Qingyu Zhang / Andy

N-number: N19903322

Prof. Pascal Wallisch

Fundamentals of Machine Learning

Capstone Project Report: Music Genre Classification

1. Introduction

This project aims to classify songs into one of ten genres based on their audio characteristics using a supervised learning model, and to explore unsupervised structures within the feature space through dimensionality reduction and clustering. The dataset consists of approximately 50,000 Spotify songs, each annotated with a genre label and 13 audio features such as tempo, energy, speechiness, and danceability.

2. Data Cleaning and Preprocessing

Initial preprocessing involved removing corrupted rows (e.g., songs with invalid tempo or duration values), dropping irrelevant columns (artist name, track name, date), and ensuring the feature columns were complete. The categorical features were handled as follows:

- **'key'** was **label-encoded** (df['key'].astype('category').cat.codes)
- **'mode'** was **one-hot encoded**, keeping only mode_Minor (pd.get_dummies(df, columns=['mode'], drop_first=True))
- **'music_genre'** was encoded into 'genre_encoded' for classification (df['genre_encoded'] = LabelEncoder().fit_transform(df['music_genre']))

3. Train/Test Split Strategy

Following instructions, a **stratified sampling** strategy was used to form a test set: exactly **500 samples per genre** were randomly selected for the test set, and the remaining songs were assigned to the training set. This avoided label imbalance while ensuring no overlap between training and test data.

4. Feature Scaling

Only continuous numeric features were scaled using StandardScaler. The binary dummy variable 'mode_Minor' was **excluded from standardization** to preserve its categorical nature. This ensured compatibility with tree-based models while avoiding distortion of categorical indicators.

5. Classification Model and Hyperparameter Tuning

A Random Forest classifier was selected for its robustness to feature scaling, ability to model non-linear interactions, and interpretability via feature importance.

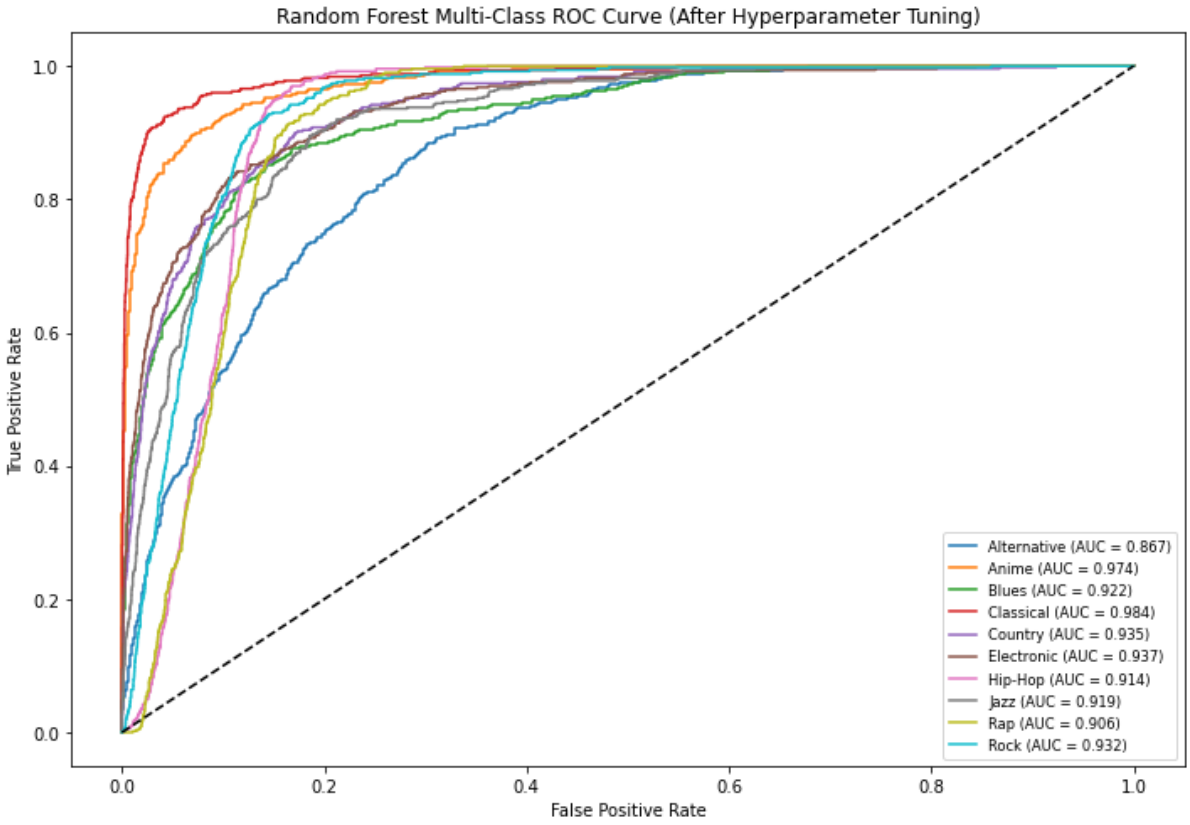
- GridSearchCV with 3-fold cross-validation was used to tune five hyperparameters: n_estimators, max_depth, min_samples_split, min_samples_leaf, max_features
- The best configuration was:

```
Best Parameters: {'max_depth': 20, 'max_features': 'sqrt', 'min_samples_leaf': 3, 'min_samples_split': 2,
'n_estimators': 300}
```

