Music Genre Classification

Abhinav Khanna
Student Researcher
akhanna@princeton.edu

Michael Yitayew
Student Researcher
myitayew@princeton.edu

Abstract

Music genres classification is the classification of music into categorical descriptions called genres. Genre classification is traditionally done manually but the need to categorize the increasing music content on the web calls for automatic classification methods. In this assignment, we build several genre classifiers and analyze their performance on 1000 music files from 10 different genres, all from the standard GTZAN dataset. We select the musical features that are most discriminant for genre classification as a feature set and use Fisher vector representation to encode time varying values of these features into a per song input vector. In our analysis, we find that Neural Nets achieve the highest classification accuracy while Blues and Rock are the most commonly misclassified music genres.

1 Introduction

Music genre classification is a challenging problem as many songs can be classified under multiple genres. Furthermore, a song is a complex audio wave that must be translated into a digital vector before being classified. This transformation process takes a continuous wave and condenses it down to a vector set, creating some information loss inherent in the problem. This work builds on the work of Tzanetakis and Cook by doing a deeper analysis of different classifiers using a similar set of features, analyze the effects of including different features in the classification task, and dissect which genres lead to the highest misclassification. Through our experiments, we discovered that Blues and Rock are the hardest for our classifiers to isolate from other genres, and thus we report results that both include Blues and Rock as well as exclude those two labels and data subsets. With Blues and Rock, the highest classification accuracy we were able to achieve was 74%. Without the Blues and Rock data subset, we were able to achieve 84% accuracy.

2 Related Work

The dataset of song files with we use is part of the GTZAN dataset, part of a standard challenge to build accurate genre classifiers. The first work in automatic genre classification was from 1997 by Dannenberg et. al[2] followed by Tzanetakis and Cook's novel approach of using multiple musical features to achieve an almost 60% accuracy[3]. Top performers on the GTZAN dataset include Bergsta et al's Adaboost classifiers[5] and Compressive sampling techniques by Chang et al[6] which are especially computationally efficient. Recent results borrow ideas from multivariate analysis(non-negative matrix factorization and computer vision to achieve 90% accuracy. Our aim, in contrast, was to achieve a high classification accuracy with the feature set at hand using known classifiers, and to identify which genres are most often misclassified and understand why.

3 Methods

3.1 Description of data and data processing

The songs in our study are 30 second clips, and each song was provided as a set of frame-level features for each 20 millisecond frame in the song. As noted in [4], frame level features do not encode temporal variation, thus Fisher vectors and exemplars were used to aggregate temporal features into a per song feature vector. We performed manual feature selection by observing the discriminant strength of subsets of features; we found MFCC(Mel-Frequency Cepstrum Coefficients) to be the most discriminant feature for genre classification. The following musical features were part of our final feature set; **MFCC, Keystrength, HCDF, Zero-cross, Chroma, Roughness, Brightness**. We combined these features through vertical stacking before processing through the Fisher Vector Kernel.

3.2 Classification Methods

We used the following 9 different classifiers; eight are from the SciKitLearn Python Libraries and the Neural Net is from Matlab.

1. *K-nearest-neighbors(KNN)*
2. *Logistic Regression with l_2 penalty(LR)* using the one-vs-rest scheme
3. *PCA followed by Logistic Regression (PCALR)*
4. *Random Forests Classifier (RF)*
5. *PCA followed by Random Forests classifier (PCARF)*
6. *Support Vector Machine (SVM)* with an exponential kernel function(rbf)
7. *Naive Bayes classifier(NB)* using the Multinomial Naive Bayes algorithm
8. *AdaBoost* with Random Forests as Weak Learners
9. *NNet Neural Network* from Matlab - A feed forward network with a single hidden sigmoid layer containing 50 neurons and an output softmax layer.

Decision Trees were not included as Random Forest builds on the Decision Tree data structure, making checking both seem redundant. We also combined PCA with LR and RFs to observe the effect of unsupervised dimensiona reduction on classifiers. It can be noted in the results table below, dimensiona reduction does not always improve classification accuracy.

3.3 Evalulation

We performed stratified 10-fold cross-validation for each classifier. In this procedure, input is divided into 10 random equal size folds. The classifier is then trained on 9 of the folds(i.e 90% train) and tested on a single fold(i.e 10% test). The accuracy of a classifier is average of the accuracy values achieved when each fold is used as a testing fold. In addition, we looked at confusion matrices to understand how the error is distributed across labels for each genre category.

