MACHINE LEARNING ANALYSIS OF SPOTIFY TRACKS

Project submitted to the University of Kerala in partial fulfillment of the requirements for the Degree of Master of Science in Statistics

Submitted By

FATHIMA KHENZA A

(Reg no: 98723615011)

Department of Statistics

University of Kerala

Thiruvananthapuram

2023-2025

DEPARTMENT OF STATISTICS
UNIVERSITY OF KERALA

Dr.E.I ABDUL SATHAR
Professor, Department of Statistics
University of Kerala

P.O.Kariavattam
Trivandrum, 695581
phone-0471-18905

CERTIFICATE

I hereby certify that this project titled "MACHINE LEARNING ANALYSIS OF SPOTIFY TRACKS" is a bonafide project work carried out by Ms. Fathima Khenza A, M.Sc.Statistics student in the Department of Statistics, University of Kerala, Kariavattom, during the period of 2023-2025, under my supervision and guidance in partial fulfillment of the requirements for the M.Sc. Degree in Statistics of the University of Kerala.

July 2025
Thiruvananthapuram

Dr E.I Abdul Sathar

DECLARATION

I hereby declare that the work presented in the project titled “Music Popularity Prediction Using Machine Learning” is completed by me under the guidance of Dr EI ABDUL SATHAR, Professor ,Department of Statistics, University of Kerala in partial fulfillment of the requirements of the Degree of Master of Science in Statistics

July 2025
Thiruvananthapuram FATHIMA KHENZA A

ACKNOWLEDGEMENT

I would like to convey my profound gratitude to Dr. EI Abdul Sathar, my project guide, for his immense assistance, support, and encouragement during this project. His suggestions and consistent motivation were critical in designing this project.

I also want to thank the Department of Statistics for enabling me the opportunity and resources for working on this project, as well as for creating a learning atmosphere that made the experience interesting and gratifying.

Finally, I want to express my heartfelt gratitude to my friends and family for their constant backing, understanding, and belief in me. Their appreciation and reassurance have consistently been my biggest strengths.

Table of Contents

List of Figures

List of Tables

CHAPTER 1:

INTRODUCTION

1.1 Background of the Study

Spotify along with additional digital music streaming platforms have entirely transformed how people listen to and enjoy music in the modern era. Spotify is a rich source of data for analytical studies in addition to being a source of entertainment, with millions of tracks available and comprehensive metadata linked to each song. Numerous audio characteristics that represent both musical structure and listener perception are available on the platform, including danceability, energy, loudness, acousticness, and tempo. For performing artists, producers, and streaming services alike, determining out what makes a song popular is an intriguing challenge. Numerous factors, such as the artist's reputation, genre, musical qualities, and outside trends, can affect popularity on Spotify, which is measured as a score between 0 and 100. By using machine learning to analyze these patterns.

Predictive modeling and tailored recommendations have become more feasible in recent years as a result to the incorporation of machine learning into music analytics. By using these methods, researchers are able to find hidden patterns and groupings in large datasets in addition to measuring the impact of individual musical features. Beyond merely forecasting popularity, machine learning can identify genre-specific characteristics, categorize songs according to mood, and give content producers insightful feedback. Studies like these are crucial in determining how music is created, distributed, and enjoyed in an online environment as the music industry depends more and more on data-driven tactics.

1.2 Motivation for the Study

In the contemporary digital landscape, music streaming services such as Spotify have emerged as the predominant means through which individuals discover and engage with music. With an extensive library of millions of tracks and tailored recommendations influencing listener habits, it is increasingly important to comprehend the fundamental patterns that contribute to a song's success. Although music is frequently perceived as an artistic and emotional journey, it is also intricately organized and rich in data — presenting a distinctive opportunity for analytical investigation.

In spite of the vast amounts of data available on streaming platforms, much of the decision-making regarding what is promoted or recommended relies on opaque algorithms. This situation creates a disconnect for artists, producers, and even data scientists who seek to understand the factors that lead to a song's success. Historically, insights within the music industry have been derived from intuition or prevailing trends, while this study aspires to enhance transparency by employing statistical and machine learning techniques on Spotify's dataset.

The impetus for this project is to reconcile the divide between musical artistry and data-informed analysis. It is driven by the desire to investigate whether aspects as subjective as musical taste and popularity can be examined, quantified, and anticipated through the identification of patterns in audio characteristics and metadata. Additionally, this research seeks to facilitate practical applications such as mood-based recommendations, a deeper comprehension of listener preferences, and feature-oriented marketing strategies for artists and producers.

1.3 Problem Statement

Despite the wealth of information available from streaming services such as Spotify, the fundamental elements that affect a song's popularity are still largely unexamined or obscured within intricate recommendation algorithms.

Artists, producers, and marketers frequently encounter a lack of clarity regarding the factors that contribute to a track's success, complicating their ability to make well-informed decisions based on listener behavior and musical composition.

Conventional evaluations of musical success have depended on intuition, prevailing trends, or social influence; however, these methods fail to utilize the complete potential of data-driven insights. Although machine learning has demonstrated its effectiveness across various predictive fields, its use in music analytics is still in a developmental phase. A significant gap exists in comprehending how measurable audio characteristics—such as energy, danceability, valence, and loudness—are related to or can forecast a song's popularity.

This research intends to fill that gap by employing statistical techniques and machine learning models to scrutinize Spotify's music data. It aims to forecast popularity, pinpoint key features, and deliver actionable insights that could be advantageous for music creators, streaming services, and recommendation systems.

1.4 Objectives of the Study

This study aims to explore the relationship between various audio features and the popularity of songs on Spotify. In addition to examining continuous musical attributes such as energy, danceability, and valence, the study also investigates whether categorical elements like explicit content and musical mode (major or minor) have a statistically significant impact on a song's popularity. These analyses provide deeper insights into how different types of musical features contribute to listener engagement.

Another key objective is to develop a regression-based machine learning model capable of predicting track popularity based on audio and metadata features. The study further aims to evaluate the predictive importance of each feature, thereby identifying which musical or contextual attributes play a dominant role in determining a song’s success on the platform.

The final objective involves performing mood-based clustering of songs using key audio characteristics, with the goal of grouping tracks into emotional categories such as Happy, Sad, Chill, and Aggressive. Based on these groupings and overall feature similarity, a basic recommendation system is implemented to suggest songs either by mood or by resemblance to a user-selected track, thereby enhancing personalized music discovery.

1.5 Scope of the Study

This project explores how machine learning and statistics can be used to understand and predict the popularity of songs using data from a Kaggle Spotify dataset. It includes in-depth exploratory data analysis, statistical testing, and the development of a regression model to predict song popularity based on audio features. The study also uses unsupervised learning to group songs by mood and builds a basic recommendation system that suggests songs either by mood or by how similar they sound to a chosen track.

The analysis is based only on the features and songs available in the Kaggle dataset. It doesn't take into account things like real-time listener behavior, marketing influence, artist fame, or social media trends. The popularity scores used in the project come directly from the dataset and are treated as reliable for the purpose of this study.

Overall, the project is intended as an academic exploration to show how data-driven techniques can help uncover patterns in music and be used to build simple, personalized recommendation tools.

1.6 Structure of the Report

This report is organized into five chapters. Chapter 1 presents the introduction, including the background, motivation, problem statement, objectives, scope, and overall structure of the study. Chapter 2 outlines the dataset and methodology, covering data preprocessing, feature engineering, tools and libraries used, and an overview of the machine learning models and recommendation systems developed. Chapter 3 focuses on exploratory data analysis, including both univariate and bivariate visualizations, along with genre-based patterns. Chapter 4 presents the results and interpretation of statistical tests, model performance, clustering outcomes, and recommendation logic. Finally, Chapter 5 provides the conclusion of the study and discusses possible directions for future work.

CHAPTER 2:

DATASET AND METHODOLOGY

2.1 Introduction

This chapter outlines the dataset used in the study and the methodologies applied to prepare, analyze, and model the data effectively. It begins with a description of the Spotify dataset, including its source, structure, and the types of audio and metadata features it contains. This is followed by a detailed explanation of the preprocessing steps undertaken to clean the data, handle missing values, encode categorical variables, correct feature ranges, and scale numerical values to a common range. Feature engineering techniques, including transformations and target-specific modifications, were also applied to enrich the dataset. The chapter further describes the statistical tools and programming libraries used throughout the analysis, providing the technical foundation for the modeling and evaluation processes in the subsequent chapters.

2.2 Description of the Dataset

The dataset used for this study was sourced from Kaggle, a well-known online platform for data science competitions and datasets. It contains information on approximately 114,000 Spotify tracks, making it a comprehensive and diverse sample for analyzing music-related trends. Each entry in the dataset represents a unique song and includes a wide array of musical and metadata features. Key attributes include tempo, energy, danceability, acousticness, loudness, valence, instrumentalness, liveness, speechiness, duration, and mode, among others. These variables are particularly useful for understanding the audio characteristics that may influence a track's popularity. In addition to audio features, the dataset also includes artist names, track genres, and popularity scores (ranging from 0 to 100) as assigned by Spotify's internal ranking algorithm. This dataset provides a rich foundation for both statistical analysis and machine learning modeling, offering insights into the musical qualities that resonate with listeners. However, it is important to note that the dataset does not include behavioral or contextual data such as playlist placements, user skip rates, or social media engagement, which could also affect a song's popularity.

2.3Data Preprocessing

Before applying statistical analysis and machine learning models, the dataset underwent a thorough preprocessing phase to ensure its suitability for modeling. This involved several essential steps, including handling missing values, correcting inconsistencies in numerical features, encoding categorical and boolean variables, performing exploratory data analysis (EDA), scaling features to a uniform range, and splitting the dataset into training and testing sets. Each step was crucial for improving data quality, enhancing model performance, and ensuring that the underlying structure of the dataset was well understood and appropriately prepared for both supervised and unsupervised learning tasks.

2.3.1 Handling Missing Values

Handling missing values is a critical step in data preprocessing, as missing or incomplete records can affect the quality of analysis and lead to unreliable model performance. In this study, the dataset originally consisted of 114,000 rows. During inspection, it was found that one row contained missing values across multiple key features, making it unsuitable for analysis. Since the proportion of missing data was extremely small (less than 0.001%), this row was removed entirely to ensure a clean dataset without compromising the overall data size. After this step, the dataset was reduced to 113,549 complete records, with no remaining null values.

2.3.2 Feature Correction

As part of data preprocessing, musical features were carefully inspected to ensure their values aligned with expected domain-specific ranges. For example, the loudness feature was found to contain some positive values, which are not valid in the context of audio analysis, where loudness is typically measured in decibels (dB) and should be negative. These values were corrected by converting them to their negative counterparts to maintain consistency.

Similarly, the time_signature feature, which typically ranges from 3 to 7 in most musical compositions (but is often encoded from 0 to 5 in datasets), was checked for zero or non-musical values. To correct unrealistic entries (like zero), a lambda function was applied to replace values less than or equal to 0 with 2, ensuring all records reflected plausible musical structures. These corrections helped preserve the quality of the dataset and ensured that the machine learning models would not be trained on invalid or misleading inputs.

2.3.3 Encoding Categorical Variables

Encoding is an essential step in data preprocessing when preparing categorical variables for machine learning models, which typically require numerical input. Features like text labels, Booleans, or categories must be converted into a numerical format to be properly interpreted by algorithms. Choosing the right encoding method depends on the nature of the feature and the model being used.

In this study, the explicit feature was originally a Boolean (True/False) value indicating whether a track contains explicit content. This was converted to a numerical format using label encoding, where True was mapped to 1 and False to 0. Additionally, the dataset contained high-cardinality categorical features such as artists and track genre. To avoid issues with too many unique values and to preserve statistical relevance, target encoding was applied to these columns. Target encoding replaces categorical labels with the average popularity score associated with each category, making it suitable for regression tasks. These encoding steps were crucial to preparing the dataset for modeling and ensured that no categorical information was lost during preprocessing.

2.3.4 Feature Scaling

Scaling numerical features is an important step in preparing data for machine learning. If the values of different features vary too much in scale, it can confuse the model and lead to poor results—especially in models that are sensitive to the size of the input numbers. Scaling helps make sure that all features are treated fairly and can improve both the accuracy and speed of training.