6. Model Performance and ROC-AUC

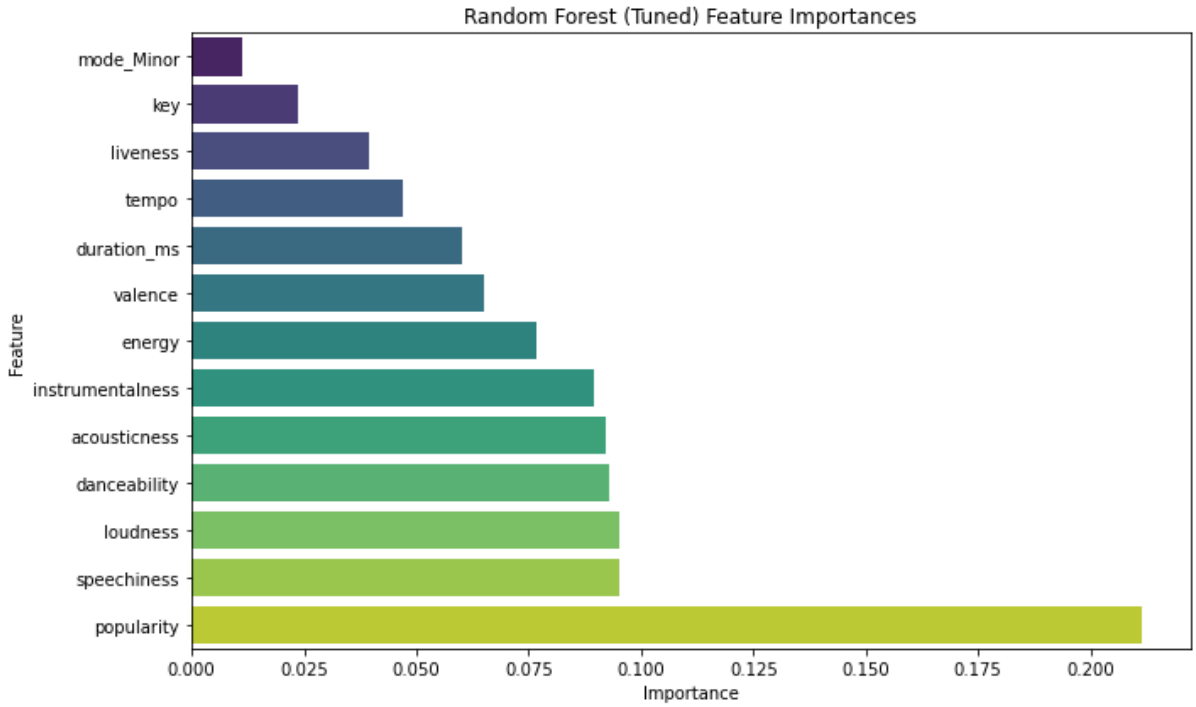
The final model achieved a **macro-averaged AUC of 0.929** on the multi-class test set. All 10 genres exhibited high individual AUCs (e.g., Classical: 0.984, Anime: 0.974, Rock: 0.932), indicating strong predictive performance. The ROC curves showed that

