Music Genre Classification

Avanti Mittal Harshika Arya Khushi Bhardwaj Rudri Jani
Sanika Narmitwar Vindhya Jain

IIT Jodhpur
Project Webpage

Abstract

In the domain of music genre classification, we delve into the utilization of the GTZAN dataset sourced from Kaggle, incorporating CSV files, audio files, and spectrograms to empower our model's training process. Our study focuses on harnessing machine learning methodologies to discern and categorize diverse musical genres embedded within the dataset. By leveraging the intrinsic features of audio spectrograms alongside accompanying textual metadata, we aim to enhance the classification accuracy and robustness of our model. Through meticulous experimentation and model refinement, we seek to optimize the classification performance across various musical genres represented in the GTZAN dataset. **Keywords:** music genre classification, machine learning, GTZAN, spectrogram.

Contents

1 Introduction

The goal of this project is to automatically assign a genre label to a piece of music (from the GTZAN Dataset) based on its audio content using traditional machine learning techniques.

1.1 GTZAN Dataset

The GTZAN dataset is a collection of 10 genres of music with **100 audio files** each, all having a length of 30 seconds. There is also a **spectrogram dataset**- a visual representation for each audio file. In addition, there are **two CSV files** - containing features of the audio files. One file has for each song (30 seconds long) a mean and variance computed over multiple features that can be extracted from an audio file. The other file has the same structure, but the songs were split before into 3 seconds audio files (this way increasing 10 times the amount of data we fuel into our classification models). With data, more is always better..

1.2 Roadmap

Seeing as we have three different types of data, we have split the problem into three parts:

- Classification using *CSV files* (30 and 3 second)
- Classification using *spectrogram images*
- Classification using *audio files*

2 Approaches Tried

2.1 CSV

2.1.1 FeatureCSV30

This CSV file had 1000 data samples and no missing values. Thus we normalized and scaled the data. We used dimensionality techniques like PCA and LDA on our dataset and then classifiers like Random Forest, SVM and KNN were used on new transformed data.

2.1.2 FeatureCSV3

This CSV file had 10000 samples as there was a further split of 30 sec audio into 3 sec increasing the dataset 10 times. Firstly, we checked for missing values and handled them. Then we proceeded just as in section 2.1.1, used PCA, LDA for reducing dimensions and then data was classified using classifiers like Random Forest, SVM and KNN were used on new transformed data.

2.2 Spectrogram

A spectrogram is a visual representation of sound frequencies over time, showing intensity variations.The dataset provided contains MEL spectrogram which shows the amplitude of each frequency bin over a function of time.

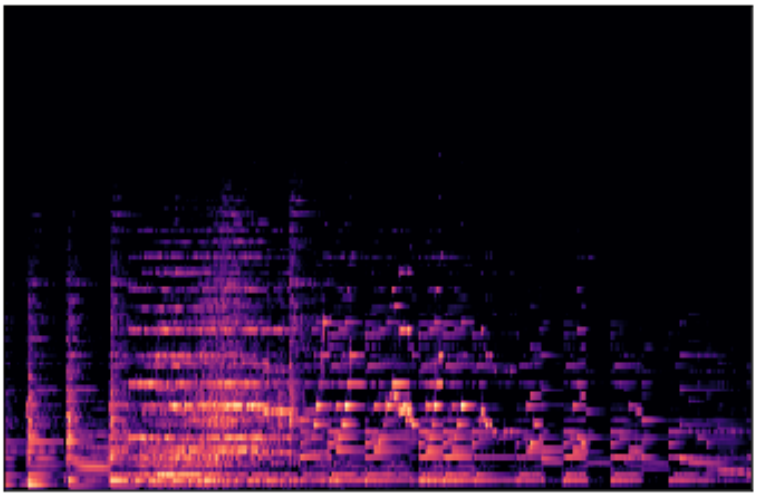


Figure 1: Spectrogram - Jazz003

2.2.1 Simple PCA

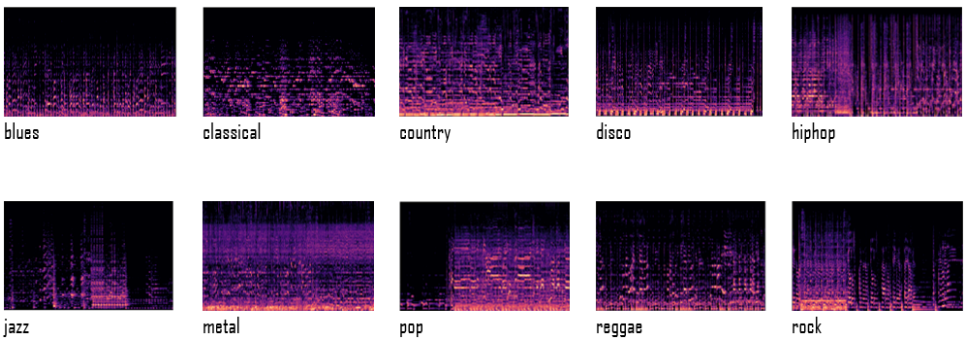
The most basic first approach we used involved directly flattening the images and applying PCA. This method predictably did not give great results as we didn't extract any meaningful features from the specrtogram and just treated them as simple images. 90 percent was only being explained at around 500 (of 700) images.

