Classification Of Music Genre

Shuvro Ahmed
dept. of CSE
BRAC University
Dhaka, Bangladesh
ID: 18201119
shuvro.ahmed@g.bracu.ac.bd

Shaima Afrin
dept. of CSE
BRAC University
Dhaka, Bangladesh
ID: 19101641
shaima.afrin@g.bracu.ac.bd

***Abstract*—Due to the abundance of digital music on the web and the most recent developments in artificial intelligence, automatic music genre classification has gained popularity in recent years.In this paper our main motive is to apply machine learning methods in Python to classify songs into genres from audio data(Rock or hip-hop) .To classify songs at first we scaled our train and test data .Than,we train different algorithm like decision trees ,logistic regression, gaussian naive bayes, support vector, random forest neural network classifier on our data .After analyzing the result we found that our project has an accuracy level of about 93% on test data through Neural Network Classifier .Though others algorithm like decision trees ,logistic regression, gaussian naive bayes, support vector, random forest we got below 93% accuracy level on our test data .**

***Index Terms*—Classification model, PCA, cross-validation, visualization , Normalization**

I. Introduction

The goal of our work is to be able to classify songs as Rock or Hip-Hop with the help of the audio data available.In doing so, we do the following: merge the data finding correlation and principal component analysis visualization of the data use ML algorithms like decision trees ,logistic regression, gaussian naive bayes, support vector, random forest neural network classifier . Nearly 18000 distinct songs' audio data are utilized. Two files hold the necessary data: a CSV file with the track's basic information and genre, and a JSON file with musical characteristics like danceability, acousticness, energy, instrumentalness, liveness, speechiness etc.

A. Research Objective

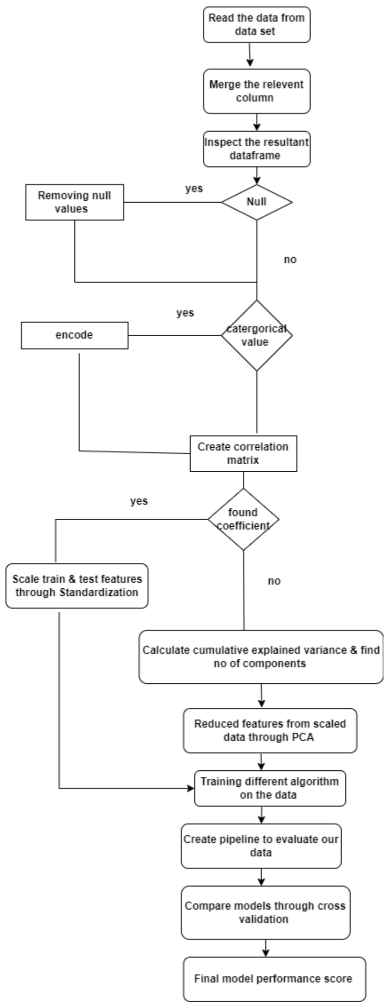
The majority of individuals now listen to their favourite music mostly through streaming services with large collections that have emerged over the past several years.However, the vast volume of music available can make it difficult for consumers to find more recent music that appeals to their tastes.Because of this, streaming providers have investigated methods for classifying music to enable customized recommendations.Without ever having listened to a single song, our objective is to analyze this dataset and categorize songs as either "Hip-Hop" or "Rock."

B. Literature Review

The majority of contemporary type classification algorithms, such as KNN, SVM, etc., involve machine learning techniques, as stated in [1]. In this paper, the GT-ZAN music dataset is presented. A convolution neural network is utilized for training and classification. The intended framework divides music into many genres by extracting the feature vector. According to our research, the project's accuracy levels for training and testing are about 97% and 74%, respectively. This will considerably stimulate and increase the classification of musical genres. In [2] it presents the PRCNN framework (2017), which parallelizes CNN and bi-directional GRU to extract both spatial and temporal data from music spectrograms.Additionally, it developed and tested model using a larger dataset (FMA) that included 8,252 pieces of music from 17 different genres.On a carefully selected dataset of 15 songs, it further verified the model. On the FMA dataset, the model obtains an overall accuracy of 88% with accuracy levels above 90% in four genre groups.

