Music Genre Classification via Advanced Machine Learning Methods

AGROUAZ Rim
rim.agrouaz@etu.uae.ac.ma

Supervised by: Pr KHAMJANE Aziz
akhamjane@uae.ac.ma

National School of Applied Sciences, Al Hoceima

Abstract- Music genre classification is a challenging and fascinating field within Music Information Retrieval (MIR), with applications in building music libraries, identifying similar tracks, and recognizing listener preferences. Due to the complex nature of musical audio data, extracting reliable features is essential for effective classification. This study compares the performance of four machine learning algorithms—K-Nearest Neighbors (KNN), Support Vector Machines (SVM), Logistic Regression, and Neural Networks—both with and without dimensionality reduction. Results show that dimensionality reduction significantly improves performance, enabling more accurate and reliable predictions. The research also highlights the strengths of each method, tailored to the specific demands of music genre classification.

I. Introduction:

Music is a major component of today's internet content, with the web serving as one of the most important sources for accessing music. Numerous platforms are dedicated to sharing, distributing, and commercializing music. Personal music libraries can contain hundreds of songs, while professional collections often include tens of thousands of files. These files are generally organized by song title or artist name, which makes it difficult to search for a specific genre.

Music engineering promotes the use of categories and family-based operators to organize audio collections more effectively. This highlights the growing need for automated methods to classify audio files into categories. Despite recent advancements in category-based organization, accurately defining genres remains a challenge, as it often relies heavily on subjective consumer input.

In this paper, we conducted a comprehensive study on the classification of audio files into genres using two distinct category sets: a smaller set consisting of 4 classes and a larger set with 10 classes. The goal was to evaluate and compare the performance of various machine learning models in classifying music genres accurately.

The study began with the extraction of audio features, a critical step in transforming raw audio signals into meaningful representations for classification. We utilized several feature extraction techniques, including Mel-Frequency Cepstral Coefficients (MFCCs), Fast Fourier Transforms (FFTs), and other statistical audio features such as spectral centroid, spectral roll-off, and zero-crossing rate. These features capture important characteristics of he audio signals, such as frequency content, timbre, and rhythm, which are essential for distinguishing between different genres.

The extracted features were then used as input for four widely applied machine learning

algorithms: K-Nearest Neighbors (KNN), Neural Networks, Support Vector Machines (SVM), and Logistic Regression. Each of these models was trained and tested on the selected datasets to examine their effectiveness in handling both simpler classification tasks (4 classes) and more complex ones (10 classes).

II. Literature Review :

In the very recent years, development of music recommendation system has been a more heated problem due to a higher level of digital songs consumption and the advancement of machine learning techniques. The foundational work by Gautam Chettiar "Music Genre Classification" (2022) [1] explores the differences between Fast Fourier Transform (FFT) and Mel-Frequency Cepstral Coefficients (MFCC) in audio feature extraction. FFT focuses on analyzing the raw frequency content of audio signals, while MFCC emphasizes perceptually relevant features using the Mel scale. Both techniques serve distinct purposes and are widely used in tasks like speech recognition and genre classification, providing complementary strengths in audio analysis.

Tom LH. Li, Antoni B. Chan and Andy HW. Chun, "Automatic Musical Pattern Feature Extraction Using Convolutional Neural Network" [2] examines the performance of music genre classification across datasets containing 4, 7, and 10 classes. For each classification task, the study evaluates and compares the effectiveness of convolutional neural network. The results highlight how the complexity of the classification problem, influenced by the number of classes, affects the performance of the models, providing detailed insights into their strengths and limitations at each level.

Michael Haggblade ,Yang Hong , Kenny Kao, "Music Genre Classification" [3] demonstrates the differences in performance between various algorithms through detailed results and analysis. The comparative study provides valuable insights into the strengths and limitations of each approach, offering a clear understanding of their effectiveness in different scenarios. The methodologies and findings presented in this paper served as a significant source of inspiration, shaping the direction and approach of this work.

Muhammad Asim Ali, Zain Ahmed Siddiqui "Automatic Music Genres Classification using Machine Learning" [4] provided valuable insights into the use of machine learning for music genre classification. It highlighted the effectiveness of algorithms like k-Nearest Neighbors (k-NN) and Support Vector Machines (SVM), showing that SVM outperforms k-NN with an overall accuracy of 77%. The paper also emphasized the importance of feature extraction using Mel Frequency Cepstral Coefficients (MFCC) and explored the impact of dimensionality reduction through PCA. One key takeaway is that while PCA simplifies data, it may reduce classification accuracy, underscoring the trade-offs between computational efficiency and performance.

III. Preparing the Dataset:

1. *Choosing the Dataset :*

The dataset used for this research was sourced from Kaggle [5] and consists of 1000 audio tracks, each 30 seconds long, intended for genre classification. The dataset includes 10 genres, with 100 tracks per genre. All tracks are 22050Hz Mono 16-bit .wav audio files. The study was conducted on both the full 10-genre dataset and a subset of 4 distinct genres: classical, jazz, metal, and pop. These four genres were selected for their clear differentiation, as previous studies have indicated that classification accuracy tends to decrease when the number of genres exceeds four. This dual approach enabled a comprehensive comparison of model performance at different levels of complexity.

