Comparative Analysis of Machine Learning Models for Spotify Genre Classification

Term Project (Data Science II)

Group 8
Anvita Yerramsetty
Austin Bell
Carter Prince
Robera Abajobir
Sanghyun An
Tyler Varma

November 25, 2025

Abstract

This project evaluates the performance of six distinct machine learning algorithms in predicting the musical genre of Spotify tracks based on tabular audio features and metadata. Using a dataset of approximately 47,000 songs across 24 genres, a preprocessing pipeline was implemented to address class imbalance and feature scaling. The models evaluated include Logistic Regression, K-Nearest Neighbors, Gaussian Naive Bayes, Random Forest, XGBoost, and a Multilayer Perceptron. Results indicate that ensemble tree-based methods yield the highest predictive performance, with XGBoost achieving an accuracy of 54.37%. This performance notably exceeds the random baseline of 4.1%, suggesting that while audio summary features contain predictive signal, the complexity of genre boundaries limits the effectiveness of tabular classification methods.

Contents

1 Problem Statement

Music genre classification is a fundamental task in Music Information Retrieval (MIR), essential for the functionality of streaming platforms such as Spotify and Apple Music. Accurate classification supports recommendation systems, playlist generation, and user personalization.

The classification task presents challenges due to the subjective nature of musical genres (e.g., the fluid boundary between "Indie" and "Alternative") and the difficulty of representing temporal audio signals as static numerical features. The objective of this project is to develop a multi-class classification model capable of predicting one of 24 distinct genres given a set of 14 high-level audio features (e.g., danceability, energy, tempo) and metadata.

1.1 Project Organization

To ensure a rigorous and fair comparison, the project adopted a modular collaborative structure. A centralized preprocessing pipeline was developed to generate a single, version-controlled dataset used by all team members. Each of the six group members took ownership of one specific modeling technique, performing independent hyperparameter tuning. Results were exported to a standardized JSON schema, allowing for the programmatic aggregation of metrics and the generation of comparative visualizations presented in this report.

2 Collection and Description of Datasets

The dataset utilized is `SpotifyFeatures.csv`, containing tracks retrieved via the Spotify Web API.

2.1 Dataset Characteristics

- **Total Samples:** 46,985 (post-cleaning).
- **Classes:** 24 unique genres, including *Alternative, Anime, Classical, Country, Electronic, Hip-Hop, Jazz, Pop, Rap, Reggaeton, Rock,* and others.
- **Input Features:** 14 total features.
  - **Audio Metrics (Continuous):** `acousticness`, `danceability`, `energy`, `instrumentalness`, `liveness`, `loudness`, `speechiness`, `tempo`, `valence`.
  - **Metadata (Continuous):** `popularity`, `duration_ms`.
  - **Musical Structure (Categorical):** `time_signature`, `key`, `mode`.

3 Data Preprocessing Techniques Applied

A standardized preprocessing pipeline was implemented using Python to ensure data quality and consistent evaluation.

3.1 Data Cleaning and Engineering

Initial inspection revealed tracks associated with multiple genres. To establish a single-label classification problem, tracks with duplicate IDs across different genre labels were removed. Additionally, the genre "Rap" was merged into "Hip-Hop" due to high similarity in feature space, and "A Capella" was excluded due to insufficient sample size.

3.2 Hybrid Balancing Strategy

To address class imbalance, a hybrid resampling strategy was applied. Majority classes were undersampled to a maximum of 2,000 samples, while minority classes were retained in their entirety. As illustrated in Figure 1, this approach reduced bias toward dominant classes while preserving information from minority classes.

