Music Genre Classification

Applied Machine Learning Final Report

Justin Choi
jc3547@cornell.edu

Andrew Park
abp73@cornell.edu

Vivian Tay
yt635@cornell.edu

Abstract

This study presents a detailed Exploratory Data Analysis (EDA) and Feature Extraction on a dataset encompassing a variety of music genres. The primary focus is on the Short-Time Fourier Transform (STFT) analysis of audio files to identify distinct patterns and outliers within genres such as blues, classical, country, rock, among others. Following this, the project proposes a method for creating music recommendations, which are specifically tailored based on the audio input from users. This approach emphasizes the diverse nature of musical genres and the potential of applying advanced data analysis in personalized music selection.

1. Introduction

We comprehensively analyzed music genres using several machine learning techniques for classification, specifically, Random Forest, K-nearest Neighbors, Support Vector Classification, and XGBoost. We decided to work with the GTZAN dataset from Kaggle, which covers 10 different music genres, each with 100 tracks, totaling 1000 songs.

From the start, we selected our sample music (blues.00000.wav) and observed the characteristics of this audio source by printing the waveform (Figure 1). With the graph, we observed the amplitude of the audio signal over time. However, we concluded that the plot only provides insights into the dynamics and loudness of the recording, not as informative for genre classification.

After recognizing the necessity of applying more advanced audio data analysis tool, we started to implement librosa library. With the library, we were able to extract some of the features each music file contains such as STFT, Spectrogram (decibels of the music), Chromagram, and BPM Boxplot.

a. Short-Time Fourier Transform (STFT) (Figure 2): The STFT graph showed the frequency content of the audio signal as it changed over time as seen in the rainbow colored plots. It enabled us to analyze the rhythm, pitch, and harmony, which are vital in classifying the genre. Most of all, the fact that the graph showed the most dominant frequencies during the time-frame was a strong characteristic we can utilize for recognizing each genre.

b. Spectrogram (Decibels of the music) (Figure 3): The graph showed us the spectral density of the signal varies with time for the sample music, blues.00000.wav. This provided information about the texture and timbre of the audio, as well as how various frequencies are distributed over time, which can be distinctive for certain genres. However, intuitively, the graph wasn't easy to understand and hard to find the potential for further uses for classification due to the hardness of catching specific trait for the sample music.

c. Chromagram (Figure 4): This graph visualized the intensity of different pitches throughout the time of the audio sample. It was particularly useful for identifying harmonic and melodic characteristics of the music, which can be genre-specific. Nevertheless, again, this graph was also challenging for getting any intuitive insights.

d. BPM Boxplot (Figure 5): This graph represented the distribution of Beats Per Minute (BPM) across different genres. BPM is a crucial feature in music analysis since it reflects the tempo of the music, which can be strongly indicative of its genre. For this graph we could see the ranges of BPM of each genre and also catch some outliers as well. We can see there are some more outliers in hip-hop and pop music, which might be due to wide range of song variation in each genre.

After this comprehensive understanding and initial analysis on the audio data, we moved onto preprocessing the audio files to standardize the data. After adequate preprocessing steps including standardizing variables and PCA analy-

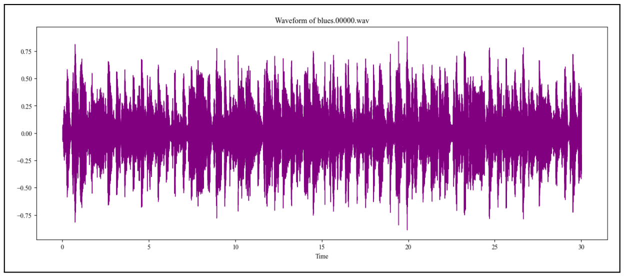

Figure 1. Waveform of Sample Music (blues.00000.wav)