2.Dataset Preprocessing:

The data was visualized by displaying the waveform of each genre along with its associated spectrogram. This approach facilitated a deeper understanding of the

acoustic features unique to each genre. The waveform provided an overview of the raw audio signal, while the spectrogram, which represents the distribution of frequencies over time, offered further insights into the frequency patterns and variations characteristic of each genre. This visualization step was essential for gaining a better understanding of the data prior to feature extraction and classification.

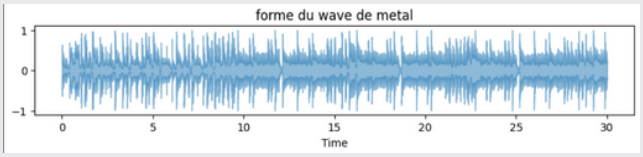

Figure 1 : Metal wave form

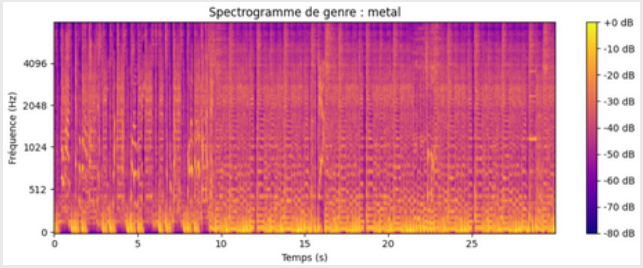

Figure 2 : Metal spectrogram

3.*Features extraction:*

a . Spectral Features:

Spectral features are derived from the frequency domain and represent the distribution of energy across different frequencies in the signal. These features are particularly useful for capturing the tonal characteristics of music and are essential for distinguishing between different types of sounds or instruments.

- Chroma features: capture the harmonic content of an audio signal by focusing on the 12 pitch classes. This feature is essential for understanding harmony and chord progressions in music, as it represents how energy is distributed across pitch classes rather than individual frequencies. It is commonly used in music genre classification tasks where harmonic structures play a critical role.

Mathematical expression:

Let X(t) be the audio signal, and S(t) the spectrogram at time t, the chroma vector C(t) at time t is computed as:

$$C(t) = \sum_{f \in F} W(f) \cdot S(t, f)$$

Where:

- F is the frequency range of interest.
- W(f) is a weighting function that corresponds to the 12 pitch classes.
- Spectral bandwidth : measures the width of the spectrum, reflecting how spread out the frequencies are. It can describe the timbral texture of the sound. Narrow bandwidth indicates a more tonal or harmonic sound, while wider bandwidth corresponds to more noise-like or percussive sounds. This feature is particularly useful in distinguishing between different types of instruments or musical textures.

Mathematical expression:

The spectral bandwidth BW is defined as:

$$BW = \sqrt{\frac{\sum_{f=0}^{F} (f - C)^2 \cdot |S(f)|}{\sum_{f=0}^{F} |S(f)|}}$$

Where:

- C is the spectral centroid,
- S(f) is the magnitude of the spectrum at frequency f,
- F is the number of frequency bins.
- Spectral roll-off is the frequency below which a certain percentage of the total spectral energy lies. It helps to capture the "sharpness" or "smoothness" of the sound and is often used to differentiate between harmonic and non-harmonic sounds.

Mathematical expression:

The spectral roll-off R is defined as the frequency fα.

$$\sum_{f=0}^{f_\alpha} |S(f)| = \alpha \sum_{f=0}^{F} |S(f)|$$

Where:

- α is a threshold value (commonly set to 0.85),
- S(f) is the spectrum at frequency f,
- F is the total number of frequency bins.
- Spectral centroid : represents the "center of mass" of the spectrum and is a measure of brightness. It is a commonly used feature to distinguish between different timbres, such as between bright (e.g., pop music) and dark (e.g., classical) sounds.

Mathematical expression:

The spectral centroid C is calculated as:

$$C = \frac{\sum_{f=0}^{F} f \cdot |S(f)|}{\sum_{f=0}^{F} |S(f)|}$$

Where:

- S(f) is the magnitude of the spectrum at frequency f,
- F is the total number of frequency bins.

b . Temporal Features:

Temporal features are derived from the time-domain signal and capture the dynamics of the audio, such as its loudness, transient characteristics, and the rate of signal changes over time.

- RMSE: measures the energy of an audio signal by quantifying its loudness over time. It is a common feature used for dynamic analysis, as it reflects the variation in volume across an audio track.

Mathematical expression:

The RMSE for a signal x(t) is given by:

$$\text{RMSE}(t) = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x(t_i)^2}$$

Where:

- N is the number of samples in the window.
- x(ti) is the signal amplitude at sample ti
- Zero-Crossing Rate (ZCR) is the rate at which the audio signal changes sign (from positive to negative or vice versa). It is a simple yet effective feature for detecting percussive sounds or identifying noisy sections of a track.

Mathematical expression:

The zero-crossing rate Z is calculated as:

$$ZCR = \frac{1}{N}\sum_{i=1}^{N-1} |\text{sign}(x(t_i)) - \text{sign}(x(t_{i+1}))|$$

- Fast Fourrier Transform (FFTs) :

The Fast Fourier Transform is a mathematical algorithm used to convert a signal from the time domain to the frequency domain. It represents the signal as a sum of sinusoidal components, each with a specific frequency, amplitude, and phase. This transformation is particularly useful for analyzing the frequency content of a signal, making it an essential tool in music and audio analysis.

for analyzing the frequency content of a signal, making it an essential tool in music and audio analysis.

