Audio Classification using Machine Learning techniques

Sandeep Reddy Komatireddy (101950696)

Smita Sanjay Deore (101925234)

Introduction

For a given set of randomly named MP3 files which contain music, our task is to sort them according to their music genre. This project aims to design a learning experiment capable of predicting music genres given their audio.

The dataset used is a sample of the Free Music Archive (FMA). The sample data consist of 2400 training instances and 1200 testing instances, which are equally divided into the following six genres:

Rock:0	Instrumental:3
Pop:1	Electronic:4
Folk:2	Hip-Hop:5

Further this project is divided into two parts:

Part 1: Data Processing and Representation

We have chosen to represent the data in three different ways in this project namely Principal Component Analysis (PCA), Mel-Frequency Cepstral Coefficients (MFCCs), and Mel-Spectrograms.

1. Principal Component Analysis (PCA):

This is a dimension reduction technique used to reduce the dimensionality of a large data set. This is done by transforming large dataset variables into smaller variables that contain the most information. We need to find a vector that summarizes the two very important components namely the direction of the data and the spread of the data. To find such a vector we need to find the direction and dimension that is most important i.e., find a vector with the highest eigenvalue. We can use this vector to rotate the data or change the data axis. This creates new uncorrelated variables that minimize the variance. The covariance matrix defines both the spread (variance) and orientation (covariance) of our data. So to represent the covariance matrix of the vector and its magnitude, we find a vector that points to the direction of the largest spread of data and whose magnitude equals the spread of the direction.

The largest eigenvector of the covariance matrix always points in the direction of the data, and the magnitude of this vector equals the corresponding eigenvalue. The second-largest eigenvector is always orthogonal to the largest eigenvector and points in the direction of the second-largest spread of data.

For population

$$Cov(x, y) = \frac{\sum \left(x_i - \bar{x}\right) * \left(y_i - \bar{y}\right)}{N}$$

For Sample

$$Cov(x, y) = \frac{\sum \left(x_i - \bar{x}\right) * \left(y_i - \bar{y}\right)}{N - 1}$$

The initial Principal Component (PC1) is a linear combination variable that is built to find the direction and magnitude of the highest variance in the dataset. This component has the most information because it has the highest variability compared to other components. To maximize the variance, the initial weight vector $w(1)$ must satisfy

$$\mathbf{w}_{(1)} = \underset{\|\mathbf{w}\|=1}{\arg\max} \left\{ \sum_i \left(t_1\right)^2_{(i)} \right\} = \underset{\|\mathbf{w}\|=1}{\arg\max} \left\{ \sum_i \left(\mathbf{x}_{(i)} \cdot \mathbf{w}\right)^2 \right\}$$

Where w are vectors of weights, $x(i)$ is the row vector of X, and $t(i)$ is the vector of principal component scores. Since $w(1)$ is defined to be a unit vector, it also satisfies

$$\mathbf{w}_{(1)} = \arg\max \left\{ \frac{\mathbf{w}^T \mathbf{X}^\mathbf{T} \mathbf{X} \mathbf{w}}{\mathbf{w}^T \mathbf{w}} \right\}$$

The following steps are followed in the PCA algorithm:

a. We have correlated high-dimension data
b. Center the data points, then get the dimension of the high variance vector
c. Compute the covariance matrix of the data
d. Add eigenvectors and eigenvalues
e. Pick m<d(original dimension) eigenvectors with highest eigenvalues
f. Project the data points to those eigenvectors
g. Get uncorrelated low-dimension data

The rationale behind the selection of PCA:

In general, reducing the number of features is always useful as it results in a simplified model which is what we are looking for. Since our musical dataset is not likely to have any strong correlations between the features, we used a common approach to reduce the number of features and retain as much information as possible i.e.., Principle Component Analysis (PCA) in which we extract the most important features that show the variance of the dataset.