most genres were well-separated, though some overlap existed between similar categories.



7. Feature Importance

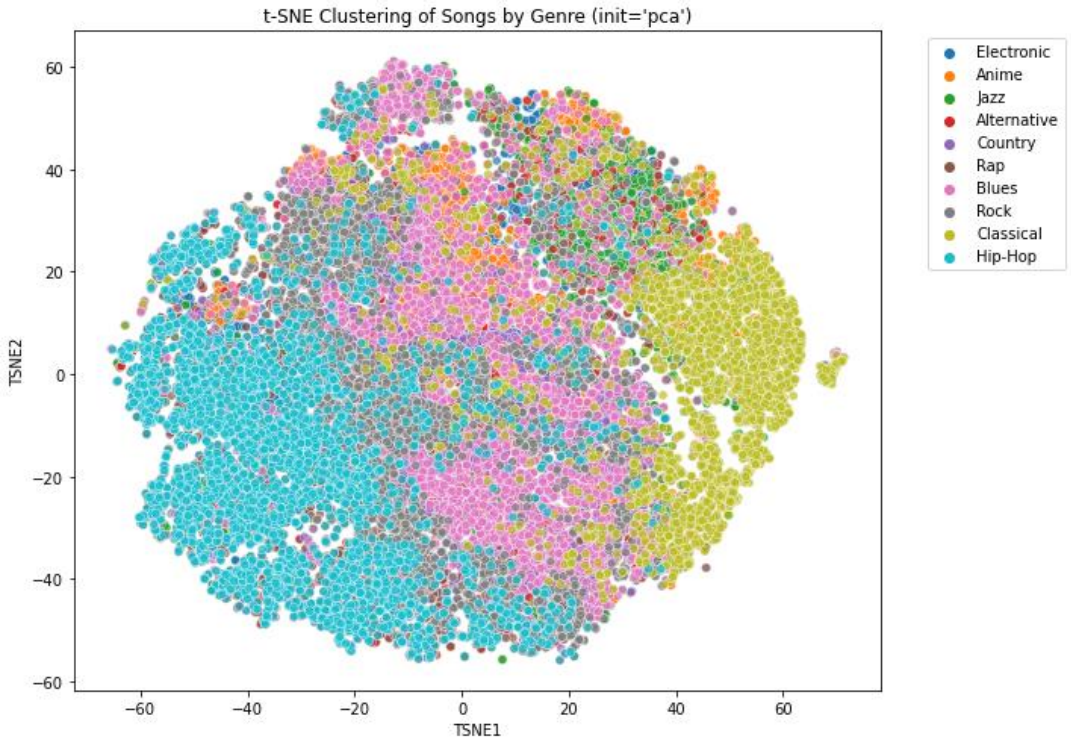
Feature importance analysis revealed that **popularity** was the most influential predictor by a large margin, followed by **speechiness, loudness**, and **danceability**. This aligns with the intuition that genre is closely tied to energy, vocals, and commercial performance.



