for popularity?
   a. This metric only shows how popular very recent artists are in general (not popularity according to genre or popularity by song/lyrical content).
3. As a result, historically VERY popular classic songs (by Earth, Wind, & Fire, The Beatles, and other "classic groups") are falsely labeled as “unpopular.” Additionally, artists who are VERY popular in their genre become ignored due to weighted artists from higher popularity genres like "pop."
4. We need a new metric for popularity. In fact, we need more than one new popularity metric.

Problem Statement - Cont.

When using Regression, Classification, and NLP Lyric Clustering to predict the popularity of a musical artist, how can evaluate whether or not to trust Spotify's ranking of popularity?

What other metrics of popularity should we define and recommend that Spotify and other top streaming sites adopt? What is our reasoning behind this logic?

Executive Summary

Giant Ordered Songlist

233,424 Songs

kaggle

149,310 Songs

Regression Modeling

Classification Modeling

Ordered Songlist

699 Songs

695 Songs

619 Songs w/ Lyrics

Sentiment Analysis

Clustering Analysis

Datasets - Ordered Songlist

Number of Songs (Before Cleaning): 699

Number of Songs (After Cleaning): 695

	Popularity	Valence	Energy	Loudness	Danceability	Liveness	Tempo	Acousticness
Mean	44.71%	48.36%	61.56%	-7.40	63.67%	17.00%	122	24.82%
Median	48.00%	47.60%	61.60%	-6.88	64.40%	12.30%	120	16.00%

Datasets - Giant Ordered Songlist

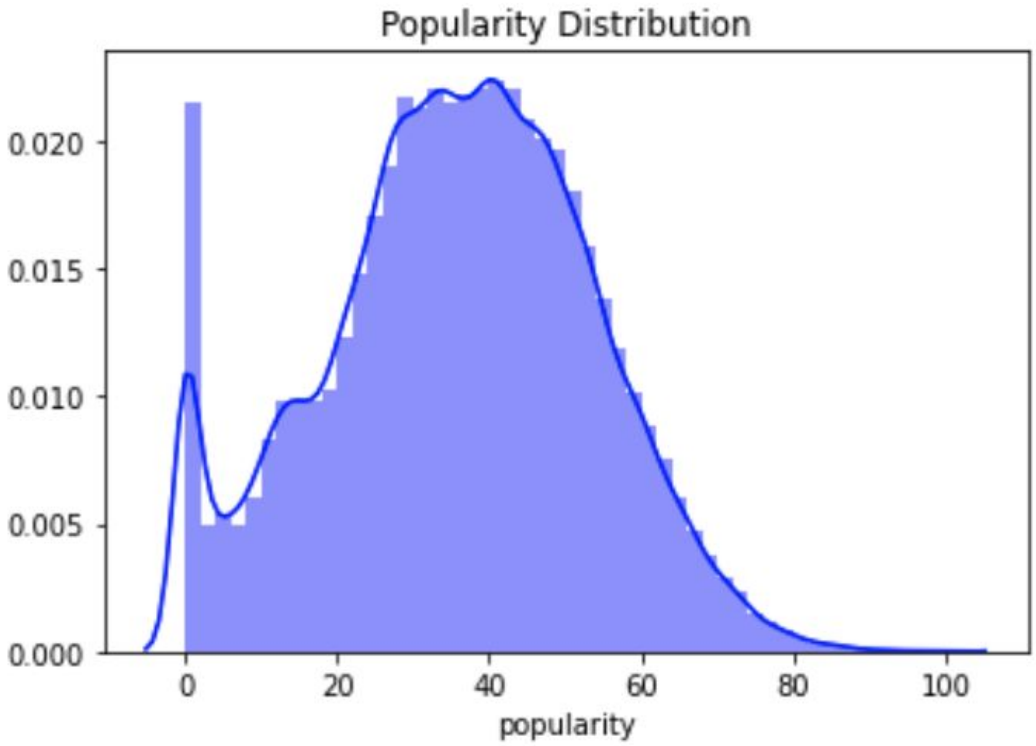
Number of Songs (Before Cleaning): 233,424