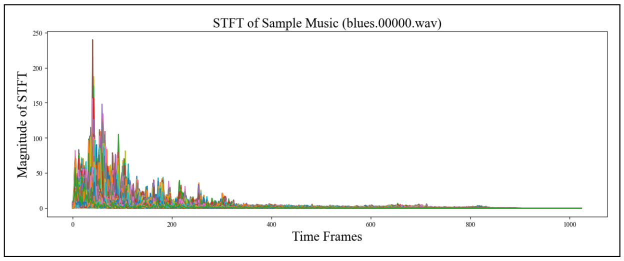

Figure 2. STFT of Sample Music (blues.00000.wav)

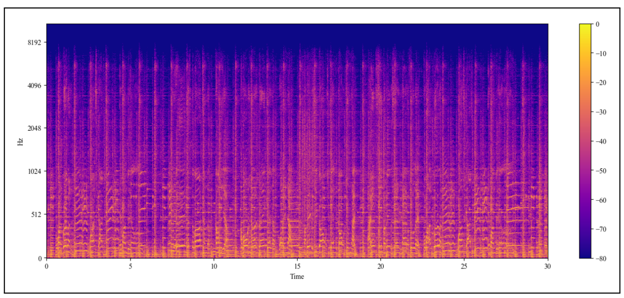

Figure 3. Decibels of Sample Music (blues.00000.wav)

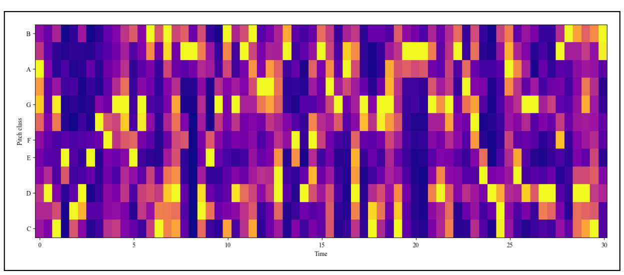

Figure 4. Chromagram of Sample Music (blues.00000.wav)

sis, we have implemented four different machine learning models to classify the files using tempo and other musical metrics. By taking a step further we aim to develop a robust system capable of making accurate music recommendations. For both steps, we would need to leverage the unique features extracted from the audio files to enhance the model's ability to discern distinct patterns across genres.