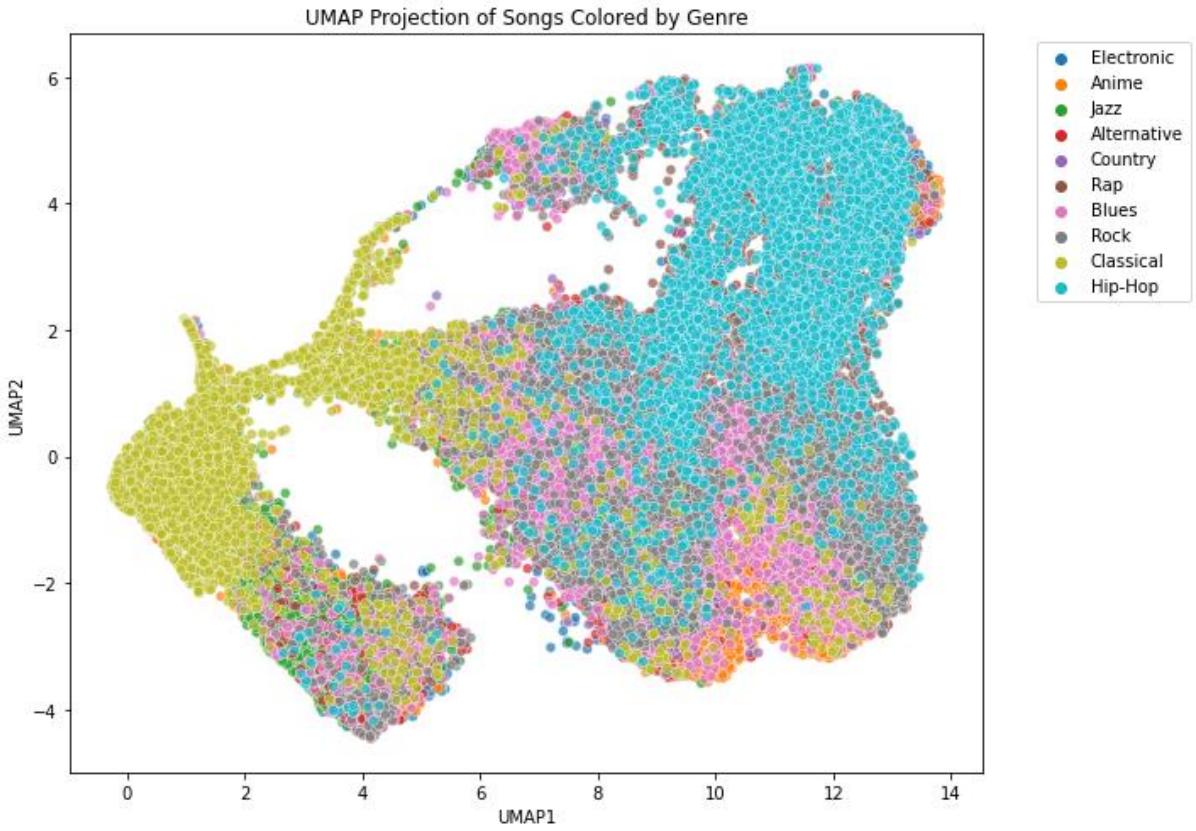
8. Dimensionality Reduction for Genre Visualization

To visualize genre separability in a low-dimensional space, two techniques were used:

- **t-SNE (PCA initialization)**: Preserved local structure well; clusters were identifiable but entangled



- **UMAP**: Produced clearer and more compact genre clusters, making it more suitable for follow-up clustering



These visualizations confirmed that genres like **Classical** and **Hip-Hop** form distinct manifolds (probably because they represent two extremes), while others like Jazz, Anime, and Alternative partially overlap.