It allows for the identification of prominent frequencies, such as musical notes, and helps differentiate between various sound sources.

The frequency domain values of the signal can be obtained by using a rather simple mathematical formula :

$$x[k] = \sum_{n=0}^{N-1} x[n] e^{\frac{-j2\pi kn}{N}}$$

- Mel-Frequency Cepstral Coefficients (MFCCs):

Mel-Frequency Cepstral Coefficients (MFCCs) are designed to closely mimic the human ear's perception of sound. The human auditory system is more sensitive to lower frequencies and less sensitive to higher frequencies, so MFCCs are designed to reflect this non-linear frequency perception. The Mel scale f is given by:

$$Mel(f) = 2595\log_{10}(1+f/700)$$

where Mel(f) is the frequency in Hz, and f is the frequency on the Mel scale.

In order to extract these features, the audio signal is first divided into overlapping frames of 20 milliseconds to capture short-term characteristics. A Hamming window is then applied to each frame to reduce spectral leakage and ensure smooth transitions. Next, the Fast Fourier Transform (FFT) is used to convert the signal from the time domain to the frequency domain. The resulting spectrum is passed through a Mel-scale filterbank, which emphasizes frequencies in a manner similar to human hearing by providing higher resolution for lower frequencies and compressing higher frequencies. The energy of each filter is computed and transformed into a logarithmic scale, reflecting the human perception of loudness. Finally, the Discrete Cosine Transform (DCT) is applied to the log energies to decorrelate the features and generate a compact representation. This process results in the MFCCs, which encapsulate the most perceptually relevant aspects of the audio signal for applications like music genre classification or speech recognition.

Figure 3 : MFCC flow

4 . Feature Importance Analysis :

Feature importance is a crucial aspect of machine learning that helps understand the contribution of each feature in predicting the target variable. In this study, the importance of features was analyzed using two tree-based algorithms: DecisionTreeClassifier and RandomForestClassifier. These algorithms are well-suited for evaluating feature importance due to their inherent ability to measure the impact of each feature on the model's performance during the training phase.

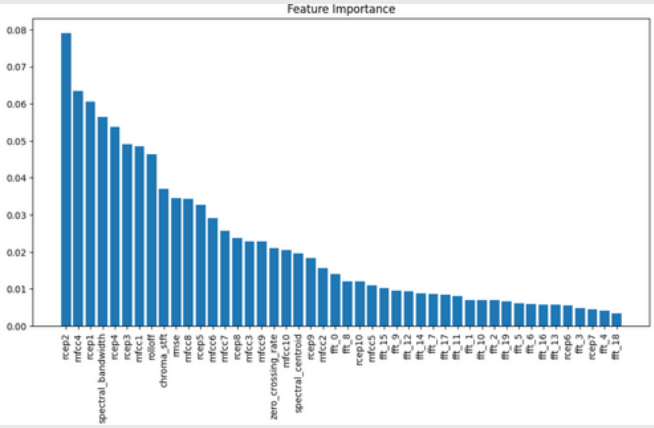

Figure 4 : Features importance using DecisionTreeClassifier

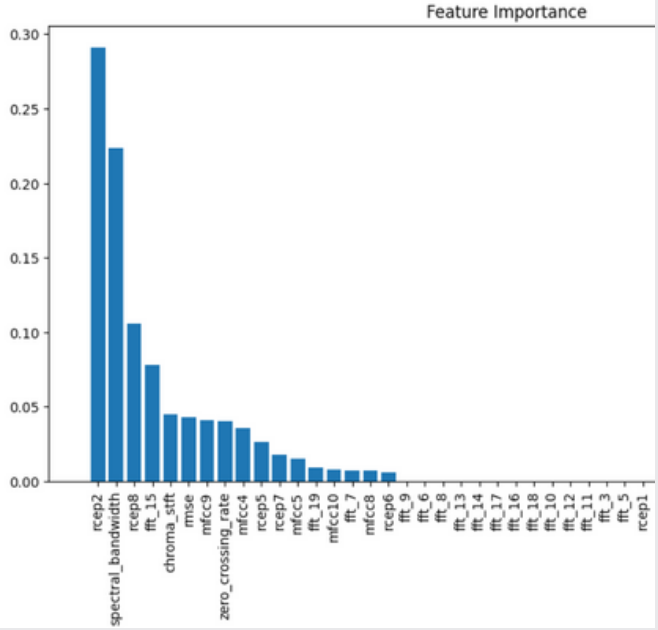

Figure 5 : Features importance using RandomForestClassifier

To gain deeper insights into the music genre classification model, we visualize the three most important features for each class. These features are selected based on their contribution to the decision-making process of the model. By identifying and displaying these key features for each genre, we can better understand which characteristics of the audio data are most indicative of different musical styles. This analysis not only helps in interpreting the model's behavior but also guides further feature engineering efforts, ensuring that the classification system focuses on the most relevant aspects of the music for each genre.