2.2.2 Most frequent frequency

A spectrogram is basically a frequency vs time graph of the audio. In this method, we took the row-wise mean i.e. the mean of intensity of a particular frequency across time. The highest value would denote the most frequent occuring frequency.

2.2.3 Count of Coloured Pixels

Each genre has a distinctive coloured region/area as can be viewed below.



The idea in this method was to record how many pixels are coloured and using that data, using knn to classify the images. This rough attempt at classifying the data gave about 41 percent accuracy (k=8 neighbours).

2.2.4 Using Time Domain and Frequency Domain Features

This time we focused on extracting meaningful features from the spectrogram

- Frequency Domain: Describes the distribution of frequencies. Includes 6 attributes - Chroma STFT, Spectral Centroid, Spectral Bandwidth, Spectral Roll-off, Harmonic and Percussive components, MFCCs
- Time Domain: Captures variation over time. Includes 2 features - Root mean square and Zero crossing rate

After extracting the mean and variance of these features, classifiers like SVM, Random Forest, KNN, Decision Tree, ANN were used to label the data.

2.3 Audio Files

2.3.1 Using Mel-Frequency Cepstral Coefficients (MFCC)

The first step was to extract features and we started with extracting MFCCs only. MFCCs capture important spectral characteristics of the audio signals, which are relevant for distinguishing between different genres. This was followed by labelling of data using classifiers like SVM, Random Forest and ANN.

2.3.2 Using Other Frequency Domain Features

This time we included other features while performing Feature extraction. The 7 features that were extracted and used to make a single Feature Vector are given below:

- Tempo: It provides crucial rhythmic information, aiding in distinguishing genres based on characteristic speed variations inherent in different musical styles.
- Spectral Centroid: This parameter characterizes the spectrum of the signal. It indicates the location of the centre of gravity‘ of the magnitude spectrum.
- Spectral Rolloff: It represents the frequency at which high frequencies decline to 0.
- Spectral Bandwidth: It measures the width of the frequency range occupied by a signal, thus assisting in identifying genre-specific spectral characteristics
- Spectral Contrast: It highlights differences in spectral energy distribution across frequency bands.
- Chroma STFT : It represents the distribution of pitch classes (notes) in a short-time frame of the audio signal.
- MFCCs : We started with using 20 coefficients. Now , we are taking into consideration 40 coefficients as the part of the final feature vector.

The second step we performed is the classification. Once the feature vectors are obtained, we trained different classifiers on the training set of feature vectors. Following are the different classifiers that were used: K Nearest Neighbors, Random Forest, Linear Kernel SVM, RBF Kernel SVM, Gradient Boosting and ANN.

3 Experiments and Results

3.1 CSV

For CSV30, a dataset containing 60 features and 1000 samples, overfitting might be a concern, Random Forest is known for its robustness to overfitting due to the averaging effect of multiple decision trees. KNN achieves slightly lower accuracy compared to SVM and Random Forest, suggesting that it may not capture the underlying structure of the data as effectively. SVM achieves the highest accuracy with both PCA and LDA.

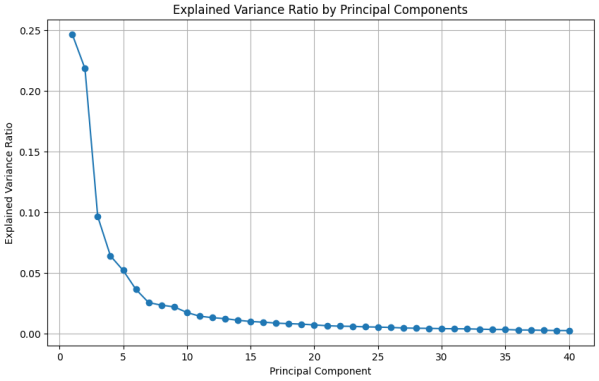


Figure 2: Explained variance ratio by Principal Components graph for CSV30

We can observe the results for CSV30 in the table below :

Classifier	**Accuracy (PCA)**	**Accuracy (LDA)**
SVM	0.7	0.7
Random Forest	0.705	0.675
KNN	0.67	0.7

Table 1: Comparison of Classifiers using PCA and LDA

For CSV3, we observe that KNN did a better job, this could be as the dataset grows, KNN gets better because it has more nearby points to compare with, which helps it understand the data better. This, along with its ability to pick out the most relevant features and deal with noisy data, gives KNN an advantage over SVM and Random Forest, making its classification more accurate.