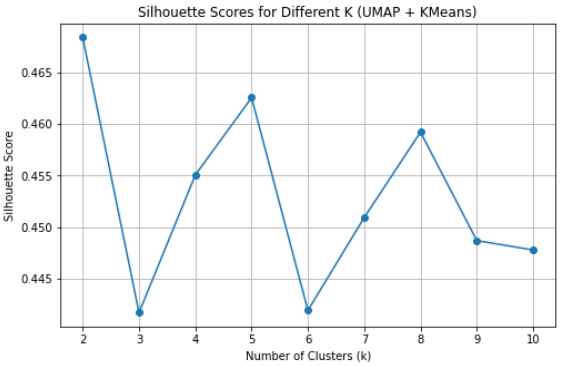
9. Clustering Analysis

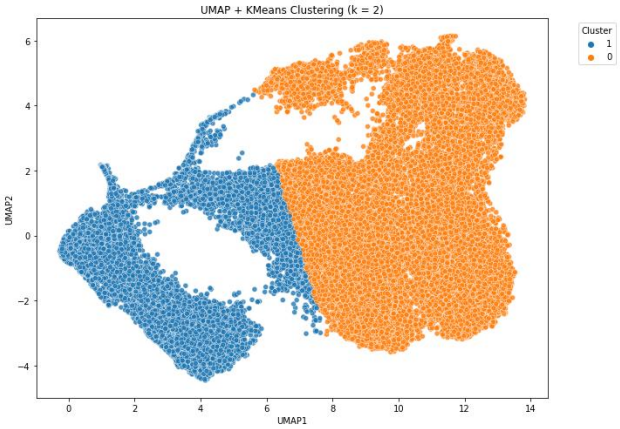
After reducing the feature space to 2D using UMAP, KMeans clustering was applied to identify latent group structures. Two clustering strategies were explored:

- **(a) Data-driven k selection:**

  The number of clusters was determined by maximizing the **silhouette score**. The best value was **k=2** with a score of **0.468**, indicating that the dataset has a prominent

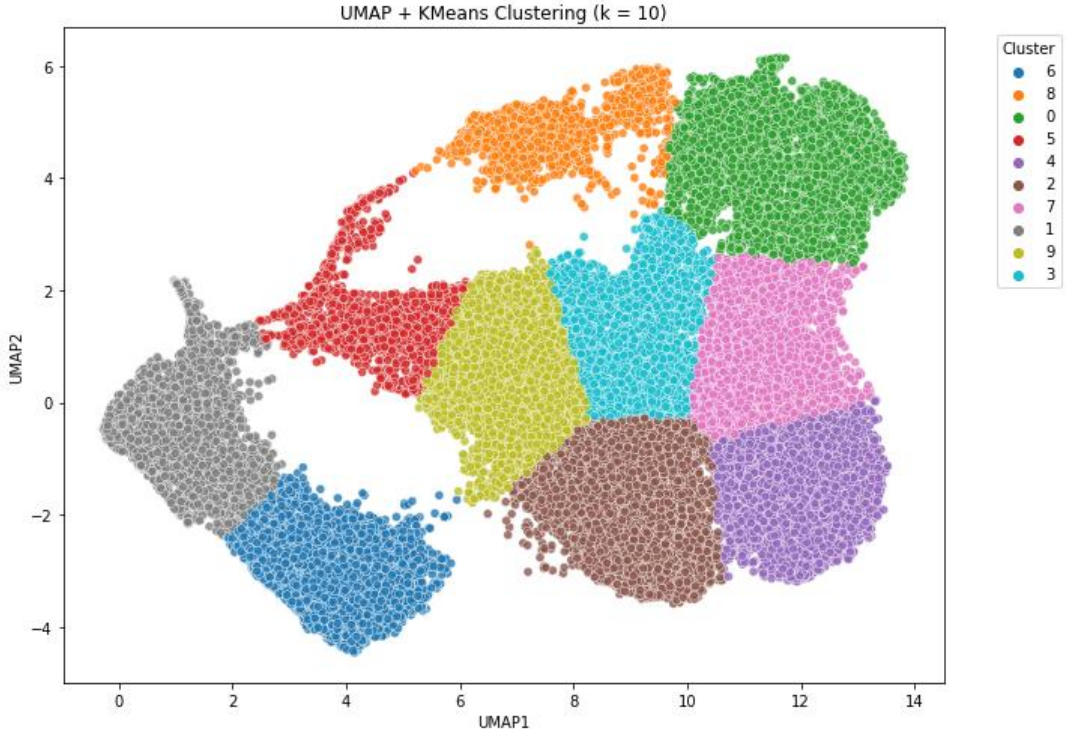
binary-level structure (e.g., vocal vs. instrumental, or high vs. low energy songs), but does not clearly separate into 10 genre-like clusters without supervision.