Number of Songs (After Cleaning): 149,310

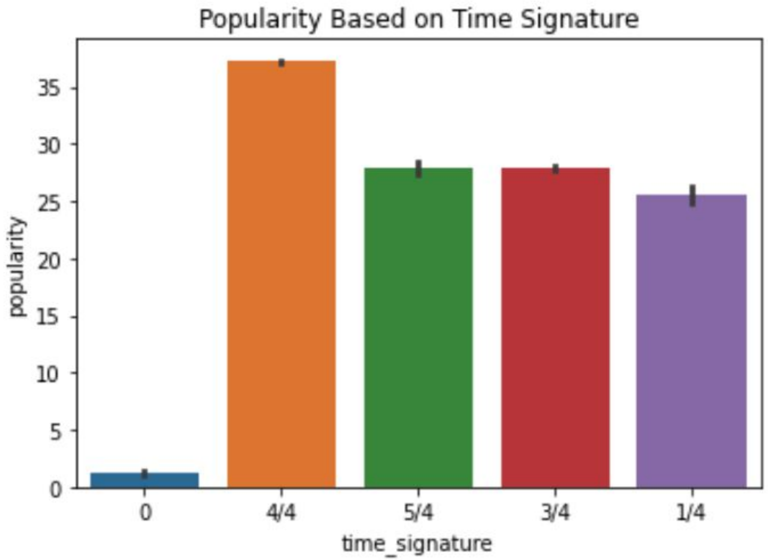
	Popularity	Valence	Energy	Loudness	Danceability	Liveness	Tempo	Acousticness
Mean	35.72%	44.87%	51.26%	-10.37	53.61%	22.85%	117	41.43%
Median	36.00%	43.70%	58.90%	-8.31	55.30%	13.10%	115	30.30%