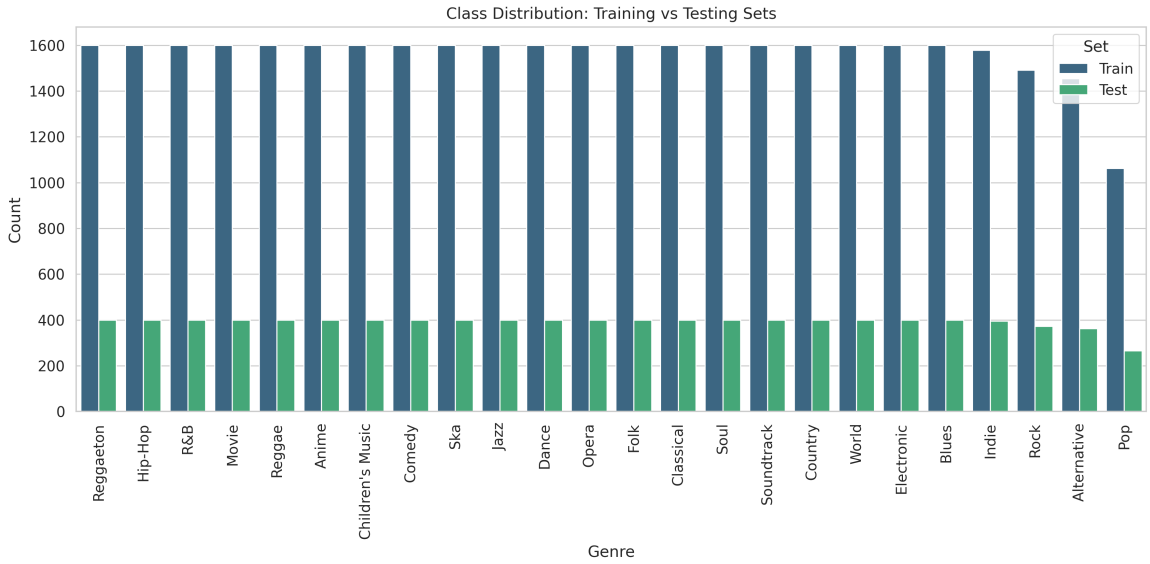

Figure 1: Class distribution in Training and Testing sets following hybrid balancing.

3.3 Scaling and Encoding

`StandardScaler` (Z-score normalization) was applied to all continuous features to facilitate the convergence of gradient-based models (MLP, Logistic Regression) and the accuracy of distance-based models (KNN). Categorical features were encoded using Label Encoding.

4 Visual Examination

Exploratory Data Analysis (EDA) was conducted to assess feature relationships and class separability.

4.1 Correlation Analysis

The Pearson correlation matrix (Figure 9) indicates multicollinearity among specific audio features. A correlation coefficient of 0.82 was observed between `energy` and `loudness`, while `acousticness` and `energy` exhibited a negative correlation of -0.74. These correlations suggest that models assuming feature independence, such as Naive Bayes, may experience performance degradation.

