MONU-MEL-DATA-PT-11-2020-U-C

Project Proposal – Project 1

Dancing Through The Decades

Objectively measuring music for the last hundred years using Spotify

Presentation date: 21-JAN-2021

**Group**:

Brett Scotland

Darren McMurtie

Jesse Tan

Kailyn Yong

Will Bobzin

Premise:

Introduce the topic, start with something broad - all about music e.g. Music is a common thing that unite people…

Then specific – about danceability

Shorten the hypothesis, make it more concise

Last part – tools you plan to use e.g. pandas, matplotlib (top 3)

Spotify offers a powerful database, containing data for over 160K songs. The database includes some ad-hoc metrics, such as *loudness*, *danceability* and *key*. As a group, we identified “Music” as one of our common interests. When exploring what kind of data project can be developed around that topic, the opportunity to use the Spotify database came up.

After a preliminary exploration, we were impressed by the thoroughness of the database, which has every single song categorized by year of release.

~~By Ryan’s suggestion,~~ we decided to implement a “local twist” to make for a more compelling story while adding a level of complexity that meets the standards require for this project. To this avail, we have included a database of Triple J Top 100 lists for 1993 to 2017. This database will act as a guide to reflect Australian music preferences. By mashing these two databases, we will explore the following areas:

* What makes a Triple J winner? Mashing data between Spotify and Triple J databases for 1993 to 2017. Get technical/numerical information for each song based on Spotify analytics. Compare some key elements in all triple j winners to determine commonalities – For example: Liveness, loudness, danceability. Hypothesis: explore first.
* What is the perfect time to release a hit? Using the release date informed by Spotify, we can find what are the most common release dates for the winners? Hypothesis: perfect launch date is March to June, as it takes a few months for people to get to know a song and start liking it enough to vote for it.
* Australia songs – What kind of Australian songs do people vote for in the Top 100? We will make a subset of Australian artists and measure their speechiness and Liveness and compare those aggregate values with the rest of the song universe in the historical top 100 list. Hypothesis: Top 100 voters will prefer Aussie songs with higher speechiness and liveness as they will choose songs they’ve enjoyed live and that showcase similar accents to theirs.
* Do Aussies like long songs? We will compare average duration for top 100 songs for each year, against the average duration of all songs released that year. This will tell us if Aussies prefer relatively short songs. Hypothesis: COMPLETE

**Limitations and challenges:**

* Mashing is relatively complex as it has to be performed by pivoting *artist* and *title* fields, which have differences in punctuation and usage of special characters. This is particularly challenging in terms of spelling of foreign names (eg. Björk) and multiple artists.
* Currently between 2 and 4 million votes are cast for the Australia date top 100. As significant as that number is, it pales in comparison to the 138 million active Spotify subscribers. In other words, the Triple J dataset is heavily skewed towards young Australians. While we can use the available data to extract trends about Australian music tastes, none of them could be extrapolated to the Spotify listener base.
* Technical data for each song comes from the Spotify algorithmic analysis. While it’s reliable and more importantly, consistent, we have no access to its inner workings and therefore can’t corroborate their accuracy.

**Sources**:

Categories explained - <https://developer.spotify.com/documentation/web-api/reference-beta/#category-tracks>

Database, extracted by a KAggel user from the Spotify API: <https://www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks?select=data.csv>

Triple J Top 100 Database - <https://github.com/majames/hottest100/>