Stream-Popularity Distribution

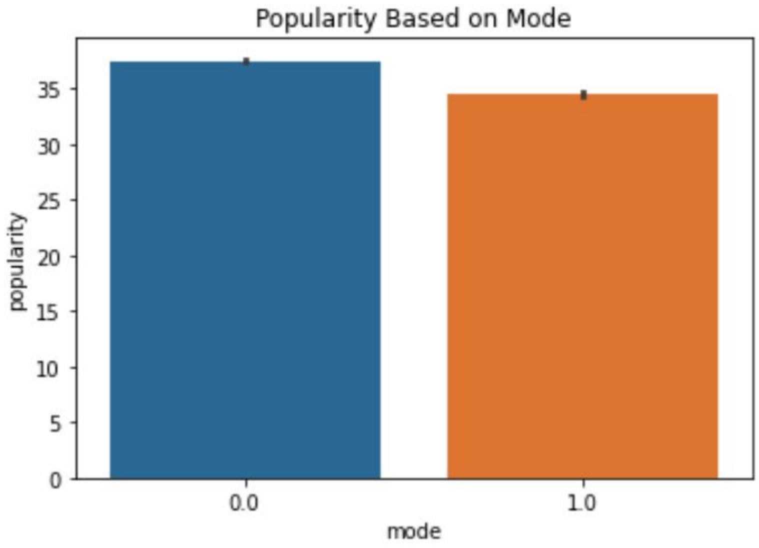
Distribution for Giant Ordered Songlist



Time Signature & Mode

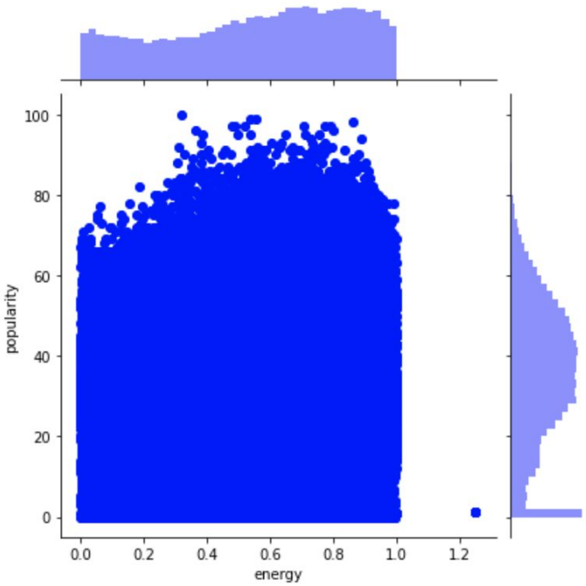


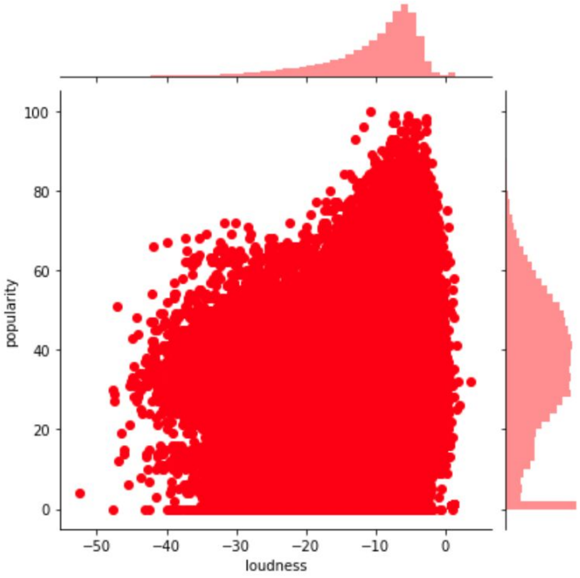
Neutral Correlation with Popularity

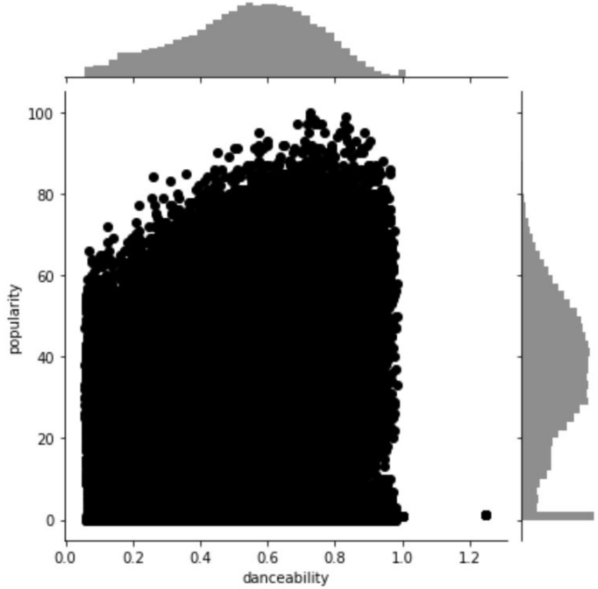


Neutral Correlation with Popularity

Energy, Loudness, & Danceability





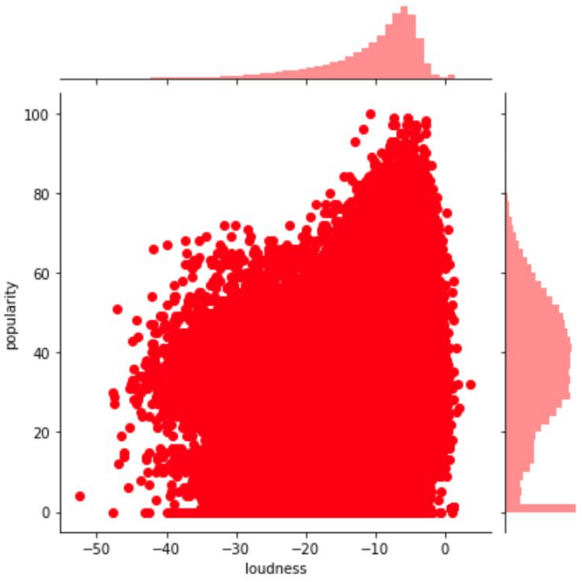


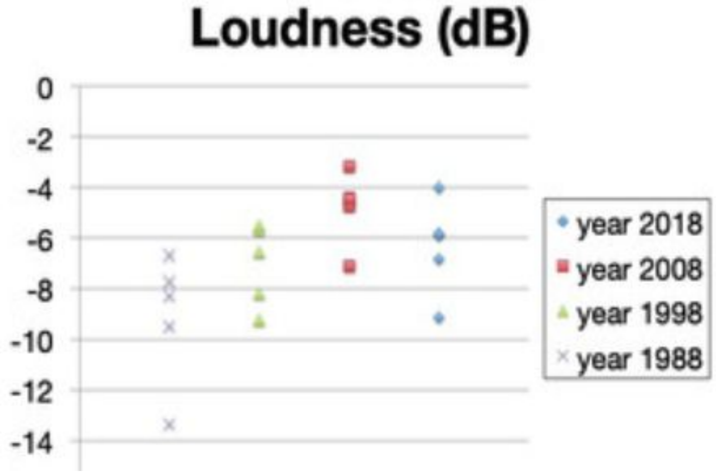
Correlation with Popularity: 24%

Correlation with Popularity: 35%

Correlation with Popularity: 21%

Loudness, Cont.

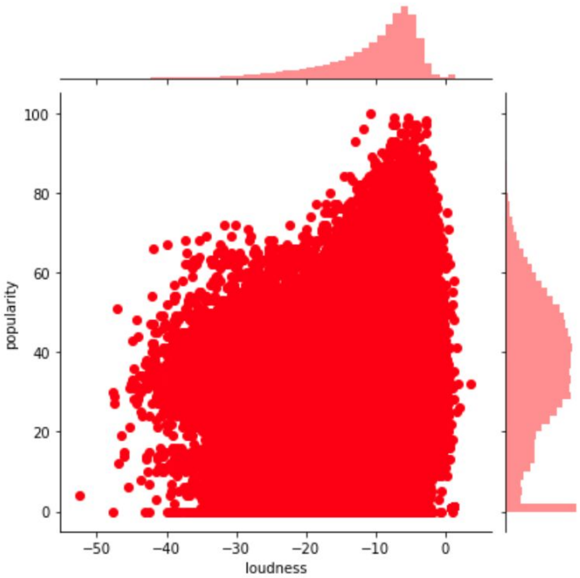




Correlation with Popularity: 35%

Elena Georgieva, Stanford University Data Scientist
from "HitPredict: Using Spotify Data to Predict Billboard Hits"