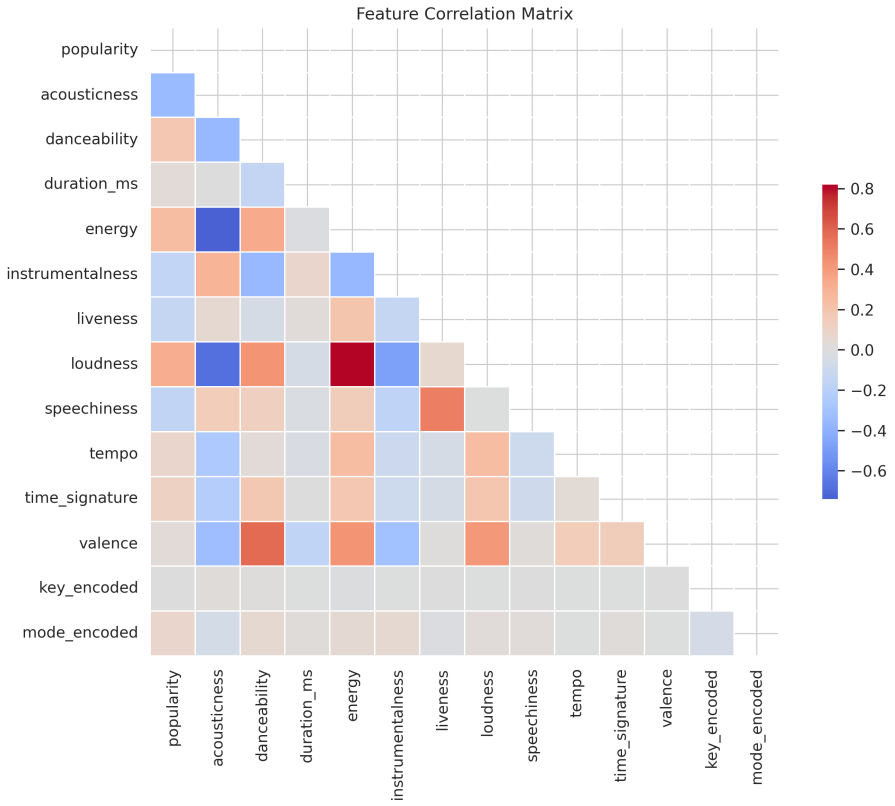

Figure 2: Feature Correlation Matrix indicating strong relationships between `energy`, `loudness`, and `acousticness`.

4.2 Feature Separability

Boxplot analysis (Figure 3) demonstrates that `acousticness` effectively discriminates Classical music from other genres, while `energy` separates Electronic music from Classical.

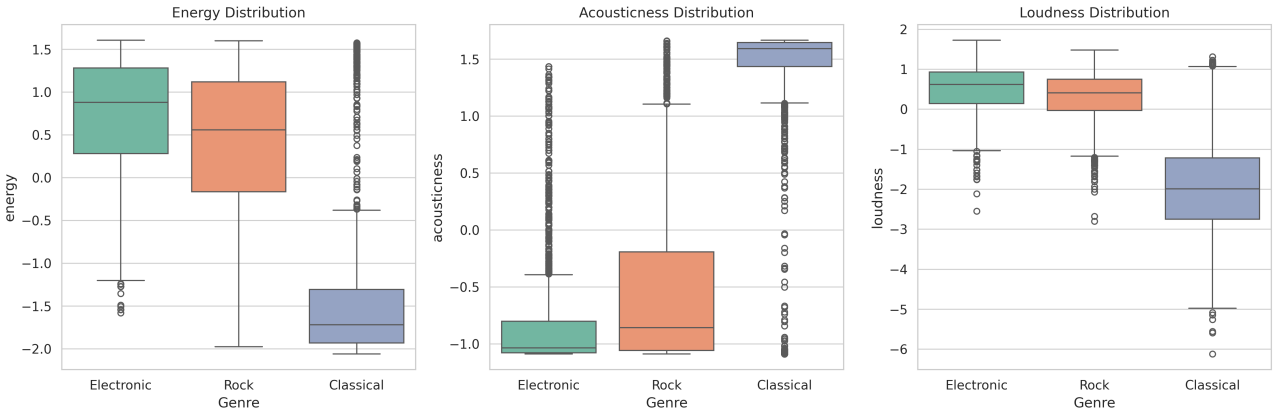

Figure 3: Distribution of key features across three distinct genres.

5 Feature Selection

Feature selection was performed in two stages: manual removal of non-predictive identifiers and algorithmic selection via model regularization.

5.1 Removal of Identifiers

The raw dataset contained metadata fields including `artist_name`, `track_name`, and `track_id`. While these fields are highly predictive of genre (e.g., a specific artist usually produces one genre), they do not generalize to unseen artists or new tracks. To ensure the model learned acoustic characteristics rather than memorizing specific catalog entries, these high-cardinality identifiers were removed prior to training.

5.2 Retention of Correlated Features

As noted in the Visual Examination, `energy` and `loudness` exhibited high correlation ($r = 0.82$). Standard procedure in linear modeling often involves removing one of these to reduce multicollinearity. However, we chose to retain both for the following reasons:

- **Tree-Based Robustness:** Our primary models (XGBoost, Random Forest) utilize embedded feature selection. They inherently select the most discriminative feature at each split, rendering manual removal unnecessary.
- **Interaction Effects:** Deep learning models (MLP) can often exploit subtle differences between correlated features (e.g., a track that is loud but low energy might indicate a specific sub-genre like drone metal) that would be lost if one feature were dropped.

Consequently, the final feature set consisted of 14 variables: 11 audio metrics and 3 structural/metadata features.

6 Modeling Techniques Chosen

Six modeling techniques were selected to evaluate a range of mathematical approaches to the classification problem.

6.1 Logistic Regression

Rationale: Selected as a linear baseline to determine if genre boundaries are linearly separable within the feature space.