Systematic Analysis of PCA[1]:

Pros:

- Removes correlated features
- Improves algorithmic performance
- Reduces overfitting
- Improves visualization

Cons:

- Independent variables become less interpretable
- Data Standardization is a must before PCA.
- Information Loss
- Expensive in terms of computing for high dimensional data

2. Mel-Frequency Cepstral Coefficients (MFCCs):

Mel-frequency cepstral coefficients are coefficients that collectively make up a Mel-frequency Cepstrum (MFC) which is a representation of the short-term power spectrum of a sound on a non-linear Mel scale frequency based on a linear cosine transform of a log power spectrum. They are derived from a type of cepstral representation of the audio clip. The difference between the cepstrum and the Mel-frequency cepstrum is that in the MFC, the frequency bands are equally spaced on the mel scale, which approximates the human auditory system's response more closely than the linearly spaced frequency bands used in the normal spectrum.

The procedure for creating the MFCC feature is as follows:

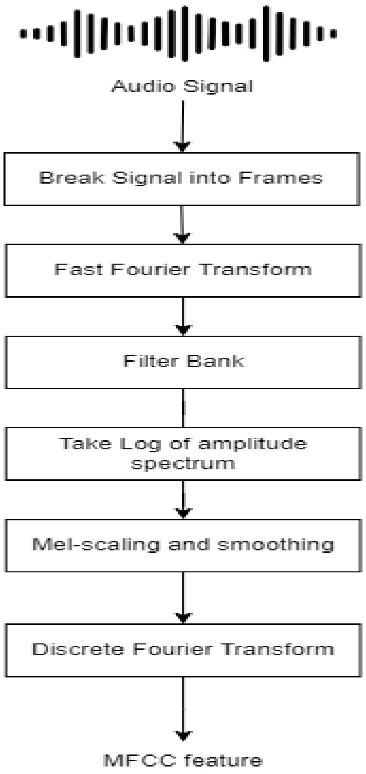

Audio Signal
Break Signal into Frames
Fast Fourier Transform
Filter Bank
Take Log of amplitude spectrum
Mel-scaling and smoothing
Discrete Fourier Transform
MFCC feature

Figure 1: MFCC Feature extraction Process

The Audio signal is broken down into overlapping frames using a windowing function at fixed intervals. Now Discrete Fourier Transform of every frame is taken and maps the powers of the spectrum obtained above onto the mel scale, using triangular overlapping windows. Next, apply the mel filter bank to the power spectra, and sum the energy in each filter. Then take the logs of the powers at each of the mel frequencies. The next step is to smooth the spectrum and Mel scaling. The last step is the discrete cosine transform of the list of mel log powers as if it were a signal. The MFCCs are the amplitudes of the resulting spectrum.

MFCCs can be used in speech recognition as features and for genre classification, and audio similarity measures in music information retrieval.

The rationale behind the selection of MFCC:

MFCC is the widely used technique for extracting the features from the audio signal for the voice recognition system. They extract the featured data which has audible characteristics from the music content, due to which they can identify consistent features even if two musical contents use different digitizing specifications. It is observed that rather than using a raw audio signal as input extracting features from the audio signal and using it as input will produce much better performance.

Systematic Analysis of MFCC[2]:

Pros:

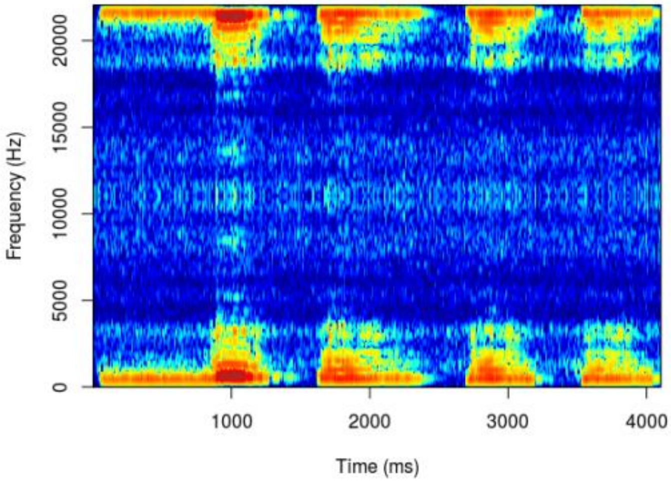
- High recognition accuracy
- Low coefficients correlation
- Good discrimination
- Not based on linear characteristics; hence, similar to the human auditory perception system
- Important phonetic characteristics can be captured

