**Mini Project Report on**



**Music and Genre classification by AI&ML Models**  


**Submitted in partial fulfillment of the requirement for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted by:**

**Student Name**  **University Roll No.**

**Kanuri Pranay 2019417**

***Under the Mentorship of***

**Dr.Tanupriya Choudhury**

**Professor**



**Department of Computer Science and Engineering**

**Graphic Era (Deemed to be University)**

**Dehradun, Uttarakhand**

**July-2024**



**CANDIDATE’S DECLARATION**

I hereby certify that the work which is being presented in the project report entitled **“Title of the project”** in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineeringof the Graphic Era (Deemed to be University), Dehradun shall be carried out by the under the mentorship of **Mentor Name, Designation**, Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun.

Name    University Roll no1

Kanuri Pranay 2019417

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Description** | **Page No.** |
| Chapter 1 | Introduction | 4 |
| Chapter 2 | Literature Survey | 5-8 |
| Chapter 3 | Methodology | 9-13 |
| Chapter 4 | Result and Discussion | 14-17 |
| Chapter 5 | Conclusion and Future Work | 18-19 |
|  | References | 20 |

**Chapter 1**

**Introduction**

**Overview of Project**

Classifying music by genre involves identifying the type of music, like jazz, classical, or rock, based on its audio characteristics. This is a tough task because finding and using the right audio features to distinguish genres is complex. While there's a lot of music available, tracks that are properly tagged with their genres are relatively rare.

The objective of the genre classification is to analyze the sound signal to forecast the category of music under which it falls. The process automation is good for people who like to find new songs and make their own music catalogs and providers who must tag and structure tunes equally.

In order to build these kinds of systems, one must implement some specific procedures for extracting acoustic representations which would serve as a good proxy for genres that might be of particular interest. This procedure requires the use of some signal processing techniques in order to examine the audio contents and extract some certain characteristics belonging to either time or frequency domains from it. Afterwards they are employed by some machine learning model.

Machine learning, a subset of artificial intelligence, enables systems to learn and improve from experience without explicit programming. By accessing and learning from data, machine learning algorithms can be used to classify and predict the genre of music based on the extracted audio features.

**Chapter 2**

**Literature Survey**

**Music Genre Classification and Recommendation by Using Machine Learning Techniques**

**Authors: Ahmet Elbir ; Hilmi Bilal Çam ; Mehmet Emre Iyican ; Berkay Öztürk ; Nizamettin Aydin, 2018 Innovations in Intelligent Systems and Applications Conference (ASYU)**

Music genre prediction is a prominent area in the field of digital music processing. Specifically, it consists of extracting acoustic features from music using digital signal processing techniques, followed by genre classification and music recommendation through machine learning. Furthermore, convolutional neural networks which belong to deep learning were applied in predicting genres and recommending music in comparison with other methods. This study employs GTZAN database where Support Vector Machine(SVM) method to be exact realized the best accuracy.  
  
  
**Music Genre Classification: A N-Gram Based Musicological Approach Authors: Eve Zheng ; Melody Moh ; Teng-Sheng Moh, 2017 IEEE 7th International Advance Computing Conference (IACC)**

The digitalization of music has deeply integrated into our daily lives, making services like recommendation systems and similarity tests essential for online platforms and marketing. A key component of these services is music genre classification. Traditionally, researchers focused on low-level features of the music, but few have tackled the problem from a more interpretable, musicological perspective. The classification process's intermediate stages are hard to understand through this method. As a result, music professional's domain knowledge is underutilized.

This paper takes a musicological approach to genre classification, considering high-level features with clear musical meanings, making the results more interpretable for music professionals. The research uses a dataset of symbolic piano pieces, with more than two hundred records of classical, jazz and ragtime music. Feature extraction and application an n gram text classification algorithm was carried out. The suggested technique proves its worth by attaining a mean prediction rate greater than 90% at mostreaches98%.

[**Long short-term memory recurrent neural network-based segment features**](https://ieeexplore.ieee.org/document/7918369/)[**for music genre classification**](https://ieeexplore.ieee.org/document/7918369/)

**Authors:** [**Jia Dai**](https://ieeexplore.ieee.org/author/37085608750) **;** [**Shan Liang**](https://ieeexplore.ieee.org/author/38467857900) **;** [**Wei Xue**](https://ieeexplore.ieee.org/author/37086170407) **;** [**Chongjia Ni**](https://ieeexplore.ieee.org/author/37085628462) **;** [**Wenju Liu**](https://ieeexplore.ieee.org/author/37422402500)**,** [**2016 10th**](https://ieeexplore.ieee.org/xpl/conhome/7912121/proceeding)[**International Symposium on Chinese Spoken Language Processing (ISCSLP)**](https://ieeexplore.ieee.org/xpl/conhome/7912121/proceeding)