6.2 K-Nearest Neighbors (KNN)

Rationale: A non-parametric, distance-based approach chosen to exploit potential clustering of genres in the feature space. Hyperparameter tuning indicated that $k = 19$ provided the optimal balance between bias and variance (Figure 4).

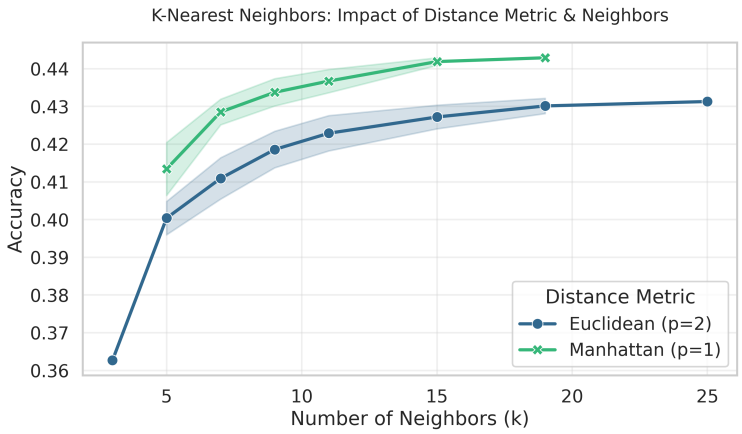

Figure 4: KNN Accuracy vs. Number of Neighbors (k).

6.3 Gaussian Naive Bayes

Rationale: Selected as a probabilistic baseline. While computationally efficient, it assumes feature independence, which serves as a control against the correlated nature of audio data.

6.4 Random Forest Classifier

Rationale: An ensemble bagging method chosen for its robustness to noise and ability to model non-linear decision boundaries. Performance plateaued after approximately 300 estimators (Figure 5).

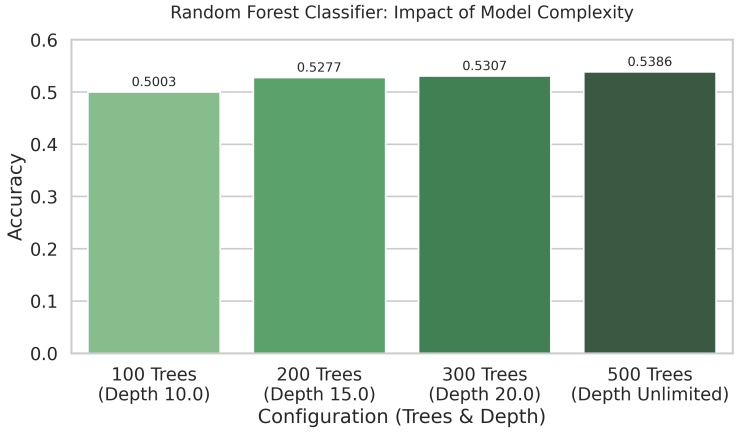

Figure 5: Random Forest Accuracy vs. Number of Trees.

6.5 XGBoost (Gradient Boosting)

Rationale: An ensemble boosting method chosen for its state-of-the-art performance on tabular data. Tuning revealed an optimal learning rate of 0.1 (Figure 7).

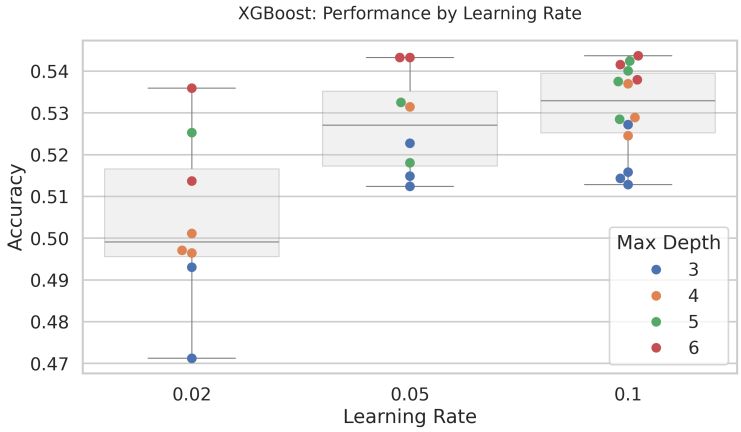

Figure 6: XGBoost Accuracy vs. Learning Rate.

