Music Genre Classification Project Report

Problem statement:

The music collection in the world is constantly evolving and expanding, and it requires time and effort for people to find music they love. Fortunately, the advent of streaming technology makes discovering music much easier. At the same time, when music becomes more accessible to people than ever, it gets messier and poses a challenge of how to organize, browse, and recommend effectively from the huge collection. Music genre classification is an essential step to disentangle the cluster and classify the music to broader groups that share similar characteristics. Practically, the need for accurate meta-data required for music streaming services such as Spotify, Apple Music and SoundCloud climbs, and the accuracy of recommendation results greatly impact their user retention. Even at physical record stores, knowledge of the music genre would help optimize the way the shelves are organized.

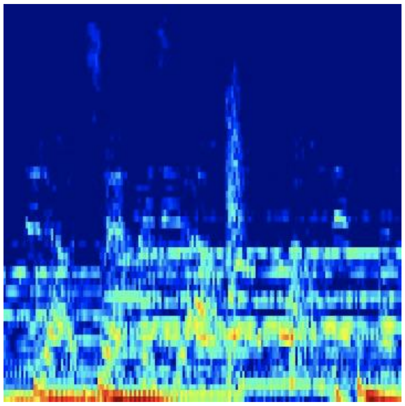
The goal of the project is to use the audio clips of music as input, apply statistical modelling techniques to classify music tracks into eight genres. (Electronic, Experimental, Folk, Hip-Hop, Instrumental, International, Pop and Rock)

Dataset and Data Preprocessing:

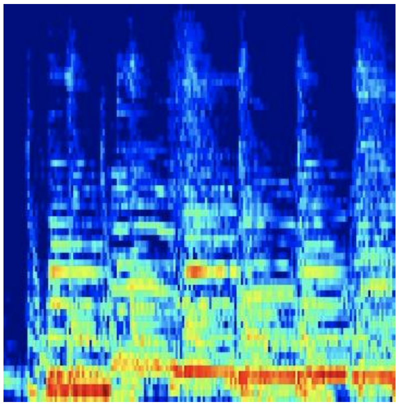
The dataset used in this project is the *Free Music Archive(FMA)* Small dataset. The reason why I chose this dataset is that the sample size is large compared to other widely used music dataset, and the classes are balanced. The dataset (8 GB) includes a combination of raw audio files and metadata, has 8 genres and 1000 songs per genre evenly distributed and the audio files length are 30 seconds each. For the scope of the analysis, only the audio files and genre label from the metadata are used.

The first step to clean the audio files is to loop through all folders and convert .mp3 audio files to the .wav files. After deleting the corrupted files and audio files that are too short for the analysis, 4 random samples of 2-second audio clips are extracted from each file.

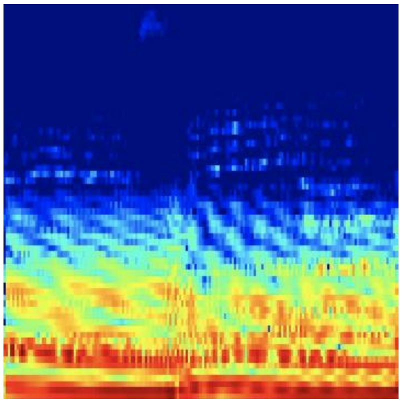
To convert the 2-second audio clips to a format that we could easily analyze, Librosa package is used. Sound is basically a sequence of vibration in varying pressure strength. After separating the audio clips to windows, fourier transformation decomposes the sound into various frequencies, and Mel Scale is applied such that sounds of equal distance from each other on the Mel Scale are also sound of equal distance from each other to humans. The plotted mel spectrograms visualize the difference in soundwaves from different music genres:



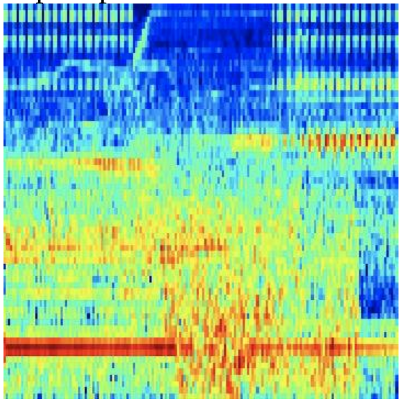
Hip Hop



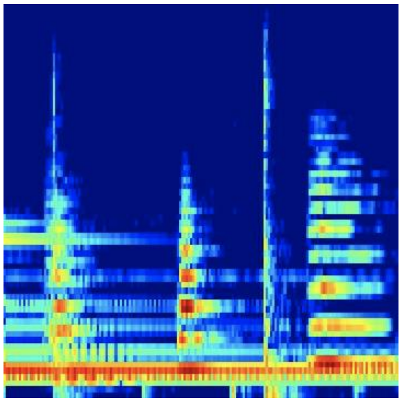
Folk



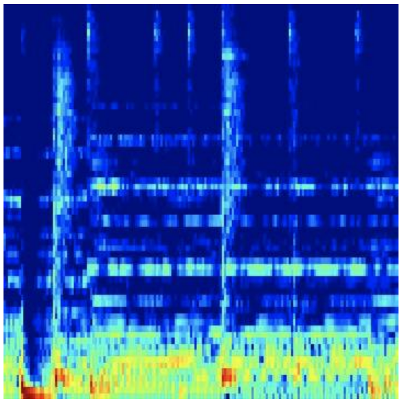
Rock



Experimental



Instrumental



Electronic

From the mel-spectrograms above, we do see different patterns for different genres. However, it is difficult for a human to identify genres based on the mel spectrograms.

For the traditional machine learning models, a 2D array of data for each audio clip after the transformation is used as input; while for the deep learning models, the 3-channel colored graphs for each audio clip is used as input. We didn't apply the data augmentation techniques because the spectrograms represent audio signals, and the information would be distorted if we flip or rotate them.

Traditional Machine Learning Models:

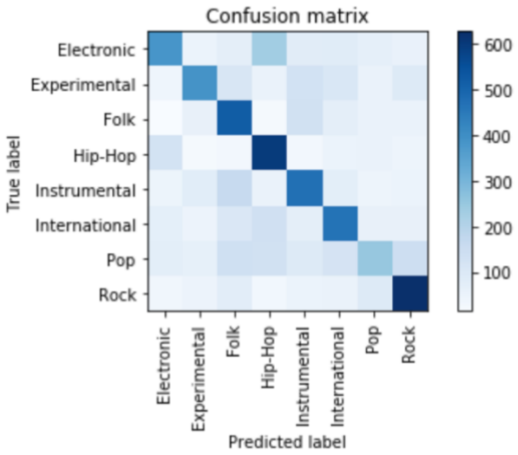
After flattening the 2D arrays representing each audio clip, the number of features greatly outnumber the number of samples. To ensure the performance of the classification models, PCA algorithm is applied to the data to extract the top PCs that explains most of the variations for each sample. The top 15 PCs are picked, which explain 76% of the variance.

The baseline model is a dummy classifier that makes predictions using simple rules. For a 8 class classification problem, the dummy classifier outputs an accuracy of 12%.

Logistic Regression, K-Nearest Neighbours, Support Vector Machine, Random Forest, Gradient Boosting and XGBoost models are constructed for the classification problem. In a multi-class classification problem, accuracy is the most straight-forward metric to measure the effectiveness of a model.A summary of the model results is listed below. Random Forest algorithm achieved the highest accuracy of 47%.

Classification Algorithm	Accuracy
Dummy classifier	12%
K-Nearest Neighbours	44%
Support Vector Machine	41%
Multi-class Logistic Regression	33%
Random Forest	**47%**
Gradient Boosting	38%
XGBoost	41%

Since these models could only capture the overall pattern of the audio data, instead of taking the correlation and variation across time into account, building deep learning models could potentially improve on the prediction accuracy.