In the conventional frame feature based music genre classification methods, the audio data is represented by independent frames and the sequential nature of audio is totally ignored. If the sequential knowledge is well modeled and combined, the classification performance can be significantly improved. The long short-term memory (LSTM) recurrent neural network (RNN) which uses a set of special memory cells to model for long-range feature sequence, has been successfully used for many sequence labeling and sequence prediction tasks. In this paper, we propose the LSTM RNN based segment features for music genre classification. The LSTM RNN is used to learn the representation of LSTM frame feature. The segment features are the statistics of frame features in each segment. Furthermore, the LSTM segment feature is combined with the segment representation of initial frame feature to obtain the fusional segment feature. The evaluation on ISMIR database show that the LSTM segment feature performs better than the frame feature. Overall, the fusional segment feature achieves 89.71% classification accuracy, about 4.19% improvement over the baseline model using deep neural network (DNN). This significant improvement shows the effectiveness of the proposed segment feature.

In conventional music genre classification methods based on frame features, audio data is divided into independent frames, ignoring the sequential nature of music. If the sequential knowledge is properly modeled and combined, classification performance can be significantly improved. The long short-term memory (LSTM) recurrent neural network (RNN), which uses special memory cells to model long-range feature sequences, has been successful in many sequence labeling and prediction tasks.  
  
In this paper, we propose using LSTM RNN-based segment features for music genre classification. The LSTM RNN is used to learn the representation of LSTM frame features. Segment features are derived from the statistics of frame features within each segment. Additionally, we combine the LSTM segment feature with the segment representation of the initial frame feature to create a fusional segment feature.  
  
The LSTM segment feature beats the frame feature, according to our tests on the ISMIR database. The fusional segment feature gave us 89.71% accuracy, with proof of an enhancement by 4.19% on the baseline of the deep neural network (DNN) model. This is a significant improvement in the sense that the suggested segment feature approach works. For instance, a segment is within an example whilst a frame represents similar segments. Small segment like-1 consisted of the fewest frames among all other small segments within small segment like-2. Our time may be divided into periods in small segment like-ob, not smaller than small segment like-I and realizing only one division. Specifically, the astrocytes have divisions, correspond to small segment like-2, in the protoplasmic astrocyte family of the striatum, but the small segment like-I astrocytes in the hippocampus do not divide. After a few days of incubation in serum-containing medium

[**Music Genre Recognition Using Residual Neural Networks**](https://ieeexplore.ieee.org/document/8929406/) **Authors;** [**Dipjyoti Bisharad**](https://ieeexplore.ieee.org/author/37086598307) **;** [**Rabul Hussain Laskar**](https://ieeexplore.ieee.org/author/37545711600)[**TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON)**](https://ieeexplore.ieee.org/xpl/conhome/8910516/proceeding)

Genre is an abstract but distinctive feature of music. Traditional automatic genre classification methods compute a set of features from the audio and design a classifier based on them, typically over a long duration of the audio.

In this paper, we propose a residual neural network-based model for genre classification, trained on short clips of just 3 seconds. Traditional genre classification algorithms usually assign a single genre to an audio clip. However, since different genres can have overlapping characteristics, this approach can be limiting.

Acknowledging this ambiguity, our model can assign three genre labels to a music clip, with each genre associated with a probability. The proposed model has an error rate of 18% for predicting the top-1 genre, 9% for the top-2 genres, and 5.5% for the top-3 genres. Our results demonstrate that the predictions made by the classifier align well with the broader, more realistic understanding of musical genres.

**On Combining Diverse Models for Lyrics-Based Music Genre Classification Authors: Caio Luiggy Riyoichi Sawada Ueno ; Diego Furtado Silva, 2019 8th Brazilian Conference on Intelligent Systems (BRACIS)**

In today's digital world, it is necessary to organize and retrieve music automatically. For different uses it is important to label songs with short descriptive texts. The most frequent marking of music recordings is the genre, which enables music platforms to arrange collections by linking pieces and composers who share commonalities.

Lyrics can provide different data source to recognize the type of music. Even though traditional text mining techniques using bag-of-words have been used extensively, modern trends apply deep learning methods in this endeavor. Nevertheless, there is no literature touching on the complementary nature of any of the two strategies. This article investigates different methods of using words to identify musical kinds and shows that straightforward mixtures of these techniques can lead to better lyrics based music category resolution.

