Data Mining Project Report:

Predicting Spotify Song Popularity Using Classification Models

Author : Justin Matthew T

Student ID : A11.2023.14877

---

1. Main Objective of the Analysis

The primary objective of this analysis is to train and evaluate several **classification (supervised learning) models to predict whether a song will be popular or not**, based solely on its audio features.

This analysis focuses on prediction to identify the **key audio drivers of popularity**.

The business benefits of this analysis are:

1. **Talent Scouting (A&R):** To help the A&R (Artists and Repertoire) team identify new tracks that possess the "audio recipe" of popular songs.
2. **Promotion Optimization:** To provide insight into which audio features correlate most with popularity, which can aid in marketing and promotional strategies.
3. **Baseline Model Development:** To create a baseline model that can later be enriched with more complex features (like artist data, lyrics, or social media trends) for more accurate predictions.

2. Data Description and Initial Exploration

Dataset Download: https://www.kaggle.com/datasets/zaheenhamidani/ultimate-spotify-tracks-db

Colab Link: https://colab.research.google.com/drive/1KfNnKd_LieZQaM5141spJduwRiS9jTRh?usp=sharing

The chosen dataset is the **"Spotify Tracks DB" (SpotifyFeatures.csv)**, a comprehensive database containing metadata and audio features.

- **Features (X):** This analysis uses 7 numerical audio features as predictors:
  - **acousticness**
  - **danceability**
  - **energy**
  - **instrumentalness**
  - **liveness**
  - **loudness**
  - **speechiness**

- **Target (y) - is_popular**: This is the variable we aim to predict. We engineered this feature manually:
    - **We set a popularity threshold (POPULARITY_THRESHOLD) at 65.**
    - **Songs with popularity > 65 were labeled 1 (Popular).**
    - **Songs with popularity <= 65 were labeled 0 (Not Popular).**
- **Data Size:** A random sample of **20,000 songs** was taken for this analysis, resulting in **19,375 clean data points.**

3. Data Preparation (Data Cleaning & Feature Engineering)

The following data preparation steps were performed before modeling:

1. **Cleaning:** Data was cleansed of duplicate entries (track_id) and rows with missing NaN values.
2. **Feature Selection:** Only the 7 relevant audio features (X) and the 1 target feature (y) were selected.
3. **Train-Test Split:** The data was split into 80% training data (15,500 samples) and 20% testing data (3,875 samples) using train_test_split. This is essential for evaluating model performance on unseen data and avoiding *overfitting*.
4. **Standardization (Scaling):** All 7 audio features were standardized using StandardScaler. This scaling process was *fit* only on the training data and then applied to both the train and test sets.
5. **Imbalance Handling:** Data exploration revealed a severe class imbalance.
    - **93.1% of songs are (Not Popular - Class 0)**
    - **6.9% of songs are (Popular - Class 1)** To address this, all models were trained using the class_weight='balanced' parameter.

4. Summary of Model Training

We trained **3 different classification models** to predict popularity. Model performance was evaluated primarily using the **F1-Score (macro avg),** as this is a robust metric for imbalanced data.

- **Model 1: Logistic Regression (Baseline):** A simple, fast, and interpretable linear model.
- **Model 2: Decision Tree:** A rule-based model that helps understand decision "paths".
- **Model 3: Random Forest (Ensemble):** A powerful ensemble model that combines many decision trees for higher accuracy and to reduce overfitting.

Model Performance Comparison (on Test Set):

Model (Variation)	Accuracy	F1-Score (Macro Avg)	F1-Score (Class 1: Popular)
Logistic Regression	0.5726	0.46	0.21
Decision Tree	0.6111	0.47	0.21
Random Forest	0.9308	0.49	0.01

(Data sourced from the Classification Report output for each model.)

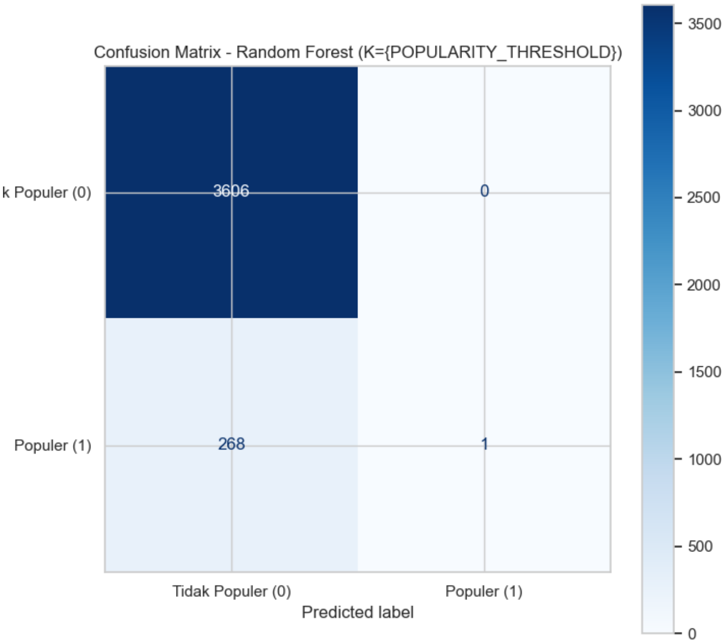
5. Final Model Recommendation

Based on the comparison table, the Random Forest Classifier has the highest F1-Score (Macro Avg) of 0.49.