Cons:

- Inaccurate recognition in noisy speech
- High dimensional features vectors
- Low robustness to noise
- In a continuous speech environment, a frame may not contain information of only one phoneme, but two consecutive phonemes
- Not flexible since the same basic wavelets have to be used for all speech signals

3. Mel-Spectrograms:

A spectrogram is a visual description of audio, we can represent time, frequency, and amplitude all in a single graph. A sound spectrogram is also called a Sonogram. A spectrogram can reveal any type of noise in the audio and can allow you to easily isolate those audio problems. Spectrograms can be used to analyze the frequency content of a waveform to distinguish different types of vibration. With the data, users can locate strong signals and determine how frequencies change over time. When data is represented in a 3D plot it's called waterfall display. Spectrograms are 2D graphs, with a 3D represented by colors. The horizontal axis represents the time, and the vertical axis represents the frequency.



Time-domain signals can be used to create spectrograms in two different ways. The Fourier transform is used to calculate the time signal and filter bank obtained from a set of bandpass filters. It can also be created digitally using the FFT. The data digitally sampled in the time domain is divided into slices, and the frequency magnitude of these slices is calculated by the Fourier transform. Each vertical line in the image corresponds to each of the chunks. These spectra are placed side by side to form a 3D image. The process cannot be reversed to generate the original signal from the spectrogram because the spectrogram does not contain any information about the phase of the signal it represents. However, if the starting phase is not important, you can generate an estimate of the original signal. The spectrogram provides a lot of information about the acoustic elements of sound.

The output spectrogram image can be used in combination with a machine learning classifier. You can use spectrogram analysis, application of multiple learning algorithms, and feature extraction to perform basic classifications that provide deeper insight into the music genre. The spectrogram of most audio clips also has many distinctive features. These properties of the spectrogram help efficiently classify music genres.

The rationale behind the selection of Spectrograms:

The spectrogram is used as it captures the important features of the audio. It is often the most acceptable way to input audio data into deep learning models. The mel spectrogram remaps the values in Hz to the mel-scale. Mel spectrograms are better convenient for applications that model human hearing perception. Signals with Lesser frequencies are at the bottom and signals with higher frequencies are at the top.

Systematic Analysis of Mel-Spectrograms[3]:

Pros:

Spectrograms provide us with audiovisual and the pressure created by the sound waves. Therefore, we can easily see the shape and form of the recorded sound. A spectrogram is plotted Frequency (y-axis) vs Time (x-axis) and uses different colors to indicate the frequency. Mel- Spectrograms use the Mel Scale instead of Frequency on the y-axis and it uses Decibel Scale instead of frequency Amplitude to indicate different colors. Hence Mel-spectrograms have an advantage when we need to model human hearing perception like audio classification.

Cons:

Audio can be a medium for conveying meaning which is fundamentally serial and more temporally dependent. Therefore, visual spectrogram representations of sounds fed into image processing networks without temporal awareness might not work optimally.

Part 2: Data Classification

We have implemented the following three different classifiers:

1. **Support Vector Machines (SVM):**

The main goal of the Support Vector Machines algorithm is to find a hyperplane in N dimension space, for N features, that classifies the data points. We need to find a plane such that it has a maximum margin, i.e., the distance between data points of the classes is maximum. Maximizing the margin distance provides some boosting such that the future data points will be classified with more confidence. Hyperplanes are the decision boundaries that e used to classify the data points. Data points separated by either side of the hyperplane can be of different classes. Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane.[2] We can use these support vectors to maximize the margin. SVM uses a method called the kernel trick to transform the training data and then based on the transformations it searches for an optimal boundary between the possible outputs.

