GENRE IDENTIFICATION

From Classical To Hip-Hop: Can Machines Learn Genres?

Dimple Kochar
Sucheta Ravikanti
Arunabh Ghosh

INTRODUCTION

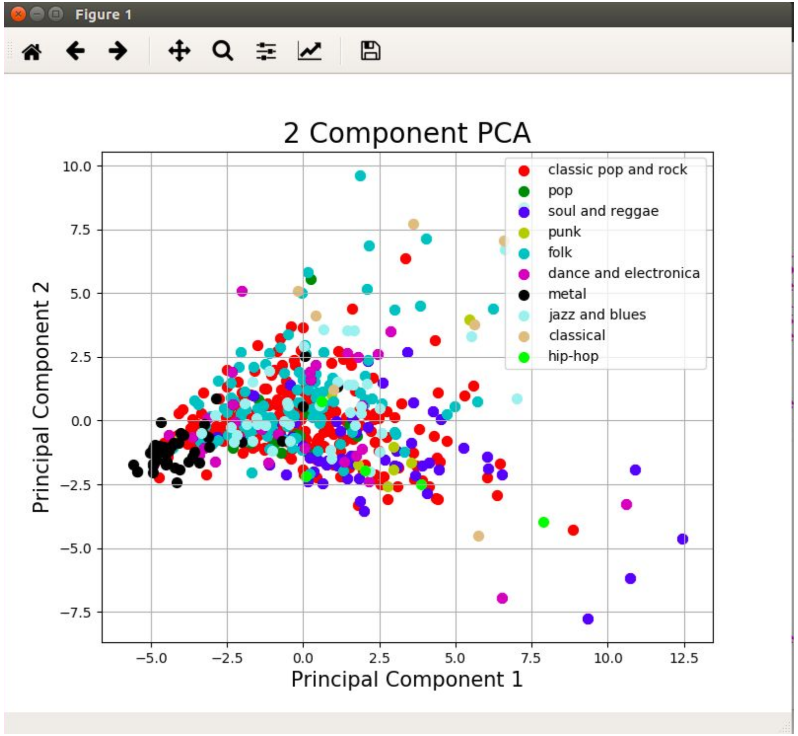
- Identifying which genre a song belongs to has been one thing at which humans are particularly good at. In just a few seconds we can tell whether we're listening to Classical music, Rap or EDM.
- We aim to develop a machine learning algorithm which automatically classifies a song into its correct genre.
- Our team explored several of these techniques over the course of this project and successfully built a system that achieves 56% accuracy in classifying music genres.
- We will now discuss the methods we used for exploratory data analysis, feature selection, hyperparameter optimization, and eventual implementation of several algorithms for classification.

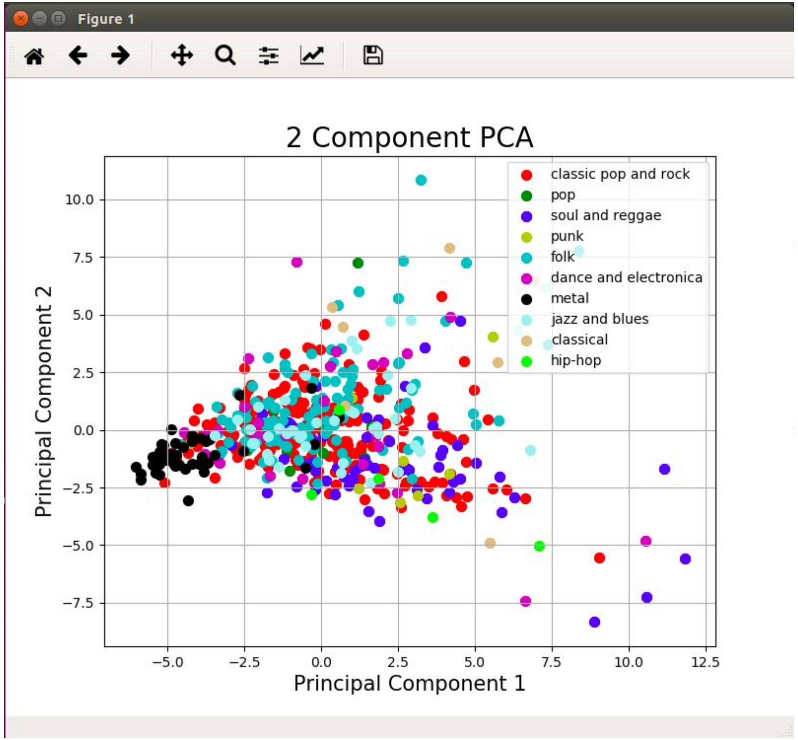
Exploratory Data Analysis

- Research shows that when a person is listening to a piece of music, the brain's attention is heightened during transitions, which frequently coincide with the loudest parts of the music.
- This indicates that the loudest part of the song might have important information regarding the genre of the dataset.
- So we decided to include the timbre feature corresponding to the loudest parts of the song.
- Including that feature, played a vital role and and increased our accuracy from 50% to 52%. With further tuning we could increase it up to 56%.