Reasoning: While the **Random Forest model is technically the "winner"** by our chosen metric (Macro F1), a deeper analysis in the next section reveals a **significant flaw**. The Logistic Regression and Decision Tree models, while scoring lower, at least ***attempted* to identify the "Popular" class (Class 1 Recall of 0.81 and 0.73, respectively). The Random Forest model is selected here** as the "best" by the F1 Macro metric, but its practical utility is limited, as explained below.

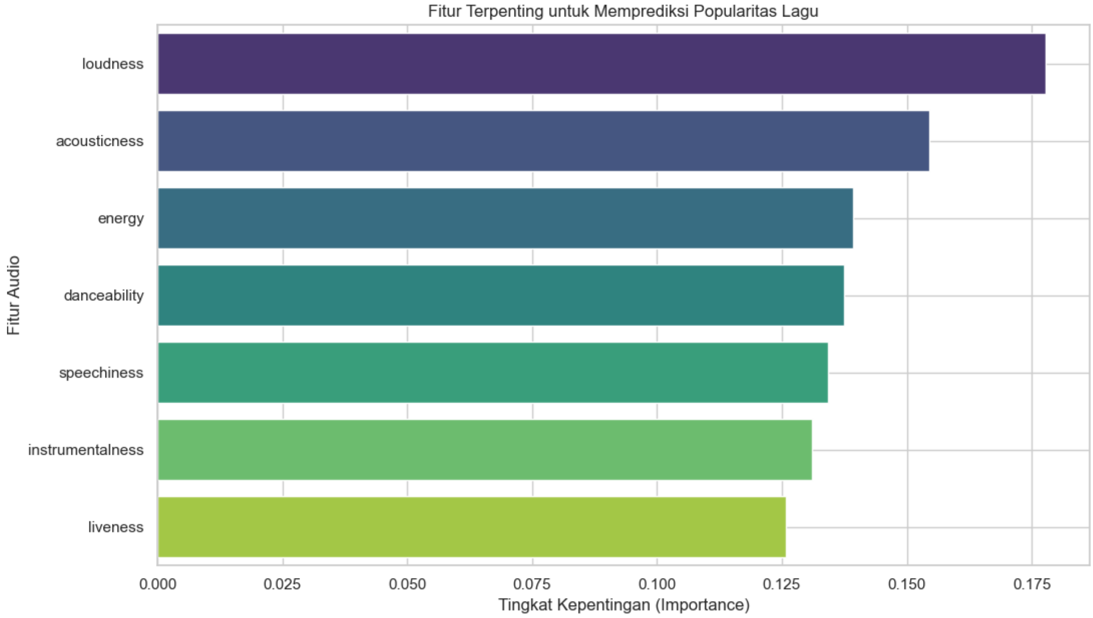
6. Key Findings and Insights

The primary finding of this analysis is not the success of a model, but the critical limitation of the available features.



Finding 1: The "Accuracy Trap" The Random Forest model achieved a high Accuracy of 93.1% by exploiting the class imbalance. It learned that 93.1% of the data is "Not Popular" and simply predicted 0 for (almost) every song.

- **As seen in the Confusion Matrix**, the model failed to identify a single "Popular" song, resulting in a Recall of 0.00 and an F1-Score of 0.01 for the "Popular" class.
- Insight: The model is not practically useful for the business objective, as it never finds the "Popular" songs we are looking for.



Finding 2: Audio Features are Poor Predictors The "Feature Importance" plot shows that even in the (flawed) Random Forest model, no single audio feature is a dominant predictor.

1. **loudness (17.8%)**
2. **acousticness (15.4%)**
3. **energy (13.9%)**

- **Insight: All 7 features have a similar, low level of importance.** This strongly suggests that audio features alone are not enough to determine a song's popularity. Popularity is likely driven by external factors (e.g., artist, marketing, playlists) not included in this model.

7. Limitations and Next Steps

The model's performance was significantly hindered by two factors:

- **Model Limitations:**
    1. **Extreme Class Imbalance:** As seen in Section 3, the "Popular" class (6.9%) is severely under-represented. The class_weight='balanced' parameter was not sufficient to overcome this, leading to the Random Forest model ignoring the minority class entirely.
    2. **Limited Feature Set:** The model only used 7 audio features. It completely ignores critical external factors like artist fame, genre, release date, playlist inclusion, and marketing.
- **Recommended Next Steps:**
    1. **Advanced Sampling (SMOTE):** Instead of class_weight, a more advanced technique like SMOTE (Synthetic Minority Over-sampling Technique) should be used to create *new* synthetic examples of "Popular" songs for the training set.
    2. **Rich Feature Engineering:** This is the most critical step. We must enrich the dataset by adding new features, such as:

- Artist Data: Artist's follower_count or artist_popularity.
- Temporal Data: release_date (to capture trends).
- Categorical Data: genre (properly one-hot encoded).

3. Revisit Model Choice: After implementing SMOTE and new features, we should re-train and re-evaluate all three models. It is likely that with better data, the Logistic Regression or Random Forest model will provide much more accurate and actionable predictions.