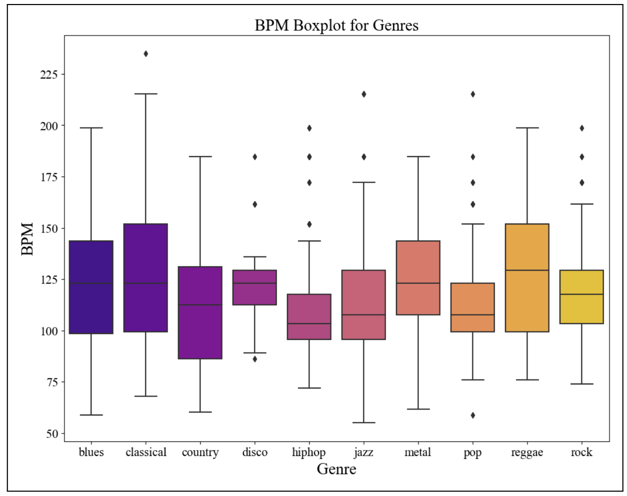

Figure 5. BPM Boxplot for Genres

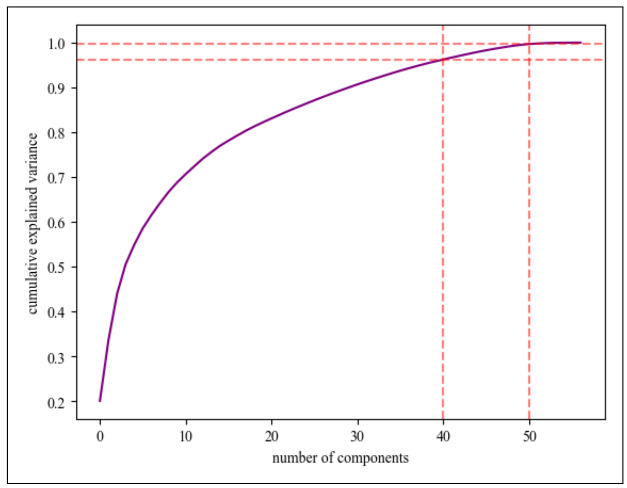

Figure 6. Cumulative Variance Explained vs. Number of PC

2. Related Work

Previous industrial endeavors in music genre classification, particularly those utilizing Principal Component Analysis (PCA), influenced our approach. These works often extend beyond classification, incorporating models to make music recommendations based on the identified genre patterns. Although we haven't used principal components as variables when training our models, the insights we got from our models and its feature, we could align it with our analysis of principal components.

2.1. PCA Extraction

a. Cumulative Variance Explained by number of principal components (Figure 6): Through this graph we can see how many principal components we need to explain the variance of original data. For example, for full 100%

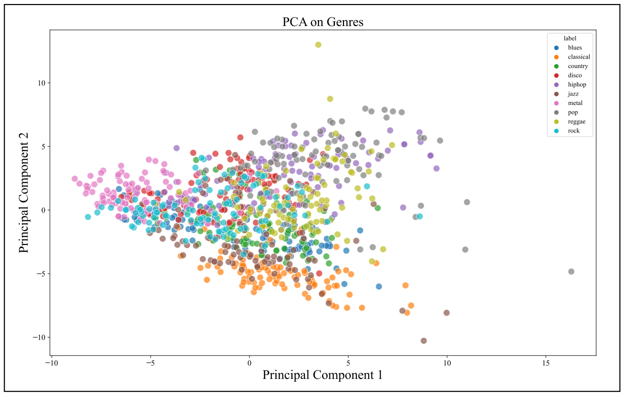

Figure 7. PC1 vs PC2 on Genres

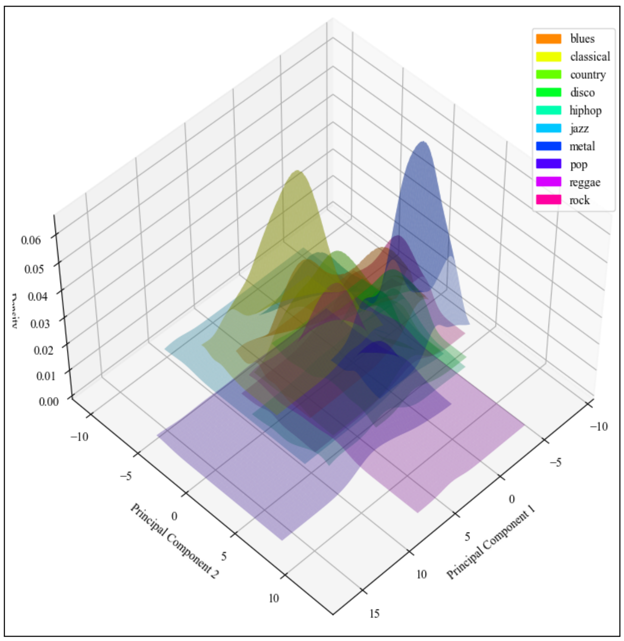

Figure 8. Density of PC1 vs PC2 in 3D

of the variance to be explained in the original data, we will need about 50 initial principal components.

b. Principal component 1 vs principal component 2 (Figure 7 & 8): These two figures demonstrate which major components contribute the most to explaining differences in different genres of music. Many of the genres have significant overlap as seen in the 3D plot, implying that they have common characteristics. This could be because PCA detected similar instruments, beats, rhythms, or other musical elements (principal component 1 range -5 to 5 and principal component 2 range -5 to 5). Some genres, on the other hand, appear to be more distinct or to have subclusters, indicating greater variety within those genres or possibly various substyles (parts of metal, classical, and pop).

In our specific methodology, we tackled music genre classification using a dataset of 1000 audio MP3 files. Our initial steps involved normalizing all audio files to ensure consistent hop length for Short-Time Fourier Transform (STFT) analysis. Subsequently, we categorized the samples and labeled the genres, leveraging the range of tempo values visualized through boxplots.