Support Vector Machines are a set of supervised learning methods used for regression and classification. SVM is a linear machine whose norm is to minimize:

$$L_p(w, \xi_n) = \frac{1}{2} b\|w\|^2 + C \sum_{n=1}^{N} \xi_n$$

With the constraints:

$$y_n(w^T x_n + b) > 1 - \xi_n, \xi_n \geq 0$$

SVM is used in various applications like text categorization, tone recognition, image classification, micro-array gene expression analysis, and many more. SVM has drawbacks such as they require full labeling of input data, uncalibrated class membership probabilities, and parameters of solved models that are hard to interpret. SVMs have flexibility in choosing a similarity function and they can handle large feature spaces. In SVMs, overfitting can be controlled by the soft margin approach.

The rationale behind using SVM:

Support vector machine is preferred as it produces significant accuracy with less computation power. SVMs are known to use kernel tricks and transform the original input into a higher dimensional space. It can be used for both regressions as well as classification problems. SVM is very effective in high-dimensional spaces. SVM can solve linear as well as nonlinear problems. We can capture more complex relations between the data without performing difficult translations. We have used the one vs rest classification task. The radial basis function (RBF) kernel trick is used to train the SVM as it is the best method to be used for a non-linear problem such as ours.

Systematic Analysis of Support Vector Machines[4]:

Pros:

- SVM is more effective in high-dimensional spaces.
- SVM algorithm is more accurate when the number of dimensions is greater than the number of samples.
- SVM is memory efficient.
- SVM works well when there is a clear margin distance between different classes.

Cons:

- SVM is not suitable for large training datasets.
- It does not perform well when data has more noise or outliers i.e., more overlapping between the classes.
- SVM will underperform when the number of samples is more than the number of features.
- The SVM classification is based on just the separation of data by a hyperplane, there is no probabilistic explanation of this classification.

2. Random Forest:

Random Forest is an ensemble learning method that uses multiple classifiers and takes the majority as result. Many trees are built using random sample data. Based on the trees, data is predicted and then using the majority of the prediction, the result is predicted. In the case of the classification majority voting between different classifiers is considered. In Regression for the average value of different classifiers is considered as the result. The key component to using the Random Forest is that error needs to be un-correlated i.e., the covariance between errors should be zero and there should be no linear relation between them. For introducing Randomness in the data following techniques can be used:

Bagging: It is also called Bootstrapping of the data. Bagging makes sample data with replacement and this data is used by different classifiers. This is also called parallel as samples do not have a dependency on the original dataset. Now each model is trained independently which generates results. Another way to introduce randomness is using random feature selection while building trees.

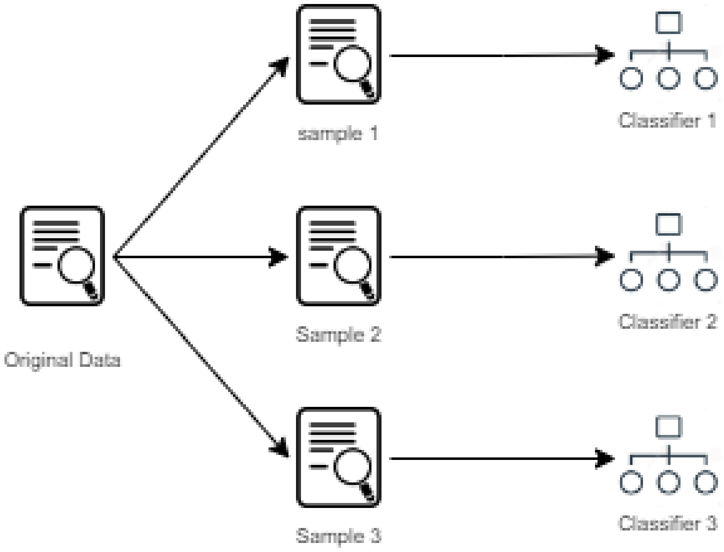


Figure 2: Bagging

Boosting: This is a second way to introduce randomness in the construction of trees. In this method, we select the sample with the same probability. This uses sequential models in which the final model will have high accuracy.

