Music Genre Classification

Omer Diab and Chi Ung, Professor Salem Othman

Wentworth Institute of Technology

Abstract. Music genres facilitate the identification of patterns and makes it easier to find music that is most satisfying to our personal taste by exploring more of their favorite music types. Music genres also help companies make recommendations and suggestions that would considerably enhance their customers' listening enjoyment. Genre helps on organize a large mass of music so that it's easier to locate and identify a song we don't know. In this project, we developed a Music Genre Classification using Machine Learning techniques. We first created our dataset but extracting music features from 1000 song using LIBROSA library. Then we used some Classification techniques such as K-nearest Neighbor, Support Vector Machine (SVM), and Logistic Regression.

Keywords: Music, Classification, Genre, GTZAN Dataset, LIBROSA, K-Nearest Neighbor, Support Vector Machine, SVM, Logistic Regression, Spectrogram, MFCC, Spectral, Chroma

1. **Introduction**

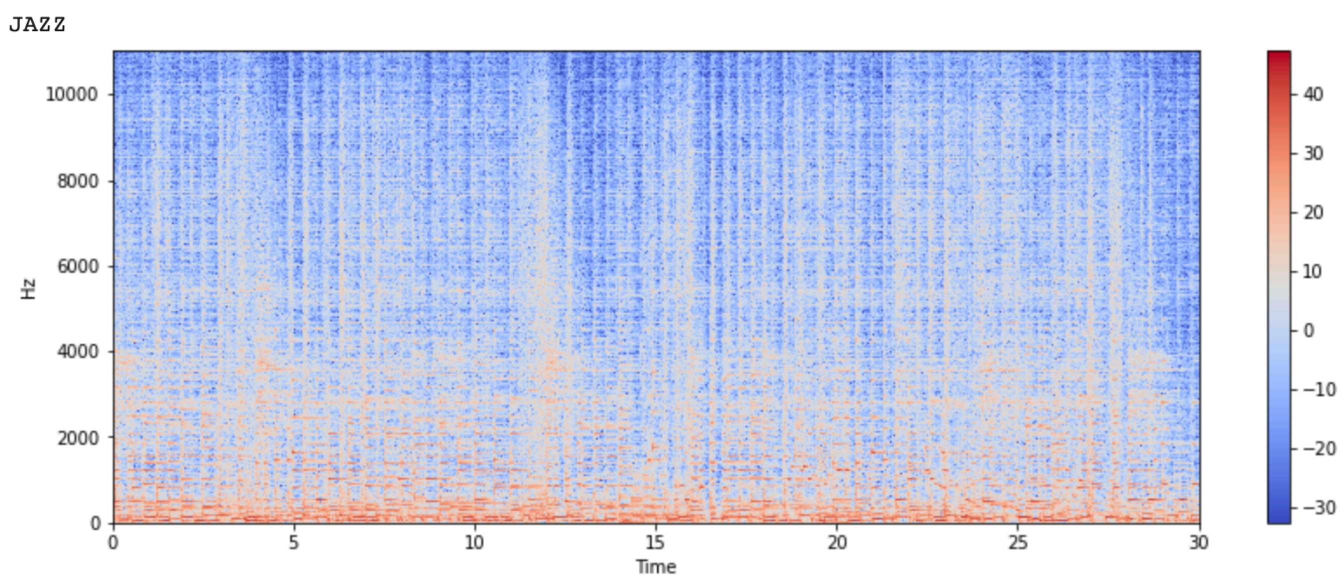
Every musical genre has distinctive features and techniques. Songs that share the same patterns can be grouped in a genre that describe these patterns. These features are a mixture of rhythm, speed progression, key, instrumentation, melody, harmony, and tempo, among others. When you discover a new song that you like, you often want to hear more songs that sound similar and add it to your favorite list of music. Nonetheless, if you do not know the genre, you will not be able to search for that. Moreover, big companies like YouTube or Spotify use music genre classification in order to make recommendations to their customer. In this paper, we will present our work of applying Machine Learning techniques towards building a classification method in order to identify a genre from music features.