To identify specific music genres in audio signal processing, we utilized traits such as zero crossings and tempo. These characteristics played a crucial role in our classification approach. Following this, we applied preprocessing techniques to standardize the data, ensuring a uniform foundation for subsequent analyses and model training.

3. Methods

Interestingly, some genres exhibited greater distinctiveness or featured subclusters, suggesting increased variability within those genres or even the presence of different substyles. Notably, this was observed in parts of the metal, classical, and pop genres, indicating nuanced substructures within these broader categories. The granularity provided by PCA allowed us to uncover these intricate relationships, contributing valuable insights to our understanding of the dataset.

3.1. Model Training

We used four different machine learning models: Random Forest Classifier, K-Nearest Neighbours (KNN), Support Vector Classification (SVC), and XGBoost. We highlighted the 3 highest accuracy to assist our analysis for all our models.

a. Random Forest Classifier: Here we set the max depth to be unlimited, used 1000 estimators, and selected a random state of 10 for ensuring proper model comparison.

While the results exhibited variability across iterations, it was evident that the Random Forest classifier excelled, particularly in recognizing classical music. We used hyperparameter tuning to fine-tune our model, with a specific focus on parameters such as the number of estimators and max depth.

This suggests that the chosen hyperparameters contributed to the model's enhanced performance, emphasizing its effectiveness in classifying this specific genre within our dataset.

b. K-Nearest Neighbors (KNN): For KNN, we initialized the model by setting the number of neighbors (n_neighbors) to 10, as we have 10 music genres in our dataset.

The effectiveness of KNN is particularly noted in classifying classical and metal music. Considering the PCA analysis graph above about the distinctiveness of classical and metal music, these results show a pattern that conforms to such analysis.

c. Support Vector Classification (SVC): In the Support Vector Classification approach, we adopted the ”ovo” (one-vs-one) decision function as a multi-class strategy for training models. This approach enables the model to create binary classifiers for each pair of classes, optimizing its ability to distinguish between various genres.

Particularly in the context of recognizing jazz music, where intricate relationships with other genres were observed in the PCA analysis, the ”ovo” strategy contributes to the model’s robustness by addressing the nuanced distinctions between multiple classes. This ensures a comprehensive and accurate classification process, especially in scenarios where genres exhibit complex interrelationships and clustering patterns.

d. XGBoost: For the XGBoost method, we employed the LabelEncoder from the sklearn preprocessing module to encode categorical labels into integer values. This preprocessing step proved essential, given our target labels are categorical, representing various music genres, while our machine learning model exclusively accepts numerical inputs.

The XGBoost model demonstrated exceptional performance, particularly excelling in accurately classifying three specific genres—classical, metal, and pop—identified as having distinct distribution patterns in the PCA analysis. Impressively, the model maintained consistently high performance across a diverse range of genres, attesting to its overall effectiveness.

To further optimize the XGBoost model, we implemented hyperparameter tuning. Specifically, we set the number of estimators to 500 and the learning rate to 0.1. This refinement aimed to enhance the model’s predictive capabilities and overall robustness in handling diverse musical genres.

4. Results

Methods	Hyperparameter Tuning	Score
Random Forest	O	0.8812
KNN	X	0.8600
SVC	X	0.8594
XGBoost	X	0.9019
	O	**0.9087**

Table 1. Comparison of machine learning methods’ accuracy.