II. Methodology

A. Workflow Diagram



B. Description of Workflow Diagram

In the dataset, each track's musical qualities, such as its danceability and acousticness, are rated on a scale of -1 to 1. These are available in two separate files in the CSV and JSON formats. While JSON is another widely used file format, databases frequently return the results of a particular query in CSV, which is a popular file format for tabular data.We start by creating two pandas DataFrames out of these files that we can merge so we have features and labels. After merging the data we inspect the resultant dataframe to check all the info null values of our dataframes .We can also check the null value of the category through x.is null().sum() function .If we found null value we can remove null values by dropping the column or rows .As we found no null values so we move to the next step which is creating correlation matrix(corr_metrics = echo_tracks.corr()) .Due to the size of our datasets, using fewer characteristics can significantly shorten the computation time. So to keep the model simple and improve interpretability we want to avoid using variables that have strong correlations with each other . We couldn't find any strong correlation among variables. We are importing train_test_split() function to split our data. We are scaling our train test features through standardization by importing StandardScaler such that all features have a mean = 0 and standard deviation = 1. Now we are going to get our explained variance ratio from PCA using all features of train data & visualizing the data using bar plot .Visualization of PCA does not appear to be a clear elbow in this scree plot, which means it is not straightforward to find the number of intrinsic dimensions using this method.Now we are calculating cumulative explained variance to reduce feature .so that we can perform PCA on that reduced data .We are training decision tree and logistic regression on the data and compare among their classification report . As we can see our data is not balanced (around 4000 for hip-hop others for rock) so we are balancing our data & redefining the train and test set with the pca_projection from the balanced data. Training different algorithms like gaussian naive bayes, support vector, random forest neural network classifier along with decision tree and logistic regression on our balanced data and comparing among the models . Now we are creating pipeline to evaluate our data . At last we are training our model using k-fold CV to get a good sense of how well our models are actually performing .Finally the result shows that we get 93% accuracy through neural network classifier on our test data .

C. Data Description

Here, we have two type of datasets imported from Kaggle which are echonest-metrics in json format and fma-rock-vs-hiphop in csv format. In tracks, we have 17734 entries and 21 columns among which there are 13 columns with categorical values. While in echonest-metrics we have 13129 entries and 9 columns among which there are no categorical values.

So, we have surface level knowledge about our tracks along with track metrics compiled by The Echo Nest.A song is more than just its name, artist, and listen count. In a different dataset, each track's musical qualities, such as its danceability and acousticness, are rated on a scale of -1 to 1. These are available in two separate files in the CSV and JSON formats.

D. Algorithm Description

The algorithms implemented here are PCA, logistic regression, decision trees, Naive Bayes Classifier, Support Vector Classifier, Ensemble Classifier (Random Forest), eural Network Classifier.

To get the greatest results, it might be very helpful to streamline our models and use the fewest possible elements. We can instead utilize a popular strategy known as principal component analysis to minimize the number of features because we didn't identify any particularly strong relationships between our features (PCA). It's feasible that only a small subset of the dataset's attributes may adequately account for the variation between genres. We may assess the relative contributions of each feature in our data to the variance across classes using PCA, which spins the data along the axis with the highest variance. However, because PCA rotates the data based on the absolute variance of a feature, a feature with a wider range of values will overwhelm and bias the algorithm.

A data item is classified into one of two or more categories using decision trees, which are rule-based classifiers that take into account features and follow a ”tree structure” of binary decisions. Decision trees not only make it simple to use and understand, but also let us see the ’logic flowchart’ that the model creates from the training data.

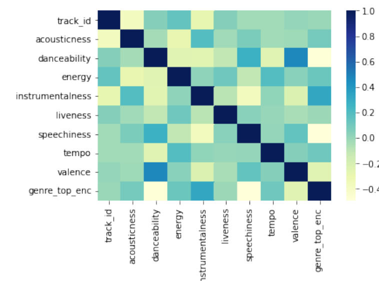
Logistic regression makes use of what’s called the logistic function to calculate the odds that a given data point belongs to a given class.

Rest of the algorithms such as Neural network or Random forest are part of classification algorithms.

E. Data Pre-Processing

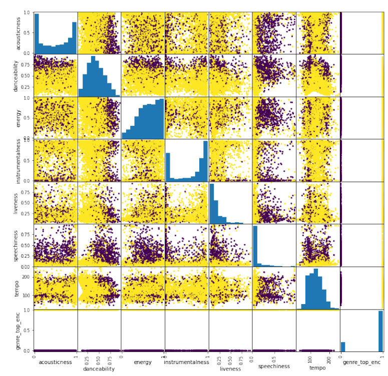
In order to have features and labels (sometimes also referred to as X and y) for the classification later on, we first split these files into two pandas DataFrames that we may merge.
Then we Explored correlations in our dataset using pandas corr function.
As was previously said, it can be especially helpful to streamline our models and employ the least number of features required to provide the greatest results. We can now divide our data into an array containing our characteristics and another containing the labels — the genre of the audio — because we didn’t uncover any especially strong relationships between our features.
Then we normalized the data using standard scaler
We can use PCA to see how much we can reduce the dimensionality of our data now that we have preprocessed it. To determine the amount of components to use in additional studies, we can use scree-plots and cumulative explained ratio plots. In descending sequence of variance, scree-plots show the number of components versus the variation explained by each component. Scree-plots provide us a better understanding of which factors adequately account for the variance in our data. On order to select an appropriate cutoff when utilizing scree plots, the ’elbow’ (a steep drop from one data point to the next) in the plot is often employed.