Loudness & Compression



Correlation with Popularity: 35%

Timeline of Compression:

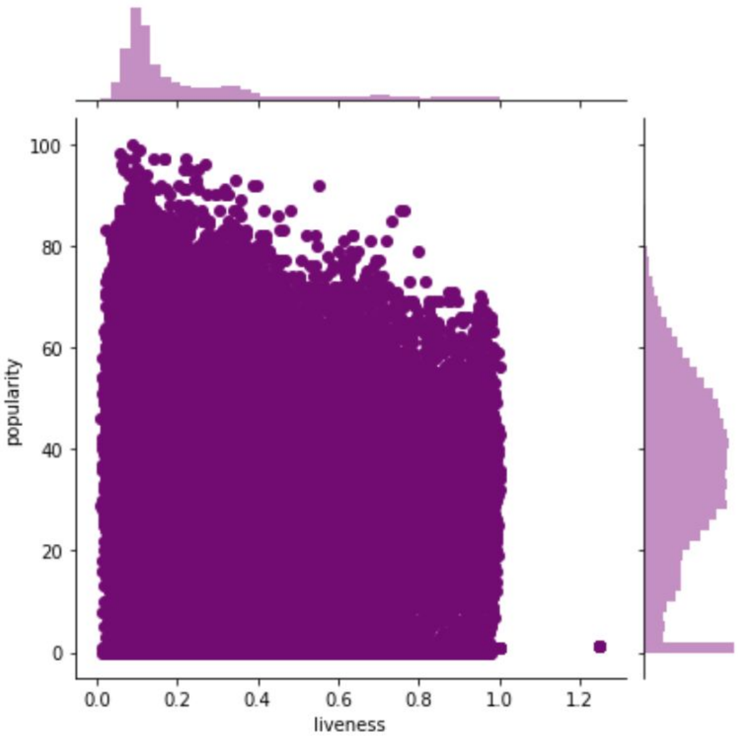
1930s-40s – Compression created to control volume of TV/Radio presenters whispering/talking quietly as-well as an entire audience clapping.
The first compressors just had In/Out functions and no other controls.

1937 – 'The Western Electric 110 limiting amplifier created, being considered to be the first commercially available compressor.

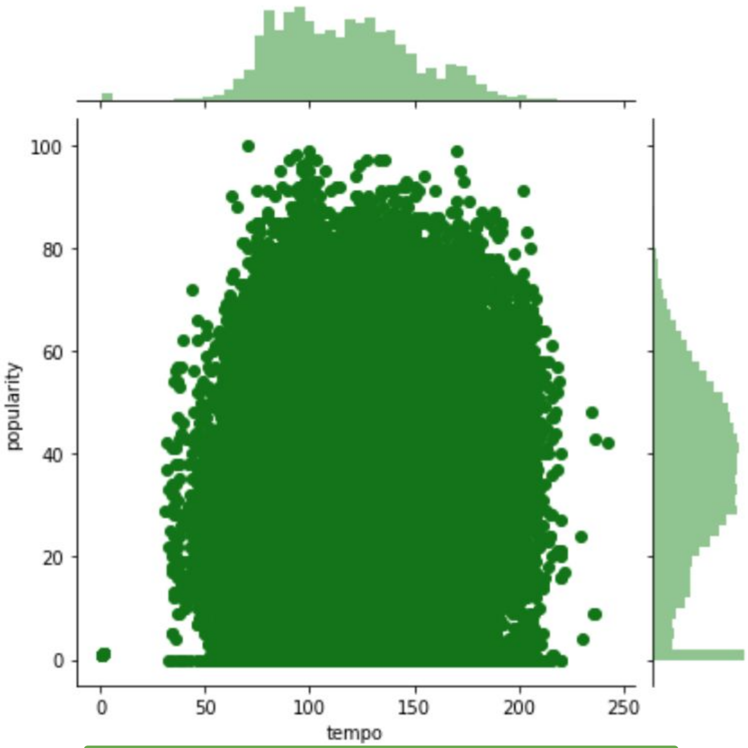
1980s – DAWs included Compression as standard in their software, rather than needing Compression Racks.