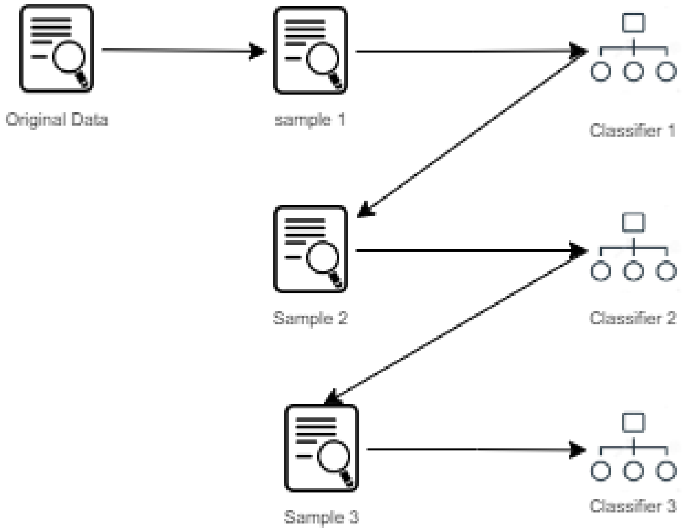


Figure 3: Boosting

Random Forest introduces diversity as each tree created will be different due to the random sample data used for classification.

The rationale behind using Random Forest:

Random Forest classifiers run efficiently on large datasets and can handle many input variables without deleting the variables. Random Forest makes it easy to measure the relative importance of each feature to the prediction, and the default parameters are used to give good results. Random forest is immune to the curse of dimensionality as each tree does not consider all features thus reducing feature space. A random forest can be used for classification as well as regression problems. No overfitting of data as we use majority voting/averaging. Efficient estimation of missing data is possible, and accuracy is high even when there is a large amount of missing data.

Systematic Analysis of Random Forest[5]:

Pros:

- Random Forest can perform both regression and classification.
- It works well with large datasets efficiently.
- Random forest reduces overfitting issues as it uses many classifiers, and each classifier is fed random samples of the original data.
- It handles both categorical as well numerical data easily. It can also handle linear and non-linear relationships.

Cons:

- For large datasets, Random Forests can be computationally intensive.
- Random forest is not easily interpretable. It provides feature importance, but it does not provide full visibility into the coefficients as linear regression.
- Also using many classifiers can make slow down the algorithm and makes it ineffective for real-time predictions.

3. 2-Dimensional Convolutional Neural Network:

Convolutional neural networks are a type of artificial neural network that uses a mathematical operation known as convolution instead of general matrix multiplication in one of their layers. 2D Neural network is specifically designed to process image data and is applied in image recognition and image processing because they learn the pattern that is translation invariant. We can also use a 2D Neural network for audio classification by extracting features that look like images and shaping them specifically to feed them into a CNN. It can detect characteristics like edges, and the distribution of colors in the image which makes it very robust in image classifications and other data that has spatial properties. It is known as a 2D Convolutional neural network because the kernel slides along two dimensions of the data.

The rationale behind using Convolutional Neural Network:

To classify an audio spectrogram, you need a classifier that detects the spectrogram's frequency pattern. We know that convolutional neural networks are best suited for matching patterns between data, so we are using this classifier for this data representation. The 2D convolution can achieve better accuracy, but the has a local pattern of meaninglessness and thereby random luck. 2D Convolutional neural networks can extract special features from the training data using the kernel which is not possible in other neural networks. Convolutional neural networks can detect the edge, distinguish the distribution of the colors, etc.

Systematic Analysis of 2D CNN[6]:

Pros:
CNN can automatically identify important features of the dataset without any human supervision. This is called feature learning. 2D-CNNs can perform large database fine-tuning, and achieve high accuracy and robustness. CNN is more efficient in terms of memory and complexity. Convolutional Neural Networks can be computationally less expensive.

Cons:
In 2D CNN, if we perform a frequency shift of a sound, it will change its spatial extent. Therefore, the spatial invariance that 2D CNNs provide might not perform as well for such data. Audio files can contain different frequencies which are not locally grouped but move together as per their common relationship. This further makes the task of finding local features using 2D convolutions more complex because they might be spaced unevenly even if they move as per the same factors.