In this project, I used Min-Max Scaling to bring certain features into a consistent range between 0 and 1. Most of the musical features in the dataset, like danceability and valence, were already within this range, so Min-Max Scaling helped match the rest with them. I applied this scaling to duration_ms, loudness, and tempo, since their raw values were on a much larger scale. This made sure that all features were in harmony with each other before feeding them into the machine learning models

2.3.5 Train-Test Split

Before training the machine learning model, the dataset was divided into training and testing subsets. This step is essential to evaluate the model's performance on unseen data and ensure

it generalizes well beyond the training set. By holding out a portion of the data for testing, we can measure how accurately the model predicts new examples and avoid overfitting to the training data.

In this study, the features (X) were selected from the processed dataset, excluding the target variable popularity, which was stored separately as y. The dataset was then split using an 80:20 ratio, where 80% of the data was used for training and 20% for testing. The split was performed using scikit-learn's train_test_split() function, with a random state of 42 to ensure reproducibility. This prepared the dataset for training and allowed for a reliable evaluation of model performance.

2.4 Feature Engineering

As part of the feature engineering process, transformations were applied to enrich the dataset and improve its suitability for machine learning. One such transformation addressed the skewness of the target variable, popularity. Since the popularity scores were heavily right-skewed, a Yeo-Johnson power transformation was applied. This method can handle zero and negative values, making it appropriate for stabilizing variance and making the target distribution more symmetrical. This transformation was intended to improve model performance by helping the regression model better capture patterns across the full range of popularity values.

Another key step in feature engineering involved categorizing tracks based on dominant and non-dominant genres of the artists. For each artist, their most frequently appearing genre in the dataset was identified as their dominant genre. Tracks matching this dominant genre were labeled as dominant, while others were labeled as non-dominant. This classification allowed for deeper analysis of popularity trends within and outside an artist's typical musical style. It also helped reveal that dominant-genre tracks generally achieved higher popularity compared to non-dominant ones.

2.5 Programming Libraries Used

This project was implemented using the Python programming language, which is widely used in data science due to its simplicity, flexibility, and strong ecosystem of analytical libraries. All data analysis, modeling, and visualization tasks were carried out in the Jupyter Notebook environment, which allowed for an interactive and well-documented workflow.

2.5.1 Pandas

Pandas is open-source Python library which is used for data manipulation and analysis. It consists of data structures and functions to perform efficient operations on data. Pandas was used for data loading, cleaning, manipulation, and organization. Its DataFrame structure made it easy to handle large tabular datasets.

2.5.2 Numpy

Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. NumPy supported numerical operations and array-based calculations, providing the foundation for mathematical processing and integration with other libraries.

2.5.3 Scikit-learn

scikit-learn was the primary machine learning library used for preprocessing (scaling, encoding, train-test split), regression modeling (Random Forest), clustering, dimensionality reduction (PCA), and evaluation metrics.

2.5.4 Scipy

SciPy is a Python library useful for solving many mathematical equations and algorithms. Scipy were used for statistical testing, such as hypothesis testing, p-value computation, and calculating the Variance Inflation Factor (VIF).

2.5.5 Matplotlib

Matplotlib is a powerful and versatile open-source plotting library for Python, designed to help users visualize data in a variety of formats. Matplotlib is an open-source visualization, widely used for creating static, animated and interactive plots.

2.5.6 Seaborn

Seaborn is an amazing visualization library for statistical graphics plotting in Python. It provides beautiful default styles and color palettes to make statistical plots more attractive. Seaborn was used for data visualization and exploratory analysis, including distribution plots, box plots, heatmaps, and correlation matrices.

2.6 Statistical Tools

In this section, the key statistical methods applied during the analysis are outlined. These techniques were used to understand the data more deeply, assess assumptions, and ensure the reliability of the modeling process

2.6.1 Correlation Analysis

Correlation analysis measures the strength and direction of linear relationships between pairs of variables. In this project, Pearson's correlation coefficients were calculated to explore the relationships between numerical audio features such as energy, loudness, valence, and danceability. A heatmap was used to visually highlight positive and negative associations. For example, a strong positive correlation was observed between energy and loudness, consistent with known acoustic patterns in music

2.6.2 Mann-Whitney U Test

The Mann-Whitney U test is a non-parametric test used to determine whether two independent samples originate from the same distribution. It is often preferred over the t-test when the data does not follow a normal distribution (Nachar, 2008). In this study, the test was used to evaluate whether popularity scores differed significantly between explicit and non-explicit tracks, and between tracks in major vs. minor musical modes. In both cases, p-values were close to 0, indicating statistically significant differences in popularity between the groups.

2.6.3 Variance Inflation Factor

The Variance Inflation Factor (VIF) is used to detect multicollinearity — a condition where predictor variables in a regression model are highly correlated with each other. High VIF values suggest redundancy and can lead to unstable regression coefficients. In this study, VIF analysis was conducted on all numerical features included in the regression model to ensure that multicollinearity was minimized and model interpretation remained valid (O'Brien, 2007). All selected features had VIF values below critical thresholds, indicating no serious multicollinearity issues.

2.7 Graphical Tools

Visualizations are a fundamental part of data analysis, as they allow patterns, trends, and relationships to be seen more clearly than through raw data alone. In this study, various graphical tools were used

throughout the exploratory and analytical phases to understand the structure and behavior of the dataset. These tools supported distribution analysis, outlier detection, relationship assessment, and preliminary comparisons between variables. The following subsections describe the key types of plots used and their role in uncovering meaningful insights from the data.

2.7.1 Boxplot

A box plot is a graphical tool used to display the distribution of a numerical variable and identify potential outliers. It summarizes the data through five key values: minimum, first quartile (Q1), median, third quartile (Q3), and maximum. Box plots also highlight values that fall significantly outside the typical range (1.5×IQR), which are considered potential outliers. In this study, box plots were used to visually inspect features such as instrumentalness, speechiness, and others to identify unusual values. This step supported outlier detection.

2.7.2 Histogram

A histogram is a graphical representation that shows the distribution of a numerical variable by grouping values into continuous intervals, or "bins." It helps in understanding how data is spread across different value ranges. In this study, histograms were used to analyze the distribution of features such as popularity and energy. This allowed for an initial understanding of whether the data was concentrated around specific values, skewed toward one end, or uniformly distributed. Such insights are essential for identifying data imbalance and deciding whether transformations (like normalization or power transforms) are needed in the preprocessing phase.

2.7.3 Scatter plot

A scatter plot is a commonly used graphical tool for visualizing the relationship between two continuous numerical variables. Each point on the plot represents an individual observation, with its position determined by the values of the two variables being compared. Scatter plots are particularly useful for identifying patterns, clusters, or potential correlations in the data. In this study, a scatter plot was used to examine the relationship between loudness and energy, helping to assess how these features interact and whether any consistent trend or association could be observed

2.7.4 Line Plot

A line plot is a basic yet effective graphical tool used to display the relationship between two continuous variables, typically showing trends or patterns across a range of values. It connects individual data points with lines, making it especially useful for identifying smooth changes, fluctuations, or general movement between variables. In this study, a line plot was used to visualize the relationship between acousticness and energy, helping to explore how these two features behave in relation to each other. This type of plot supported the understanding of their interaction and guided further analysis

2.7.5 Correlation heatmap

A correlation heatmap is a graphical representation of the pairwise correlation coefficients between numerical variables in a dataset. It uses color gradients to indicate the strength and direction of linear relationships, making it easy to visually detect patterns, strong associations, or potential redundancy among features. In this study, a correlation heatmap was used to assess how various musical features — such as energy, acousticness, valence, and loudness — relate to one another. This helped guide feature selection, supported statistical interpretation, and provided a clearer understanding of overall data structure.

2.8 Machine Learning Models

In this project, several machine learning models were used to predict the popularity of songs based on their musical features. The models were chosen to represent a balance between performance, interpretability, and variety — combining both ensemble-based approaches and a simpler, distance-based method. In addition to regression models, Principal Component Analysis (PCA) was used to reduce dimensionality and explore feature relationships more effectively. A brief overview of each technique is given below

2.8.1 Random Forest Regressor

The Random Forest Regressor is an ensemble model that builds a collection of decision trees and combines their outputs to make predictions. It's especially good at handling complex, nonlinear relationships and doesn't overfit as easily as a single decision tree. This model was chosen as a strong baseline because it performs well even without much tuning and gives useful insights into feature importance (Breiman, 2001).

2.8.2 XGBoost Regressor

XGBoost (short for Extreme Gradient Boosting) is a powerful and efficient model known for its speed and accuracy. It works by building a series of trees that learn from the errors of previous ones and includes techniques like regularization to prevent overfitting. It has become one of the most widely used models in real-world data science tasks because of its consistent performance (Chen & Guestrin, 2016).

2.8.3 Gradient Boosting Regressor

Gradient Boosting is a more traditional boosting method where trees are added sequentially, and each new tree focuses on correcting the mistakes made by the earlier ones. It's a flexible and effective method for regression, especially when the data has complex patterns. Though slower to train than Random Forest, it often gives highly accurate results when tuned carefully (Friedman, 2001).

2.8.4 K-Nearest Neighbors (KNN) Regressor

KNN is a simple, intuitive algorithm that makes predictions based on the average of the closest examples in the data. It doesn't learn a model beforehand — instead, it uses the whole dataset during prediction, which makes it easy to understand but slower with large datasets. It's useful for checking how well simple, non-parametric models perform in comparison to more complex ones.

2.8.5 Principal Component Analysis (PCA)

PCA is not a prediction model but a technique used to reduce the number of input features by combining them into fewer, uncorrelated variables (called principal components). It helps simplify the dataset while keeping most of the important variation. In this project, PCA was used to visualize high-dimensional data and support clustering by making the patterns easier to see and work with (Jolliffe & Cadima, 2016).

2.9 Evaluation Metrics

This section outlines the key metrics used to evaluate both the regression models and the clustering algorithm in this study. These metrics were selected based on their relevance to the type of model used and the nature of the task (supervised vs. unsupervised learning).

2.9.1 P value

The p-value, or probability value, is a fundamental concept in hypothesis testing that helps determine the strength of evidence against the null hypothesis. It represents the probability of observing the given result, or one more extreme, assuming that the null hypothesis is true. A smaller p-value indicates stronger evidence against the null hypothesis, often leading to its rejection, while a larger p-value suggests weaker evidence and typically results in retaining the null hypothesis. Common thresholds, such as 0.05 or 0.01, are used to assess significance, but the p-value should always be interpreted within the context of the study, considering factors like sample size and study design.

2.9.2 Mean Squared Error (MSE)

The Mean Squared Error is a widely used metric for evaluating regression models. It measures the average squared difference between the actual and predicted values. MSE is sensitive to large errors because it squares the residuals, giving more weight to outliers. A lower MSE value indicates a more accurate and reliable model, as it means the predictions are closer to the true values.

2.9.3 R-squared Score (R^2)

The R-squared score, or coefficient of determination, indicates how well the regression model explains the variance in the target variable. An R^2 value of 1 represents perfect prediction, while a score of 0 suggests that the model fails to explain any of the variance. In this project, R^2 was used to understand the explanatory power of different regression models in predicting song popularity.

2.9.4 Silhouette Score

The Silhouette Score was used to evaluate the clustering process, which aimed to group songs based on audio features for mood-based analysis. This score measures how similar each data point is to its own cluster compared to other clusters. It ranges from –1 to +1, where a higher score indicates well-defined and meaningful clusters. The silhouette score was especially useful for assessing whether the mood-based song groupings were clearly separated and internally consistent.

2.10 Clustering of Songs Based on Emotion-Related Audio Features

To uncover hidden patterns in musical characteristics and group songs based on their emotional tone, unsupervised clustering was performed using the K-Means algorithm. A selected set of audio features — namely valence, energy, acousticness, danceability, tempo, and loudness — were identified as the most relevant for reflecting a song's emotional nature. These features were extracted into a separate dataset and used as input for the clustering process.