**Chapter 3**

**Methodology**

**Dataset Selection**

Correctly selecting a dataset is critical to the success of any machine learning project. A person may usually have various datasets that could be suitable for them but each dataset has its good and bad points. Therefore, the decision on the dataset that is to be used can lead to either a successful project or otherwise. This part of the paper will present some of such options in order to explain why a particular dataset was selected for this study.

**GTZAN Dataset**GTZAN is a dataset, that is known in the Music Information Retrieval (MIR) field, as described in Tzanetakis (2002). It features 10 types of 100 30-second song clips, each representing a different genre, which accumulates to 1000 audio files. The dataset is suitably organized into genre-specific folders, which, in contrast, makes it relatively easy to navigate, and its well-proportioned composition is one of the reasons the dataset is attractive and does not pose that much of a challenge for machine learning projects. However, GTZAN has its drawbacks. Among other datasets used in machine learning, the set is very small- can boast only 100 samples per genre. Deeper learning typically benefits from larger datasets (Ng, 2015), and it allows the classifier to build more robust models because more data gives the classifier to learn more. Subsequently, it was very natural to explore large datasets.

**Data Preprocessing**

**Preliminaries**

Several preprocessing steps were undertaken to prepare the dataset for classification, using Microsoft Excel rather than code. Initially, the 'file.csv' file was obtained from the FMA website (Defferrard et al., 2017b). Entries lacking information in the 'genre\_top' field were removed, as some tracks were labeled with multiple genres rather than a single 'top genre'. This cleanup aimed to focus on clearly classifiable songs.

Afterward, some genres failed to feature in the datasets. Removal of ‘Spoken’ pieces followed for the simple reason that non-musical lyrics were never considered part of this project. Furthermore, we ruled out any songs classified as ‘Experimental’ since this kind covers such broad territory that it’s hard to find alike in sound terms making its categorization very complicated. In the same fashion, ‘Instrumental’ recordings were done away with since all types of music may have versions without lyrics with the end product having minimal similarity in terms of its kind.

### The Standard Dataset

Creating a standard dataset was an easy thing. There were ten genres. In order to have an equal distribution of each different kind of music, undersampling was used. Each about 100 Rock, Hip-hop or Electronic music were used as a source of data along with 100 Folk and 100 Pop because no more could be found; the latter number being the maximum available for that category. Selection was done from 100 songs per category hence balancing the\_datasets."

**Models**

This section will first detail the CatBoost procedure used to train and validate each machine learning model. It will then detail how each of the seven classifiers was implemented, and how hyperparameters were selected, as well as details of the training process.

**Train-Test Split**

A test size of 30% was used in all experiments. When using the standard dataset, the split was performed using Scikit-learn’s train-test split function. For the augmented dataset, the train set and test set had to be imported manually and separately.

**Neural Network**

One of the main features of the project is the sequential Keras model that was selected since it is a simple and configurational model and was very easy to tune, thus it is very convenient for the research, instead of having to concentrate on the one classifier and being able to tune various classifiers.

**Network Structure**

A layer of the neural network was tailored to 58 nodes which signifies the number of features in what we are working on. The output layer, which makes Softmax activation, was formed of a genre with some units. Other network internal architectures were checked. In general, networks with more layers and more units can catch more complex features from the data. Nevertheless, in addition to the use of the regularization techniques (discussed later on), overfitting has been one of common errors. In discovering the best configuration of layers, it was arrived at that a 5-layer structure was the best option. Besides, neurons that are used in the layers one after another are decreasing, which is a standard practice of neural networks (Smith, 1997, ch. 26). The 3 hidden Dense layers and the input layer applied the Rectified Linear Unit (Relu) activation function. This function, also known for its computational efficiency and gradient-related issue handling, is better than tanh(By a significant margin).

**Optimizer**

The Adam optimizer was the one that was demonstrated for its performance in various applications and therefore, it was our choice. Consequently, it aligns well with the Relu activation function.

**Regularization**

To avoid overfitting, the effective measure was to introduce the Dropout layers at the middle of networks. The number of dropouts was experimented and the end result is that 0.3 dropout rate has been found as the most efficient in avoiding overfitting without drifting towards underfitting.

**Training & Testing**

The model was optimized through grid search using different epochs and batch sizes to train and test it. The most optimum value to which the standard dataset achieved the best results was 100 epochs with a batch size of 32. On the other hand, a 100-epoch training regimen coupled with larger batch sizes, around 32, demonstrated the enhanced diagnostic tool of the augmented dataset. Testing involved a step in the process aside from the training process, which was comparing the model's predictions to the true labels of the test data and the accuracy as a percentage score was mentioned. In fact, the model was carefully validated over diverse data conditions in order to confirm the correctness of its results.

**Random Forest Classifier**

The RandomForestClassifier from sklearn.ensemble was utilized because it is easy to implement and tune, just like SVC provided in sklearn's toolkit.

