Assignment Project 2: Audio Classification Using Machine Learning

Report by:
Sanjaya Gunawardena
Liyang Fang

Introduction

Audio classification is a fundamental task in various applications such as music genre classification, speech recognition, and environmental sound classification. In this project, we aim to develop and evaluate different machine learning classifiers to classify audio samples into various genres using a dataset containing audio features.

Objective

The objective of this assignment is to:
- Load and preprocess the provided audio dataset.
- Build and evaluate multiple machine learning classifiers.
- Compare the performance of these classifiers.
- Analyze and visualize the results to determine the best performing model.

About the dataset:

You have been provided with a CSV file features_30_sec.csv which contains the following:
Columns:

- The first column is the filename.
- The last column is the label (genre of the audio).
- The remaining columns are various features extracted from the audio files.

1.1 Data Loading and preprocessing

- We loaded the dataset using Pandas to understand its structure.
- Feature Extraction: We extracted the features and labels from the dataset and encoded the labels using LabelEncoder from scikit-learn.
- Train-Test Split: We split the data into training and testing sets (80% training, 20% testing) using train_test_split.
- We standardized the features using `StandardScaler`.Standardization is a preprocessing step in data preparation that involves scaling the features of a dataset so that they have a mean of zero and a standard deviation of one. This is particularly important for machine learning algorithms that rely on the distance

between data points, such as K-Nearest Neighbors (K-NN) and Support Vector Machines (SVM).

Model building and Evaluation

K-Nearest Neighbors (K-NN)

We initialized the K-NN classifier with n_neighbors=5, trained it, and evaluated its performance.

Random Forest

We initialized the Random Forest classifier with `n_estimators=100`, trained it, evaluated its performance, and visualized feature importances.

Support Vector Machine (SVM)

We initialized the SVM classifier with kernel='linear', trained it, and evaluated its performance.

Gradient Boosting

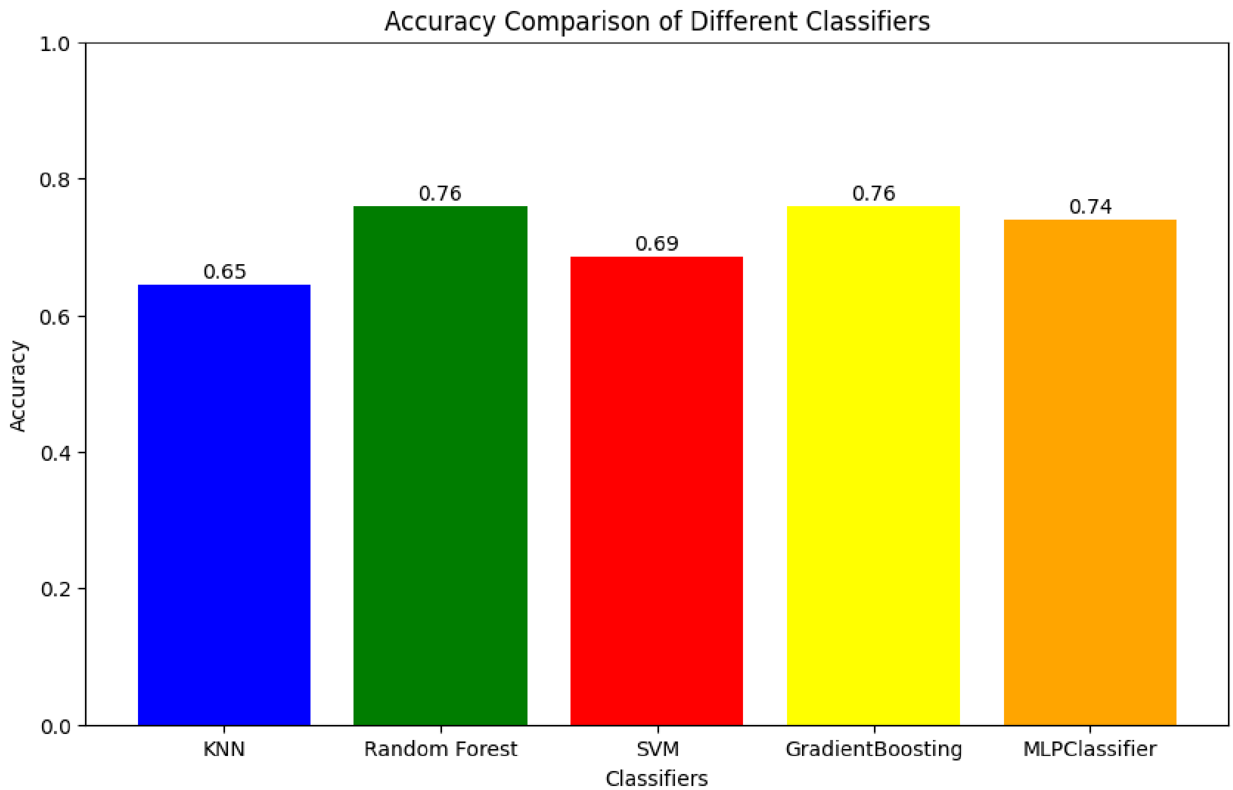
We initialized the Gradient Boosting classifier with n_estimators=100, trained it, evaluated its performance, and visualized feature importances.