The number of clusters was set to four, based on domain intuition and experimentation. After applying K-Means, each track was assigned to one of the four clusters. To better interpret the characteristics of these clusters, the mean values of the mood-related features were computed for each group. Based on these averages, clusters were labeled with approximate emotional categories: Happy, Sad, Aggressive, and Chill. These mood names are meant to simplify interpretation and do not reflect precise psychological or genre-based classifications.

To support visualization and better understand how the tracks were grouped, Principal Component Analysis (PCA) was applied to reduce the six-dimensional feature space to two principal components. This allowed the clusters to be plotted on a two-dimensional graph, providing an intuitive view of how songs were spread across the emotional spectrum. Color coding was used to distinguish between the four mood categories in the resulting scatter plot.

The quality of clustering was evaluated using the Silhouette Score, a common metric in unsupervised learning that reflects how well each data point fits within its assigned cluster. The score obtained was 0.47, indicating moderate clustering performance. This suggests that while there is some structure and separation in the data, the clusters are not highly distinct — which is understandable given the subjective and overlapping nature of musical emotion. As such, the mood-based clustering in this project should be viewed as an exploratory tool rather than a strict classification system.

2.11 Recommendation System Design

This project also included the development of two simple recommendation systems to demonstrate how audio features can be used to support personalized music suggestions. These systems were built on top of the previously explored clustering and feature extraction methods

and served as applications of the insights gained from unsupervised learning and feature analysis.

2.11.1 Mood-Based Recommendation System

The first recommendation system was built using the mood labels generated from the K-Means clustering results. After assigning each song to one of four approximate mood categories — Happy, Sad, Aggressive, or Chill — a basic function was developed to allow users to receive recommendations based on their current emotional state. When the user selects a mood, the system randomly retrieves ten songs that belong to the corresponding mood cluster. This provides a quick and interactive way to explore music that aligns with a desired emotional tone. While the mood labels are only approximations based on audio features, they offer a playful and intuitive entry point for music discovery.

2.11.2 Content-Based Recommendation System Using Cosine Similarity

In addition to mood-based suggestions, a second system was developed using content-based filtering. This system allows users to receive recommendations based on the musical similarity of a selected song. A set of audio features — including danceability, energy, valence, tempo, acousticness, speechiness, instrumentalness, and liveness — was used to represent each track. Given a reference song name, the system computes the cosine similarity between its feature vector and all other songs in the dataset. It then ranks songs by similarity and returns the top five most similar tracks, excluding the reference song itself.

This method allows users to discover songs with a similar sound or structure, regardless of genre or mood category. It demonstrates how audio-based similarity metrics can be applied to build simple yet effective recommendation systems that respond to user input and musical context.

Chapter 3

Exploratory Data Analysis (EDA)

3.1 Introduction

This chapter presents the exploratory data analysis (EDA) carried out to understand the characteristics and underlying patterns in the dataset. It includes both univariate and bivariate analyses to examine the distribution of individual features and the relationships between them. Genre-based patterns and outlier detection are also explored. Visual tools such as histograms, box plots, scatter plots, and heatmaps are used throughout to support interpretation. The findings from this chapter help guide the modeling and testing steps that follow.

3.2 Univariate analysis

Univariate analysis involves the examination of a single variable at a time to understand its distribution, central tendency, and spread. It is often the first step in exploratory data analysis and helps in identifying the basic structure of the data. In this study, univariate analysis was conducted on key features such as popularity, energy, loudness, valence, and speechiness, using tools like histograms and box plots. This allowed for a clearer understanding of how each feature is distributed, whether the data is skewed or balanced, and the presence of any outliers. The results from this stage provide important context before moving into more complex relationships between variables.

Univariate analysis was performed to understand the individual behavior of key features within the dataset. This includes examining the distribution of popularity scores, which provides insight into how song success is spread across the platform, and the distribution of energy levels, which reflects the general intensity and liveliness of tracks. Additionally, the frequency of explicit versus non-explicit content was analyzed to identify content-based imbalances. These visual summaries help highlight dominant patterns, skewness, and potential biases within the dataset, forming the foundation for further comparative and predictive analysis.

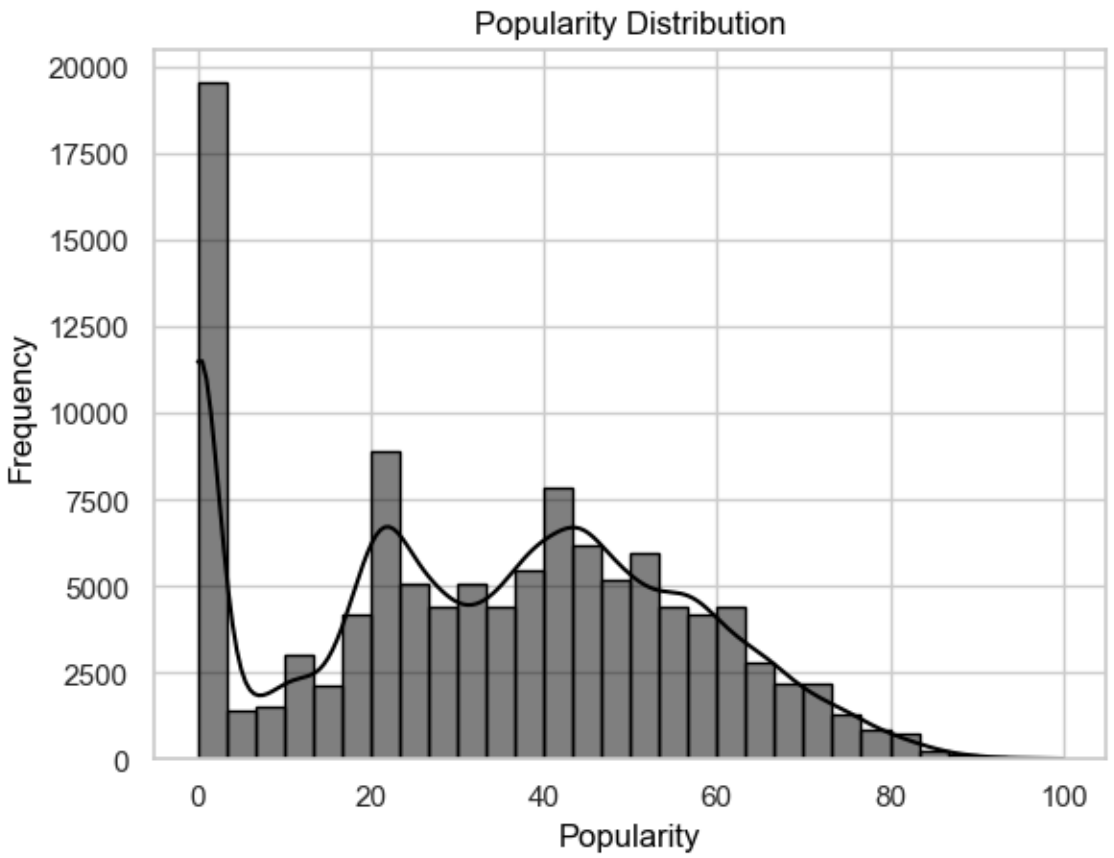

Figure 3.1. Popularity Distribution

From Figure 3.1, the histogram illustrates the distribution of song popularity scores across the dataset. A striking concentration of tracks falls within the lowest popularity range, particularly between 0 and 10, indicating that a significant portion of songs receive very little user engagement. As the popularity score increases, the number of tracks gradually declines, with relatively few songs achieving high popularity scores above 70. This right-skewed distribution reveals that most songs on the platform do not attain mainstream popularity, suggesting either niche appeal or limited exposure. The shape of the distribution highlights the imbalanced nature of the dataset, which could potentially influence the performance and fairness of predictive models built on this data.

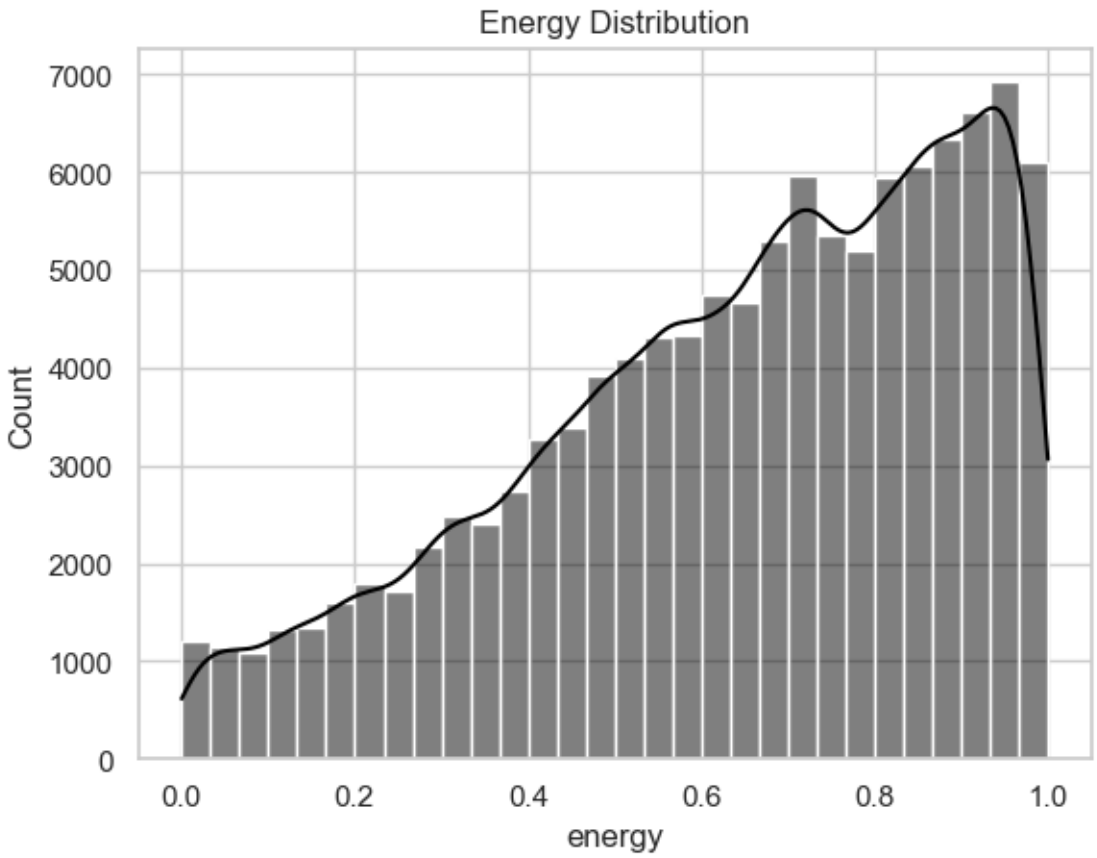

Figure 3.2. Energy Distribution

From Figure 3.2, the histogram reveals the distribution of energy values across the dataset, showing a clear upward trend. The majority of tracks exhibit high energy levels, with counts steadily increasing as energy approaches its maximum value of 1.0. This suggests that the dataset is predominantly composed of energetic and intense songs, possibly reflecting user preferences or the characteristics of commercially successful tracks. Low-energy tracks are relatively rare, indicating that calm or mellow music is underrepresented in the dataset. The shape of the distribution may influence modeling by introducing a bias toward high-energy features when predicting popularity or classifying song moods.