**Number of Estimators**

Just to note, the process of determining the optimum number of tree should be taken into account in the process of specifying the number of estimators (trees) in the forest. On the other hand, the trade-off involves just simulation which made computational time depend on the number of trees. A set of different numbers were evaluated to determine the best trade-off.

**Criterion**

The criterion parameter is the priceless needy attribute to building a decision tree. It is the function that measures the quality of different node parting methods, built on the ICMLDT model. The research was shown as the best comparison between Gini and Info Imp based criterion over different search points.

**Max Features**

This parameter determines how many features will be considered in the splitting of the nodes. It is important to note that different values were tried, with the default setting of 10 the most effective one in this case.

**Max Depth**

The technique of increasing the maximum number of leaves is in principle bound to learning where the purpose is to include deeply specific patterns at the expense of the increasing probability of the estimation errors arising from the process. The increments were necessary in certain cases during grid search to reach the peak performance of the model.

**Training & Testing**

Having a grid search has led to the fact that new best hyperparameters are n\_estimators=1000, criterion='gini', max\_features='auto', max\_depth=100. The above-mentioned settings were kept unchanged while applying both standard and augmented datasets.

**Gradient Boosting Machine (XGBoost)**

Many machine learning algorithms have been developed for various tasks and XGboost is the most commonly used algorithm for the latest machine learning algorithm. In the case of the data we are analyzing here, the XGBClassifier proved to be the best.

**Min Child Weight**

The min child weight refers to the ‘Minimum sum of instance weight (hessian) needed in a child’ (XGBoost, 2019a). Higher values of ‘min child weight’ will result in models that are less complex, so as a parameter it can be used to prevent overfitting. Several values were tested.

**Learning Rate**

What learning rate essentially does is to govern the share of input values that would be used for individual trees in the boosting model and has a say in both the velocity of learning and the degree of overfitting. Experiments were carried out using many different combinations of such values between 0 and 1 in the process of hyperparameter tuning.

**Max Depth**

Consequently, the max depth parameter constitutes constraints to the boosting process as it determines the number of layers the tree has. The use of higher values will increase model complexity, thereby increasing learning capacity and the likelihood of overfitting.

**Training & Testing**

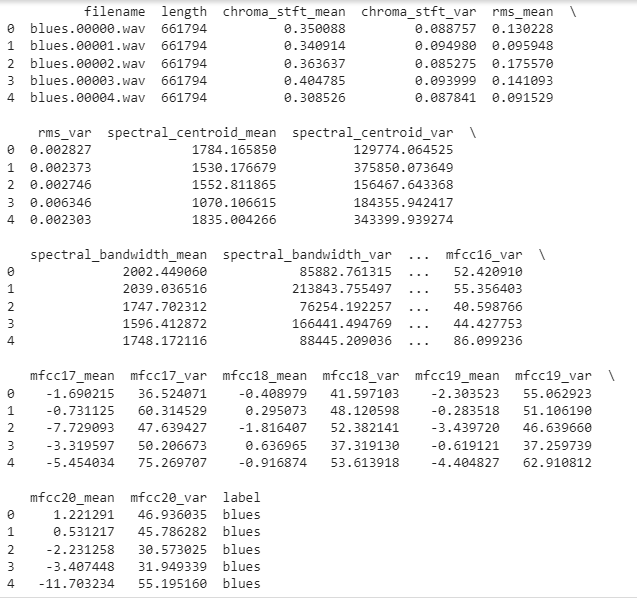
The best hyperparameters setting for both datasets were: min\_child\_weight=5, learning\_rate=0.05, max\_depth=10. Let me confirm these values for both datasets were the best one with respect to the model complexity and pattern recognition across various datasets.