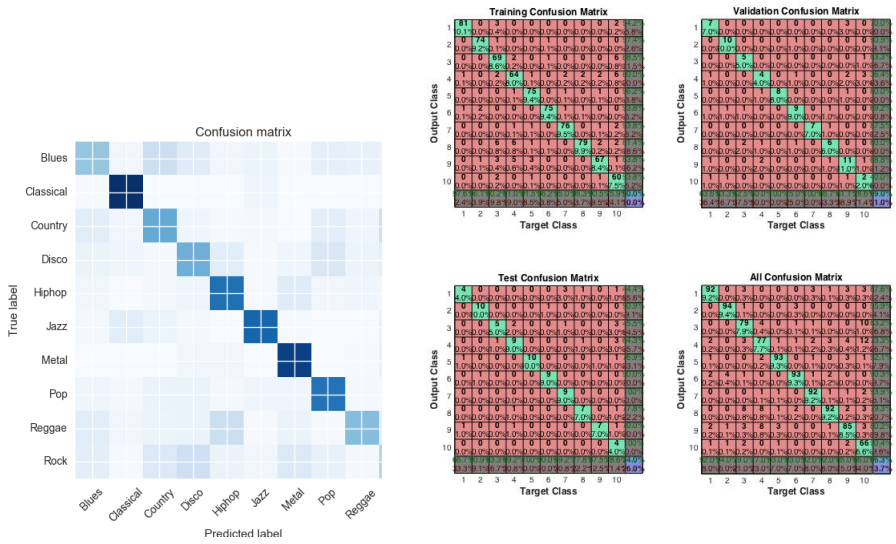
4 Results

We tested the classifiers against a data set containing all ten class labels as well as a data set that did not include Blues and Rock. The following error-rates were achieved at the classifiers' optimal parameter settings against the complete data set. Optimal parameter settings were found by plotting the error-rate vs parameter graph and finding the minimum. Performance was variable across the classifiers with Neural-Nets, Random Forest and AdaBoost with Random Forests performing the best.

Classification accuracy		
Classifier	**Error-rate %**	**Error-rate % w/o Blues & Rock**
KNN	42.6	32.2
LR	41.1	32.0
PCALR	39.4	25.2
RF	**35.8**	**25.3**
PCARF	39.3	**29.1**
SVM	46.6	40.1
NB	45.3	40.9
Ada	**35.4**	**24.7**
NNet	**26.5**	**16.1**

In order to better understand what the distribution of misclassifications looked like, we took the top classifiers, and created confusion matrices of their classifications. As you can see in figure 1, the confusion matrix for Random Forest Classifier suggests that Blues and Rock both are often mislabeled as other classes, while the rest of the genres appear to be far more accurately classified. This trend reappeared for AdaBoost, and Neural-Net based classifiers. In order to verify this, we reran the top classifiers with Blues and Rock removed. The right column in the above table showcases

Figure 1: Confusion Matrix for Random Forest and Neural Network - With Blues and Rock



the accuracy rates on the data subset that does not contain Blues and Rock data. Without Blues and Rock labels, the error rates decrease, and in the best case scenario the feed forward neural network decreases by almost 10%. This seems to agree with our observation that Blues and Rock are heavily misclassified, and that without them in the data set, the other classification labels are relatively discriminate.

4.1 Parameter Optimization

For the purposes of brevity, we will only talk about the parameter optimization for the top 3 classifiers, the feed forward neural network, the random forest classifier, and the AdaBoost classifier. We focused on optimizing the random forest for two of its parameters: max depth and max learners. In the end, values of 110 and 31 were used for max learners and max depth respectively. The final values for the AdaBoost classifier's hyperparameters were 10 for the number of estimators and 0.01 for the learning rate. The feed forward neural network was tried with the following hidden layer sizes: 10, 30, 50, 75, and 100. After trying these values, it was discovered that post 50, the error rate does not decrease significantly but the time needed for the net to converge grows dramatically.