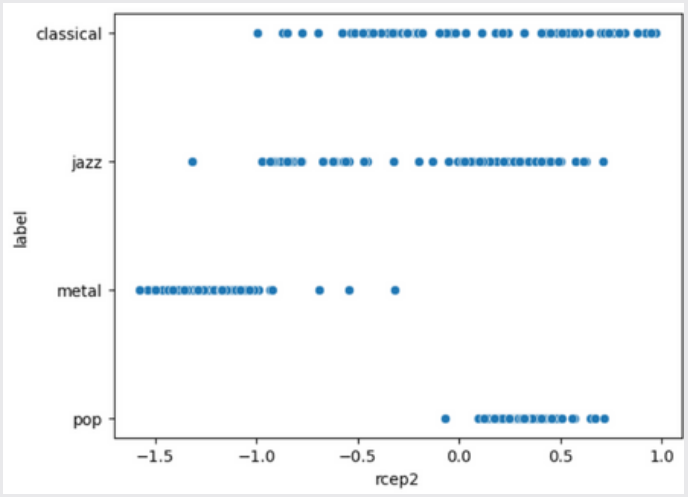

Figure 6 : Distribution of RCEP2 across classes

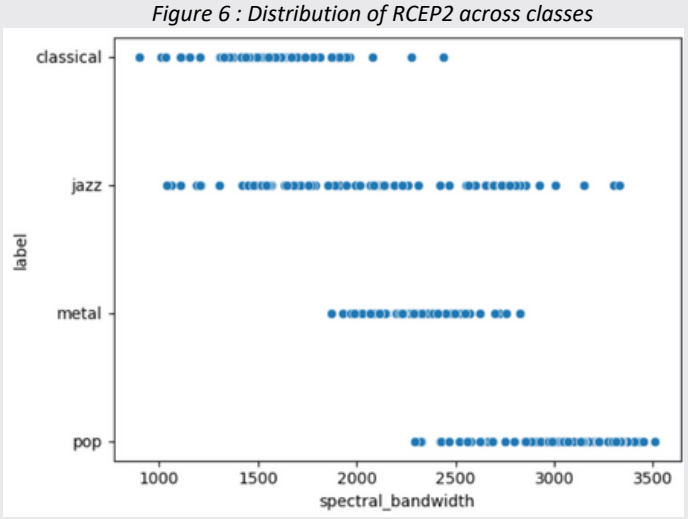

Figure 7 : Distribution of Spectral_Bandwidth across classes

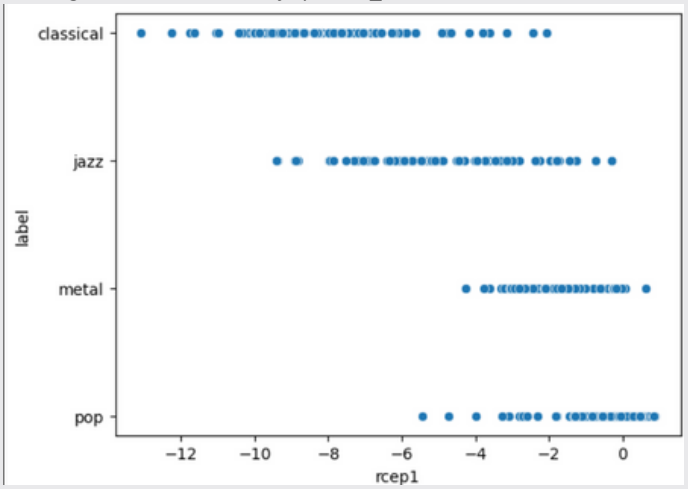

Figure 8 : Distribution of RCEP1 across classes

IV. System Architecture:

This architecture outlines a framework for processing and classifying multimedia data, such as images or songs. The input undergoes preprocessing to enhance quality, followed by feature extraction to identify key attributes like

MFCCs for songs or edges for images. These features are passed to a classifier, which assigns the data to specific categories, such as musical genres or emotional expressions. The recognized category then triggers an action, like playing a related song. Additionally, a database can provide stored data for analysis, with its features contributing to improving classification accuracy and system knowledge.

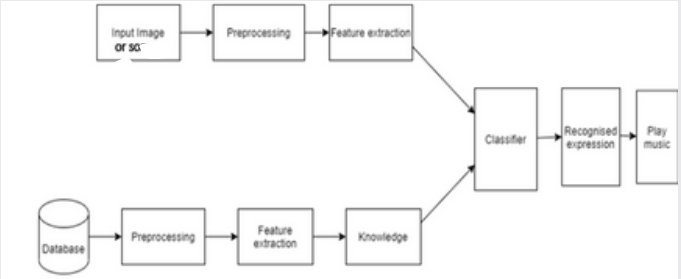

Figure 9 : System architecure

V. Algorithms :

- *K-Nearest-Neighbors :*

KNN is a simple and effective classification algorithm based on proximity. To classify a data point, KNN examines the "k" closest neighbors in the feature space and assigns the data point the majority class among its neighbors. For this study, we performed hyperparameter tuning using GridSearchCV to determine the best values for "k," the distance metric, and the weight function. The tuning process revealed that the optimal hyperparameters for our case were metric='manhattan', n_neighbors=4, and weights='distance', which resulted in the best classification performance. This algorithm is particularly useful for non-linear data, but its performance can be affected by the size of the dataset and the choice of the value for "k."