2007 – Tracks started using less heavy compression, experimenting with different compression levels

Liveness & Tempo

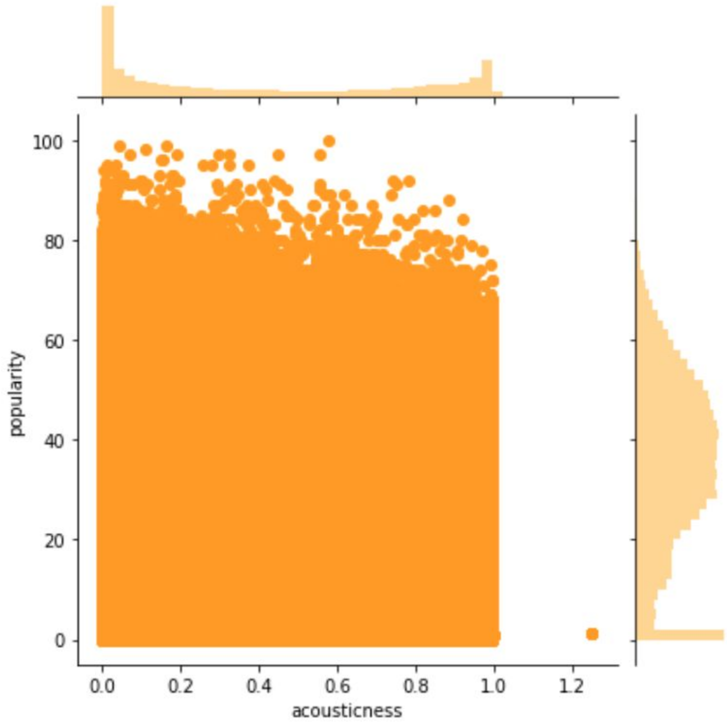


Correlation with Popularity: -19%

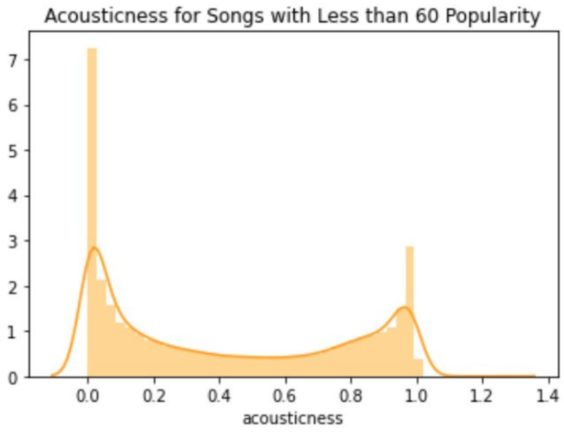


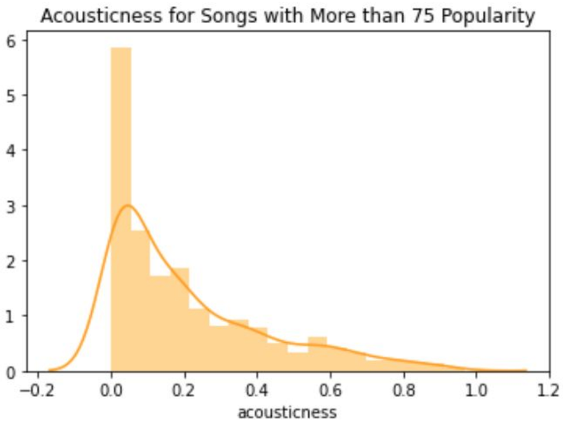
Correlation with Popularity: 12%

Acousticness

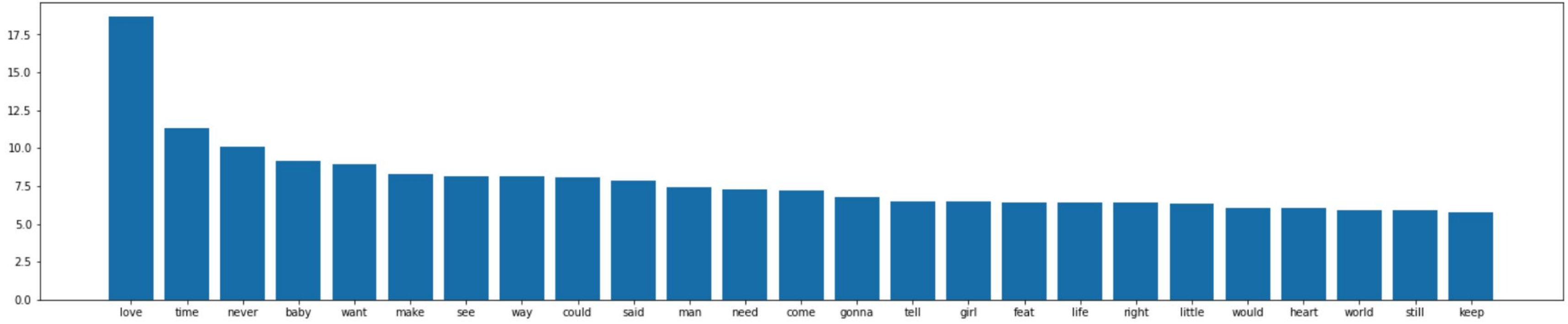


Correlation with Popularity: -0.39





Most Common Lyrics (Top 25)