Figure 2: Error vs Learners and Error vs Max Depth

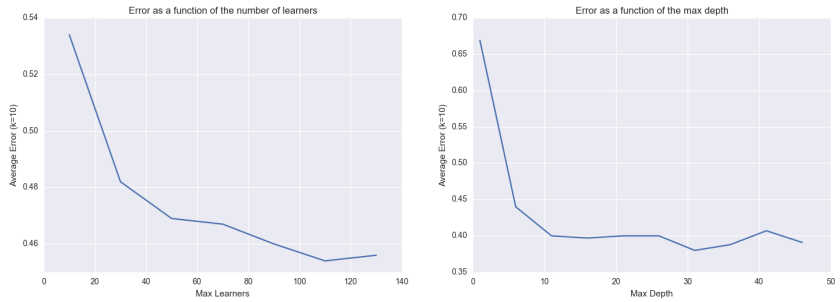
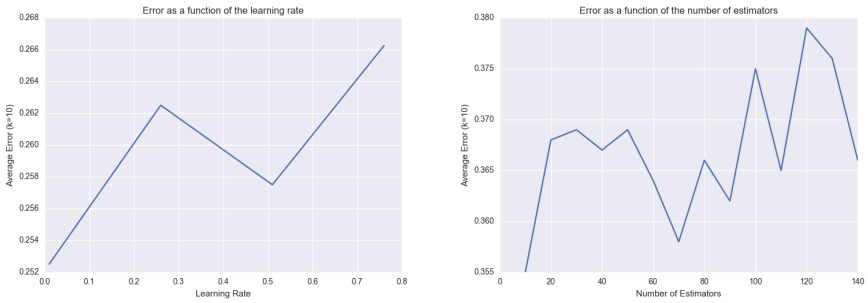


Figure 3: Error vs Learning Rate and Error vs Max Depth



5 Discussion and Conclusion

In this work, we compared 10 classification methods to predict the music genre for a particular 30 second clip after converting the audio signal to a digital vector representation of the features we listed above. After performing manual feature selection and picking the most discriminatory features, we also compared the classification methods with and without Rock and Blues data subset entries from the GTZAN data set. Considering the confusion matrices and the classification error rates of the different classifiers with and without the Rock and Blues data subsets, the feed forward neural network performs the best on the data including Blues and Rock subsets, and the data excluding Blues and Rock. Because Blues and Rock form the foundation for many of the musical genres that come afterwards (like Disco and Pop), it makes sense that the classifiers often misclassify these two genres. Including more Frame-level music features in the Fisher Vector creation process helped yield higher accuracy rates for the neural networks. From a human analysis perspective, it makes sense that the features that pertain specifically to musical elements in the song help further discriminate genres beyond just what MFCC can provide.

There are a number of directions we could go to improve this work. Seeing as the Neural Network performed the best on this data set, and that the performance of neural networks increases with the increase in training data, gathering more training data could immensely improve accuracy. Furthermore, different types of neural networks may do a better job at capturing the differences between these classes than the standard feed forward neural network. Recurrent Neural Networks and other custom neural networks may provide increased accuracy and are worth trying. Focusing on better feature extraction, it has been mentioned in literature that instrument details about a given song can be greatly beneficial to genre classification. Adding in instrument recognition to our Fisher Vectors may allow further distinction between these classes. However, because Blues and Rock originate from similar roots and form the basis for many of the other genres, instrument details may not help us resolve the issue of mislabeling Blues and Rock. The real question that remains is how do we discriminate Blues and Rock from the other genres more effectively, and if we can find a feature that allows for this discrimination then we can greatly decrease the difficulty of our task.

Acknowledgments

We would like to acknowledge Princeton University for offering the wonderful COS 424 class that provided us with the opportunity to take on this research. We would also like to thank our Professor, Professor Englehart, and her TA team for providing us with guidance and the tools necessary to make this project happen.

References

[1] - Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, et al. (2011) Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12: 28252830

[2] - Roger B. Dannenberg, Belinda Thom, and David Watson. A machine learning approach to musical style recognition. In Proc. International Computer Music Conference, pages 344 347, 1997.

[3] - George Tzanetakis and Perry Cook. Musical genre classification of audio signals. Speech and Audio Processing, IEEE transactions on, 10(5):293302, 2002.

[4] - T. Brundage, G. Gliner, Z. Jin, K. Wolf (2016) Music Genre Classification

[5] - James Bergstra, Norman Casagrande, Dumitru Erhan, Douglas Eck, and Balzs Kgl. Aggregate features and adaboost for music classification. Machine Learning, 65(2-3):473484, 2006.

[6] - Kaichun K. Chang, Jyh shing Roger Jang, and Costas S. Iliopoulos. Iliopoulos: music genre classification via compressive sampling. In Proceedings of the 11th International Conference on Music Information Retrieval (ISMIR, pages 387392, 2010)