**Project Topic Idea: Popularity of a Song**

Of the project ideas you submitted in the previous deliverable, select the one you want to work on throughout this project. In making your decision, please refer to the feedback you have on the previous deliverable. If all ideas were found plausible, feel free to select the one that you liked the best as a team. You will be working on this project idea during the remainder of the semester.

Complete this section based on the previous deliverable. After deciding on your topic idea, simply copy and paste the same information from the Topic Proposals document here.

### **Problem Statement**

### **Describe the problem you would like to tackle.**

### This project will help users find out how popular a new song would be if they released it today.

### **What is the topic of your project?**

### A user can enter a song, and we would then run a sentiment analysis on its features. Spotify’s API can provide information on a song’s tempo, valence, speechiness, liveness, instrumentalness, energy, danceability, acousticness and mood tags. Using that information, we would give an estimated popularity level for that song.

### **What do you want to learn about it**

### What factors are more important in determining how popular a song is? What types of songs are popular? Why are certain songs more popular than others?

### **Significance of the Problem**

### **Why is it important to tackle this problem in your project?**

Knowing what music to release can be difficult for artists and producers, so this project can help users explore how popular a given song would be today. Artists would then know what kind of music is the most popular.

### **In what ways could the insights from this project be useful?**

### As time passes, we will be able to update the popularity index for songs and see how popularity changes for songs over time. This could help give insight into what features are most important for predicting a song’s popularity. It would also be interesting to track the rate of popularity growth for a song, which could be used to predict which songs are likely to become popular.

### **Potential Datasets**

### Spotify API: <https://developer.spotify.com/documentation/web-api/reference/tracks/get-audio-features/>

**Dataset File**

Download or scrape your data from the source you identified above. Save your dataset as a CSV file. The first row of the file should contain variable names.

**Your dataset should have at least 1000 rows, corresponding to samples/records, and 10 columns, corresponding to features and target variables. This is the bare minimum. The more, the better!**

Describe your variables below (add more rows if necessary):

Note: All song trait descriptions were retrieved from Spotify: <https://developer.spotify.com/documentation/web-api/reference/tracks/get-audio-features/>

|  |  |  |
| --- | --- | --- |
| **Variable Name in File** | **Description** | **Feature/**  **Outcome** |
| artist | The artist of the song. | N/A |
| track | The title of the song | N/A |
| id | The unique SpotifyID of the song. | N/A |
| danceability | How suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable. The distribution has a slight negative skew | Feature |
| energy | A measure from 0.0 to 1.0 which 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. The distribution has a negative skew | Feature |
| key | The estimated overall key of the track. 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. | Feature |
| 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 typical range between -60 and 0 db. | Feature |
| mode | The modality (major or minor) of a track, the type of scale from which its melodic content is derived. Major is represented by 1 and minor is 0. | Feature |
| 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. | Feature |
| acousticness | A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic. The distribution of values is positively skewed | Feature |
| 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. | Feature |
| 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. | Feature |
| 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). | Feature |
| 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. | Feature |
| duration\_ms | The duration of the track in milliseconds. | Feature |
| popularity | A number from 0-100 corresponding to the popularity index of the given song. A score of 100 is the highest possible level of popularity. | Outcome |
| In the Feature/Outcome column, indicate whether the variable is a feature or outcome variable. You need to have at least one outcome variable, with several feature variables. | | |

**Based on what we discussed regarding machine learning (Week 07 Day 02), does your dataset include a set of feature variables and one outcome variable that you can use for a supervised machine learning task? Please explain. You need to meet this requirement and show us you understand that you are required to use a predictive model in your project.**

Our dataset does include a set of feature variables and one outcome variable. Our feature variables are the 12 traits of the songs that we pulled using the Spotify API. Our outcome is the popularity index, which is also found through the API. These variables can be used for a supervised machine learning task, because we can map our feature variables to our outcome variable and determine the popularity of a given song. The songs can be classified into a given popularity number based on their features. Therefore, if we provided a brand new song to the tool, it would be able to predict how popular the song would be.

**Further info on submitting the dataset:**

Submit a CSV file, or multiple files, containing your data. If the dataset is too large, you can upload it to Github or any other online repository, and provide a public link.

If you have scraped your data, you should also submit a Jupyter Notebook containing your Python code used to scrape the data. Please be reasonable and comment your code out whenever it makes sense to do so.