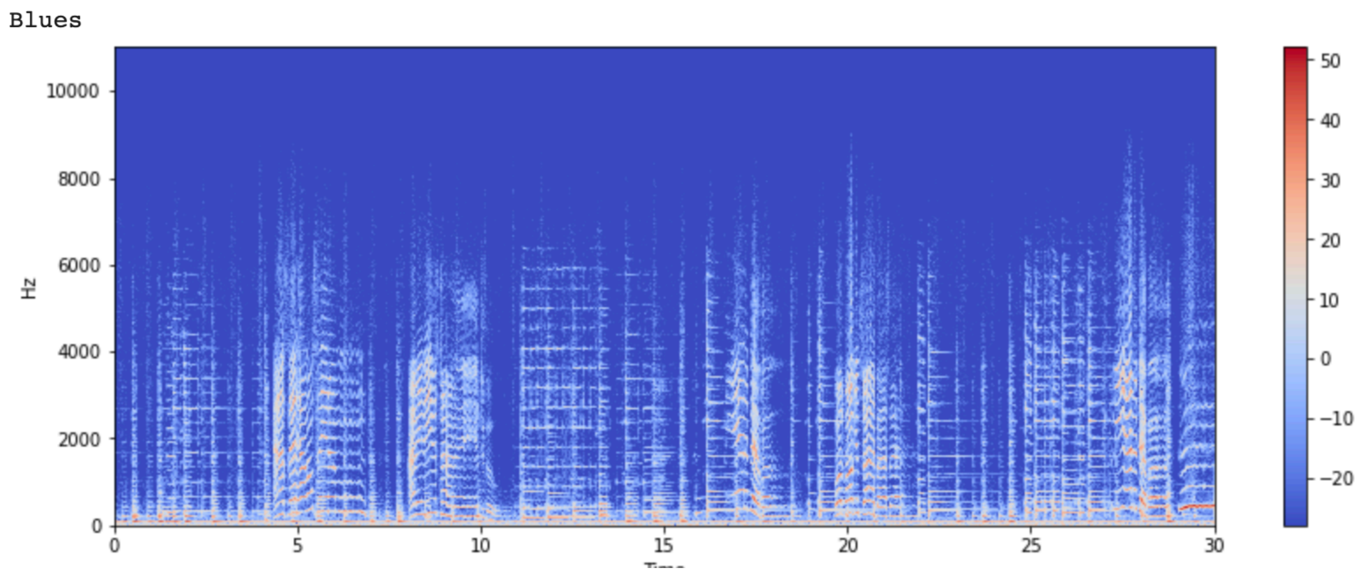
2. **Implementation**

2.1. **Datasets and Features**

After an intensive research on the internet, we could not find a dataset for music features labeled by a genre. We had to create our own dataset from scratch. Luckily, there is GTZAN dataset out there that provides 1000 of 30 seconds music clips, a 100-music clip for each genre and each music clip was labeled with its own genre.

Fortunately, Python has great libraries for audio processing. One of these libraries is LIBROSA which we used to visualize and extract features from the music clips and create our own dataset. For visualization, LIBROSA provides Spectrogram which is a visual representation of the spectrum of frequencies of sound or other signals as they vary with time. Below are examples of two audios from two different genres.





Every audio signal consists of many features. However, we must extract the characteristics that are relevant to the problem we are trying to solve. The features used for this project are:

1- **Spectral Centroid:** Indicates where the ”Centre of Mass” for a sound is located and is calculated as the weighted mean of the frequencies present in the sound.
2- **Spectral Rolloff:** It iss a measure of the shape of the signal. It represents the frequency below which a specified percentage of the total spectral energy
3- **Mel-Frequency Cepstral Coefficients (MFCC):** Which is a small set of features which concisely describe the overall shape of a spectral envelope.
4- **Chroma Frequencies:** Chroma features are an interesting and powerful representation for music audio in which the entire spectrum is projected onto 12 bins representing the 12 distinct semitones (or chroma) of the musical octave.

2.2. **Classifications**

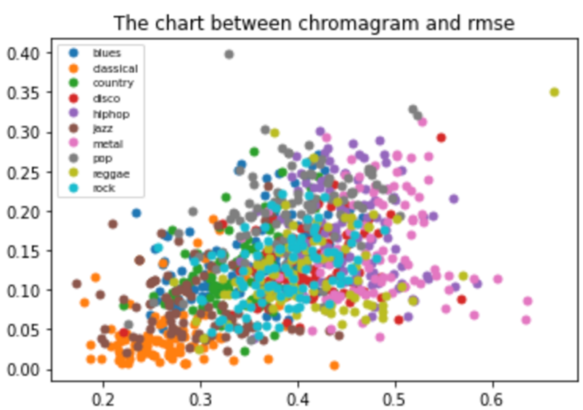
After extracting the features, we used several classification algorithms in order to classify the songs into different genres.