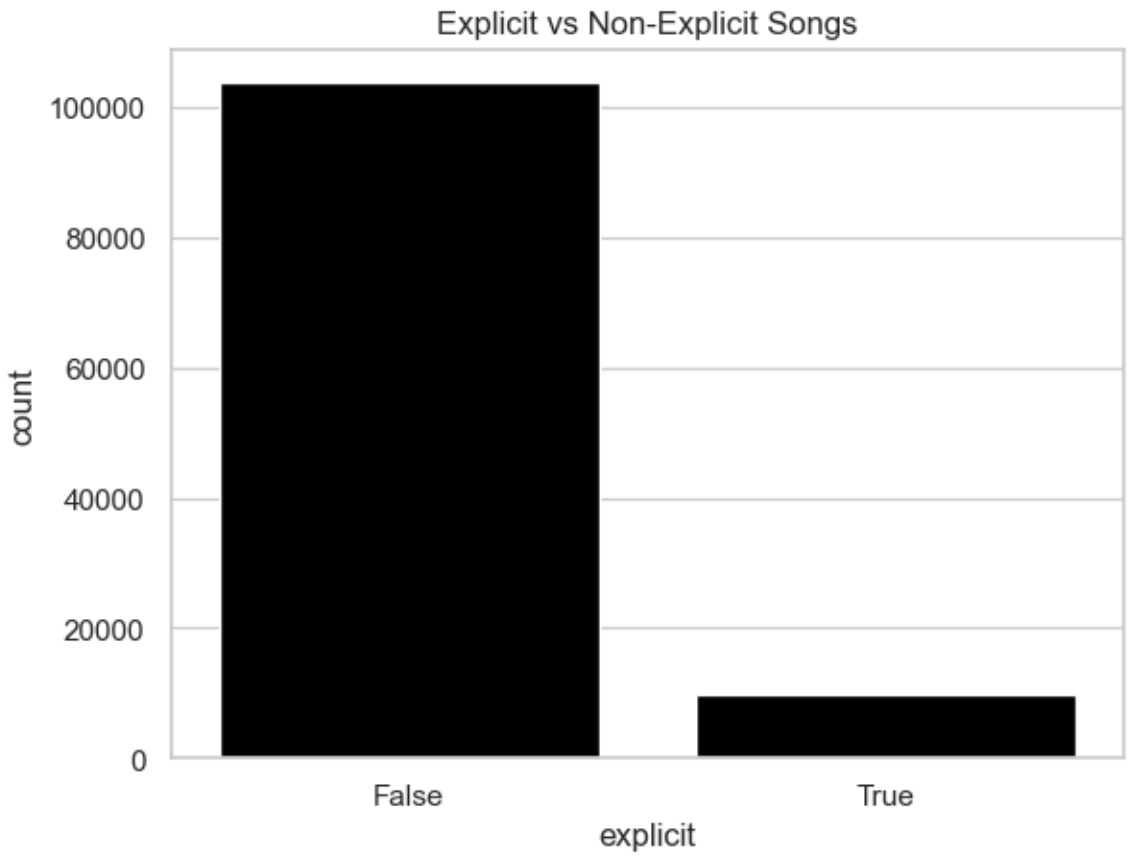

Figure 3.3. Frequency of Explicit and Non-Explicit Tracks in the Dataset

Figure 3.3. presents the distribution of songs based on explicit content. The count plot reveals a striking imbalance, with non-explicit tracks overwhelmingly dominating the dataset. The number of explicit songs is significantly lower, indicating that the majority of music in the dataset adheres to general or family-friendly content standards. This pattern could be due to broader market appeal, platform policies, or simply the characteristics of the curated dataset. The underrepresentation of explicit songs suggests that explicit content is not the norm, which may influence model behavior by reducing the exposure to such examples during training.

3.3 Bivariate Relationships

Bivariate analysis involves examining the relationship between two variables to understand how they may influence or interact with one another. Unlike univariate analysis, which focuses on a single feature at a time, bivariate analysis allows for the discovery of correlations, trends,

or contrasts between pairs of variables. This type of analysis is especially useful for identifying linear or non-linear associations, potential dependencies, or patterns that may inform predictive modeling and feature selection.

In the context of this study, bivariate analysis was used to investigate how certain audio features interact with one another and with the target variable, popularity. For example, we explored whether features like energy and loudness show a direct relationship, or whether acousticness tends to decrease as energy increases. Understanding these relationships helps reveal the musical dynamics behind popular tracks. Scatter plots, line plots, and box plots were used to visually capture these interactions and provide deeper insight into the structure of the dataset.

This helped in assessing whether changes in one feature were associated with shifts in another and laid the groundwork for more complex modeling later. By using visual tools such as scatter plots, box plots, and correlation heatmaps, bivariate analysis helped uncover structural insights and guided important decisions in feature selection, statistical testing, and overall interpretation. It provided a necessary foundation for evaluating whether certain audio characteristics or metadata elements have meaningful connections with track popularity or with each other.

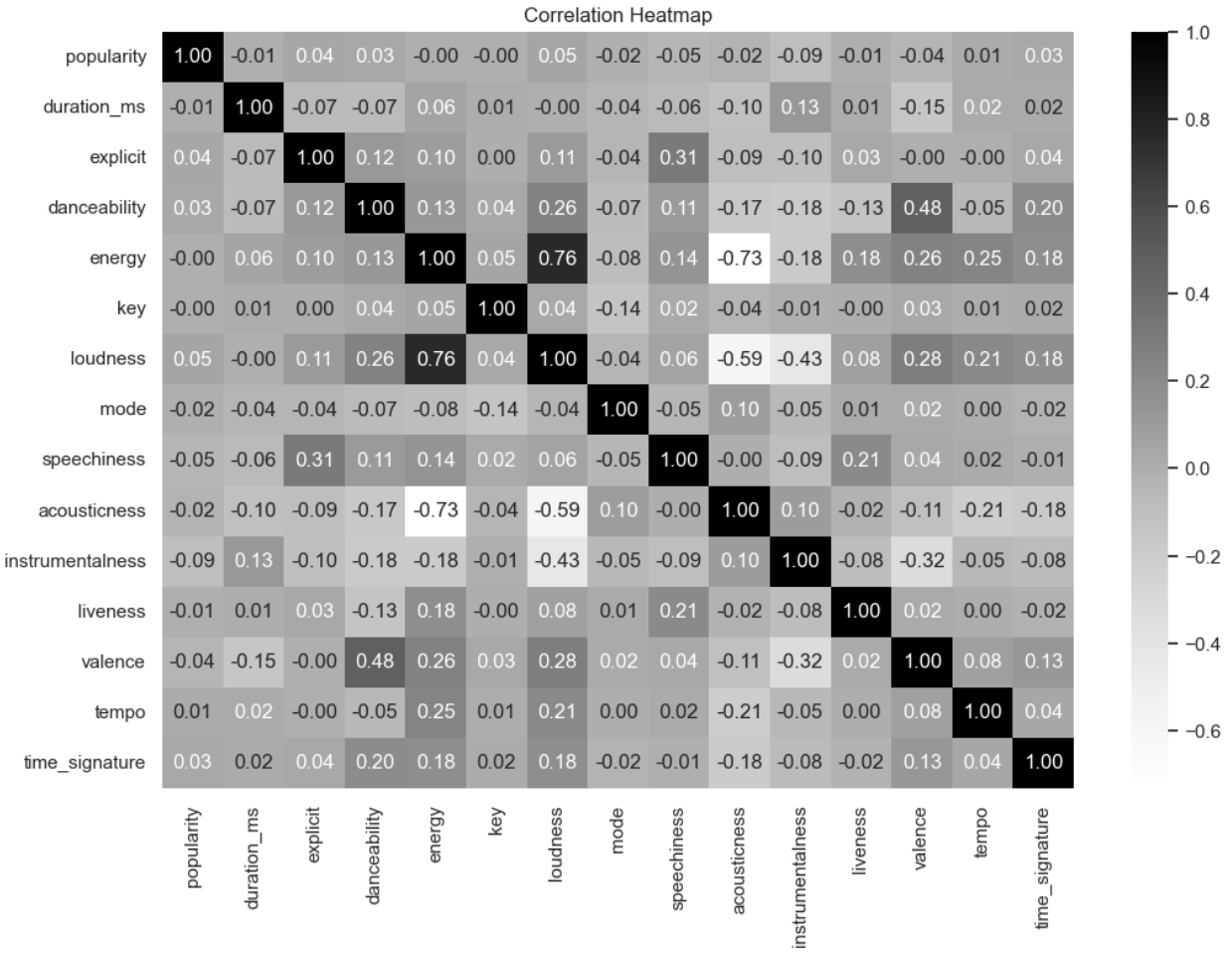

Figure 3.4. Correlation heatmap of numerical features

Figure 3.4. shows the correlation between various numerical features in the dataset. A strong positive relationship is seen between energy and loudness (r = 0.76), indicating that louder songs tend to be more energetic. In contrast, energy and acousticness are strongly negatively correlated (r = –0.73), suggesting that high-energy songs are rarely acoustic. A moderate positive correlation also exists between danceability and valence (r = 0.48), meaning happier songs are often more danceable. Most other correlations, including those with popularity, are weak, highlighting the complex and multifactorial nature of track popularity.

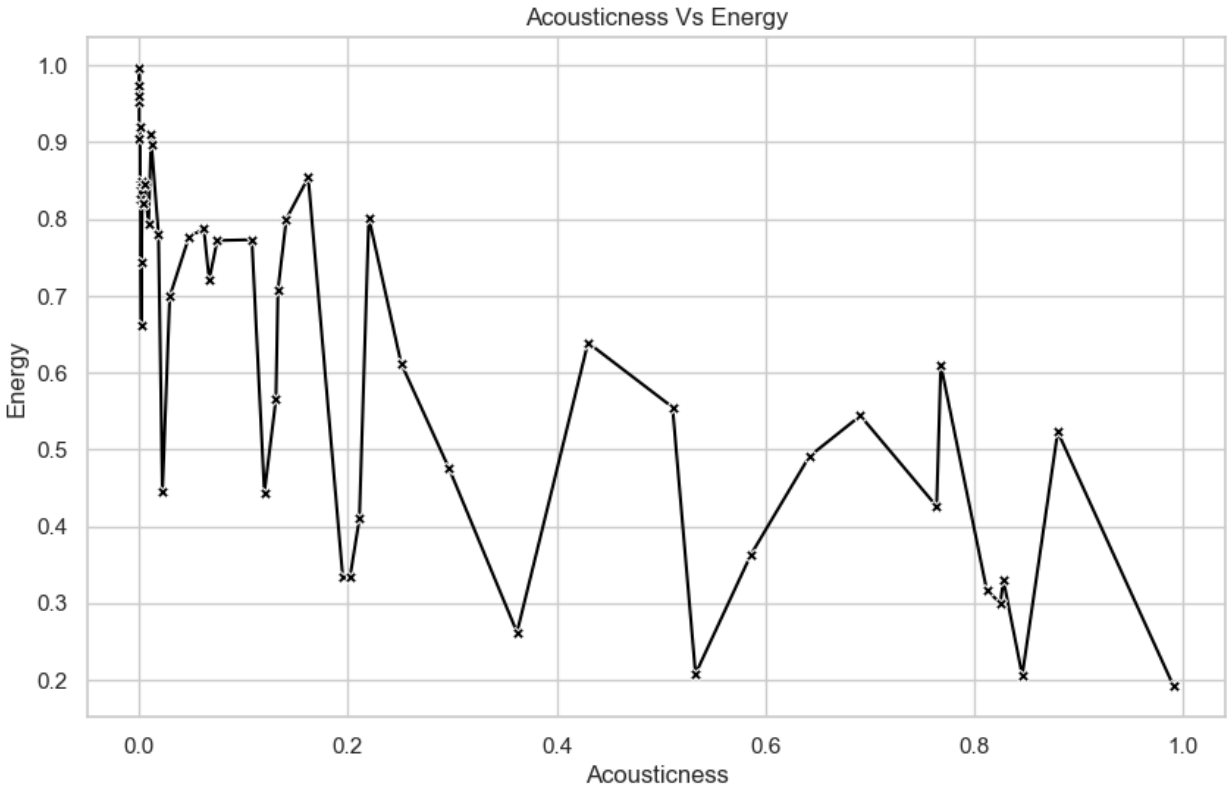

Figure 3.5. Relationship Between Acousticness and Energy

Figure 3.5 visualizes the relationship between acousticness and energy, revealing a clear negative trend. As acousticness increases, energy generally decreases, indicating that songs with more acoustic characteristics tend to be less energetic. While the line shows some local fluctuations, the overall pattern supports the strong negative correlation seen in the heatmap. This suggests that acoustic songs are more likely to be mellow or softer in nature.