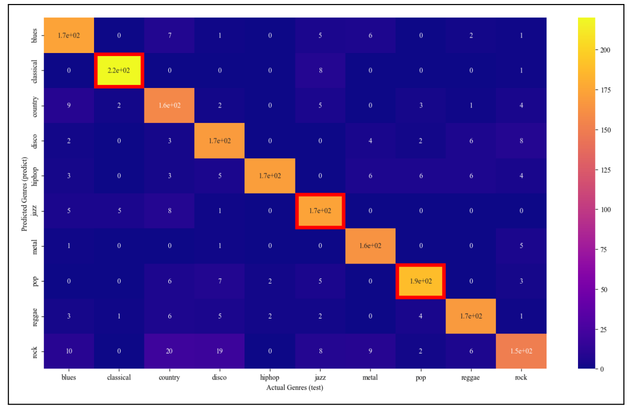

Figure 9. Random Forest Classifier

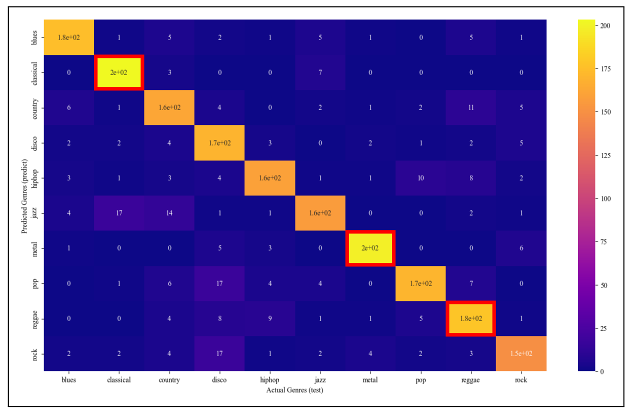

Figure 10. K-Nearest Neighbors (KNN)

5. Discussions

5.1. Findings

a. Random Forest Classifier (Figure 9): The Random Forest Classifier demonstrates notable proficiency in recognizing classical music. While results exhibit variability across iterations, it is evident that Random Forest classifiers excel particularly in the classification of classical music.

b. K-Nearest Neighbors (KNN) (Figure 10): The KNN model exhibits elevated accuracy in the classification of both classical and metal genres, aligning seamlessly with the distinctive patterns observed within these genres through PCA analysis.

c. Support Vector Classification (SVC) (Figure 11): The SV Classification model excels in classifying classical, metal, and jazz music. However, the disparity in performance is not significant when compared to its prowess in recognizing other genres. Noteworthy is the model’s adept recognition of jazz music, a particularly commendable feat given the PCA analysis’s revelation of jazz having a high

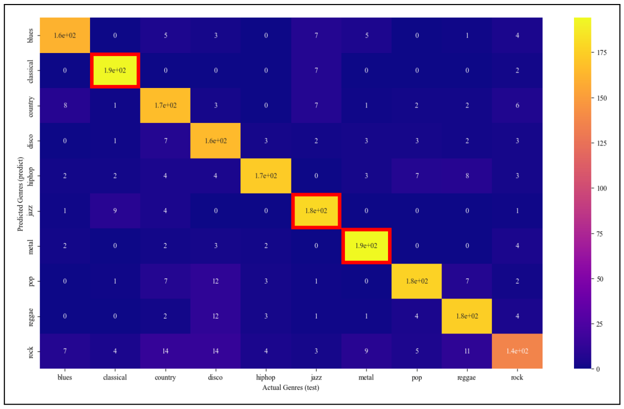

Figure 11. Support Vector Classification

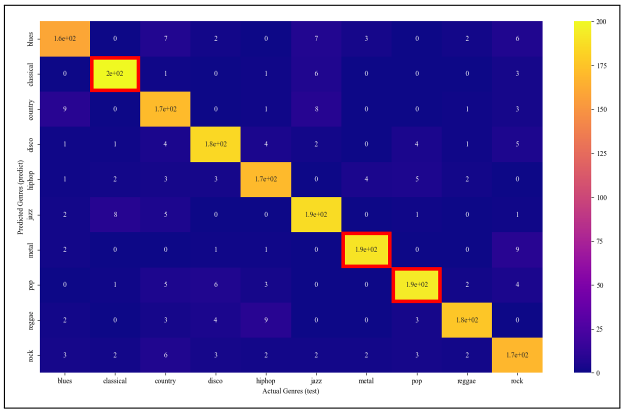

Figure 12. XGBoost

density of data points clustered with other genres, hence increasing the complexity of classification.

d. XGBoost (Figure 12): The XGBoost method performs exceptionally well on the precise three genres (classical, metal, and pop) that exhibit distinct distribution patterns, as identified in the PCA analysis. Moreover, the model demonstrates consistent and robust performance across various genres, indicating an overall high level of proficiency and effectiveness.