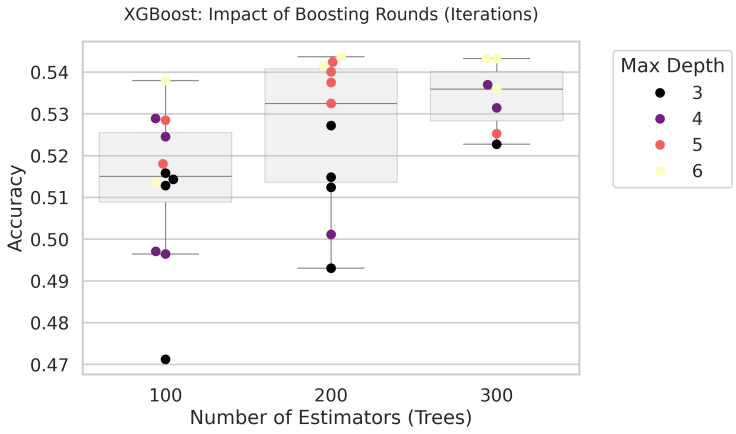

Figure 7: XGBoost Accuracy vs. Iterations.

6.6 Multilayer Perceptron (MLP)

Rationale: A deep learning approach using PyTorch, selected to capture complex, high-dimensional feature interactions via a 4-layer architecture.

7 Reporting of Results

Model performance was evaluated on the held-out test set ($N = 9,397$). Figure 8 displays the comparative accuracy of all models.

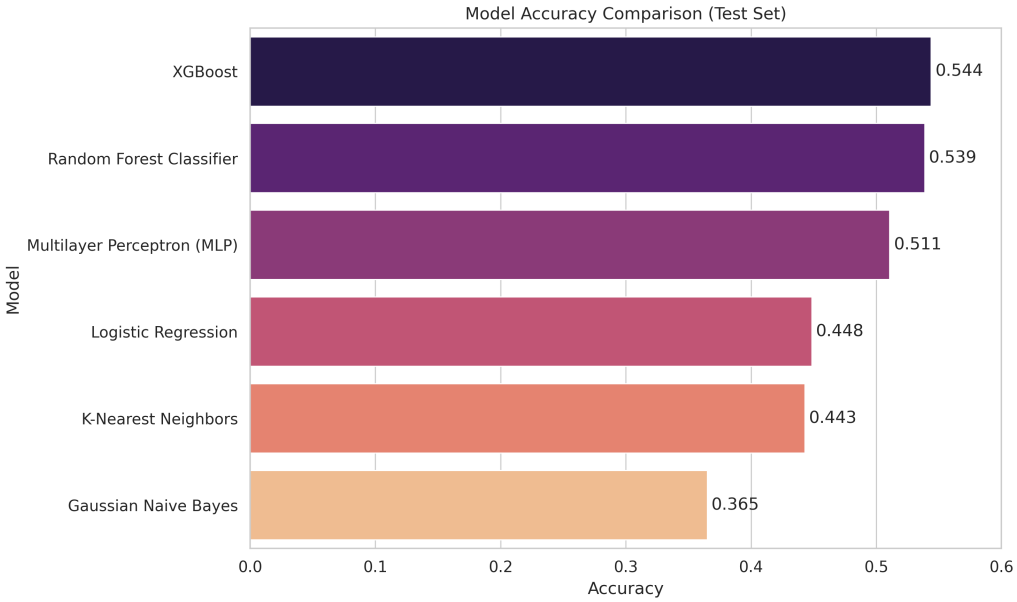

Figure 8: Comparative accuracy of the six evaluated models.