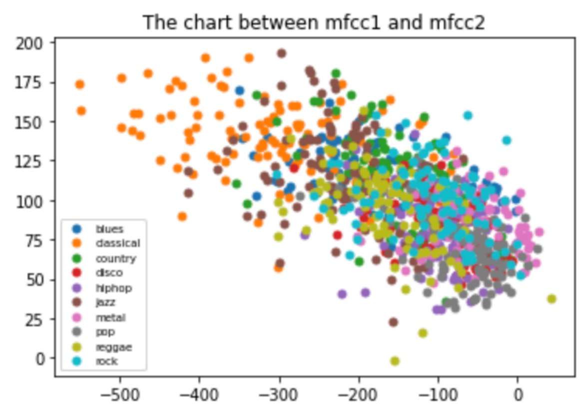
1. Our first approach for classification was K-Nearest Neighbors which is a widely used technique. K-nearest neighbor is a naive search on the training data to get the best label for the data in a classification problem. So, to classify a data point we look at its K nearest neighbors and we classify the point as the majority class in those neighbors. We applied GridSearchCV to find the best K value which was 5.
2. Our second Approach for classification was Support Vector Machine (SVM). It is a supervised machine learning algorithm that follows a technique called the kernel trick to transform the data and based on these transformations, it finds an optimal boundary between the possible outputs. In addition, we applied three different kernels to check what was the best approach for training the GTZAN dataset. Importantly, using a pipeline to pipe the data normalization, and check if the features have low variance, remove it, and select the most relevant features with LGBM Classifier to find the best params for the model through GridSearchCV.
3. Our final approach for classification was Logistic Regression. It measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic/sigmoid function.

3. **Results**

3.1. **Analysis of MFCCS and Chromagram**

Algorithms	**Chromagram**	**MFCC**
Logistic Regression	20%	47%
Linear_SVC	19%	41%
RBF_SVC	25%	49%
Poly_SVC	18%	45%
KNN (neighbors = 3)	23%	53%





For analyzing the differences between MFCC and Chromagram, we decided to use 3 machine learning algorithms for training and calculating the accuracy of the model. We used Logistic Regression, SVC, and KNN; however, in SVC, we applied three different kernels to do the work, which are Linear SVC, Rbf SVC and Polynomial SVC.

As we can see from the table, MFCC has a better performance than Chromagram. The reason might be due to the roles of Chromagram and MFCC. To be specific, according to charts below, the Chromagram is related to 12 pitch classes. These are very common and widely used in all genres. This causes the training model and prediction for the genres to be very low. However, MFCC is vital and it is related to musical instrument and human voice, which truly relates to the genres. Consequently, the predictions and accuracy are higher than Chromagram.