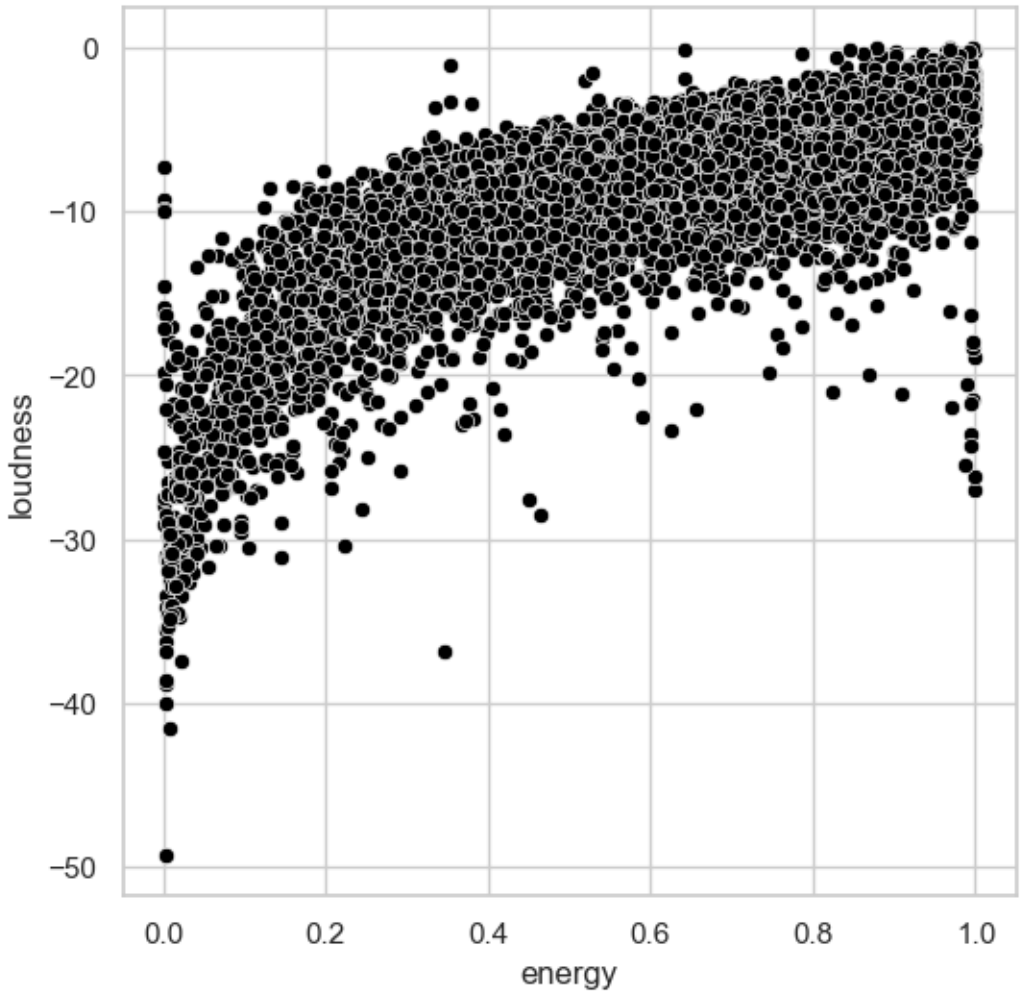

Figure 3.6 Relationship Between Loudness and Energy

Figure 3.6 shows a positive relationship between loudness and energy. As energy increases, loudness values tend to move closer to 0 dB, indicating louder sound levels. Most high-energy tracks cluster in the upper-right area, suggesting that energetic songs are typically louder. This visual trend aligns with the strong positive correlation observed earlier, reinforcing the connection between perceived intensity and volume in music.

3.4 Genre-Based Patterns

This section investigates the role of musical genres in shaping both track popularity and distinctive audio characteristics. Genres often reflect not only stylistic preferences but also patterns in composition, instrumentation, and production. To better understand these relationships, the dataset was analyzed from two complementary angles. First, a genre-based classification of artists was developed by identifying each artist's dominant genre — the genre in which they had the highest number of tracks — and comparing it to their non-dominant genres, which represent genres they explored less frequently. This allowed for a comparison of popularity distributions across dominant and non-dominant genres, helping determine whether artists tend to be more successful when performing within their primary genre category. The classification was performed using a track count–based grouping method that tallied each artist's genre contributions and assigned the most frequent one as their dominant genre.

In addition to analyzing popularity, this section also uses the track genre to interpret outliers in several musical features. Specifically, outliers in instrumentalness, speechiness, tempo, and liveness were detected using the interquartile range (IQR) method, and then mapped back to their corresponding genres to investigate whether certain genres are more likely to produce songs with extreme values. For example, some genres might consistently generate high-instrumentalness tracks, while others may lean toward highly live or speech-heavy compositions. This multi-dimensional approach to genre analysis highlights not only how genre influences commercial success but also how it shapes the sonic character of a track, offering useful insights for both data-driven exploration and musical interpretation.

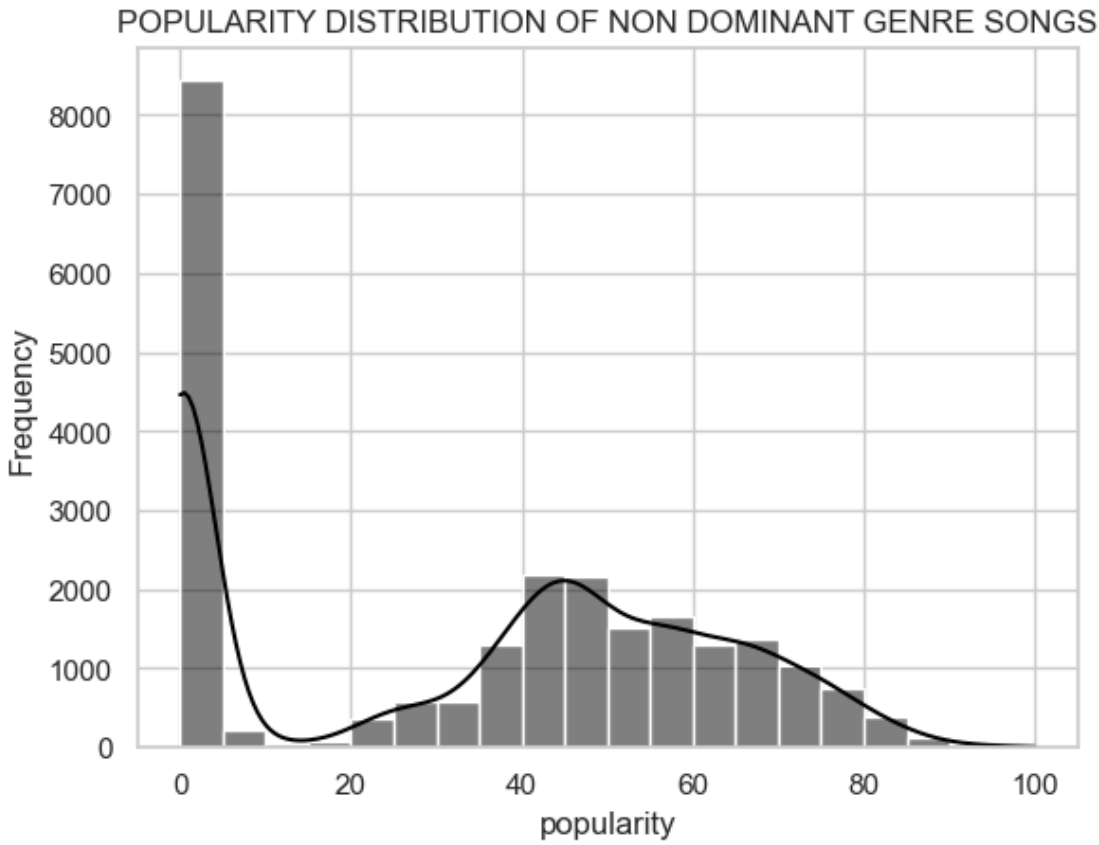

Figure 3.7. Popularity of non-dominant genre songs

Figure 3.7 shows the distribution of popularity scores for songs that fall outside of an artist's dominant genre. The distribution is highly right-skewed, with a large concentration of tracks having very low popularity. Only a small proportion of these non-dominant genre songs achieve high popularity.

This pattern is similar to the overall popularity distribution observed in the dataset, where most tracks tend to have low popularity scores. However, the skew is even more pronounced in non-dominant genre songs, suggesting that artists are generally less successful when experimenting outside their main genre, possibly due to reduced listener familiarity or weaker fanbase association.

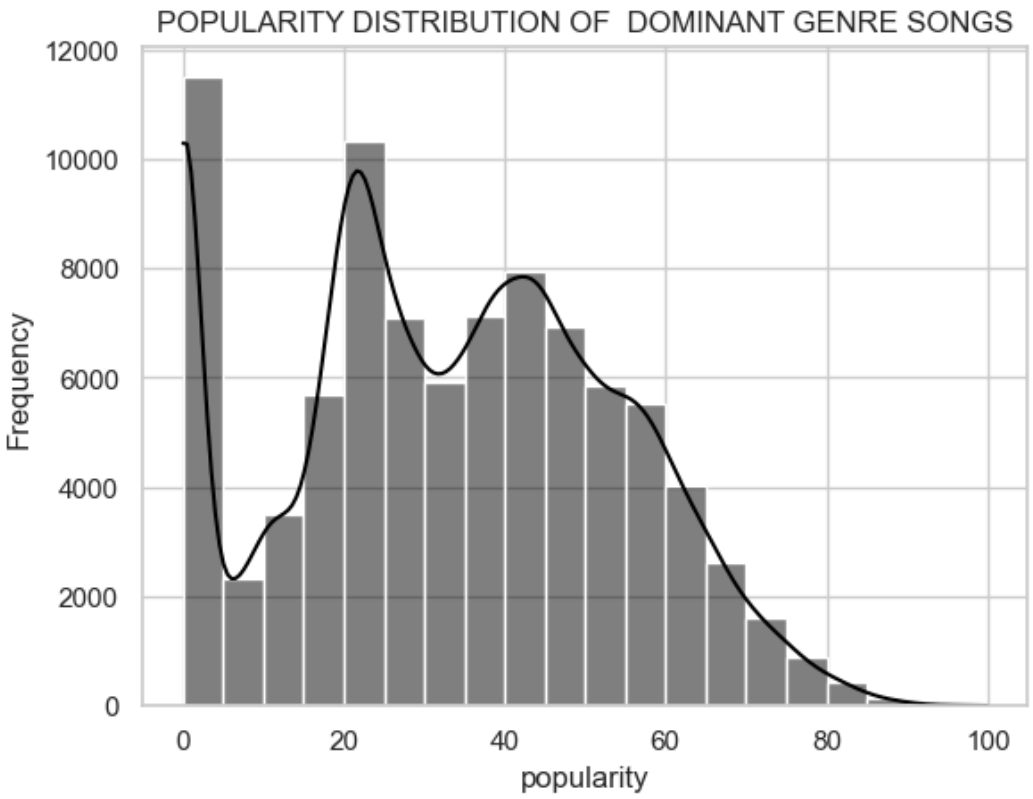

Figure 3.8 Popularity of dominant genre songs

Figure 3.8 reveals key insights about audience preferences and artist performance within established genres. The distribution shows that songs aligning with an artist's dominant genre tend to achieve more consistent popularity levels, with a significant proportion reaching moderate to high popularity scores. This pattern suggests that listeners demonstrate stronger engagement with artists working within their recognized musical style, likely due to established fan expectations and algorithmic reinforcement from streaming platforms. The concentration of tracks in higher popularity ranges underscores the commercial advantage of genre consistency, where artists benefit from audience familiarity and targeted recommendations. However, the presence of some lower-popularity tracks within the dominant genre indicates that even stylistic consistency does not guarantee universal appeal, pointing to additional factors influencing song reception such as quality, promotion, or timing. These findings highlight the importance of genre alignment in music strategy while acknowledging the complex interplay of elements that ultimately determine a track's success.