**Chapter 4**

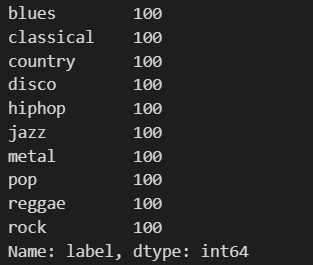
**Result and Discussion**

**Results**

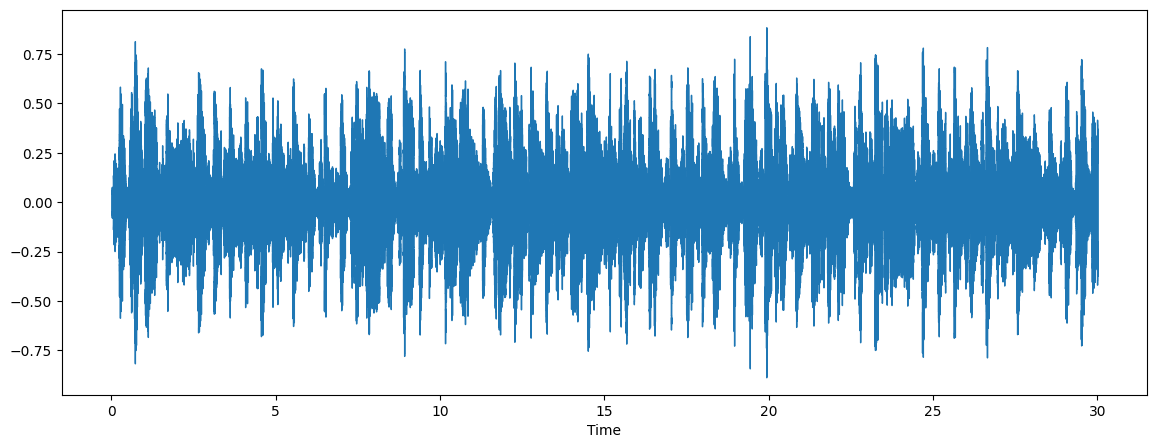
Given below are the results of the project

All of the fields present in the dataset are given below  


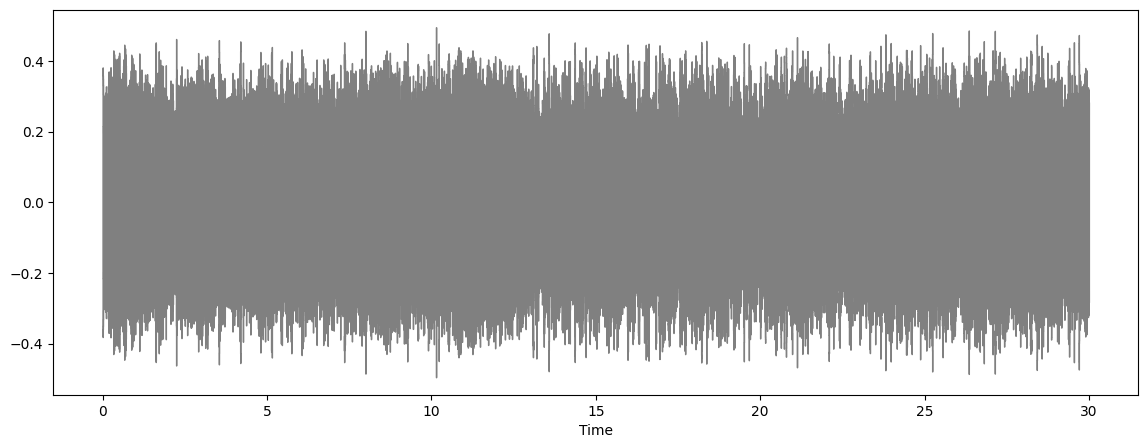
Number of samples present in each genres are given below

****

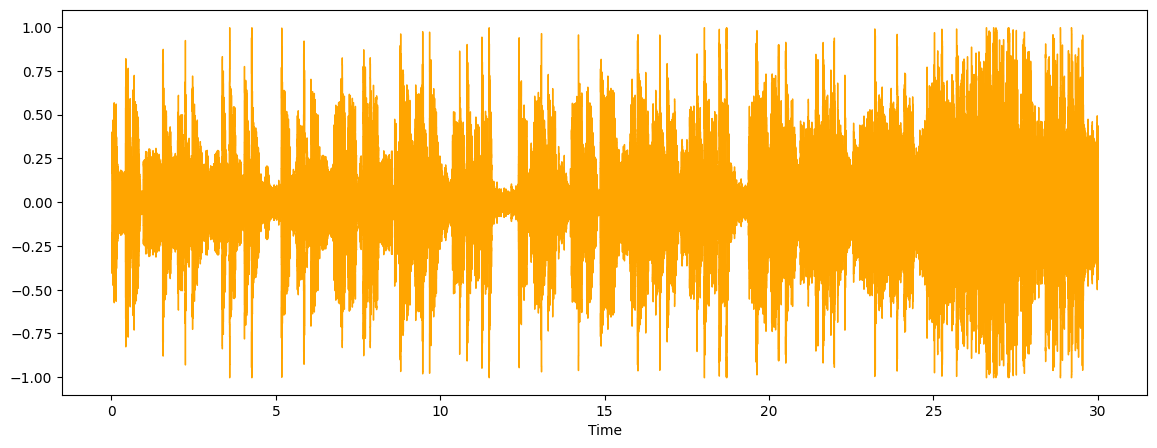
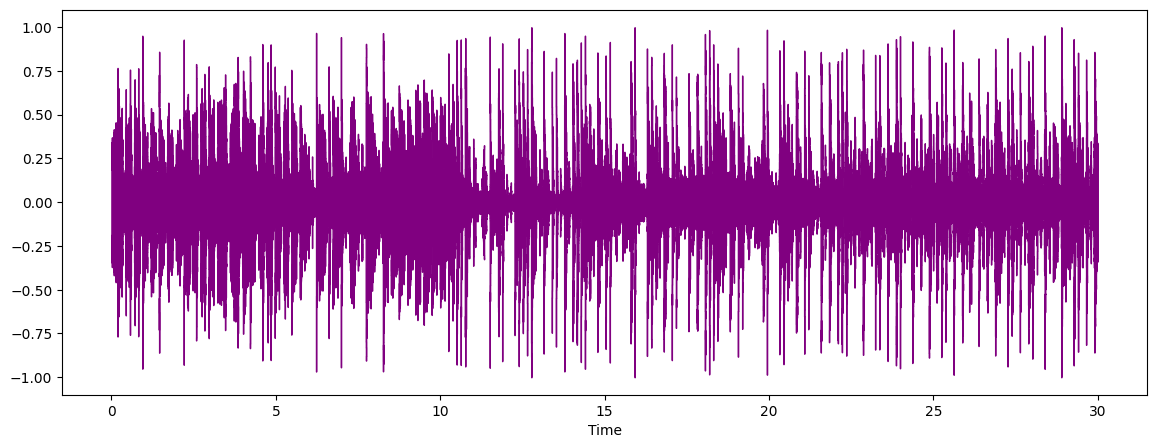
Given below of the wave graphs of the different genres