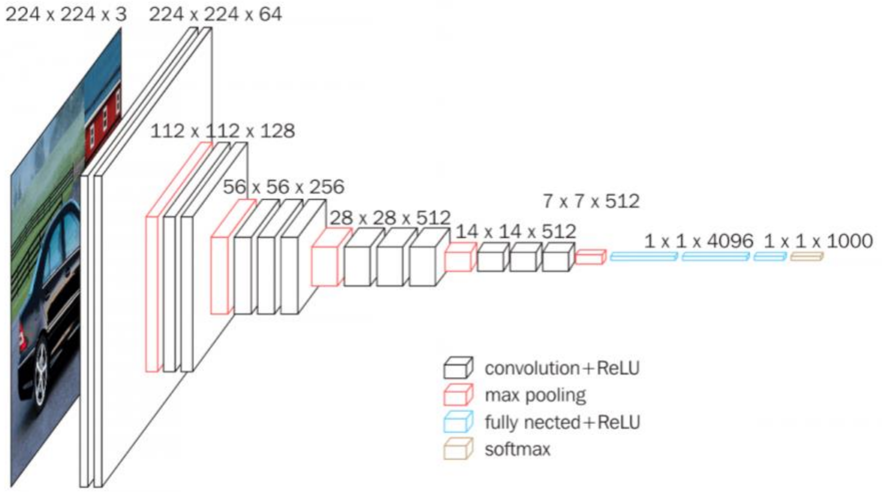
(confusion matrix for Random Forest classifier)

Deep Learning Models:

For the deep learning part, three model architectures are chosen to make predictions:

1. Feed-Forward Neural Network: 3 fully connected layers, with the first two followed by dropout layers and activation layers.

2. VGG16 Convolutional Neural Network: 16 layers, combination of convolutional layers and max-pooling layers, followed by fully connected layers at the end. VGG framework uses fixed kernel size to reduce the number of trainable parameters and reduce overfitting.



3. ResNet34/ResNet50 Convolutional Neural Network: ResNet framework introduces the identity shortcut and projection shortcut to address the vanishing gradient problem.

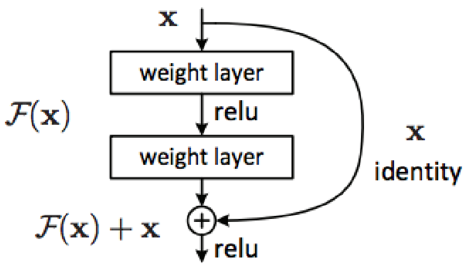

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

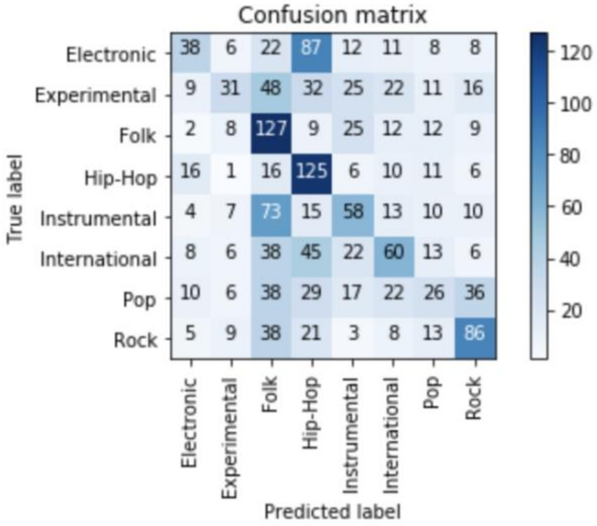
Figure 2. Residual learning: a building block.

VGG16 and ResNet are classic and popular CNN networks which typically generate high accuracy for image classification tasks. The pretrained model parameters are available in Keras package, which are used as the starting points of our training process.

For VGG16 and ResNet based architectures, the first step is to fix the pre-trained parameters and only tuning the fully connected layers at the end using transfer learning. After obtaining the parameters that give the highest accuracy in the validation set, we fix the fully connected layers and fine tune the VGG16 and ResNet convolutional layers.