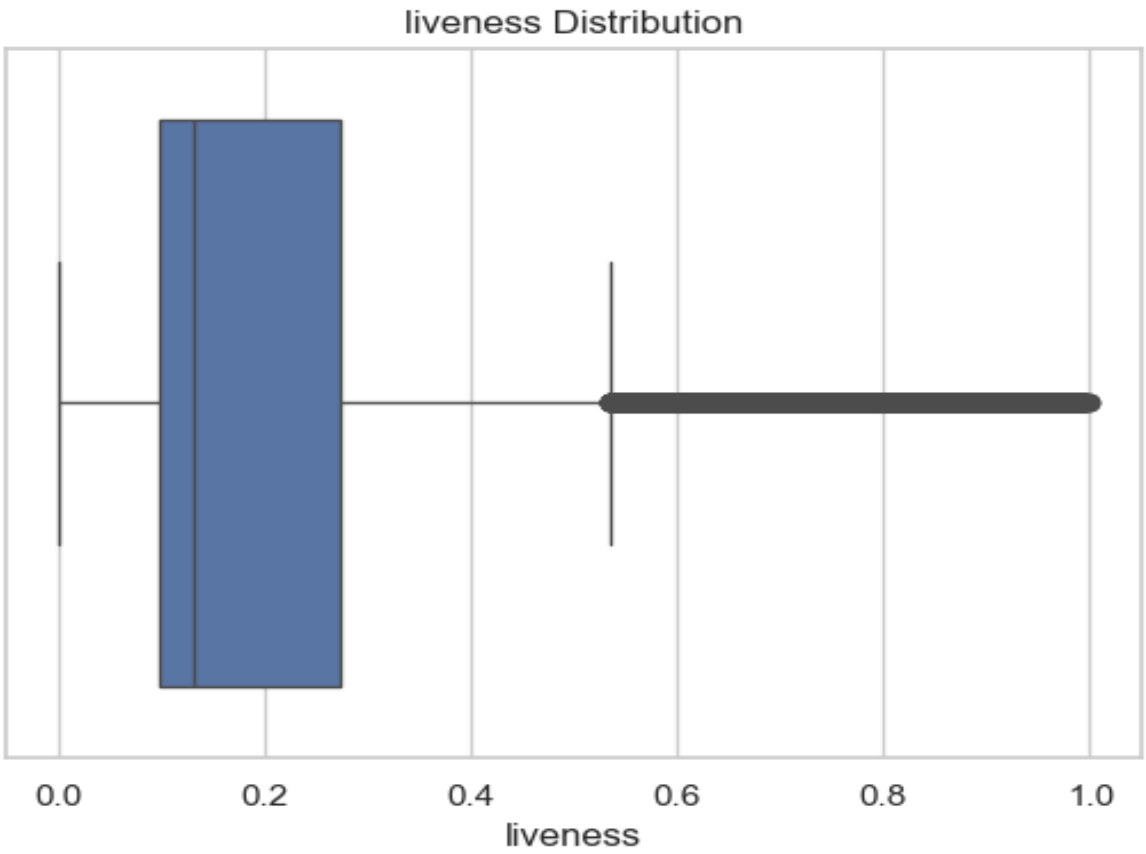

Figure 3.9 Liveness Value Spread with Outliers

Figure 3.9 shows the distribution of liveness values, which is heavily skewed toward the lower end, with most tracks falling between 0.0 and 0.2. A small group of tracks displays unusually high liveness values, identified as outliers using the interquartile range (IQR) method.

To further interpret these outliers, the genres of the high-liveness tracks were extracted. A significant number belonged to performance-oriented or culturally specific genres such as comedy, pagode, sertanejo and samba. This genre-based pattern suggests that liveness outliers are not random but reflect the nature of certain musical styles that include live audience elements or ambient performance settings.

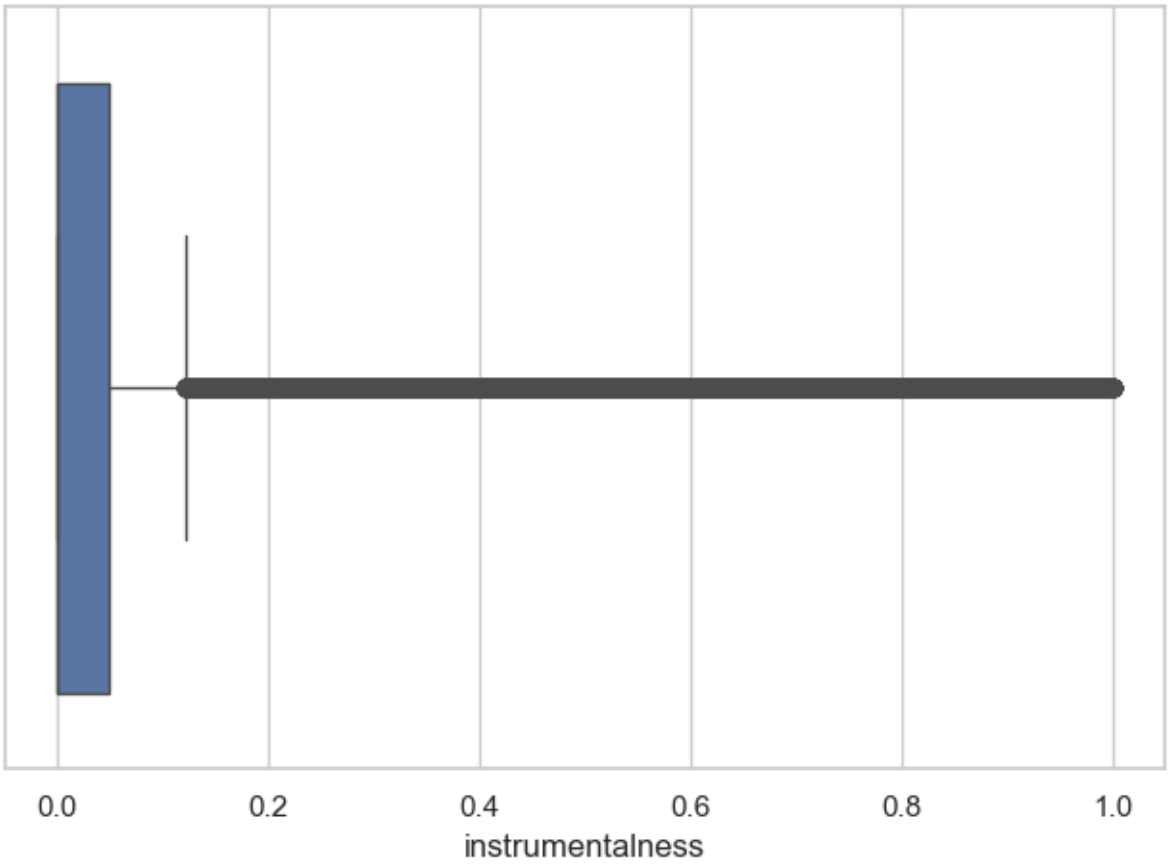

Figure 3.10. Instrumentalness Value Spread with Outliers

Figure 3.10 shows a clear bimodal distribution of instrumentalness values, with strong concentrations near 0.0 and 1.0. This indicates that most tracks are either highly vocal or fully instrumental, with very few falling in the mid-range. The dominant peak at 0.0 suggests that vocal-driven tracks remain the standard in popular music, while the secondary peak at 1.0 reflects the presence of purely instrumental genres such as classical, electronic, or ambient music.

Tracks with extreme instrumentalness values were further examined by genre. Many high-instrumentalness outliers belonged to genres like minimal techno, Detroit techno, sleep, and new age — styles known for their atmospheric or instrumental focus. This reinforces the idea that instrumentalness is not randomly distributed but strongly tied to specific genre characteristics.

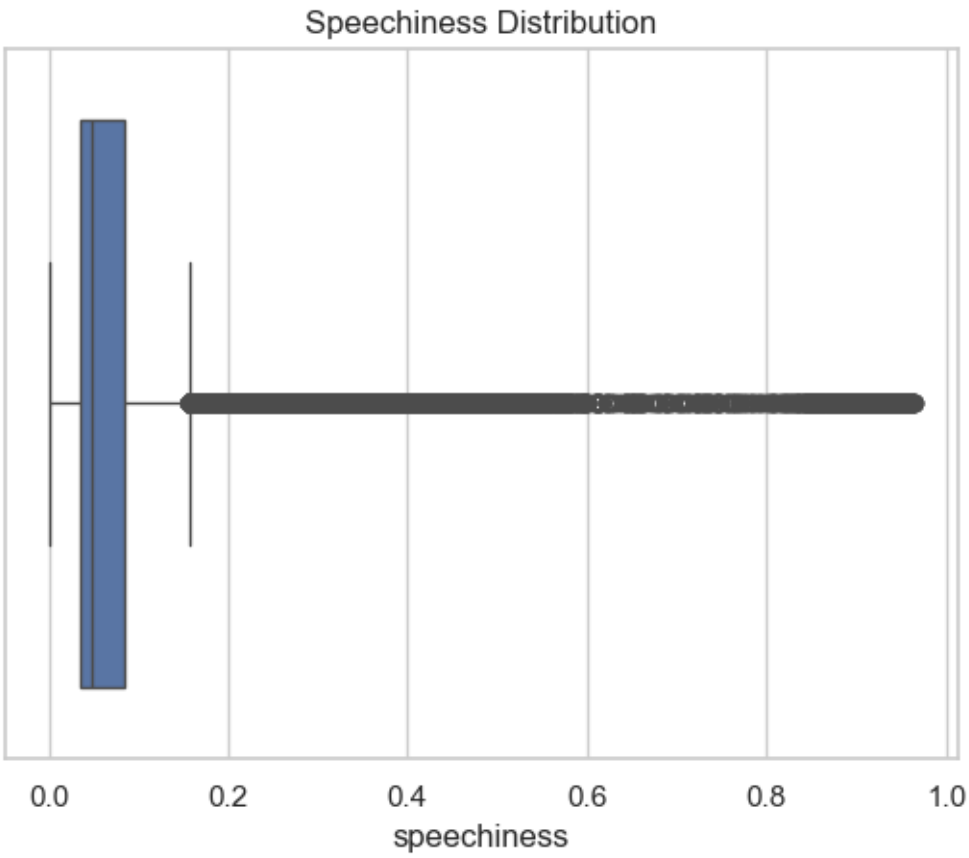

Figure 3.11. Speechiness Value Spread with Outliers

Figure 3.11 displays a strongly right-skewed distribution of speechiness values, with the vast majority of tracks falling below 0.3. This reflects a general trend in mainstream music toward sung or melodic vocals rather than speech-driven content. Only a small cluster of tracks appears in the high range (above 0.66), which likely includes genres such as rap, spoken word, or other speech-heavy formats.

Outlier analysis revealed that tracks with unusually high speechiness values often belonged to genres such as comedy, j-dance, dancehall, hardcore, and kids, many of which naturally incorporate spoken vocals, narration, or character voices. The mid-range (0.4–0.66) remains sparsely populated, suggesting that tracks tend to be either predominantly musical or clearly speech-based, with little middle ground.