Multi-layer Perceptron (MLP)

We initialized the MLP classifier with hidden_layer_sizes=(100,) and max_iter=300, trained it, and evaluated its performance.

Evaluation

We compared the accuracies of the different classifiers using a bar plot. We printed the confusion matrix and classification reports for the models.



Based on the bar plot we can see random forest and gradient boosting classifiers have the same accuracies which is 0.76.

The Random Forest (RF) and Gradient Boosting (GB) classifiers have the same accuracy based on the classification reports; it suggests that both models perform similarly on the dataset in terms of overall accuracy. However, it's important to look beyond just the accuracy metric to get a more comprehensive understanding of each model's performance. Here are some key points to consider:

Precision, Recall, and F1-Score

- **Precision**: The proportion of positive identifications that were actually correct. High precision indicates that the classifier has a low false positive rate.
- **Recall**: The proportion of actual positives that were identified correctly. High recall indicates that the classifier has a low false negative rate.
- **F1-Score**: The harmonic mean of precision and recall, providing a single metric that balances both concerns.

Analyze the performance for each class (genre in this case) individually, as some models may perform better on certain classes even if their overall accuracy is similar.

Random Forest Classification Report:

	precision	recall	f1-score	support
0	0.67	0.70	0.68	20
1	1.00	1.00	1.00	13
2	0.79	0.70	0.75	27
3	0.71	0.71	0.71	21
4	0.57	0.87	0.68	15
5	0.87	0.91	0.89	22
6	0.86	0.96	0.91	25
7	0.89	0.62	0.73	13
8	0.72	0.57	0.63	23
9	0.65	0.62	0.63	21
accuracy			0.76	200
macro avg	0.77	0.77	0.76	200
weighted avg	0.77	0.76	0.76	200

Gradient Boosting Classification Report:

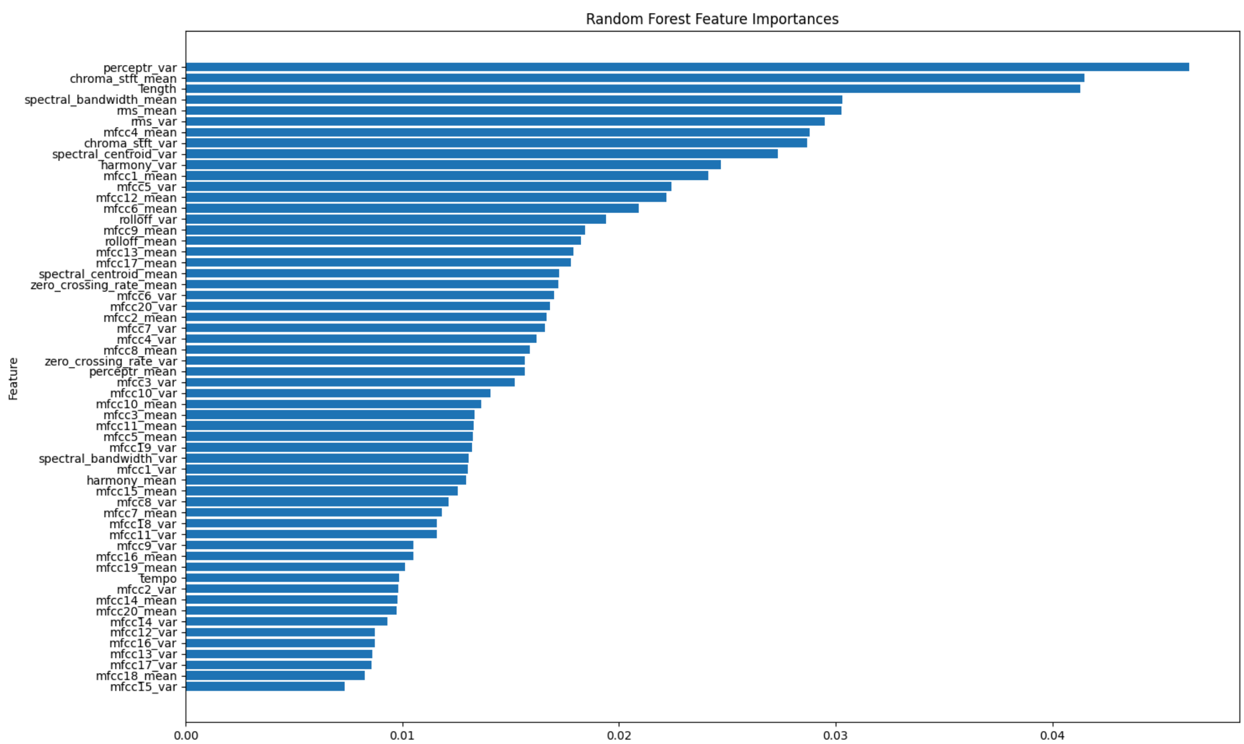
	precision	recall	f1-score	support
0	0.71	0.85	0.77	20
1	1.00	1.00	1.00	13
2	0.70	0.59	0.64	27
3	0.80	0.76	0.78	21
4	0.60	0.80	0.69	15
5	1.00	0.91	0.95	22
6	0.92	0.92	0.92	25
7	0.83	0.77	0.80	13
8	0.57	0.57	0.57	23
9	0.60	0.57	0.59	21
accuracy			0.76	200
macro avg	0.77	0.77	0.77	200
weighted avg	0.77	0.76	0.76	200