****

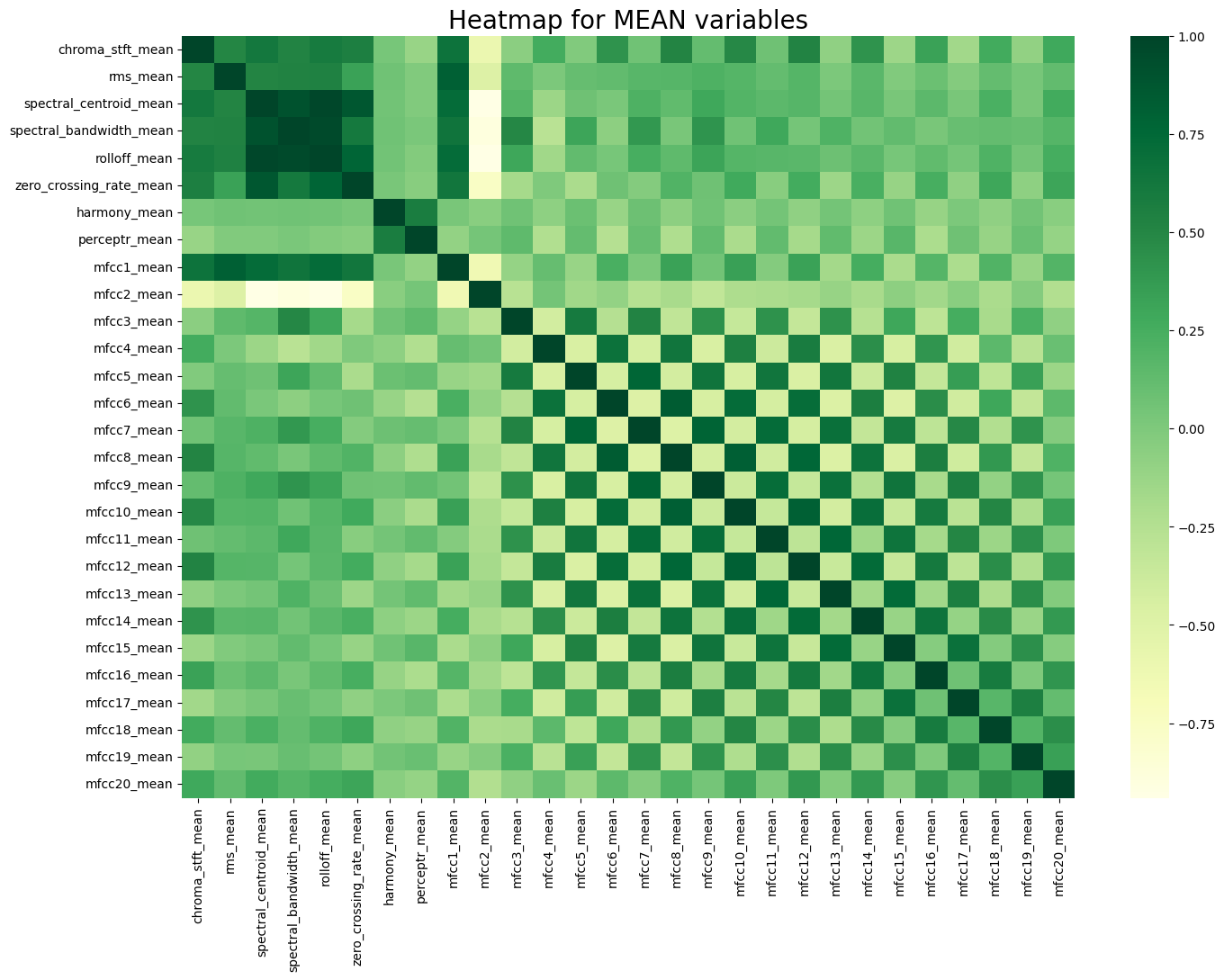
Genre: Blue

****

Genre: Metal

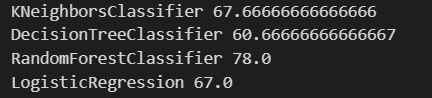
Genre: PopGenre: Hiphop

Given below is the heatmap of the mean variables of the dataset



Given below are the of the different models namely

KNeighnors Classifier, Decision Tree Classifier, Random Forest Classifier, Logistic Regression, Cat Boost Classifier, Extreme Gradient Boost (XGD) Classifier

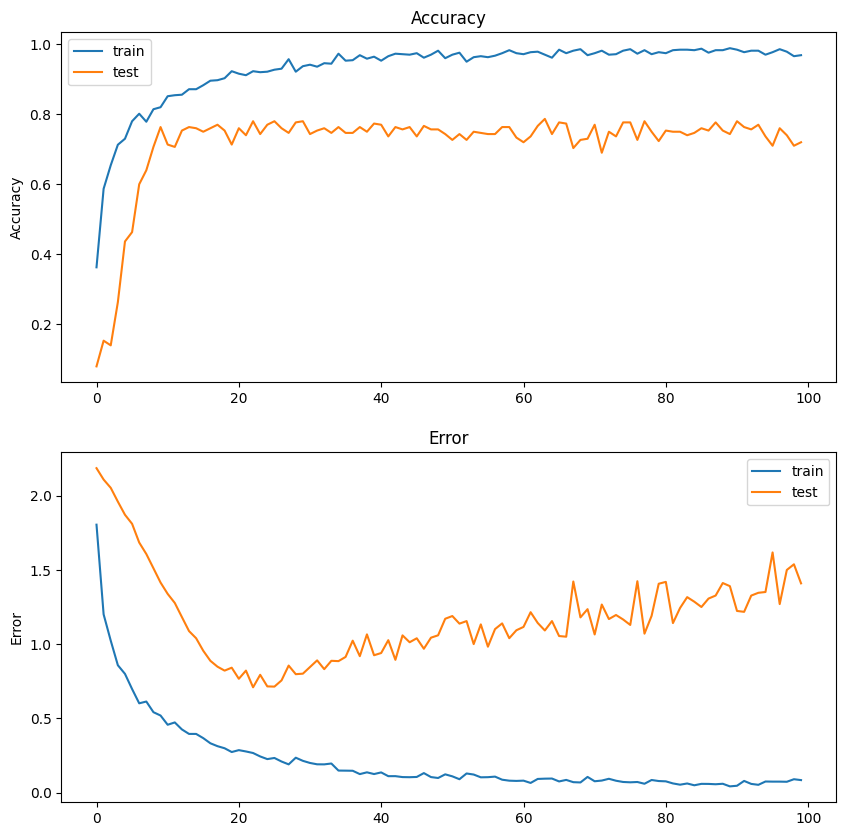




Given below is the test accuracy of Neural Network Model with 100 epochs



Given is the line-plot of accuracy and error of the train and test models



**Chapter 5**

**Conclusion and Future Work**

We can derive certain conclusions by odds of their accuracy among various models presented:

1. Cat Boost Classifier: This classifier with 83.33% accuracy is listed on the top. Cat Boost algorithm is an algorithm for handling categorical data effectively in a given efficient way. Quite on the contrary, models such as tree-based ones a—you can go from worse hyperparameters to better ones with virtually any manipulation of the tree.

2. Random Forest Classifier: The Random Forest classifier which had an accuracy of 78% demonstrated, in addition to efficiency, that it was a robust and good model. And as the ensemble model is flexible enough to yield better-than-maximum results without being prey to minor oscillations, such as local maxima and minima, that is the reason for this model is still effective. Further, the ensemble technique is successful in the reduction of overfitting over the single decision tree and so on.