Comparing the accuracies of different classifiers:

1. Support Vector Machines (using PCA)

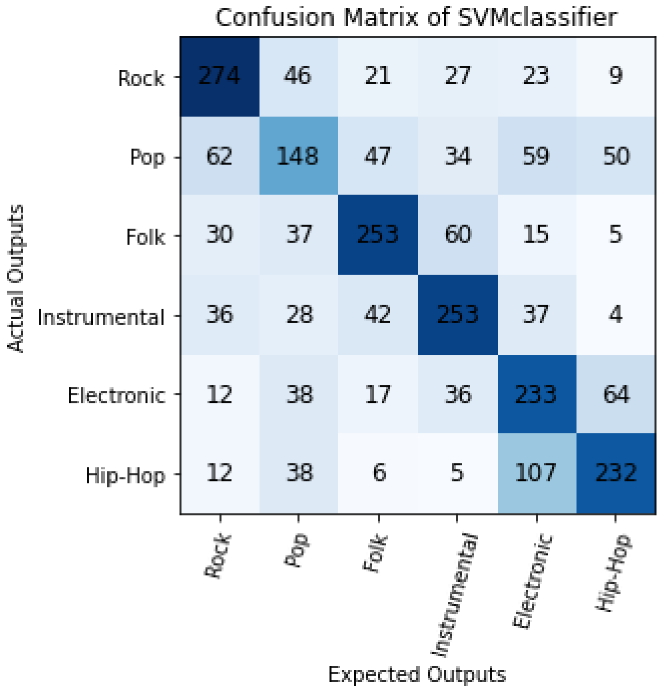
We have first modified our music data into the first data representation which is PCA. Using this data, a classifier is implemented which is a Support Vector Machine to classify the testing data into different classes based on the audio.

Accuracies obtained:

After training the model with the training data, the classifier is tested on testing data and the accuracy obtained on the original testing data is 0.61176.

The accuracy of Support Vector Machines on validation data is 0.580417.

A confusion matrix is calculated and shown below to analyze the efficiency of the classification of the model.



	Rock	Pop	Folk	Instrumental	Electronic	Hip-Hop
Rock	274	46	21	27	23	9
Pop	62	148	47	34	59	50
Folk	30	37	253	60	15	5
Instrumental	36	28	42	253	37	4
Electronic	12	38	17	36	233	64
Hip-Hop	12	38	6	5	107	232

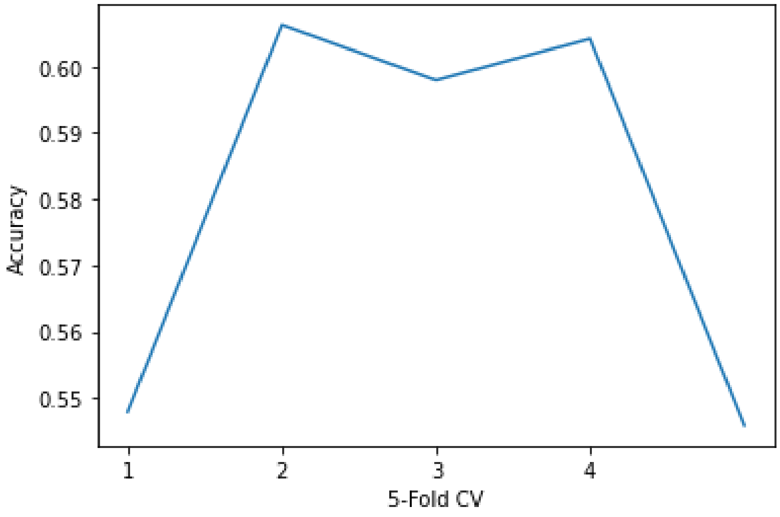
Description of the results:

From the above results, it can be inferred that the model is effective in classifying the data of Rock, Folk, Instrumental, Electronic, and Hip-Hop genres. It is also seen that the classifier struggles in determining the data of the 'Pop' genre.

The explanation for Bias:

It seems that the classifier is biased towards other classes when it encounters the data of the 'Pop' class. The classifier is not able to differentiate the 'Pop' class from other classes effectively.

Here is the graph of the accuracies obtained during 5-fold cross-validation.



Confidence Intervals:

At a 99% confidence level, the accuracy of SVM is likely between 0.55443 and 0.60640.

