1. Problem Statement

Inspired by music providing services and their algorithms for music recommendation, I wanted to design and build my own music genre classification model. To add to this project, I will also attempt to identify the types of instruments used in any audio track.

2. Data Preprocessing

I am using the GTZAN Genre Collection dataset (http://marsyas.info/downloads/datasets.html) consisting of 1000 audio tracks each 30 seconds long. It contains 10 genres, each represented by 100 tracks. The tracks are all 22050Hz Mono 16-bit audio files in .wav format. The dataset consists of 10 genres: Blues, Classical, Country, Disco, Hip-hop, Jazz, Metal, Pop, Reggae, Rock. Each genre contains 100 audio tracks.

Using the Librosa library, we extracted multiple features from each audio track spectrograms:

- Zero crossing rate: Rate of sign-changes along a signal
- Spectral centroid: It indicates where the centre of mass for a sound is located and is calculated as the weighted mean of the frequencies present in the sound.
- Chroma frequencies: Short-time Fourier transform of an audio input and maps each STFT bin to chroma
- Spectral roll-off: The frequency below which a specified percentage of the total spectral energy lies.
- Mel-frequency cepstral coefficients (20): A small set of features) which concisely describe the overall shape of a spectral envelope.

The data is stored in data.csv.

3. Machine Learning Model and Preliminary Results (and 4.)

With the above data, I first used traditional Scikit-Learn supervised learning models to get an estimation of the accuracy we are trying to achieve.

a. Support Vector Machine

Using a linear support vector machine on our dataset, we obtained the following results:
SVM Training Score: 0.53875
SVM Test Score: 0.505

b. K-nearest-neighbors

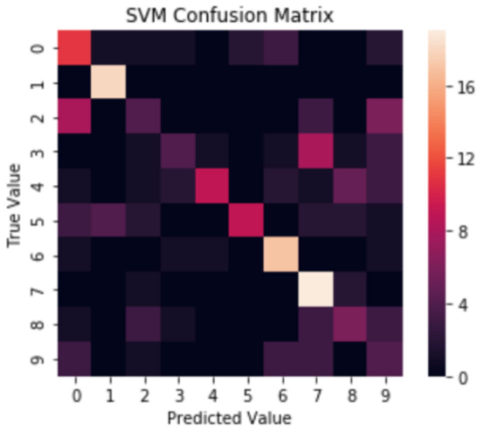
For the K-nearest neighbors classification model, I tried to find the number of neighbors resulting in an appropriate score for both the training set and the test set. Therefore, the K-nearest neighbors model was trained with n_neighbors = 6, which allowed us to obtain the following results:
KNN Training Score: 0.64375
KNN Test Score: 0.62

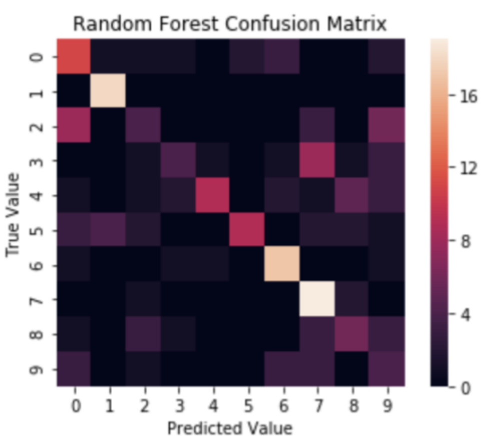
c. Random Forest Classifier

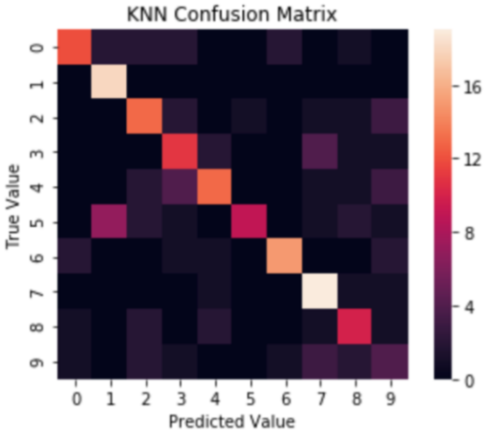
Finally, the random forest classification with the number of estimators initially set at 15. If I decided to pursue with this model or for learning, it would be interesting to determine the optimal number of estimators and other hyperparameters (for later!). Meanwhile, the model resulted in:

Random Forest Training Score: 0.99625
Random Forest Test Score: 0.56

Confusion Matrices:



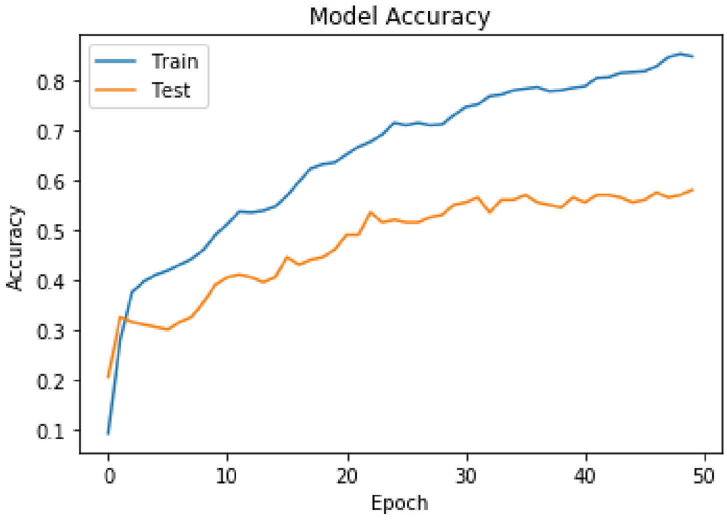


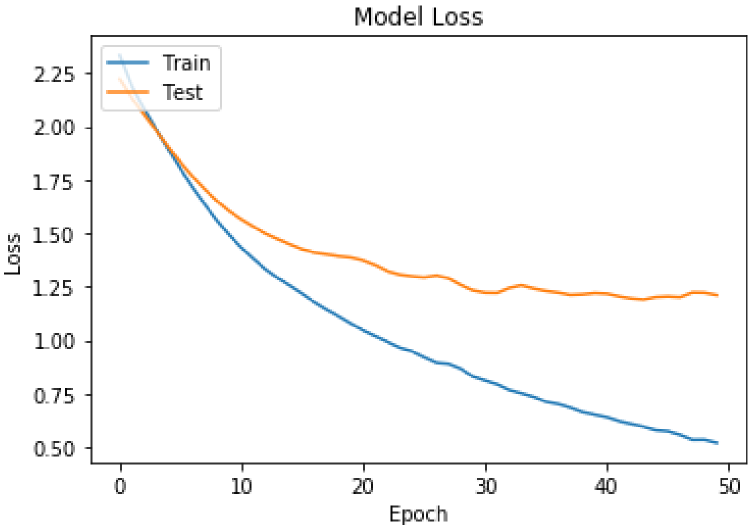


The results obtained from these traditional models are considerably lower than expected, with the best one (KNN Classifier) obtaining only a 62% accuracy on the test set.

From a research report from BAHULEYAN, Hareesh (2018) on Music Genre Classification using Machine Learning Techniques[1], a method utilizing the audio track spectrograms and deep learning techniques resulted in far better results. I decided to look into Convolution Neural Network (CNN) models for music genre classification. A first simple CNN sequential model trained with the data from data.csv is composed of three successive Dense layer (rectified linear unit activation) and one final Dense layer (Soft-max activation).

d. The simple CNN Sequential Model
CNN Test Accuracy: 0.6499999761581421





5. Next steps

Since I was not yet satisfied with the results I obtained, I decided to continue along the way in the research from BAHULEYAN, Hareesh on Music Genre Classification using Machine Learning Techniques[1]. In fact, I extracted the spectrograms from each audio tracks, which can be treated as images and used by the Convolutional Neural Network to predict the genre label. I am currently experimenting with a model consisting of 5 convolutional blocks (conv base) with relu activation, followed by a set of densely connected layers, which outputs the probability that a given image belongs to each of the possible classes. Since this model is currently training, I will report on it for the third deliverable. Some of the spectrogram samples can be found in the folder Spectrogram_Samples.

In addition, from Derek A. Huang, Arianna A. Serafini, and Eli J. Pugh's research on Music Genre Classification[2], significant performance increases across all models were obtained from proper data processing, using a 64 mel-bins and a window length of 512 samples with an overlap of 50% between windows as well as a log-scaling using the formula log(X_2). Also, I will sample a contiguous 2-second window at four random locations from each audio track in order to increase our data size from 1000 clips to 8000 clips of two seconds each.

[1] BAHULEYAN, Hareesh. (2018). Music Genre Classification using Machine Learning Techniques. eprint arXiv:1804.01149.

[2] HUANG, Derek A., Serafini, Arianna A., PUGH, Eli J. (2019). Music Genre Classification. eprint.