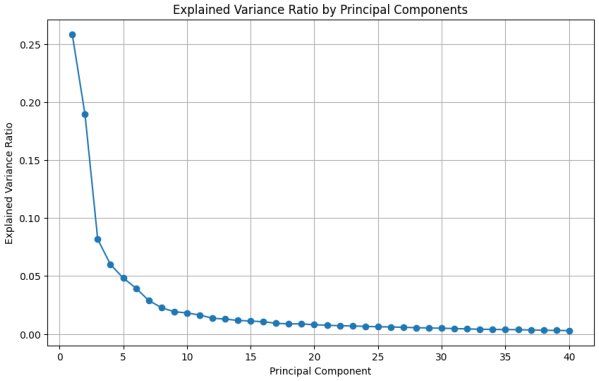


Figure 3: Explained variance ratio by Principal Components graph for CSV30

We can observe the results for CSV3 in the table below:

Classifier	**Accuracy (PCA)**	**Accuracy (LDA)**
SVM	0.9008	0.8168
Random Forest	0.8843	0.8254
KNN	0.9153	0.8280

Table 2: Comparison of Classifiers using PCA and LDA

3.2 Spectrogram

After trying various different approaches, we obtained the best results by extracting frequency and time domain features to classify the data as explained in section 2.2.4. We were able to extarct 96 features. PCA was used to extract 40 that explained the best variance.

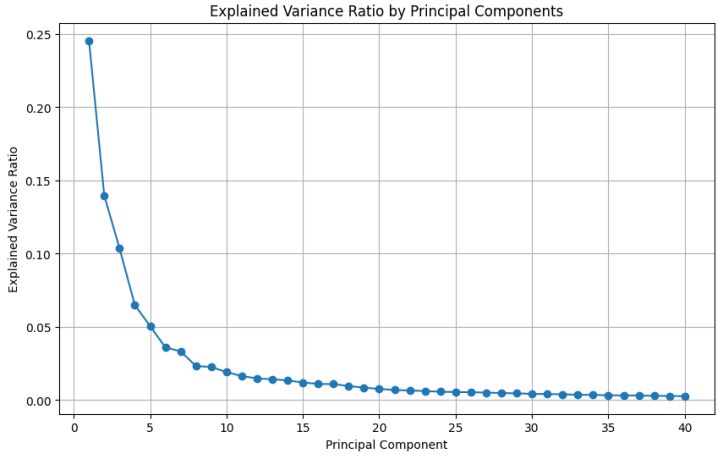


Figure 4: Explained variance ratio by Principal Components graph for Spectrogram

On classifying these extracted features with traditional machine learning models, the following results are observed. (In decreasing order of accuracy).

Classifiers	TestAccuracy
ANN	0.705
SVM	0.67
Random Forest (n estimators = 110)	0.635
KNN (k = 4)	0.595
Decision Tree	0.42

Table 3: Accuracy in decreasing order for various classifiers

The best results were for the ANN classifier (epochs = 1700). Below is the confusion matrix and classification report to visualise the performance of the classifier.

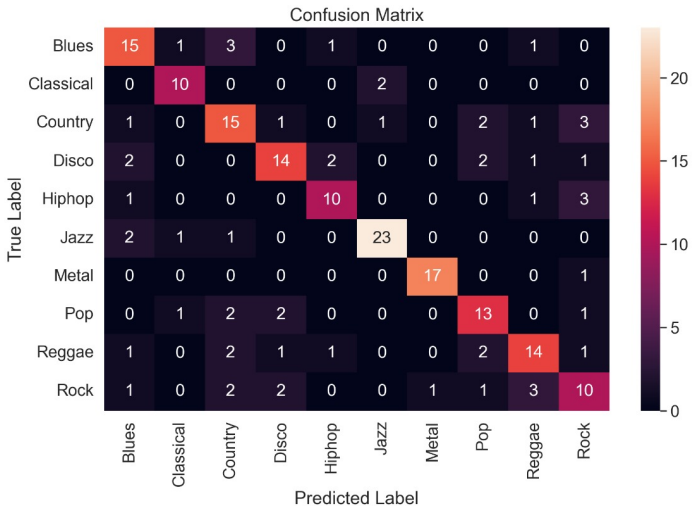


Figure 5: Confusion Matrix for ANN classifier

Genre	Precision	F1-Score
blues	0.65	0.68
classical	0.77	0.80
country	0.60	0.61
disco	0.70	0.67
hiphop	0.71	0.69
jazz	0.88	0.87
metal	0.94	0.94
pop	0.65	0.67
reggae	0.67	0.65
rock	0.50	0.50