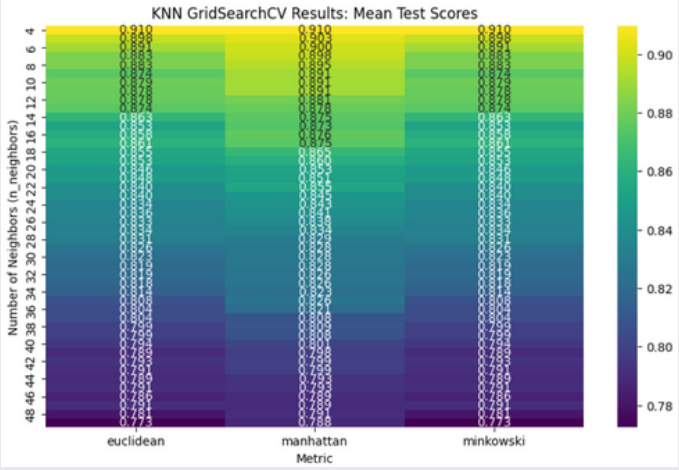

Figure 10 : KNN Gridsearch results

- *Support Vector Machine:*

SVM is a powerful classifier that seeks to maximize the margin between different classes by finding an optimal hyperplane that best separates the classes. For this study, we performed hyperparameter tuning using GridSearchCV to find the optimal values for the regularization parameter C, the kernel type, and the gamma parameter. The tuning process revealed that the best hyperparameters for our case were C=100, gamma=0.001, and kernel='rbf', which resulted in the best classification performance. SVM is particularly effective in high-dimensional spaces and for classification problems with clear class margins. By using the radial basis function (RBF) kernel, SVM can handle non-linear separations efficiently.

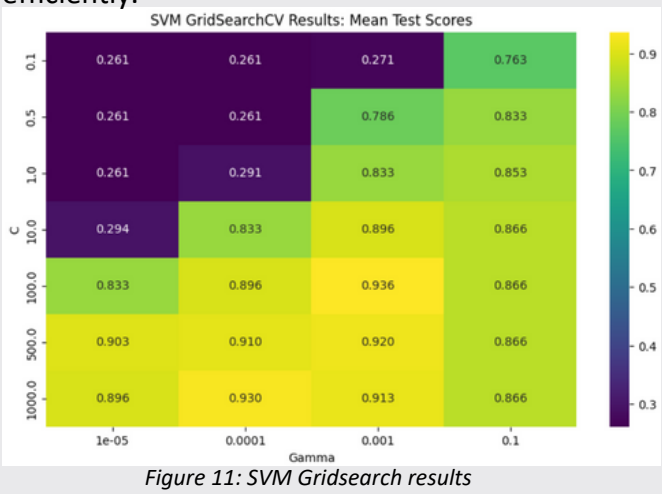

Figure 11: SVM Gridsearch results

- *Logistic regression:*

Logistic Regression is a popular classifier used for binary and multi-class classification problems. It estimates the probability of a data point belonging to a specific class by applying a linear decision boundary. In this study, we performed hyperparameter tuning using GridSearchCV to optimize the C (regularization strength), penalty (regularization type), and solver (optimization algorithm) parameters. The best hyperparameters for our case were C=1.0, penalty='l1', and solver='saga', which provided the best classification performance. Logistic Regression is particularly useful for linearly separable data and can also perform well with multi-class problems when properly tuned.

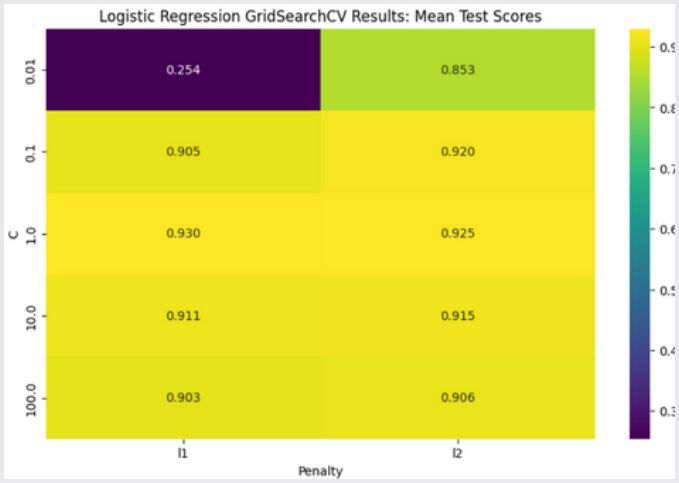

Figure 12 : Logistic Regression Gridsearch results

- *Neural Network (MLPClassifier) :*

The Multilayer Perceptron (MLP) is a type of artificial neural network that consists of multiple layers of nodes, where each node in a layer is connected to every node in the previous and subsequent layers. It is a powerful model capable of capturing complex patterns in data. In this study, we utilized GridSearchCV to optimize the hyperparameters of the MLP, focusing on the activation function and the number and size of hidden layers. After performing the tuning process, the best parameters found were activation='relu' and hidden_layer_sizes=(100, 400). These configurations resulted in the highest model performance during cross-validation. The ReLU activation function helps with faster convergence, and the specific hidden layer configuration allows the model to capture a broader range of data patterns.