Why timbre?

- It is that quality of the sound which allows our ear to distinguish two sounds which have the same pitch and loudness. It is the perceived sound quality of a musical note.
- Timbre is mainly determined by the harmonic content and envelope of the sound.
- The characteristic sound of each instrument is determined by the correct combination of the timbres they produce.
- Envelope and attack time play an important role too in characteristing a particular instrument.
- Sound of the electric guitar is a characteristic of rock music. Similarly saxophones are typically used in Jazz have their own characteristic timbre.





Plot of data before and after including the max timbre feature. The plot on the right does seem to have slightly better clustering of classes.

Classification

- After completing preliminary exploratory data analysis, we shifted our focus to various supervised learning methods, turning first to tree-based methods.
- A classification tree is “grown” by successively splitting the predictor space at a single cut point which leads to the greatest possible reduction in some chosen measure.
- For our project, we chose as our measure the Gini index (G),which is often thought of as a measure of region purity:

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$

Classification

- One single classification tree might not have extraordinary predictive accuracy in comparison to other machine learning methods, aggregation of decision trees can substantially improve performance.
- For out project we chose a random forest of classification trees as our predictor.
- We first used Bayesian optimization to tune several hyperparameters.
- The results of this optimization indicated that the **optimal minimum number of samples required to split an internal node in a tree is 12**, the **optimal minimum number of samples that must be in a newly created leaf is 1**, and the **optimal number of trees in the forest is 196**.
- The random forest achieved accuracy of **56%** and a F1 score of 50.65%.

Tuning the hyperparameters

- There are certain parameters specific to an algorithm that cannot be found from training the data set. But, the values of those parameters should be set by us either randomly or use training algorithms different from the ones used for data. These parameters are known as **Hyperparameters.**
- Eg : Length of the decision tree, thresholds used for the leaves, hidden layers in NN etc.
- One method by which we can choose the best hyperparameters is the grid search. Where we iterate over every possible value of the combination of hyperparameters and compare the results.
- Bayesian Optimization is a better way of tuning the hyperparameters as grid search fails in the case of a large number of hyperparameters and dimensions.

Bayesian Optimization

- Bayesian Optimization assumes a prior belief on the behavior of the hyperparameters and searches the space of the hyperparameters based on the belief and it is updated after every step.
- It uses the previously computed points to predict a posterior expectation of the loss.
- It then samples the loss at some new point such that the utility of expectation is maximised.
- The last step is done to ensure that the expected improvement in the loss is large and the prior is updated based on the present values of hyperparameters. For the prior distribution, we assume that f is a gaussian process.

ACQUISITION FUNCTION :

$$EI(\mathbf{x}) = \mathbb{E}\left[\max\left\{0, f(\mathbf{x}) - f(\hat{\mathbf{x}})\right\}\right]$$

Results

Here's how we performed on all the genres of music

Class	Precision	Recall	F1-Score
Classic pop and rock	0.53	0.90	0.67
Folk	1.00	0.17	0.29
Dance and electronica	0.55	0.47	0.51
Classical	1.00	0.88	0.93
Hip-Hop	0.27	0.23	0.25
Soul and reggae	0.50	0.44	0.47
Punk	1.00	0.17	0.29
Metal	0.70	0.45	0.55
Jazz and blues	0.37	0.69	0.48

Reflection on the performance :

- After the training and testing phases, we observed that some genres got clubbed with other genres or the classification was poorly done because of the confusion.
- Classical did the best when compared to other genres and this can be attributed to the distinctive timbre different from most genres.
- Some genres like Punk was confused with rock and classical pop due to the lesser training examples used for this genre. But again, they make an extensive use of electric guitar and have similar number of beats/bar ratio.
- Classical pop and folk were confused. The likely reason can be due to the use of acoustic instruments in the mentioned genres.