5.2. Summary

In conclusion, this project highlights the efficacy of machine learning models in the task of music genre classification. Through a in depth examination of various models, including Random Forest, KNN, Support Vector Classification, and XGBoost, we have uncovered patterns that illuminate the models' strengths and weakness in discerning different musical genres.

The correlation observed between the distinctiveness of genres in PCA analysis and their corresponding classification accuracy in specific models underscores the significance of feature selection and model adaptability. Notably, features capturing perceptual variance and mean emerge as crucial elements in the accurate classification of music genres. The varying importance levels of MFCC features (Figure 13) further emphasize the role of timbral characteristics in distinguishing genres—a phenomenon well-established in music genre classification.

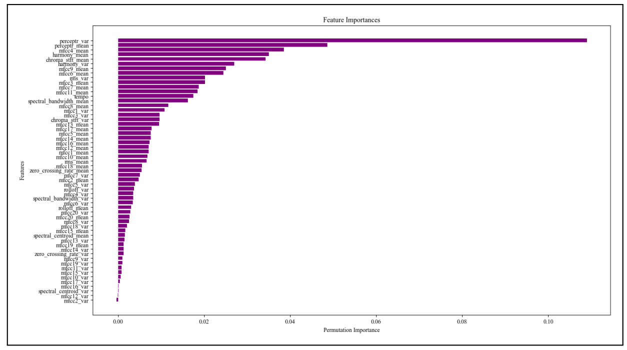

Figure 13. Feature Importances

Furthermore, the hierarchy of feature importance, with features like "harmony_mean" and "tempo" holding significance albeit to a lesser degree, contributes to a nuanced understanding of the multifaceted nature of music genre classification.

The recognition that both mean and variance play pivotal roles in feature importance reinforces the models' need to comprehend both central tendencies and the spread of data. This dual consideration highlights the complexity inherent in capturing the diverse characteristics of music genres.

However, the project also revealed limitations, particularly in the oversimplification of complex relationships between features through PCA. The vast diversity within genres, including sub-genres and fusion genres, poses a challenge for precise classification. The models may struggle to capture subtle distinctions between closely related genres or accurately identify music that defies conventional genre boundaries.

One other hardship came when using we had to use librosa library. Since it was first time using it for all members in our team, each process in this project took longer time than needed. Moreover, the versions we first had was little different among the team, resulting in different conclusions at first. Through out the project we have figured out that we had different versions and at the end, this project has gave us the chance to be more familiar with the functions.

As we look to the future, our team believes it is crucial to explore deeper dimensional-reduction techniques beyond PCA, aiming to capture even finer distinctions between genres. Additionally, investigating the impact of different hyperparameter choices on PCA analysis stands as a potential avenue for enhancing its effectiveness. Furthermore, we could also explore methods to incorporate more tedious genre definitions or consider alternative approaches to address the intricate nature of musical genres beyond the broad categories employed in this study.

In summary, this final project not only contributes to the understanding of machine learning applications in music genre classification but also identifies aspects for future research and improvement. As we navigate the complicated realm of music analysis, the insights gained here could be used as refined methodologies and advancements in the accurate classification of diverse music genres.

6. References

[1] Benjamin Murauer and Günther Specht. 2018. Detecting Music Genre Using Extreme Gradient Boosting. In WWW '18 Companion: The 2018 Web Conference Companion, April 23–27, 2018, Lyon, France. ACM, New York, NY, USA 5 Pages. https://doi.org/10.1145/3184558.3191822

7. Data Source

[1] https://www.kaggle.com/datasets/andradaolteanu/gtzan-dataset-music-genre-classification/data