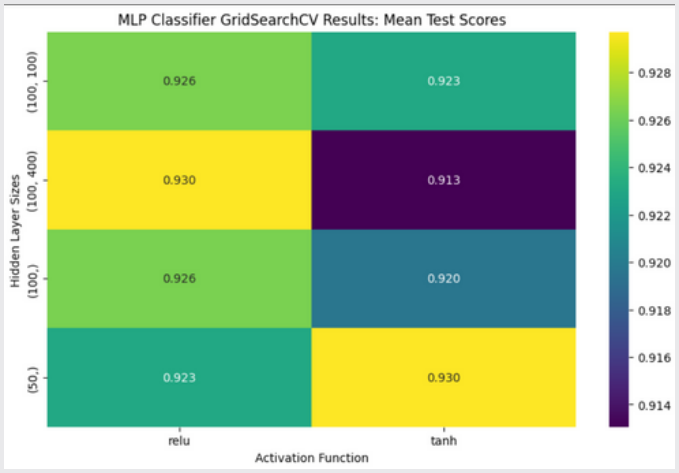

Figure 13 : Neural Network Gridsearch results

VI. Classification and Evaluation :

At first we used KNN for classification on the scaled features.

Result:

- 4 Classes Recognition Rate = 0.91
- 10 Classes Recognition Rate = 0.64

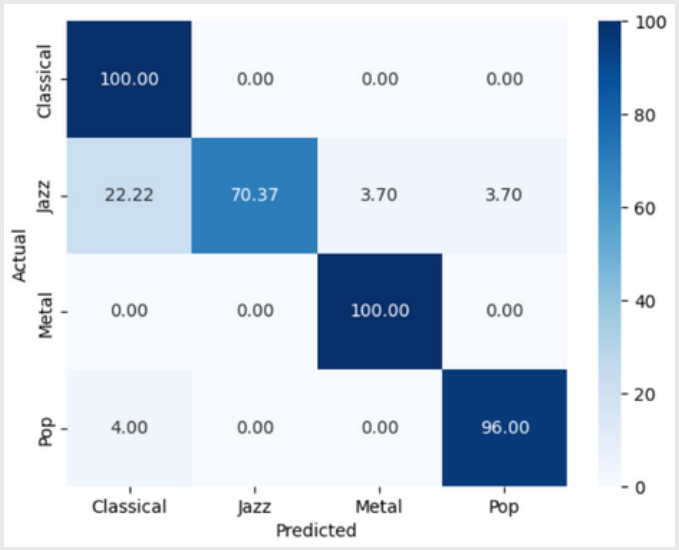

Figure 14: KNN Confusion Matrix with 4 classes

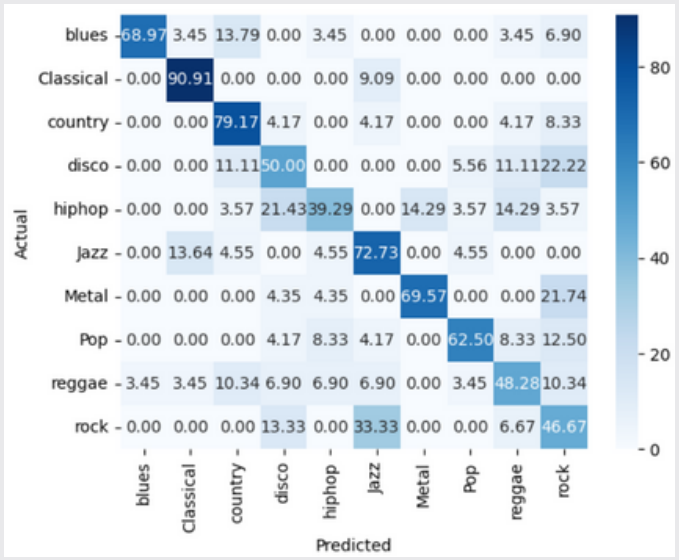

Figure 15: KNN Confusion Matrix with 10 classes

The recognition rate for the k-Nearest Neighbors (kNN) model increased significantly from 0.64 to 0.91 when reducing the number of classes, highlighting the model's higher accuracy when dealing with a smaller class set. This improvement demonstrates the model's ability to perform well under simplified classification scenarios. Next, we will analyze the performance differences for other algorithms under the same conditions.

- Support Vector Machine :

Result:

- 4 Classes Recognition Rate = 0.92
- 10 Classes Recognition Rate = 0.63

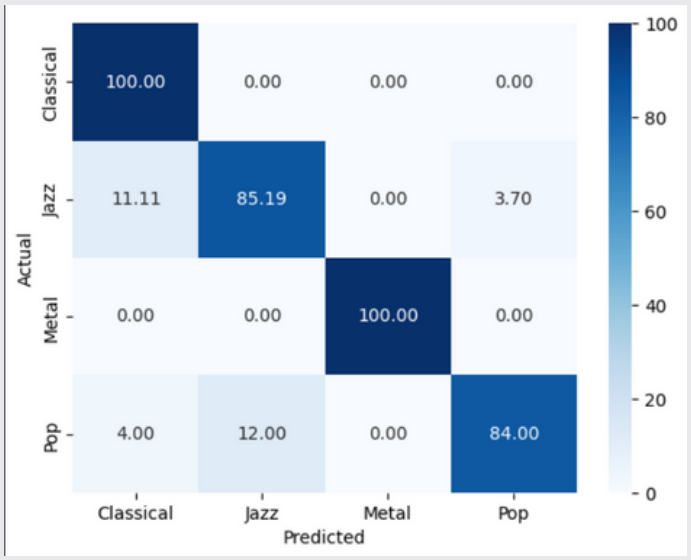

Figure 15: SVM Confusion Matrix with 4 classes

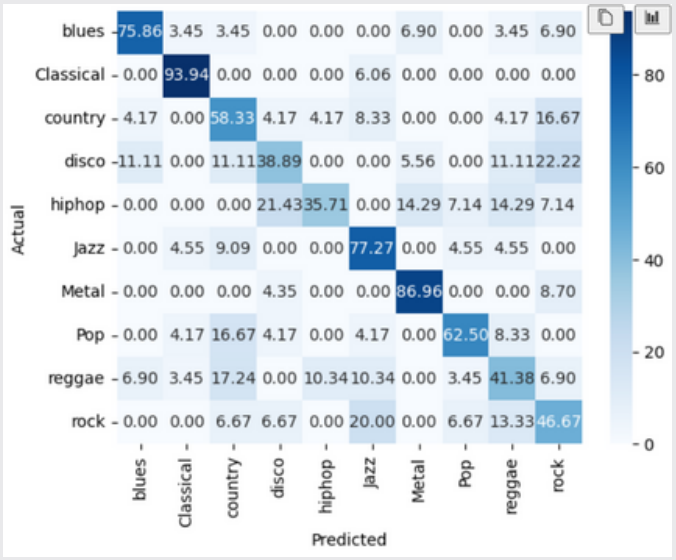

Figure 16: SVM Confusion Matrix with 10 classes

- Logistic Regression :

Result:

- 4 Classes Recognition Rate = 0.91
- 10 Classes Recognition Rate = 0.57

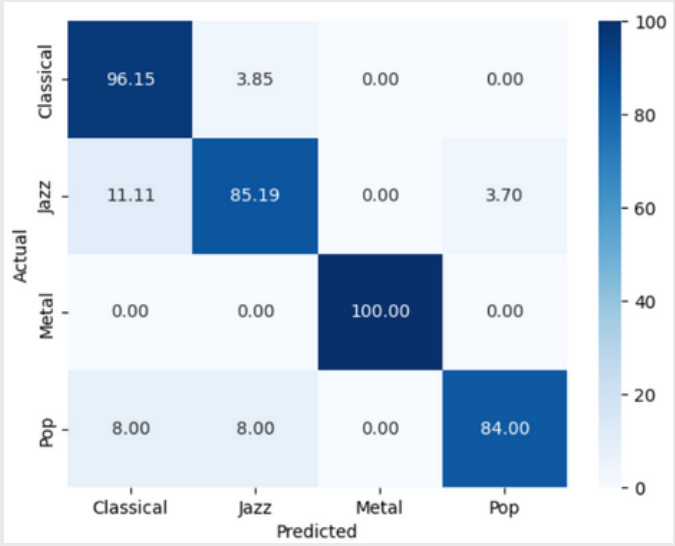

Figure 17: Logistic Regression Confusion Matrix with 4 classes

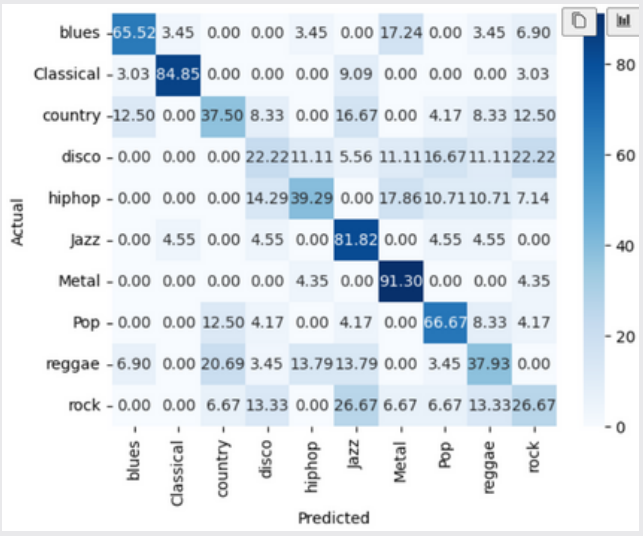

Figure 18: Logistic Regression Confusion Matrix with 10 classes

- Neural Network :

Result:

- 4 Classes Recognition Rate = 0.93
- 10 Classes Recognition Rate = 0.64

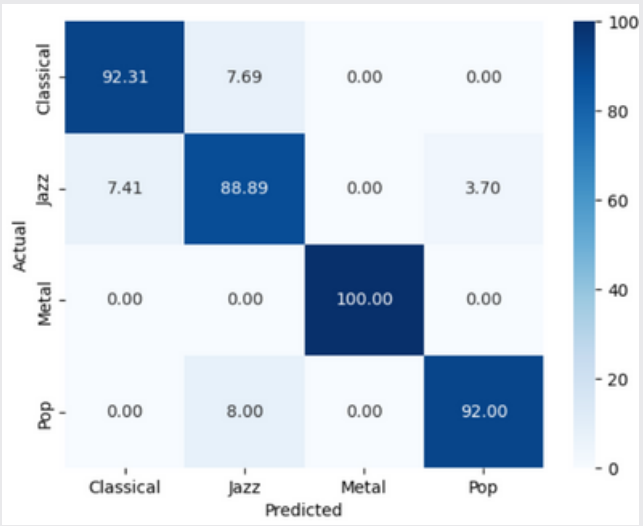

Figure 19 : Neural Network Confusion Matrix with 4 classes

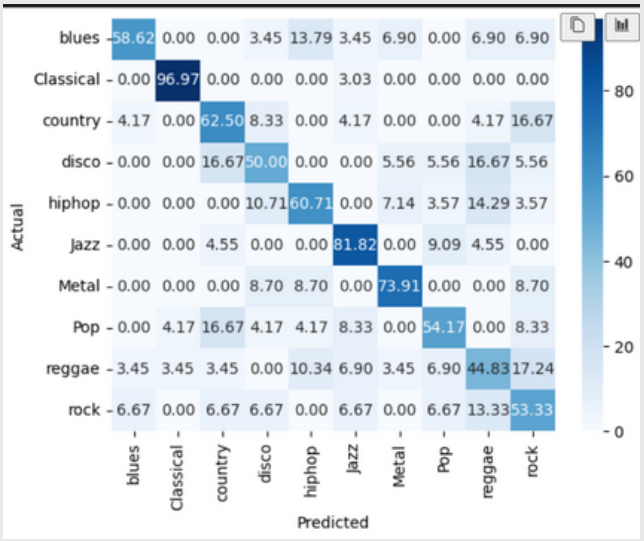

Figure 20: Neural Network Confusion Matrix with 10 classes

The overall recognition rate improved from 0.60 to 0.91 across all algorithms when reducing the number of classes, demonstrating a significant boost in classification performance. Among these algorithms, the neural network emerged as the best performer with four classes, making it the most suitable model for deployment in our application.

VII. Model Deployement:

Flask, combined with HTML and CSS, was used to deploy the model and develop a graphical user interface for genre detection. This interface enables users to classify music genres either by uploading audio files or providing YouTube links, offering a seamless and interactive solution for accessing the model's functionality.

After detecting the genre, the interface displays the identified genre, the confidence percentage, and provides five YouTube recommendations of the same genre, offering a comprehensive and interactive user experience.

Figure 21: User Interface

Figure 22 : Audio file genre detection

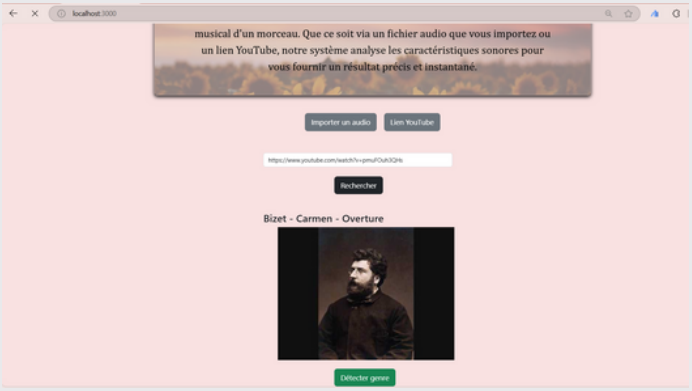

Figure 23 : Youtube url uploading

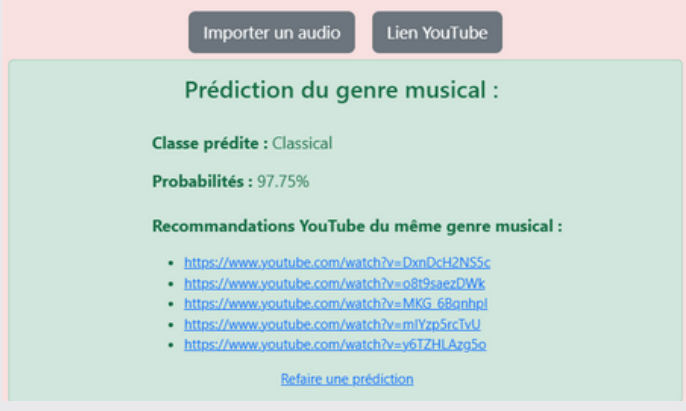

Figure 24 : Youtube video genre detection

VIII. Model limits:

One notable limitation of the model lies in its difficulty in accurately classifying audio tracks that belong to mixed or hybrid genres, such as Classical/Jazz or Jazz/Pop. These genres often exhibit overlapping acoustic features, such as similar rhythms, harmonies, or instrumentations, which can confuse the model and lead to misclassifications. Additionally, the dataset may not contain sufficient examples of these mixed genres, making it challenging for the model to learn their unique characteristics effectively.

IX. Conclusion:

In conclusion, this study on music genre classification evaluated four different algorithms: Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Neural Networks. The classification was conducted over two different sets of classes, 4 and 10. The results revealed that reducing the number of classes significantly impacted the accuracy of the models. Among the evaluated algorithms, the Neural Network performed the best when applied to the 4-class classification task, demonstrating superior accuracy compared to the other models. This finding highlights the importance of class reduction in improving model performance and suggests that neural networks can be highly effective for music genre classification when optimized for fewer classes.

X. References :

1. *Gautam Chettiar* : Music Genre Classification (2022). [researchgate.net]
2. *Tom LH. Li, Antoni B. Chan and Andy HW. Chun* : Automatic Musical Pattern Feature Extraction Using Convolutional Neural Network. [researchgate.net]
3. *Michael Haggblade ,Yang Hong , Kenny Kao* : Music Genre Classification. [stanford.edu]
4. *Muhammad Asim Ali, Zain Ahmed Siddiqui* : Automatic Music Genres Classification using Machine Learning. [thesai.org]
5. GTZAN dataset . [kaggle]