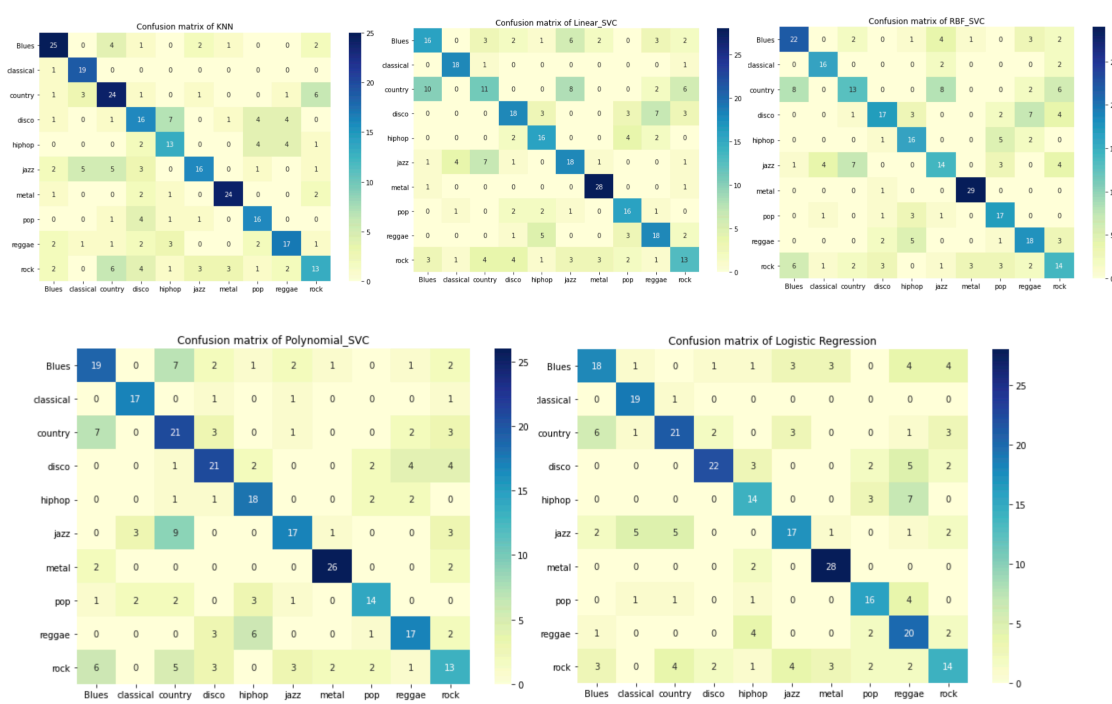
3.2. Analysis the dataset

Models / Algorithms	Training Accuracy	Test Accuracy
Logistic Regression	76%	63%
Linear_SVC	70%	57%
RBF_SVC	79%	59%
Polynomial_SVC	83%	61%
KNN (best k value is 5)	78%	61%

In order to analyze the training model, we used three algorithms (Logistic Regression, SVC, and KNN) and in the case of SVC, we used three different kernels to train the model, which are similar to the one used to analyze the MFCC and Chromagram. According to the table, we could see that the accuracy for 4 models is quite similar beside the Linear_SVC. Based on the score on training accuracy and test accuracy, we believe that the best approach is Polynomial_SVC, which are 83% and 61% respectively. The table also show that the model has the lowest accuracy, evidently, it is the Linear_SVC with 70% in training accuracy and 57% in test accuracy. For the other models, the percentage are quite close to the Polynomial_SVC, with the errors are between 5% - 7%

in training accuracy and less than 2% in test accuracy. Specifically, the KNN models, which we chose the best k value from using GridSeaechCV, has the same testing accuracy with Polynomial_SVC and only 3% difference in training accuracy. Overall, KNN model has the closest accuracy with Polynomial.

Last but not least, the confusion matrix graphs below represent all the matrix of 4 models. Overall, the accuracy for all the models are not really bad; however, as we explained above, Linear_SVC has the lowest accuracy and based on the Linear_SVC graph, we realize that for the genre, Blues, it mistakenly predicted with different Country. This has an impact on the performance of training model using Linear_SVC kernel in SVC algorithm. Besides, the genre Blues was not the only one mistaken in most of the algorithms with Country genre (8 in RBF_SVC or 6 in Logistic Regression), the Country genre was mistaken with Jazz genre and Polynomial_SVC was affected the most. Overall, the accuracy score for all the algorithms we trained the model is pretty good and it reaches up to 83% in Polynomial_SVC algorithm.



4. **Conclusion and Future Work**

In Conclusion, to proceed with this project, we had to create our own dataset by using GTZAN dataset of 1000 labeled music files, and LIBROSA to extract the music features needed for classification. We used different approaches for the classification of music genres. These approaches are K-Nearest Neighbor, SVM, and Logistic Regression. We found that the best approach was Polynomial_SVC kernel in SVC algorithm with accuracy of 83% in training accuracy and 61% in testing accuracy.

For future work, we would like to make the dataset bigger by extracting features from more songs. We would also enhance our classification models to get a better accuracy and implement different options such as Music Emotion Recognition and then develop a Mobile App for music.

5. **References**

- G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," in IEEE Transactions on Speech and Audio Processing, vol. 10, no. 5, pp. 293-302, July 2002, doi: 10.1109/TSA.2002.800560.
- Sanket Doshi; “Music Feature Extraction in Python”. Dec 2018. https://towardsdatascience.com/extract-features-of-music-75a3f9bc265d#:~:text=The%20zero%20crossing%20rate%20is,recognition%20and%20music%20information%20retrieval.
- Deekshith Raya, Vinod Patidar, Yash Khatri, Sai Bhaskar Devatha. “Music Genré Classification”. https://www.saibhaskardevatha.co.in/projects/dsp/
- Antonio Jose Homsi Goulart, Rodrigo Capobianco Guido, Carlos Dias Macie. “Exploring different approaches for music genre classification”. Released: 18 April 2012. https://www.sciencedirect.com/science/article/pii/S1110866512000151?via%3Dihub