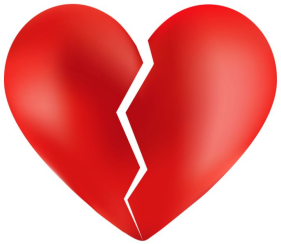
Most Common Lyrics involving Love (6/25):
Love, Baby, Man, Need, Girl, Heart

N.L.P.: "Love vs. Heartbreak"

Correlation with Popularity: 16%

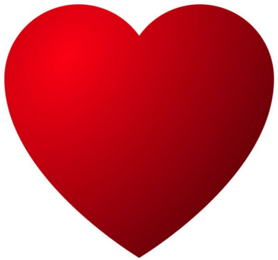
Heartbreak Words (-1):

- Cry
- Tears
- Distant
- Crazy
- Horrible
- Die
- Cold
- Pain
- Lonely
- Lost
- Fight
- Last
- etc.



Love Words (+1):

- Love
- Loved
- Baby
- Shawty
- Girl
- Boy
- Guy
- Date
- Kiss
- Good
- Great
- Together
- etc.



N.L.P.: "Hype vs. Chill"

Correlation with Popularity: 2.4%

Chill Words (-1):

- Chill
- Bed
- Coffee
- Make
- Friend
- Sleep
- Rest
- Alone
- Space
- Food
- Hang
- Hangout
- etc.

Hype Words (+1):

- Hype
- Yeah
- Baby
- Party
- Punch
- Tonight
- Livin / Living
- Together
- Wine
- Drink
- Alcohol
- Buzz
- etc.

N.L.P.: “Independent vs Cuffed”

Correlation with Popularity: -1%

Cuffed Words (-1):

- Need
- Cuffed
- Us
- Together
- Forever
- Love
- Bed
- Sleep
- Goodnight
- Sweetie
- Baby
- Sweetheart
- etc.

Independent Words (+1):

- Never
- Independent
- Bad
- Cheats / Cheated
- Rude
- Own
- Make
- Win / Winning / Won
- Money
- Power / Powerful
- Feminist
- Flawless
- etc.

Regression Modeling

	LinReg	KNN-R	Dec. Tree Reg	Bag. Dec. Trees	Rand. For.	AdaBoost
Training Score:	21.17%	48.62%	99.89%	90.82%	90.81%	24.34%
Testing Score:	20.94%	21.95%	-31.12%	34.63%	34.62%	23.58%
Train RMSE:	15.49	12.51	24.63	17.27	17.04	17.55
Test RMSE:	15.56	15.46	24.56	17.35	17.12	17.62

- Linear Regression is the Least Overfit
  - But still has a RMSE of 15.5 points (each prediction is on average that far away from the true value)
- Random Forest has the highest Testing Score
  - But still only predicts the song’s stream popularity accurately one-third of the time
- We CANNOT use Regression with Song Features to accurately predict stream Popularity

Classification Modeling

	LogReg	KNN-C	Dec. Tree Reg	Bag. Dec. Trees	Rand. For.	AdaBoost
Training Score:	91.81%	92.33%	99.97%	99.96%	99.96%	91.81%
Testing Score:	91.93%	91.02%	85.95%	91.89%	91.93%	91.93%
Train RMSE:	0.2860	0.2860	0.2897	0.2860	0.2861	0.2860
Test RMSE:	0.2839	0.2839	0.2876	0.2839	0.2839	0.2839

- We can predict whether a song is popular (greater than 75%) w/ 91.9% accuracy
- It's easy to tell whether a song is popular or unpopular
    - You CANNOT predict a song's exact stream popularity with just musical features
    - It's hard to tell WHY a song's stream popular with just its musical features
    - Lyrics and Branding are more important for song stream popularity than one might think
    - We CAN use musical features to inform us loosely why stream popularity might be higher