F. Data Visualization

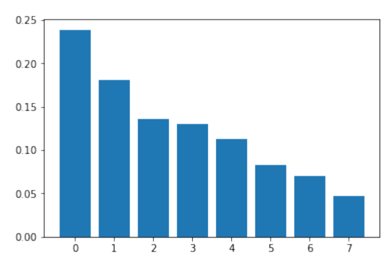


Correlation Matrix
Here, we can see that there are no strong correlation between any features. Hence, we didn’t find any particularly strong correlations between our features and we will use PCA to reduce number of features.

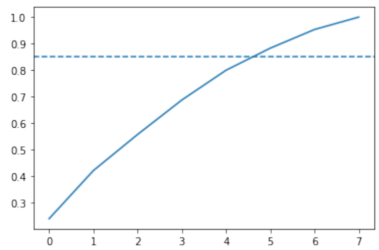
‘



Scatter Matrix
Here, datas can’t be seperated visually. So, no linearity is visible. Hnece, we can come to the conclusion that non linear models such as decision tree are applicable here.

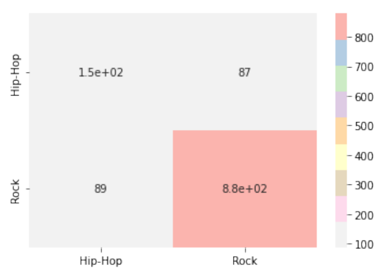


scree-plots
Scree-plots display the number of components against the variance explained by each component, sorted in descending order of variance. Scree-plots help us get a better sense of which components explain a sufficient amount of variance in our data. When using scree plots, an ’elbow’ which is a steep drop from one data point to the next in the plot is typically used to decide on an appropriate cutoff.



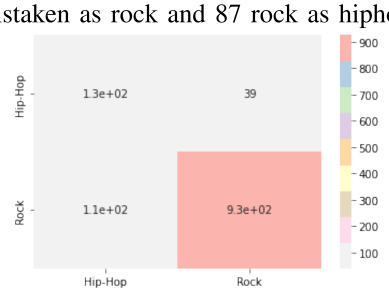
cumulative explained variance plot

To find out how many features are needed to explain, say, 85% of the variance, we can also examine the cumulative explained variance plot. In this case, the cutoffs are fairly arbitrary and typically determined by rules of thumb. We can do PCA with that many components once we've established the right amount, hopefully bringing down the dimensionality of our data.



Confusion matrix for decision tree

Here, overall result is quite good as only 87 hip hop are mistaken as rock and 87 rock as hiphop.



Confusion matrix for Logistic regression

Here, overall result is quite good for hip hop as only 39 hip hop are mistaken as rock but for rock the result is not up to the mark

G. Model Implementation

First, we implemented only decision tree and logistic regression then compared both. But it seems there is a huge bias of rock datas when compared. Thus, we reduced datas of rock based on amount of hip hop rows available. After that we implemented Naive Bayes Classifier, support vector classifier and etcetras.
Later we cross validation to evaluate out models.

III. Result Analysis

We must build pipelines to scale our data, apply PCA, and instantiate our preferred model before we can run cross-validation. Since how our data is divided into train and test sets might have an impact on how well the model performs, CV makes an effort to split the data in a variety of ways and test the model on each split. Although there are many various CV systems, each with its own benefits and drawbacks, we will utilize the K-fold CV in this instance. K-fold divides the data first into K distinct subgroups of equal size. The remaining data is then used as train sets while repeatedly employing each subset as a test set. The results from each fold can then be combined to determine the final model performance.

A. Analysis

Here, decision tree, Logistic Regression, Naive Bias,SVC, Random Forest and Neural network Classifier gives us accuracy for rock respectively 0.84,0.94,0.94,0.93,0.93,0.93 and for hiphop it is 0.90,0.92,0.91,0.93,0.91,0.93.

For weighted average in both case, we get around 0.89-0.94

B. Cross validation

After cross validation we get, Decision Tree: 0.8989010989010989 Logistic Regression: 0.9170329670329671 Naive Bias: 0.9186813186813187 Support Vector: 0.9214285714285715 Random Forest: 0.9241758241758242 Neural Network Classifier: 0.9274725274725275. K fold is given 10 as to get 10 test subset for each validation process.

IV. Conclusion

In this project, we propose multiple architectures for music genre classification, based on digital signal processing techniques and machine learning methods. All architectures give decent results with more than 89% whole accuracy.By modelling the data as a network of layers, where each layer holds more abstract (and occasionally semantic) information about the data, neural networks can learn rich representations of the data.They outperform all other classifiers because they can estimate functions that are naturally extremely nonlinear.

References

[1] Lakshman Kumar Puppala, Siva Sankar Reddy Muvva, Sudarshan Reddy Chinige, P.Selvi Rajendran,(2021),"A Novel Music Genre Classification Using Convolutional Neural Network"

[2] Hui Yuan, Wenjia Zheng, Yun Song, Yijun Zhao,(2021),"Parallel Deep Neural Networks for Musical Genre Classification: A Case Study"