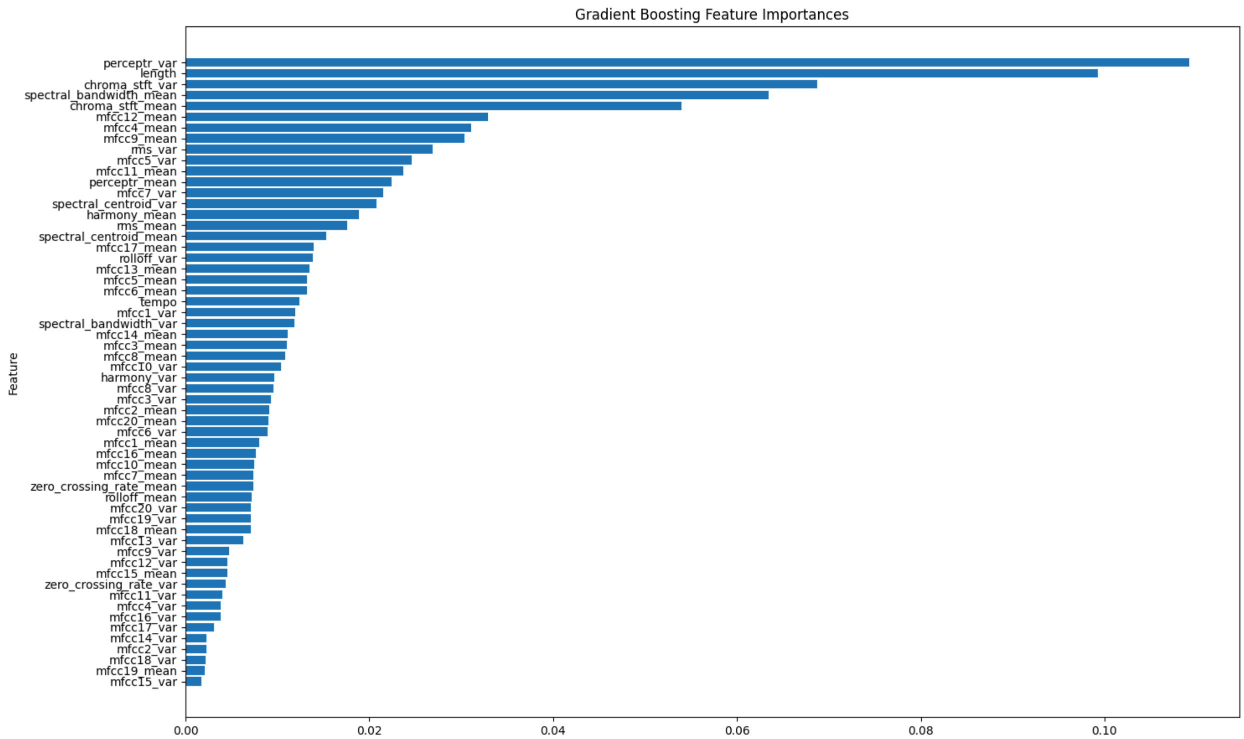
1. **Precision and Recall**:
    - Both classifiers have similar macro and weighted average precision and recall scores.
    - Class 0: GB has higher precision and recall than RF.

    - Class 2: RF performs better in terms of precision and F1-score, but GB has a better recall.
    - Class 4: GB has better precision and recall than RF.
    - Class 8: RF has better precision and F1-score, but the same recall as GB.
2. **Consistent Performance**:
    - Both models perform perfectly on class 1 (100% precision and recall).
    - Classes like 5 and 6 have high precision and recall for both models.
3. **Class-Specific Differences**:
    - RF has higher precision for classes 2 and 8, while GB has higher recall for these classes.
    - GB generally shows higher precision and recall for class 0 and class 4.

Feature Importances:

For models like RF and GB, it's beneficial to look at feature importances to understand which features contribute most to the prediction. This can help in interpreting the model and potentially improving it.





Comparative Analysis

1. **Common Important Features**:
    - Both classifiers identify `perceptr_var`, `chroma_stft_mean`, `spectral_bandwidth_mean`, and `length` as critical features, indicating these features are consistently valuable across different models.
2. **Differences in Importance Distribution**:
    - The Gradient Boosting model places relatively higher importance on the `length` feature compared to the Random Forest model.
    - Some MFCC features are ranked slightly differently in importance between the two models, though both consider them significant overall.
3. **Model Interpretation**:
    - **Random Forest**: The importance scores are more evenly distributed among the top features. This could imply that the Random Forest model leverages a more diverse set of features for making predictions.
    - **Gradient Boosting**: The importance scores for the top features are more pronounced, suggesting that Gradient Boosting may rely more heavily on a few key features to improve predictive accuracy.

Conclusion

Both models highlight the importance of spectral, perceptual, and MFCC features in audio genre classification. The slight differences in the ranking and distribution of feature importance scores provide insights into how each model processes the features to make predictions.

For practical purposes, if interpretability and training speed are crucial, Random Forest might be preferred. If marginally better performance and handling of complex patterns are more important, Gradient Boosting could be the better choice.

In this scenario, since both models have equal performance metrics, we can consider the computational efficiency and interpretability as deciding factors. Thus, **Random Forest** can be highlighted as the classifier with the best balance of performance and practical utility for this audio classification task.