Comparing Classical Pattern-Recognition Methods for Music Genre Classification

Esomchukwu Too-Chiobi
Department of Computer Science and Engineering
Texas A&M University
esomtoochiobi@tamu.edu

Abstract

This project explores classical pattern recognition methods applied to music genre classification using audio features extracted from the GTZAN dataset. We implement and evaluate statistical (LDA, QDA) and function-approximation-based classifiers (SVM with linear and RBF kernels) using 5-fold stratified cross-validation. Preliminary results show that even with a simple feature set, classical models achieve accuracies well above chance, with the RBF-kernel SVM performing best. Additional analysis includes PCA visualization of feature separability and fold-level performance statistics.

1 Introduction

Music genre classification is a longstanding task in music information retrieval, involving the assignment of audio samples to predefined genre categories. Despite the dominance of deep learning in recent years, classical pattern recognition methods remain valuable for their speed, interpretability, and alignment with theoretical learning frameworks.

In this project, we investigate several classical models in the context of multiclass music genre classification. The work is motivated by both pedagogical alignment with lectures in pattern recognition and personal research interests in audio classification. Our goals are to evaluate performance, interpret model behavior, and visualize feature-space structure using established tools from pattern recognition.

2 Dataset and Feature Extraction

We use the GTZAN Genre Collection, a benchmark dataset consisting of 1,000 audio clips (30s each) evenly distributed across 10 genre labels: blues, classical, country, disco, hip-hop, jazz, metal, pop, reggae, and rock. We downloaded the data via Kaggle and used the "genres_original" directory.

To convert raw audio to usable feature vectors, we used the `librosa` library in Python. From each audio file, we extracted the following:

- **MFCC (Mel-Frequency Cepstral Coefficients)**: 13-coefficient mean and standard deviation
- **Chroma Features**: averaged across time
- **Spectral Contrast**: mean contrast values per frequency band
- **Zero-Crossing Rate (ZCR)**: mean value
- **Tempo**: estimated beats per minute

These were aggregated into a fixed-length vector per clip. The dataset was standardized and cleaned to yield 999 usable samples. Features were saved to `features/features.csv`.

3 Methods

We implemented and compared four classifiers:

- **Linear Discriminant Analysis (LDA)**: Assumes Gaussian class-conditional densities with shared covariances.
- **Quadratic Discriminant Analysis (QDA)**: Allows individual covariance matrices per class.
- **Support Vector Machine (SVM - Linear)**: Margin-based linear classifier.
- **Support Vector Machine (SVM - RBF)**: Nonlinear kernel classifier using the radial basis function.

All models were implemented in `scikit-learn`. Each was wrapped in a pipeline that included standardization via `StandardScaler`. Evaluation was performed using `StratifiedKFold` with 5 splits and accuracy as the scoring metric.

We extended basic evaluation by computing:

- Mean and standard deviation of accuracy across folds
- Total number of correct and incorrect predictions
- Raw fold-by-fold accuracy table
- PCA visualization of the standardized feature matrix

3.1 Code Organization

Code was modularized into:

- `extract_features.py` for feature generation
- `train_models.py` for training, evaluation, and visualization

The repository is publicly hosted at: `https://github.com/yourusername/649-project`

4 Preliminary Results

4.1 Classifier Performance Summary

Model	Mean Acc.	Std Dev	Correct	Incorrect	Total
LDA	0.4455	0.0161	445	554	999
QDA	0.4775	0.0194	477	522	999
SVM (Linear)	0.4615	0.0184	461	538	999
SVM (RBF)	**0.4925**	**0.0078**	492	507	999

Table 1: Summary of classifier accuracy and prediction counts

4.2 Fold-by-Fold Accuracy

Fold	LDA	QDA	SVM (Linear)	SVM (RBF)
1	0.4500	0.4450	0.4800	0.4950
2	0.4600	0.4850	0.4600	0.5000
3	0.4150	0.4750	0.4300	0.4800
4	0.4450	0.5050	0.4800	0.5000
5	0.4573	0.4774	0.4573	0.4874

Table 2: Accuracy per fold across classifiers

4.3 PCA Visualization

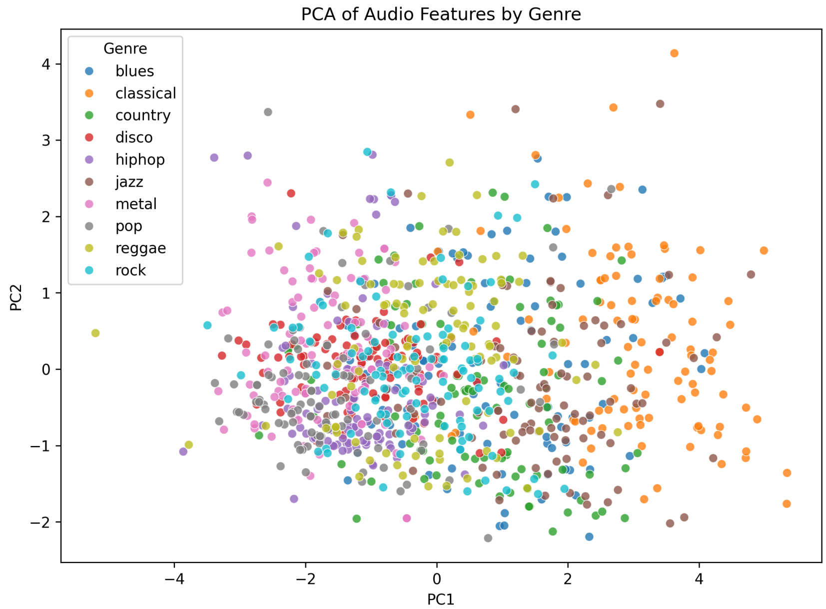

Figure 1: PCA projection of audio features colored by genre

The PCA plot shows moderate genre separability. Genres like classical and metal form more distinct clusters, whereas others such as pop, rock, and hip-hop overlap significantly. This supports the relatively low accuracies of linear models and highlights the utility of nonlinear classifiers like SVM-RBF.

5 Next Steps

We plan to:

- Add more temporal features (e.g., MFCC deltas)
- Visualize confusion matrices to understand genre-level misclassifications
- Explore dimensionality reduction using LDA
- Tune SVM hyperparameters using grid search
- Expand report with related work, final analysis, and insights

References

[1] Ho Kin Pou, J., Rao, H.K., Bhambhani, G., Joseph, J., & Prakash, S.P.J. (2023). *Music Genre Classification using Machine Learning*. In Proceedings of the 4th International Conference on Communication and Computational Technologies (ICCCT 2023). https://doi.org/10.1007/978-981-99-2056-1_22

[2] Gourisaria, M.K., Agrawal, R., Sahni, M., & Singh, P.K. (2024). *Comparative Analysis of Audio Classification with MFCC and STFT Features Using Machine Learning Techniques*. International Journal of Information Technology. https://doi.org/10.1007/s41870-024-01501-w