Clustering Lyric Vectors with SpaCy

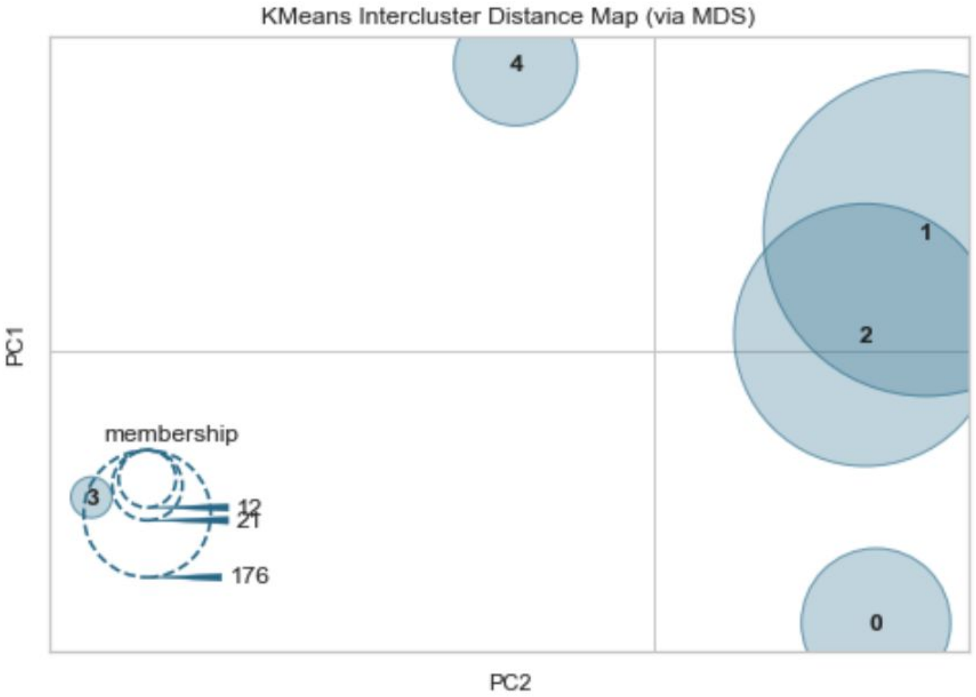
0 = Soft Pop

1 = Pop / Rock

2 = Rock / Hip - Hop

3 = Spanish Music

4 = (No Lyrics)



General Recommendation:

- Increase Energy and Danceability to be around average values (60%)
- Decrease Acousticness and use digital instruments / music production
- Only increase Loudness to make it easy to listen to on a mobile phone
- If you mention “love” more in your song, it can’t hurt

Recommendation 1: All-Time Stream Popularity

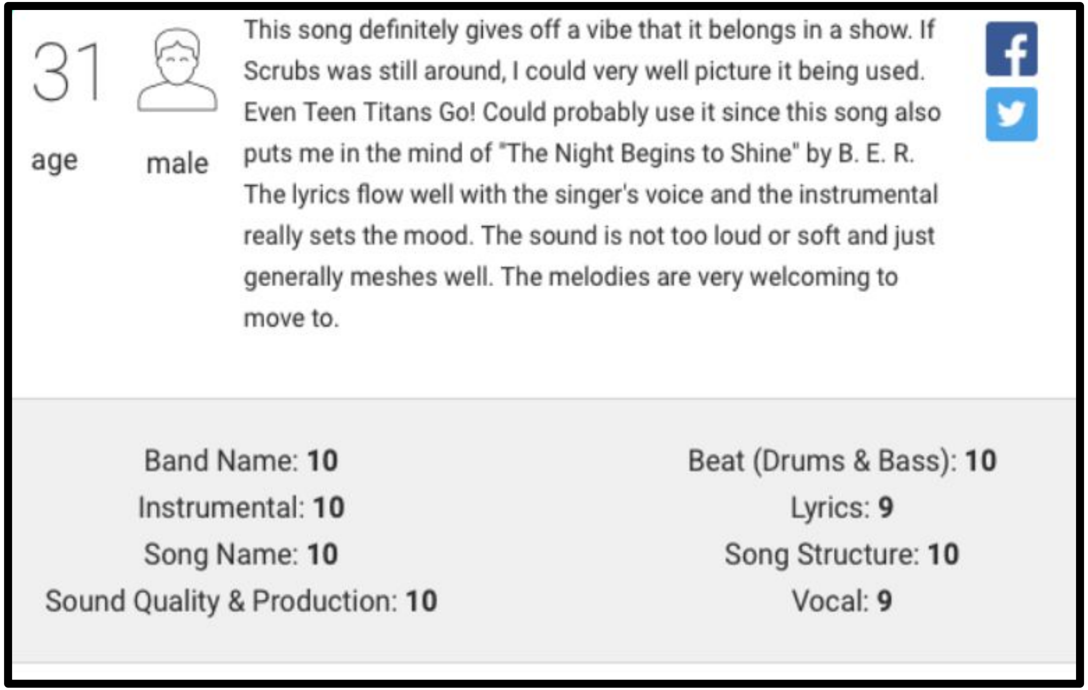
- A new popularity metric based on:
  - “Total Number of Streams of All Time”
- This will let us grade older songs comparably with newer songs
- We could compare historical trends in music with current trends without improper scaling worries from Stream Popularity