- **(b) Prior knowledge with k=10:**

  Knowing that the dataset contains 10 genres, we explicitly set **k=10** in KMeans to examine whether clustering can recover genre-level separability. While this produced clean and well-spaced clusters in UMAP space, **the cluster boundaries did not strongly align with the actual genre labels.**



This was made especially clear when comparing the **UMAP + KMeans clustering (k=10)** result with the **UMAP genre visualization** from Step 8. In the latter, points were colored using actual genre labels, revealing **much more diffuse and overlapping genre distributions** (between styles like Blues, Alternative, and Rock). In contrast, the KMeans output imposed arbitrary partitions, which often split the same genre across multiple clusters or merged different genres into one.

This comparison suggests that **while KMeans can impose structure on low-dimensional embeddings, it does not faithfully recover the underlying genre taxonomy.** The discrepancy underscores the complexity of musical genre classification and highlights the limitations of unsupervised clustering without label supervision.

10. Conclusion

This project successfully combined supervised and unsupervised learning to explore the relationship between Spotify audio features and musical genre. A carefully tuned Random Forest classifier achieved strong predictive performance, reaching a **macro-averaged AUC of 0.929**, and revealed that features such as popularity, speechiness, and loudness are most important in determining genre.

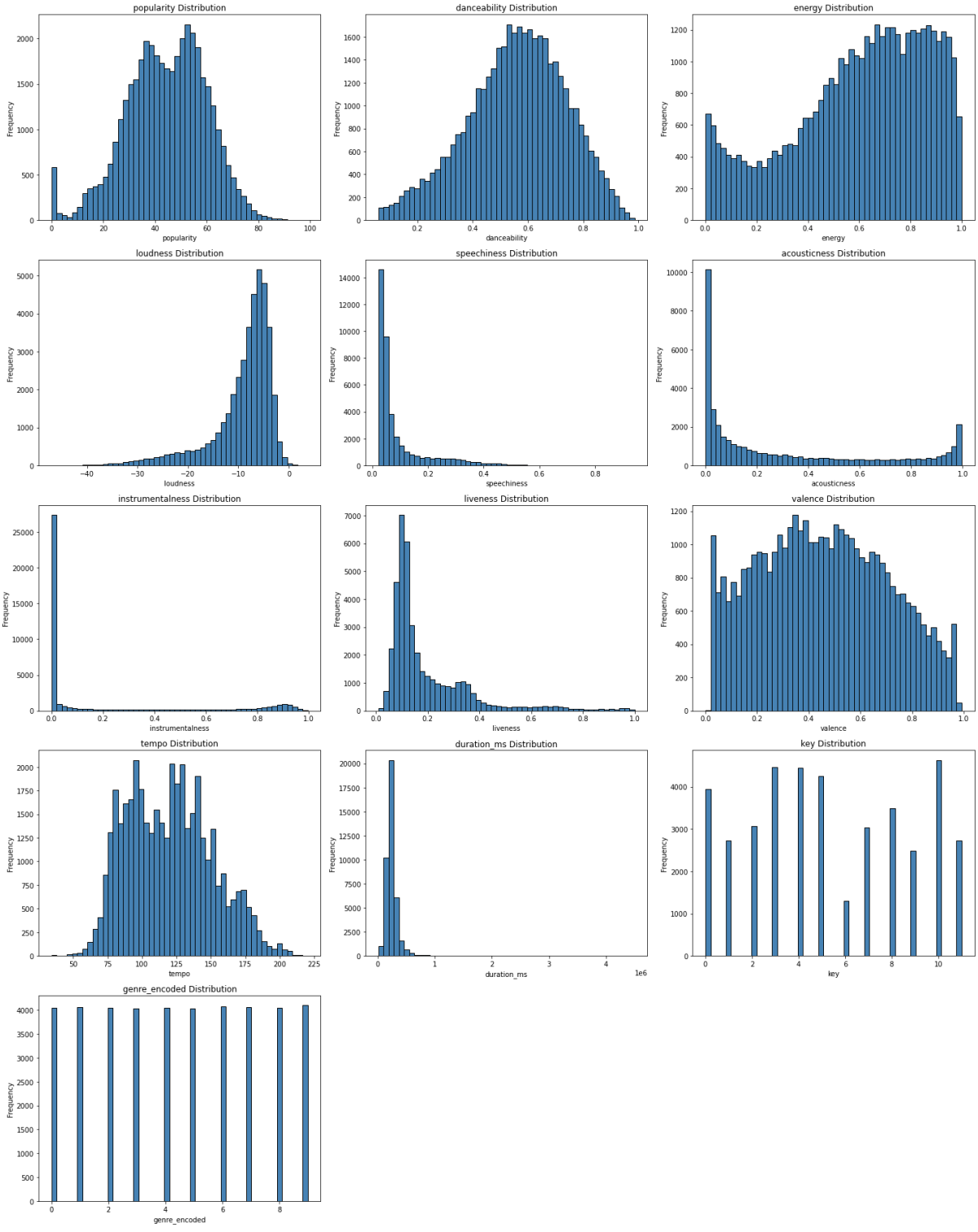
Through dimensionality reduction using both **t-SNE (with PCA initialization)** and **UMAP**, genre distributions were visualized in two-dimensional space. UMAP provided clearer and more compact clusters, making it especially suitable for downstream clustering analysis.