Table 1: Final Model Leaderboard

Rank	**Model**	**Accuracy**	**Weighted F1**
1	XGBoost	**0.5437**	**0.5400**
2	Random Forest	0.5386	0.5322
3	MLP (Neural Net)	0.5107	0.5046
4	Logistic Regression	0.4484	0.4369
5	KNN	0.4429	0.4402
6	Gaussian NB	0.3650	0.3405

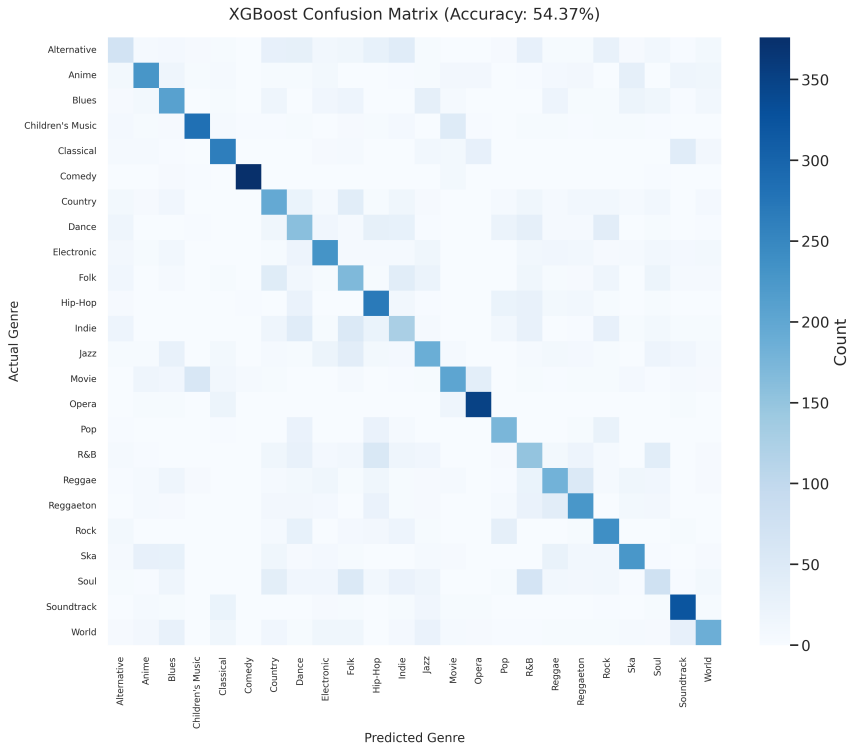

Figure 9: Confusion Matrix for the best XGB trial.

8 Interpretation of Results

8.1 Predictive Power vs. Random Baseline

Given 24 distinct classes, a random classifier would achieve an accuracy of approximately 4.1%. The XGBoost model achieved an accuracy of 54.37%. This substantial increase over the baseline confirms that the selected audio features contain significant predictive signal regarding musical genre, despite the subjective overlaps between classes.

8.2 Model Comparison

- **Tree Ensembles Dominate:** XGBoost and Random Forest outperformed linear and distance-based models by nearly 10%. This indicates that genre boundaries are highly non-linear and ragged, requiring models that can partition the feature space into complex regions.
- **Neural Networks:** The MLP performed well (51.07%), surpassing linear models, but did not exceed the performance of tree-based ensembles on this tabular dataset.
- **Linear Limitations:** Logistic Regression (44.84%) plateaued significantly lower than the top performers, suggesting that the relationships between audio features and genre are not linearly separable.
- **Impact of Correlation:** Gaussian Naive Bayes yielded the lowest accuracy (36.50%). This confirms the hypothesis derived from the EDA: the strong correlations between features (e.g., `energy` and `loudness`) violate the independence assumption of Naive Bayes, degrading its performance.

9 Recommendations of Study

Based on the analysis, the following recommendations are proposed for future research:

1. **Richer Data:** To exceed the observed accuracy ceiling of ≈55%, future iterations should incorporate Convolutional Neural Networks (CNNs) on raw audio or perhaps a pretrained audio embedding model rather than relying solely on pre-calculated summary statistics.
2. **Hierarchical Classification:** Given the taxonomical nature of music (e.g., "Rock" encompasses "J-Rock"), a hierarchical model that predicts the super-genre before refining the specific sub-genre could improve performance.
3. **Dimensionality Reduction:** For distance-based models such as KNN, applying Principal Component Analysis (PCA) or manually removing features with low separability (such as `key` and `mode`) may reduce noise and improve distance metrics.

10 Code Availability

The complete source code, data preprocessing scripts, and analysis notebooks used in this project are available in the following GitHub repository:
`https://github.com/carterprince/spotify-classification`