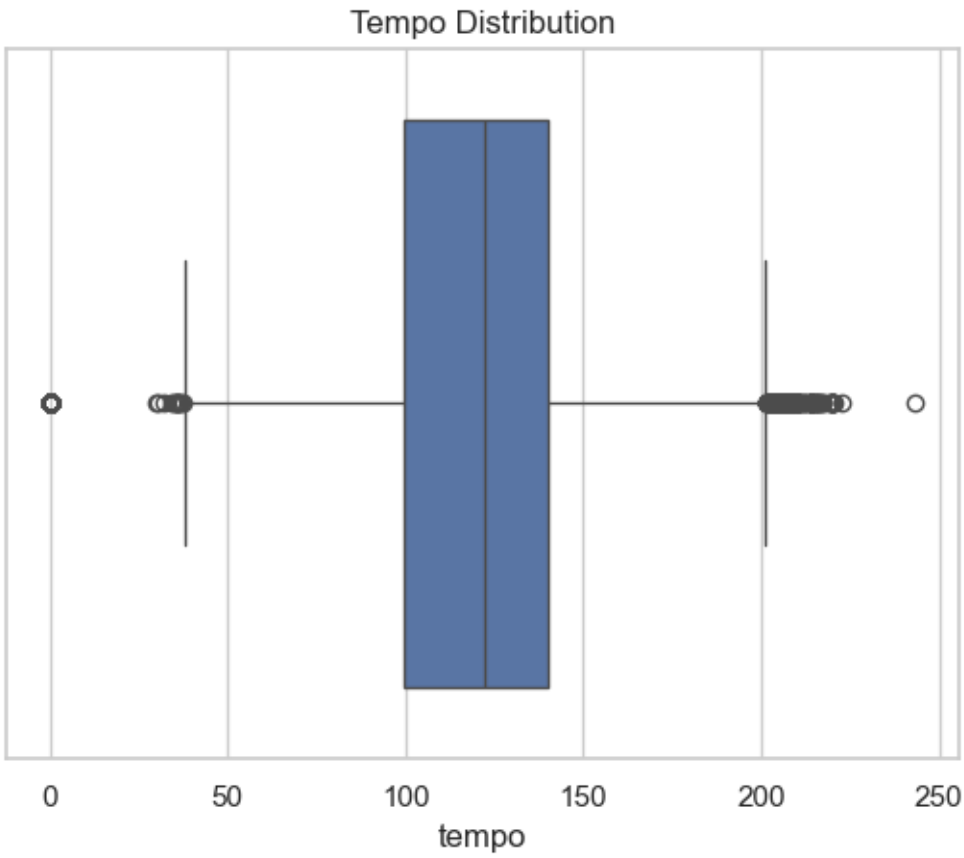

Figure 3.12. Tempo Value Spread with Outliers

Figure 3.12 shows a near-normal distribution of tempo values, with most tracks falling between 90 and 150 BPM. The central peak is concentrated around 120 BPM, aligning with the rhythmic conventions of mainstream genres such as pop, hip-hop, and electronic music. This balanced distribution reflects an industry-wide preference for tempos that support danceability and general listener comfort.

Lower tempo outliers (below 60 BPM) were largely associated with genres like sleep, which favor slow, ambient pacing. In contrast, upper tempo outliers (above 200 BPM) were primarily found in genres such as piano, rock, and songwriter, which may emphasize energetic or intricate rhythmic structures. These extremes, while rare, demonstrate the tempo diversity present in niche and emotionally expressive genres.

3.5 Summary of Observations

In the univariate analysis, features like popularity and energy exhibited skewed distributions. Most tracks had low popularity scores, indicating that viral or highly popular songs make up only a small portion of the dataset. Energy was concentrated near the upper end of the scale, suggesting a general trend toward high-energy music.

Bivariate relationships revealed a strong positive correlation between energy and loudness, and a strong negative correlation between energy and acousticness. These relationships highlight that louder tracks tend to be more energetic and that acoustic elements are more common in lower-energy music.

Tracks that belonged to an artist's dominant genre tended to have higher popularity scores compared to those in non-dominant genres, indicating stronger audience alignment and better reception when artists produce music within their primary style. Genre analysis was also applied to extreme feature values, revealing that many outliers were not random anomalies but were instead associated with specific genres that naturally exhibit those characteristics. Notably, the sleep genre appeared consistently across multiple outlier cases — including low tempo, high instrumentalness, high acousticness, and low energy — suggesting that it is inherently aligned with such musical features. This emphasizes that outliers in the dataset often reflect intentional stylistic choices rather than errors or noise.

Chapter 4:

Results and Interpretation

4.1 Introduction

This chapter presents the results of the analyses and models developed throughout the study, along with interpretations of their significance. It begins with the outcomes of statistical hypothesis testing, which evaluates whether categorical features such as explicit content and musical mode have a significant impact on track popularity. The performance of various regression models is then compared, with a focus on the best-performing model and its evaluation through metrics such as mean squared error and R^2 score. Feature importance is also analyzed to understand which audio or metadata features contribute most to predicting popularity. Additionally, the chapter includes the outcomes of mood-based clustering and the implementation of a basic recommendation system, both of which were developed using unsupervised learning techniques. Together, these results provide a comprehensive view of how musical and contextual features influence a track's popularity and how machine learning can be applied to model and interpret those patterns.

4.2 Statistical Testing

Statistical hypothesis tests were conducted to determine whether certain categorical features — namely explicit content and musical mode (major or minor) — have a significant effect on the popularity of tracks. As the data did not meet normality assumptions (based on the Shapiro-Wilk test), the Mann-Whitney U test was used for both comparisons.

4.2.1 Popularity Differences by Explicit Content

To assess whether explicit tracks differ in popularity compared to non-explicit ones, the following hypotheses were formulated:

H_0: There is no significant difference in popularity between explicit and non-explicit tracks.

H_1: There is a significant difference in popularity between the two groups.

The Mann-Whitney U test returned a p-value of 0.000, indicating a statistically significant difference between the groups.

Table 4.1: Popularity Statistics by Explicitness

Track type	Mean popularity	Median popularity
Explicit	36.52	37.5
Non-Explicit	33.02	34

As shown in Table 4.1, explicit tracks exhibit slightly higher mean and median popularity than non-explicit ones. Although the difference is modest, it is statistically significant. This suggests that tracks marked as explicit may perform marginally better in terms of listener engagement or streaming counts.

4.2.2 Popularity Differences by Musical Mode (Major/Minor)

To determine whether a track's musical mode (major or minor) influences its popularity, the following hypotheses were tested:

H_0: There is no significant difference in popularity between major and minor mode tracks.

H_1: There is a significant difference in popularity based on mode.

Again, the Mann-Whitney U test was used, resulting in a p-value of 0.000, indicating statistical significance.

Table 4.2: Popularity Statistics by Musical Mode

Track type	Mean popularity	Median popularity
Major	33.07	34.0
Minor	33.76	35.0

Table 4.2 shows that minor mode tracks have slightly higher average and median popularity scores than major mode tracks. While the difference is relatively small, it is statistically significant, implying that mode may have a subtle influence on listener preference or track performance.

4.3 Regression Analysis

4.3.1 Linear regression

A linear regression model yielded a moderate R^2 score (≈ 0.73) but was excluded due to weak linearity in the features and limited interpretability. More flexible non-linear models were explored for improved generalization.

4.3.2 Tree-Based Regression Models

To capture the non-linear relationships between musical features and track popularity, several regression algorithms were tested and compared. These included ensemble-based models such as Random Forest, XGBoost, and Gradient Boosting, as well as Linear Regression and K-Nearest Neighbors (KNN). Each model was evaluated using R^2 score and Mean Squared Error (MSE) on the test dataset.

Table 4.3: Comparison of Regression Models

Model	Key Parameters (tuned)	R^2 Score	MSE
Random Forest	n_estimators=100, max_depth=20, random_state=42	0.7788	110.95
Gradient Boosting	learning_rate=0.1, max_depth=5	0.7443	128.24
XGBoost	reg_alpha=0.1, reg_lambda=1.0, max_depth=5	0.7476	129.23

LightGBM	n_estimators=200, learning_rate=0.07, max_depth=7, num_leaves=31	0.7462	127.28
K-Nearest Neighbors	n_neighbors=5, weights='distance'	0.6638	168.59

Table 4.3. shows that the Random Forest Regressor achieved the best overall performance, with the highest R^2 score (0.7788) and lowest MSE (110.95). This model was selected for further interpretation, including feature importance analysis and prediction evaluation.

4.3.3 Feature Importance Analysis

To evaluate which features contributed most significantly to the prediction of track popularity, a feature importance graph was generated using the trained Random Forest Regressor. This plot ranks each feature according to its relative contribution in improving the model's predictive accuracy.

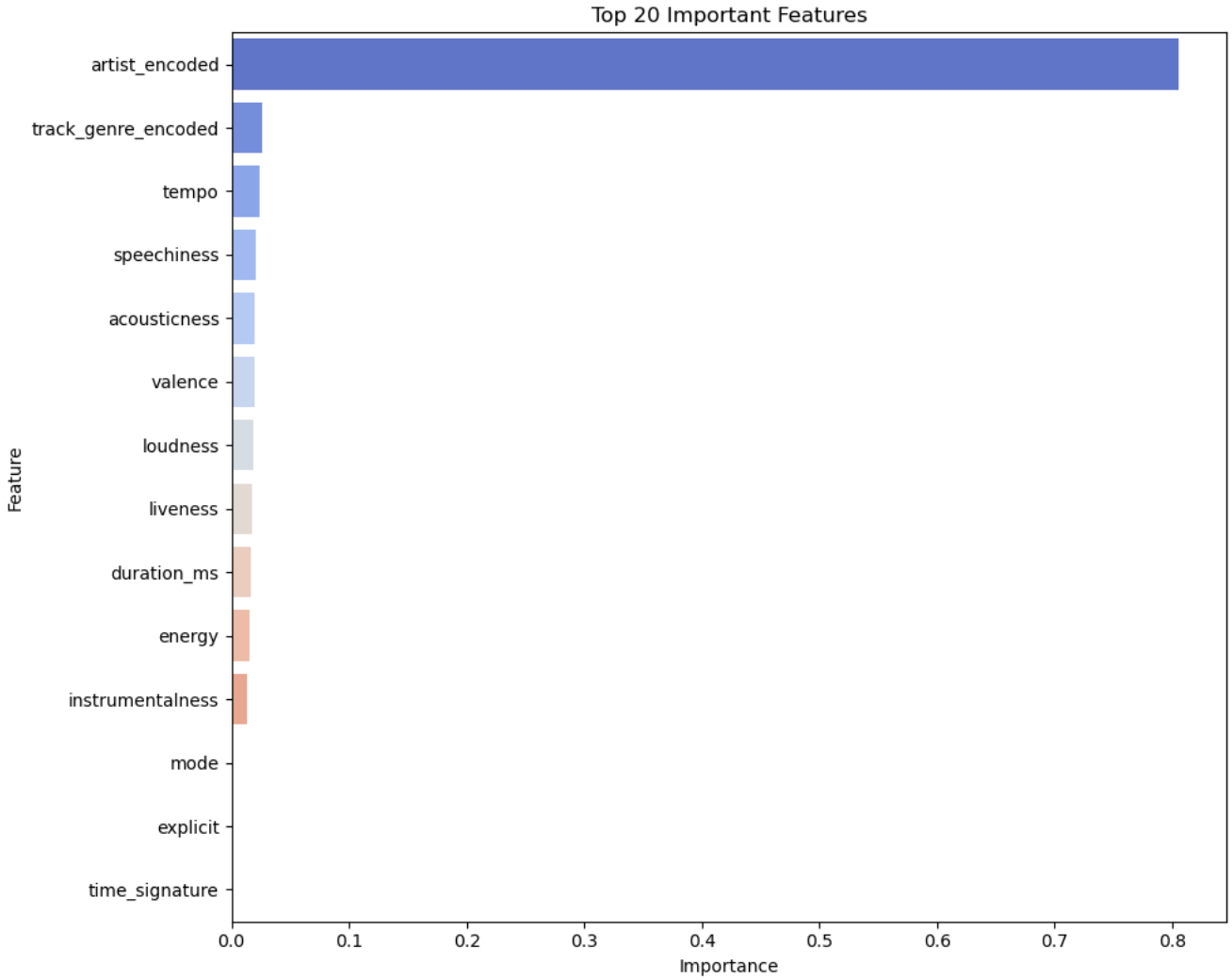

Figure 4.1. Feature Importance Plot from Random Forest Model

From Figure 4.1, it is illustrated that the feature artist_encoded dominates the model with an importance value of approximately 0.80, indicating that the artist's identity is by far the most significant predictor of popularity. All other features fall well below this, with importance values less than 0.05. The second-highest contributing feature is track_genre_encoded, followed by tempo, speechiness, acousticness, and valence, each with minimal but non-zero influence. Features like mode, explicit, and time_signature are absent from the plot, likely due to low variance or limited predictive power, as supported by earlier statistical tests.

4.3.4 Residual Plot

To evaluate the distribution of prediction errors made by the model, a residual plot was generated. This plot displays the difference between the actual and predicted popularity values for each observation, helping to identify systematic bias or irregularities in model performance. Patterns in the residuals can reveal issues such as heteroscedasticity, underfitting, or model instability across different popularity ranges.