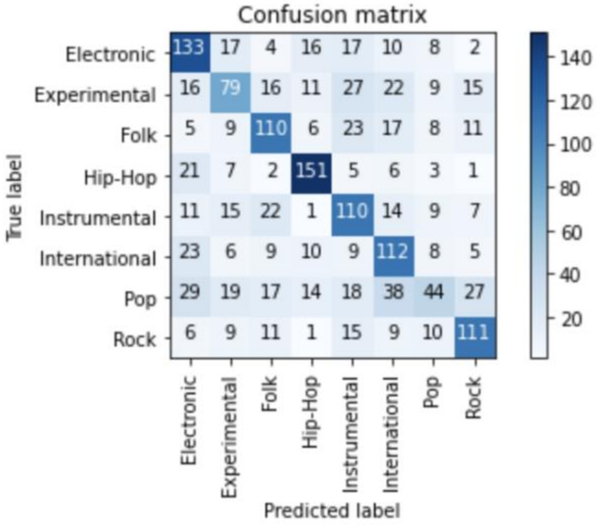
Architecture	Accuracy
Feed-Forward Neural Network	35.4%
VGG16	**55.9%**

Resnet34 and Resnet50	12.5%



To speed up the training process, the deep learning models are deployed in an AWS EC2 p2.xlarge instance, with connection to the AWS S3 storage system, to leverage an GPU in the server to maximize training efficiency. The training time was reduced by more than a half.

The tuned hyperparameters for the VGG16 model are learning rate, batch size, dropout rate, and L2 regularization factor. The VGG16 model achieved a validation accuracy of 55.9%, a great improvement from the baseline model accuracy of 12%, and also from the best performing traditional machine learning models.

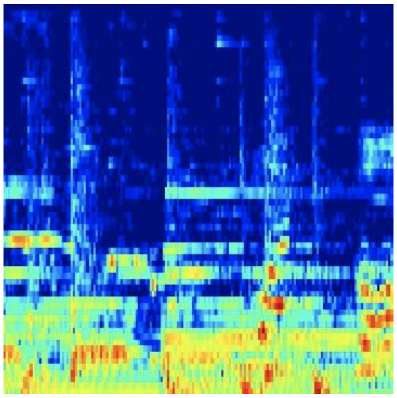


Unfortunately, the ResNet34 or ResNet50 architecture don’t work well for this classification problem. The training accuracy increases for each epoch, but the validation accuracy stays the same as the accuracy of the baseline random classification model. The VGG16 model turns out to be the best performing deep learning model for the problem.

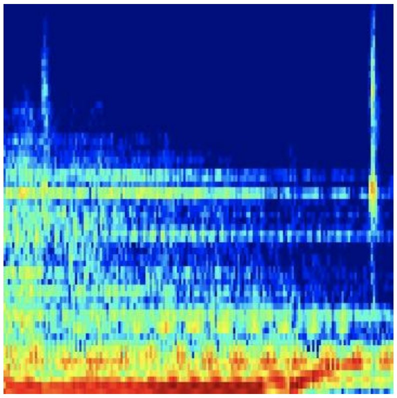
Model Results Interpretation:

The VGG16 network architecture works well for this classification problem. With 8 classes in the problem, 55.9% accuracy is a reasonably good result that we could achieve. A simple fully connected network with 3 layers doesn't work well and is likely caused by the structure not complex enough to capture the features of different genres of music. Even though I have freeze most layers of the network and added dropout layers for the ResNet, the reason why the ResNet architectures lead to overfitting too quickly might be because there are too many degrees of freedom in parameters, which make the trained parameters from the training set unable to generalize to the validation set.

