Machine Learning: Application Project Proposal

Shane Keane, Dylan Kierans, Alexander Sanfilippo
17329836, 21346261, 21343904

October 27, 2021

Song recommendations using Spotify API

Motivation

Streaming platforms are as popular as ever. While the film/TV and podcast streaming sector has seen an increase in competition in recent years, the same can not be said for music platforms. Spotify is the home of music streaming. One of the strongest features of the platform is its "My Liked Songs" playlist, which as the name suggests, is an easy way to store any mix of rag-tag songs you like. Recommended songs would be a great way to build up your own personal playlist, however Spotify has a tendency to recommend you complete and utter trash. So we thought we, the users, would give it a shot instead.

The aim of the project is to train a model to classify songs according to which volunteer (see Dataset) is most likely to enjoy a song. Applications include user recommendations, or categorizing liked songs into sub-playlists. The model will also be able to give an estimated probability that a user will enjoy a specific song.

Dataset

We first require a group of volunteers with a suitably large playlist of liked songs (over 100) on the music platform, Spotify. The aim is to collect data from the playlists of 5-10 different volunteers. We will then use Spotify's public API in order to gather information about these songs.

Spotify stores parameters for each of its songs according to audio features such as energy, danceability, instrumentalness etc. We will have access to 7 of these normalized parameters (on a scale of 0-1), along with 3 additional features tempo, song length (both continuous), and musical key (discrete). These features will be used as our training data, while the target values will be classes signifying which volunteer liked the song.

Method

We intend to use two main models;

1. MLP Neural-Network: Will require a choice of activation & output functions, with cross-validation on number of hidden layers and relevant hyperparameters.

2. k-Nearest Neighbours model: Will require choice of distance function, and cross validation on hyperparameter k.

For comparison we will also provide the results of a simpler classifier such as a linear or logistic model. As a baseline we will use a random predictor, and a simple model taking the parameters for one song, and predicting the 'nearest' volunteer class (according to some distance function).

Intended Experimentation

We plan to split our data into training/testing data and evaluation data (80:20). This will ensure we have data for model assessment once the model has been fully tuned.

In order to evaluate the model, we can treat it as a recommendation model. In this case we treat the space of potential recommended songs is massive. So User_A is only interested in the cases where the model recommends them a song (i.e. the model predicts A, True Positives or False Positives), and less so in songs which are recommended to someone else (i.e. True Negatives or False Negatives). Hence the precision of the model will be an important evaluation criterion.