Recommendation 2: Personal Popularities

- Bring back a 5-Star or "One-to-Ten" review system for each user's songs
- This will let us assess what styles each individual user prefers
- This will allow us to create a Regression Model and Recommender System for the user for their highest rated songs, improving user turnout

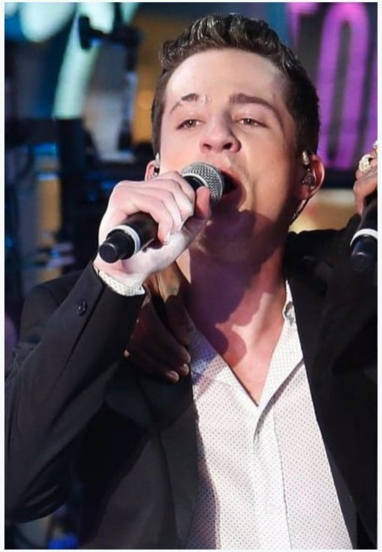
Recommendation 3: Song Features Review

- Create an optional Features Review section for each song in Spotify
- Vectorize the words used in Features Review
- Create Sentiment Analyses with these Vectors
- Create a recommender system with these Vectorized Sentiments



Recommendation 4: - Individual Research

Charlie Puth



Puth at Times Square in 2015

Lady Gaga

Lady Gaga during an interview for NFL Network in 2016

Lizzo

Lizzo performing in London; November 2019

College: Berklee College of Music
Major: Music Production & Engineering

College: NYU Tisch School of the Arts
Major: Music (Songwriting)

College: University of Houston
Major: Music (Classical Flute)

Further Research & Future Projects

1. Using Parallel Programming (AWS) not Serial Programming (Jupyter, Google)
   a. Processing all 150,000 song lyrics
   b. Extending NLP Sentiment Lists and Performing Sentiment Analysis on all 150,000 song lyrics
   c. Performing NLP Clustering with SpaCy on all 150,000 song lyrics
2. Using Public Opinion on Pop Songs for Sentiment Analysis
   a. Scraping News/Twitter/Reddit/Tumblr/etc. Posts for All Songs
   b. Using NLP to Determine if Public Opinion Towards Artist is Negative, Neutral, or Positive
3. Using Song Attributes & Reviews to Create a Recommender System for Songs
   a. Publish online or submit to Record Labels / Streaming Companies

Sources

Cho, Youngmin P. "Quantify Music and Audio." *NYC Data Science Academy*, 3 June 2019, nycdatascience.com/blog/student-works/web-scraping/spotify-x-billboard/.

Georgieva, Elena, lecturer. "HitPredict: Using Spotify Data to Predict Billboard Hits." ICML 2020, researcher by Nicholas Burton and Marcella Suta, Stanford University, 18 July 2020.

Gingeleski, Ashley. "Spotify Web API: How to Pull and Clean Top Song Data Using Python." *Ashley Gingeleski*, 11 Nov. 2019, ashleygingeleski.com/2019/11/11/spotify-web-api-how-to-pull-and-clean-top-song-data-using-python/.

Hamidani, Zaheen. Kaggle, 2019, www.kaggle.com/zaheenhamidani/ultimate-spotify-tracks-db.

Itsaam. "History and Development of Compression." *Music Tech Student*, 11 Aug. 2013, musictechstudent.co.uk/music-technology/history-and-development-of-compression/.

Loot, Rare. "Extracting Spotify data on your favourite artist via Python." *Medium*, 30 Dec. 2018, medium.com/@RareLoot/extracting-spotify-data-on-your-favourite-artist-via-python-d58bc92a4330.

Passy, Jacob. "How Spotify influences what songs become popular (or not)." *MarketWatch*, 18 June 2018, www.marketwatch.com/story/how-spotify-influences-what-songs-become-popular-or-not-2018-06-18.

Pierre, Sadrach. "Analysis of Top 50 Spotify Songs using Python." *Medium*, Towards Data Science, 27 Dec. 2019, towardsdatascience.com/analysis-of-top-50-spotify-songs-using-python-5a278dee980c.

Sahu, Apratim. "Country-wise visual analysis of music taste using Spotify's API & Seaborn in Python." *Medium*, Towards Data Science, 12 June 2020, towardsdatascience.com/country-wise-visual-analysis-of-music-taste-using-spotify-api-seaborn-in-python-77f5b749b421.

SpaCy. spacy.io/. Sept. 2020.

Spotify for Developers. Spotify, developer.spotify.com/dashboard/. Sept. 2020.

Spotipy: Read the Docs. Spotipy, spotipy.readthedocs.io/en/2.16.0/. Sept. 2020.

Tweepy Documentation. Tweepy, docs.tweepy.org/en/latest/. Sept. 2020.

Questions?