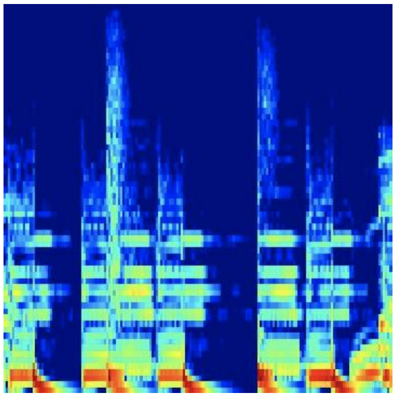
The confusion matrices I got from different algorithms shows comparable classification success rate for each genre individually. Experimental and Pop music are the hardest to successfully classify, and they're equally likely to be classified as each of the other categories. After looking into the spectrograms of Pop music clips, I found that the pop music has very different patterns in different songs, which caused it being easily misclassified as other genres. Practically, the definition of pop music is continuously evolving, and musicians often borrow signature characteristics from other genres when making pop music. Similarly, the nature of experimental music causes it to consist of an eclectic mix of elements from all music genres. For businesses that need automatic music genre classification as part of the production process, it would be helpful to include other features such as lyrics and album cover as input to the classification model in order to achieve more accurate results.



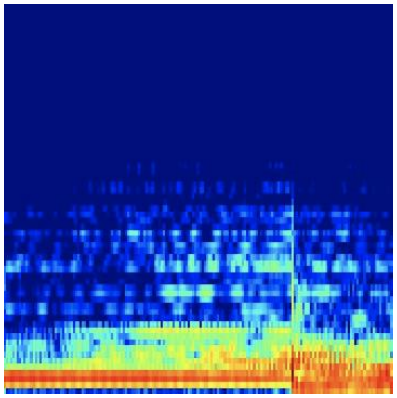
Pop 1

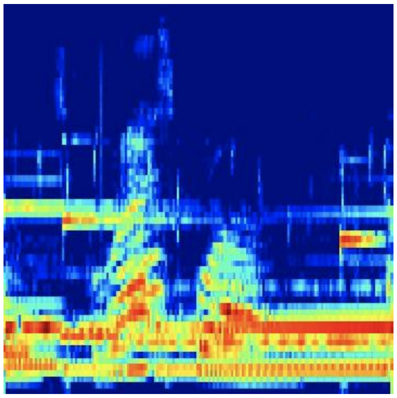


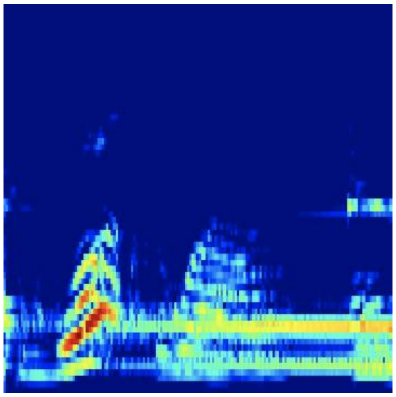
Pop 2



Pop 3







Pop 4 Pop 5 Pop 6

Additionally, the quality of the music audio files used to train the model is crucial to the success of the classification result. For the traditional machine learning part of the project, I trained the same models on another popular GITZAN dataset, but the models achieved much higher accuracy on GITZAN dataset, keeping everything else the same. (I didn’t choose GITZAN dataset for this project because many researchers had worked on that dataset for similar purposes) The difference is probably due to the collection of the music files. It would be beneficial if we get to compare the time span of when the songs are produced, and the popularity of the songs in the two datasets, in order to understand the difference in the prediction accuracy.

Further Work:

In order to enhance the predictive power of the model and allow the model to be more effectively implemented in a music streaming business. There’re several aspects we could work on:

1. Collecting better quality music audio data. The dataset used for this project is a balanced and labeled dataset, with a size large enough to build a deep learning model on. However, the sound in the audio clips is not very clear, and we couldn’t be sure if the data labelled for each genre is a representative sample of the music in the genre. Higher quality data would lead to better accuracy and enhance credibility of the model applied to new dataset.
2. Experimenting with more deep learning frameworks. For the scope of the project, I only experimented with VGG and ResNet based convolutional neural network models. However, more creative model structures and introduction of Recurrent Neural Network could help improve on the model accuracy. Like languages, the sequence of notes/rhythm/beats is a meaningful property of music data. It makes sense to use RNN model for this problem.
3. Using ensemble techniques to combine models. Ensemble building is a typical winning strategy in data science competitions. Combining various models by averaging the predicted probabilities of the eight genres from different models could also potentially lead to improved accuracy.