In the unsupervised stage, KMeans clustering was applied to the UMAP embeddings using both a silhouette-based k and a fixed k=10 informed by prior knowledge of the number of genres. However, **neither clustering strategy was able to accurately recover the actual genre structure**, as shown by comparison with genre-labeled UMAP plots.

Overall, the project demonstrated a well-rounded and rigorous analysis pipeline, balancing effective modeling with thoughtful reflection on the limitations of unsupervised learning for semantic tasks like music classification.

Extra Credit: Non-Trivial Feature Distribution Observations

To explore the dataset further, I plotted the marginal distributions of all numerical features, including the target variable genre_encoded.



Several non-obvious and potentially meaningful patterns emerged:

1. Popularity Is Bimodal

Unlike most audio features, popularity displays a clear **bimodal distribution**. One mode centers around ~30, and another around ~65, suggesting that the dataset may contain two

distinct populations of songs—perhaps underground/independent vs. mainstream. This confirms and visually supports why popularity had the highest importance in the Random Forest model.

- **Insight**: Genre prediction may partially act as a proxy for audience size or commercial exposure rather than musical content.

2. Instrumentalness and Acousticness Are Highly Skewed

Both features have **heavy right-skew**, with the majority of songs clustering near zero. This suggests that most tracks in the dataset are not instrumental and not acoustic, which may bias the classifier towards genres like Pop and Hip-Hop. A small number of outliers with high values likely belong to Classical or Jazz.

3. Key Distribution Is Uneven

Despite there being 12 possible keys (0–11), some keys like 0, 5, and 9 are clearly more common than others. This reflects real-world musical preferences and may influence genre classification in subtle ways.

4. Genre Distribution Is Uniform by Design

As expected from the stratified sampling procedure (500 samples per genre for the test set), genre_encoded appears almost uniform, ensuring balanced training/testing and fair model evaluation.

Conclusion of Extra Credit:

These observations go beyond surface-level summary statistics. They support earlier findings (e.g., the dominance of popularity), raise new questions about feature bias, and provide

tangible visual evidence of how musical and non-musical properties influence genre classification. Such explorations improve both **model interpretability** and **data understanding,** aligning with the goals of responsible and thoughtful machine learning.