CS216 Project Proposal: Group 26

Jack Lichtenstein, Shannon Houser, Libba Lawrence, and Linda Tang

Part 1: Introduction and Research Questions

Spotify is one of the world's largest music streaming service providers with over 365 million monthly active users. The Spotify database includes a variety of information on the characteristics and metrics of song tracks. Each song on Spotify is associated with a popularity index ranging from 0 to 100 reflecting how popular a song is relative to other songs. In this project, we plan to apply the analysis techniques that we learned in class to the Spotify music database. The **research question** is how can we predict the popularity of a song using characteristics such as danceability, energy, loudness and artist (descrbed below in the data sources section). We plan to explore different machine learning algorithms such as linear regression, decision trees and support vector regression and compare their relative performance on prediction. The scope of the project is feasible since the dataset we will use is readily avaiable online through Kaggle, the dataset contains a variety of songs and variables for analysis and it's in a relatively clean format. The algorithm from our project could be helpful for the artist community to predict the popularity of new release songs ahead of time and could inform future researches on understanding what factors influence popularity on Spotify.

Part 2: Data Sources

The data used for this project comes from an episode of the Sliced data science challenge held in summer 2021. The data is accessible publicly on Kaggle, and was initially extracted from the Spotify Web API. The competition provides three datasets, each of which we will make use of to answer our research questions:

- train.csv - the training set
- test.csv - the test set
- artists.csv - data about artists in the dataset

The train and test datasets are formatted the same way. We read in `train.csv` and `artists.csv` below. Additionally, we provide an in-depth data description as well.

In [1]:

```
# import some libraries
import os
import pandas as pd
import numpy as np
```

In [2]:

```
# change to one directory back (only run this once)
os.chdir(os.path.normpath(os.getcwd() + os.sep + os.pardir))
```

In [3]:

```
# read in spotify data
spotify_train = pd.read_csv("data/train.csv")
spotify_train.head()
```

Out[3]:

	id	name	popularity	duration_ms	artists	id_artists	danceability	energy	key	loudne
0	269	blun7 a swishland	63	167760.0	['tha Supreme']	['19i93sA0D7yS9dYoVNBqAA']	0.692	0.792	7	-5.984
1	27504	Que Me Perdone Tu Señora	42	150640.0	['Manoella Torres']	['4JRKcLbpjobmoOVoOXPd6y']	0.608	0.447	6	-12.15
2	16082	愛唄 ~since 2007~	42	242373.0	['whiteeeen']	['6v3VFX2qIWthj4Lr5QIxts']	0.572	0.782	8	-5.275
3	14585	Let me be your uncle tonight	12	202989.0	['Tvíhöfði']	['6rmrk3Jk0Ecf8fjioCCZmV']	0.855	0.470	7	-9.252
4	14740	Never Going Back Again - 2004 Remaster	40	134400.0	['Fleetwood Mac']	['08GQAI4eElDnROBrJRGE0X']	0.654	0.336	6	-12.82

In [4]:

```
# read in spotify artist data
spotify_artists = pd.read_csv("data/artists.csv")
spotify_artists.head()
```

Out[4]:

	id	followers	genres	name	popularity
0	55CXG5KDJpRYwBopfYAJHa	21756	['country blues', 'country rock', 'piedmont bl...	Jorma Kaukonen	40
1	08mjMUUjyTchMHCW7evc3R	640993	['turkish pop']	Hande Yener	62
2	3Ebn7mKYzD0L3DaUB1gNJZ	161509	['celtic', 'irish folk']	Christy Moore	56
3	7GfaHcpmNcrcHoyGnOBsAz	9578	['kindermusik', 'kleine hoerspiel']	Die Biene Maja	56
4	1DYXGLnfNDt8mO2aK9k83j	48876	['opm', 'vispop']	Jay-R Siaboc	39

train.csv/test.csv

- id (Unique identifier of track)
- name (Name of the song)
- popularity (Ranges from 0 to 100)
- duration_ms (Integer typically ranging from 200k to 300k)
- artists (List of artists mentioned)
- id_artists (Ids of mentioned artists)
- danceability (Ranges from 0 to 1)
- energy (Ranges from 0 to 1)
- key (All keys on octave encoded as values ranging from 0 to 11, starting on C as 0, C# as 1 and so on...)
- loudness (Float typically ranging from -60 to 0)
- speechiness (Ranges from 0 to 1)
- acousticness (Ranges from 0 to 1)
- instrumentalness (Ranges from 0 to 1)
- liveness (Ranges from 0 to 1)
- valence (Ranges from 0 to 1)
- tempo (Float typically ranging from 50 to 150)
- release_year (Year of release)
- release_month (Month of year released)
- release_day (Day of month released)

artists.csv

- id (Id of artist)
- followers (Total number of followers of artist)
- genres (Genres associated with this artist)
- name (Name of artist)
- popularity (Popularity of given artist based on all his/her tracks)

These data are extremely detailed and rich. The `train` set has 21,000 rows, while the `artists` set has 17,718 rows. Note the ease in which we can join the two together as well, specifically using `id_artists = id` to join any of `test/train` with `artists`. No doubt, we will be able to address our research question on how and why a given song is popular/unpopular on Spotify. We may also be able to explore the similarity of songs within artists and across artists based on a number of the features present in their songs (energy, key, loudness, etc.).

Part 3: Collaboration Plan

- How will you divide responsibilities?

We will all take part in cleaning, wrangling, visualizing, and modeling these data. This will be a collaborative effort. Specifically, each person will explore on their own before we come together as a group to share our findings and consolidate.

As an example, we plan to have two people explore the numerical features of these data (danceability, loudness, etc.), while the other two explore the categorical and time series features (genres, release year/month).

- About how much time do you expect every group member to spend on the project each week, on average?

Each member will spend around 2 hours per week, though this may vary by week.

- When and how will you meet?

On zoom after classes on Wednesdays and Friday.

- What platform(s) will you use to communicate between meetings?

We have a text message groupchat.

- Where will you store data, code, writing, etc., so that all group members have easy access to shared materials?*

We have a shared GitHub repository here.