3. XG Boost Classifier: Thereafter, the XG Boost model also outperformed the Random Forest with a bit of the accuracy, i.e., 78.33%. XG Boost is a faster gradient boosting algorithm used to high-scale processing and performance. Consequently, this makes XG Boost a usual high-ranking algorithm for the implementation of the entire systems.

4. Neural Network: The Neural Network potentially got through with a 76.67% accuracy. Given that the parameters that you can be more likely to play with is the number of neurons, the higher closer one gets to the true error, so they can get quite better with both more data and hyperparameter focusing.

5. Logistic Regression and K-Nearest Neighbors: These two were given a 67% and 67.67% score respectively In the area they were considered for. Their USP is their simplicity and transparency. Probably the main limitation thereof, is that they may not be so capable to capture the complexity of the data as the other models of higher precision would do.

6. Decision Tree Classifier: Among all classifiers, the Decision Tree classifier proved to be the worst with an accuracy rate of 60.67%. This scenario often happens when the model is overfitting the training data as the number of samples becomes less. However, this model is not represented in this graph and thus we cannot be sure of this statement and thus we will have to remove it.

**Future Work:**

1. Hyperparameter Tuning: Not only can the random forest be tested, but also other models like neural networks and XG Boost. Moreover, in each model, the best performance state should be set as a target to be achieved.

2. Feature Engineering: Indulge in a more profound feature extraction by simply dividing signals into short time periods and then computing the spectrogram density. As shown in the example, no background noise is added to the artificial pitch therefore there are no mistakes such as miss-classifications.

3. Data Augmentation: For miking, I would like to meet with anyone to get more data and then use an AI-based system to produce new data there or ground, it may be either really time-intensive or even pointless if it does not make sense in the end.

4. Ensemble Methods: The combination of various methodologies cannot be overlooked, and the most flexible way of upscaling data is to pile a bunch of different methods (e.g. 50 methods out of the 199) and delegate them to a second layer which is called a consolidator.

5. Deep Learning Models: Experiments can take various forms depending on what data is dealt with - almost endless opportunities may come. This is not supposed to be what exactly the experiments are going to look like and therefore we have at the very beginning contemplated implementing the following ideas and finally, the decision has been made to go with the AAB pattern.

6. Cross-Validation: In order to cross validate the forecast, it is mandatory to send transaction data checkpoints which are then sorted time-wise. We should not expect that the entire dataset will be used for the validation process but rather some of its parts will be separated for training and the model will be tested with the other parts.

**References**

[1] Music Genre Classification Using Independent Recurrent Neural Network, Wenli Wu ; Fang Han ; Guangxiao Song ; Zhijie Wang, 2018 Chinese Automation Congress (CAC)

[2] Music Genre Classification and Recommendation by Using Machine Learning Techniques, Ahmet Elbir ; Hilmi Bilal Çam ; Mehmet Emre Iyican ; Berkay Öztürk ; Nizamettin Aydin, 2018 Innovations in Intelligent Systems and Applications Conference (ASYU)

[3] Improve Music Genre Classification with Conv Nets, Rafael L. Aguiar ; Yandre M.G. Costa ; Carlos N. Silla, 2018 International Joint Conference on Neural Networks (IJCNN)

[4] Genre Classification using Word Embeddings and Deep Learning, Akshi Kumar ; Arjun Rajpal ; Dushyant Rathore, 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)

[5] Music Genre Classification: A N-Gram Based Musicological Approach, Eve Zheng ; Melody Moh ; Teng-Sheng Moh, 2017 IEEE 7th International Advance Computing Conference (IACC)

[6] Long short-term memory recurrent neural network based segment features for music genre classification, Jia Dai ; Shan Liang ; Wei Xue ; Chongjia Ni ; Wenju Liu, 2016 10th International Symposium on Chinese Spoken Language Processing (ISCSLP)

[7] Music Genre Recognition Using Residual Neural Networks, Dipjyoti Bisharad ; Rabul Hussain Laskar

TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON)

[8] Parallel Recurrent Convolutional Neural Networks Based Music Genre Classification Method for Mobile Devices, Rui Yang ; Lin Feng ; Huibing Wang ; Jianing Yao ; Sen Luo IEEE AccessYear: 2020

[9] On Combining Diverse Models for Lyrics-Based Music Genre Classification ,Caio Luiggy Riyoichi Sawada Ueno ; Diego Furtado Silva, 2019 8th Brazilian Conference on Intelligent Systems (BRACIS)

[10] Automatic Classification of Music Genre Using Masked Conditional Neural Networks, Fady Medhat ; David Chesmore ; John Robinson, 2017 IEEE International Conference on Data Mining(ICDM)