Table 4: Classification Report of ANN Classifier

3.3 Audio Files

We measure the performance using various metrics like accuracy. Accuracy is defined as the ratio of the number of correctly classified results to the total number of classified results. We have provided the Accuracy of the various classifiers we used in the table below:

Classifiers	TestAccuracy
ANN	0.71
RBF SVM	0.685
Random Forest (n estimators = 100)	0.67
Linear SVM	0.64
Gradient Boosting	0.625
KNN (k = 5)	0.575

It can be observed from the table that ANNs were better classifiers with accuracy of 0.71 following the approach explained in section 2.3.2. The results obtained from ANNs are given below:

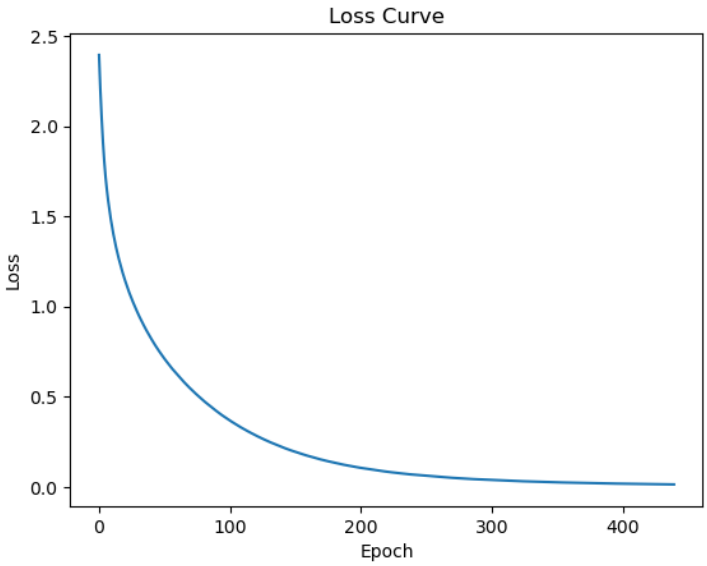


Figure 6: Change in the loss function over training epochs

Genre	Precision	F1-Score
blues	0.89	0.85
classical	0.73	0.81
country	0.62	0.62
disco	0.63	0.59
hiphop	0.71	0.75
jazz	0.88	0.82
metal	0.81	0.87
pop	0.81	0.85
reggae	0.55	0.52
rock	0.45	0.45

Table 5: Classification Report of ANNs

4 Summary

Through the course of this project, we have succeeded in classifying the CSV, spectrogram, and audio files provided by the GTZAN dataset. By navigating through these different file types, we've gained insights into various classification techniques and their application across different data.

We presented an automated system for music genre classification. MFCC features, Chroma features, spectral centroid, spectral roll-off, ZCR are used as the feature vectors and trained the system using traditional classifiers like KNN, SVM, random forest, decision tree and also ANN.

To further better our solution, we can explore deep learning techniques and end-to-end models. Also, we can combine audio features with visual or textual data to leverage additional information sources.

A GitHub Repository

All our code has been pushed in this repository:
https://github.com/sanikanarmitwar/MusicGenreClassification

B Contribution of each member

1. Avanti Mittal: Classification for CSV, Audio dataset
2. Harshika Arya: Classification for Audio dataset, report
3. Khushi Bhardwaj: Classification for CSV, Audio dataset
4. Rudri Jani: Classification for Spectrogram dataset, webpage
5. Sanika Narmitwar: Classification for Spectrogram dataset, ppt
6. Vindhya Jain: Classification for Spectrogram dataset, report

References

[1] Music genre classification using mfcc, k-nn and svm classifier. https://ijcert.org/ems/ijcert_papers/V4I206.pdf.

[2] Abdulrahman H. Altalhi and Ahmad M. Almutairi. Music genre classification using machine learning: A comparison study. *ResearchGate*, 2021. URL https://www.researchgate.net/publication/362619781_Music_Genre_Classification_using_Machine_Learning_A_Compa.

[3] Nikhil Nair. Music genre classification using mel spectrograms. https://medium.com/@niknair31898/music-genre-classification-using-mel-spectrograms-88ce05400d3c, July 2021. Medium.

[4] Andrada Olteanu. Gtzan dataset for music genre classification. https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification, February 2022.

[5] Justin Salamon and Juan Pablo Bello. Deep convolutional neural networks and data augmentation for environmental sound classification. *arXiv preprint arXiv:1804.01149*, 2018. URL https://arxiv.org/pdf/1804.01149.pdf.