2. Random Forest (using MFCC)

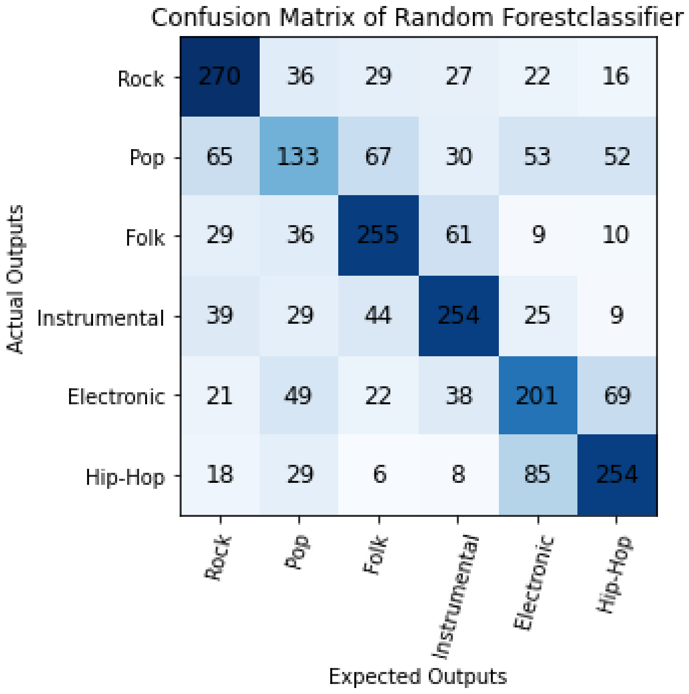
We have modified the data into another data representation which is MFCC. Using this data, a classifier is implemented which is Random Forest to classify the testing data into different classes based on the audio.

Accuracies obtained:

After training the model with the training data, the classifier is tested on testing data and the accuracy obtained on the original testing data is 0.58137.

The accuracy of Random Forest on validation data is 0.56958.

A confusion matrix is calculated and shown below to analyze the efficiency of the classification of the model.


Confusion Matrix of Random Forestclassifier

	Rock	Pop	Folk	Instrumental	Electronic	Hip-Hop
Rock	270	36	29	27	22	16
Pop	65	133	67	30	53	52
Folk	29	36	255	61	9	10
Instrumental	39	29	44	254	25	9
Electronic	21	49	22	38	201	69
Hip-Hop	18	29	6	8	85	254

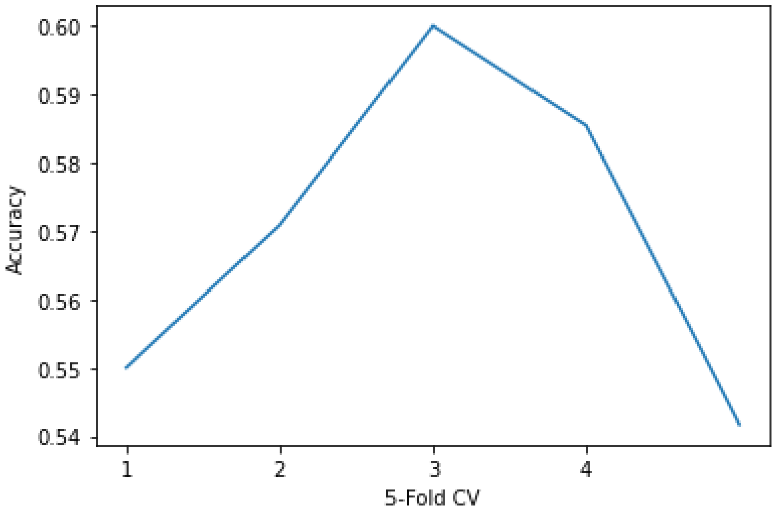
Description of the results:

From the above results, it can be inferred that the model is effective in classifying the data of Rock, Folk, Instrumental, and Hip-Hop genres. It is also seen that the classifier struggles in determining the data of Pop and Electronic genres.

The explanation for Bias:

It seems that the classifier is biased towards other classes when it encounters the data of Pop and Electronic music data. The classifier is not able to differentiate the Pop and Electronic classes from other classes effectively.

Here is the graph of the accuracies obtained during 5-fold cross-validation.



Confidence Intervals:

At a 99% confidence level, the accuracy of Random Forest is likely between 0.54350 and 0.5 9566.

3. 2-Dimensional Convolutional Neural Network (using Spectrograms)

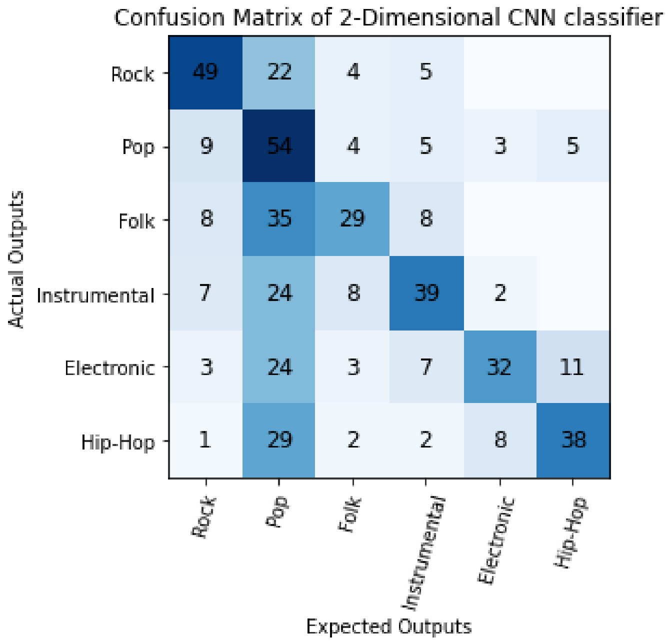
The original data of audio files is modified into Spectrograms which can be used to design a model of a Neural Network. The classifier which is implemented using these spectrograms is a 2-Dimensional convolutional neural network.

Accuracies obtained:

After training the model with the training data, the classifier is tested on testing data and the accuracy obtained on the original testing data is 0.49901.

The accuracy of the 2D Convolutional Neural Network on validation data is 0.50208.

A confusion matrix is calculated and shown below to analyze the efficiency of the classification of the model.



Description of the results:

For validation purposes, we have considered only 480 samples of the data. Out of these, it can be inferred from the results that the classifier is accurate in determining the samples from Rock and Pop genres. All the remaining genres are less accurately determined by the classifier due to the overfitting of the data.

The explanation for Bias:

It seems that the classifier is biased toward Rock and Pop genres. All the other genres are not as accurately classified as these two and one of the main reasons for this bias is the overfitting of the data.

Confidence Intervals:

At a 99% confidence level, the accuracy of 2D CNN is likely between 0.44320 and 0.56096.

Further improvements to this classification task:

The training data given for the project has only 2400 instances for 6 different genres of music. We believe that the accuracy of the classifiers can be improved if we have more training data. There may be some extra features like artist, year, etc. which can result in better accuracy for the classifiers. Building CNN-RNN models also might give a better performance as RNNs are good at understanding sequential data by making the hidden state at t-1 time dependent on the hidden state at time t-2 and they can do a good job in recognizing the long term and short-term temporal features in the song. We believe that more systematic hyper-parameter tuning and cross-validation could potentially improve the overall performance of our models.

References:

[1] https://www.i2tutorials.com/what-are-the-pros-and-cons-of-the-pca/

[2] Automatic Speech Recognition Features Extraction Techniques: A Multi-criteria Comparison. International Journal of Advanced Computer Science and Applications, Vol. 12, No. 8, 2021

[3]https://towardsdatascience.com/whats-wrong-with-spectrograms-and-cnns-for-audio-processing-311377d7ccd

[4]https://dhirajkumarblog.medium.com/top-4-advantages-and-disadvantages-of-support-vector-machine-or-svm-a3c06a2b107

[5] https://medium.datadriveninvestor.com/random-forest-pros-and-cons-c1c42fb64f04

[6] https://www.quora.com/What-are-the-pros-and-cons-of-convolutional-neural-networks