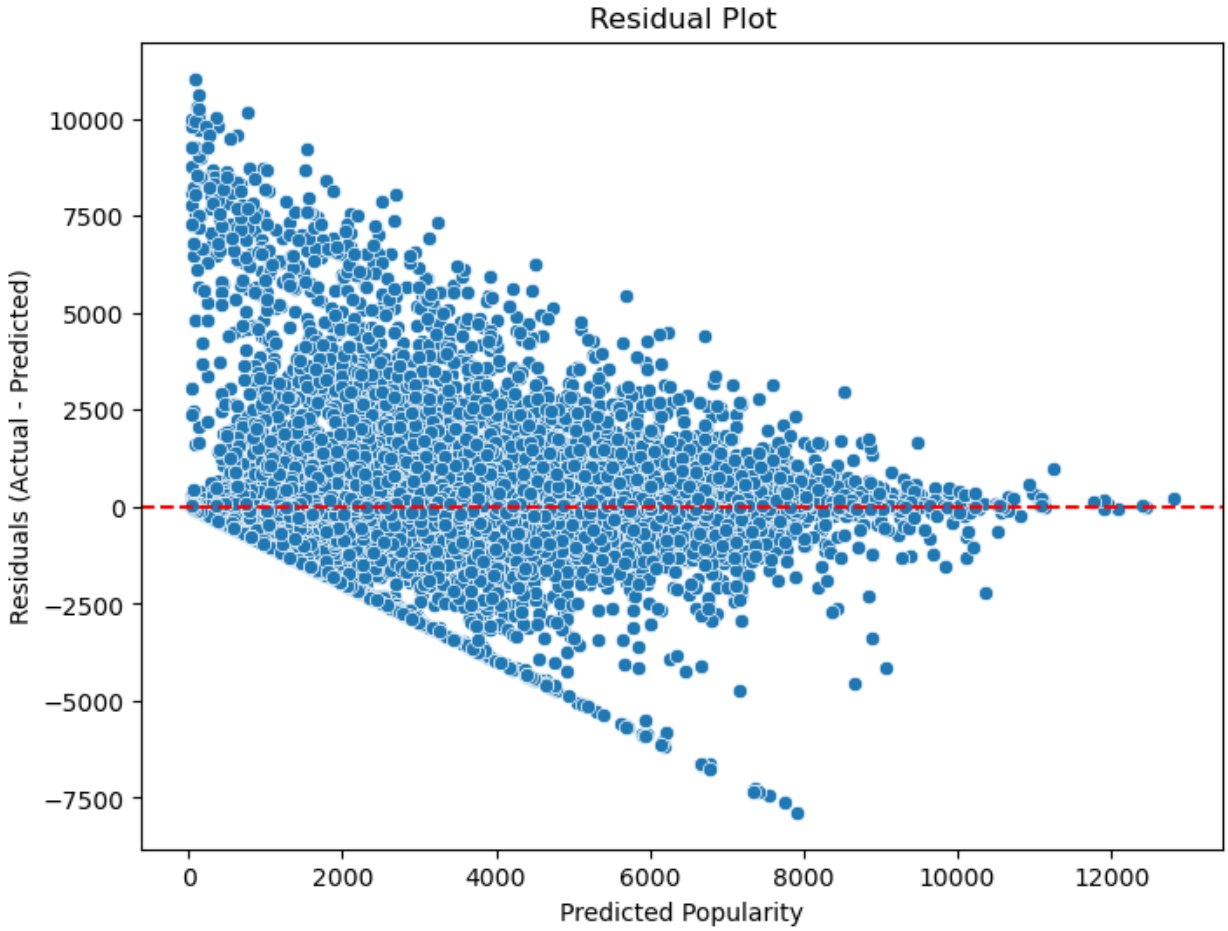

Figure 4.2. Residual Plot of Popularity Errors

Figure 4.2 shows the residuals from the Random Forest model. Most are centered around zero, but the spread widens with higher predicted popularity, indicating greater error for very popular tracks. This suggests the model performs better on less popular songs. A few large residuals likely reflect outliers influenced by factors not captured in the data.

4.3.5 Actual vs Predicted Plot

The predicted vs actual plot provides a direct visual comparison between the popularity values predicted by the model and the actual observed values. Ideally, if predictions are accurate, the data points should align closely along a diagonal line (y = x). Deviations from this line reveal where the model tends to overestimate or underestimate, offering insights into its precision and generalization ability.

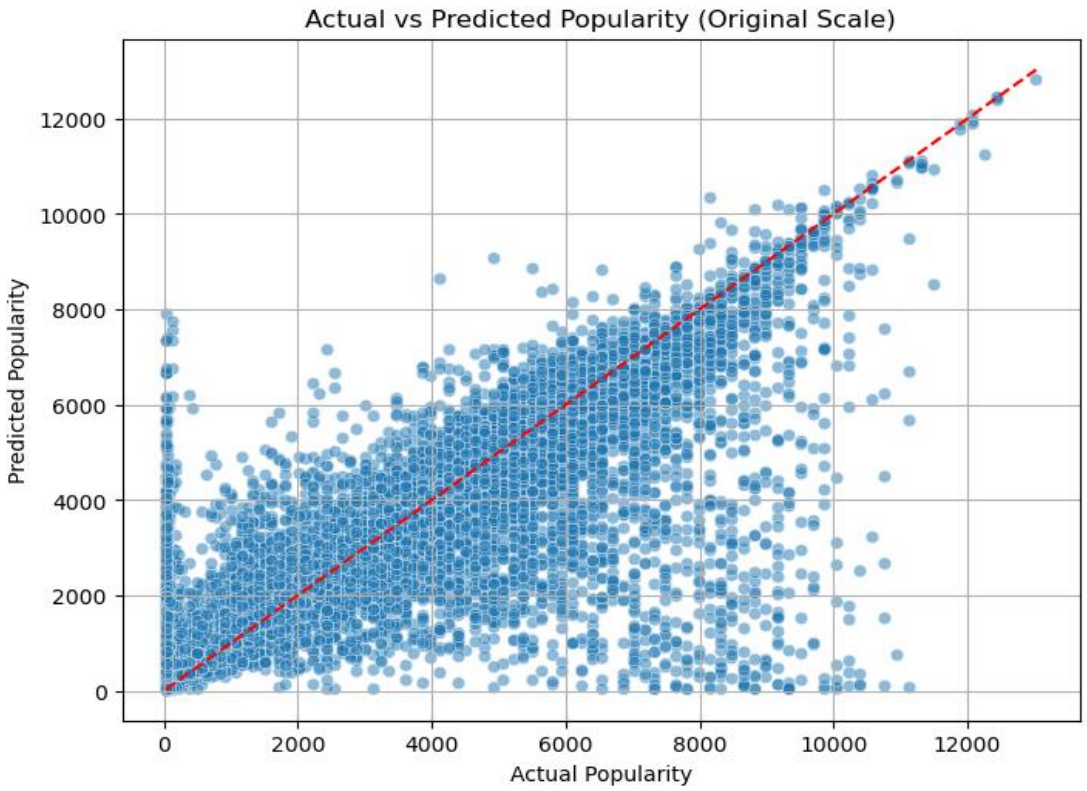

Figure 4.3. Predicted vs Actual Plot

Figure 4.3 shows a scatter plot of predicted vs. actual popularity scores. Points mostly follow the diagonal, indicating decent overall performance. However, the spread increases with popularity—highly popular songs are often underpredicted, while mid-range scores show slight overprediction. This suggests rising uncertainty at the extremes, likely due to missing influential factors behind viral tracks.

4.3.6 Cross-Validated R^2 Score

To ensure the model's performance was not specific to a single train-test split, cross-validation was performed using 5-folds. The R^2 scores obtained in each fold were plotted to assess the consistency of the model's predictive ability.

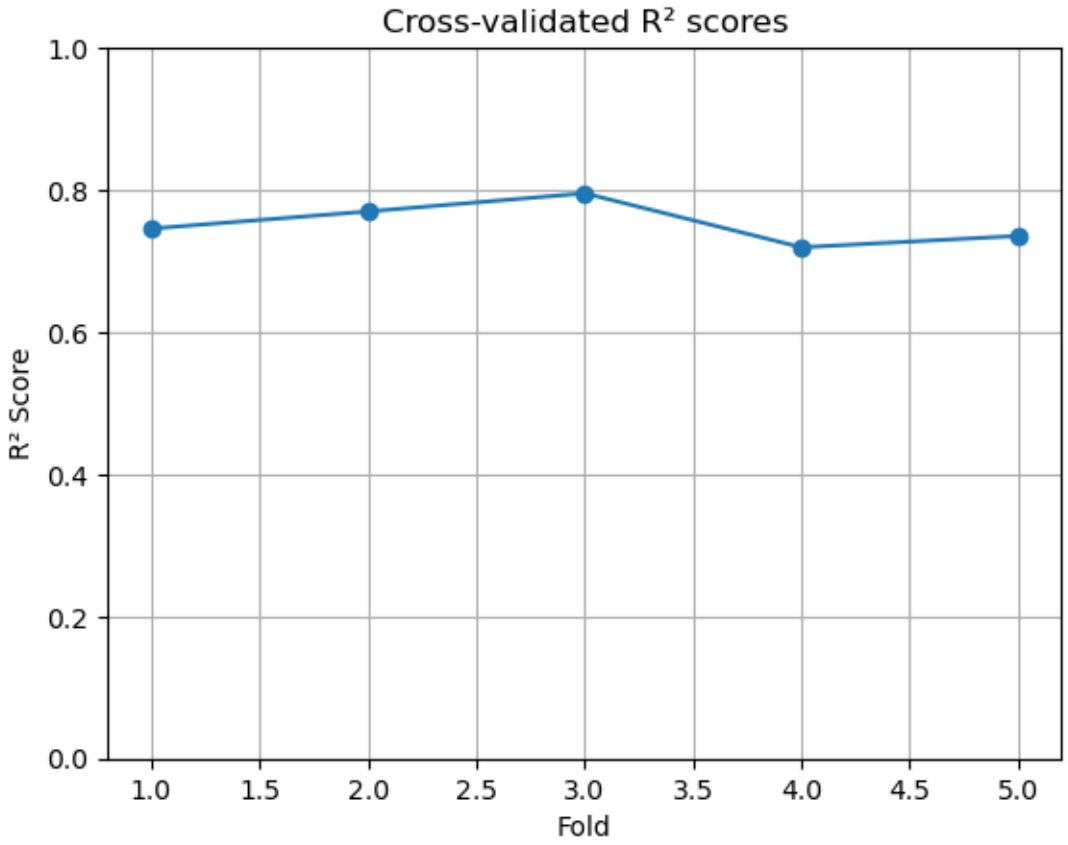

Figure 4.4. Predicted vs Actual Plot

Figure 4.4 displays the R^2 scores obtained from 5-fold cross-validation of the regression model. The values range between 0.60 and 0.80, indicating that the model consistently explains a substantial portion of the variance in track popularity across different subsets of data. While slight variations are observed between folds, the overall performance remains stable, suggesting that the model generalizes well and is not overly sensitive to data partitioning. This level of consistency reinforces the reliability of the selected features and modeling approach, while also highlighting potential headroom for further improvement.

4.4 Clustering and Analysis

To uncover emotional groupings in the music dataset, unsupervised clustering was performed using the K-Means algorithm. A subset of mood-relevant audio features — valence, energy, acousticness, danceability, tempo, and loudness — was selected to represent the affective characteristics of songs. Based on these features, the model divided the dataset into four distinct clusters.

Each cluster was manually interpreted and assigned a mood label — Aggressive, Happy, Sad, and Chill — based on the average feature values in each group. For instance, tracks in the Aggressive cluster typically showed high energy and loudness, while Chill songs were lower in tempo and more acoustic.

4.4.1 Dimensionality Reduction with PCA

To visualize the clusters in two dimensions, Principal Component Analysis (PCA) was applied. This technique reduced the six mood features to two principal components, making it easier to interpret the structure of the clusters.

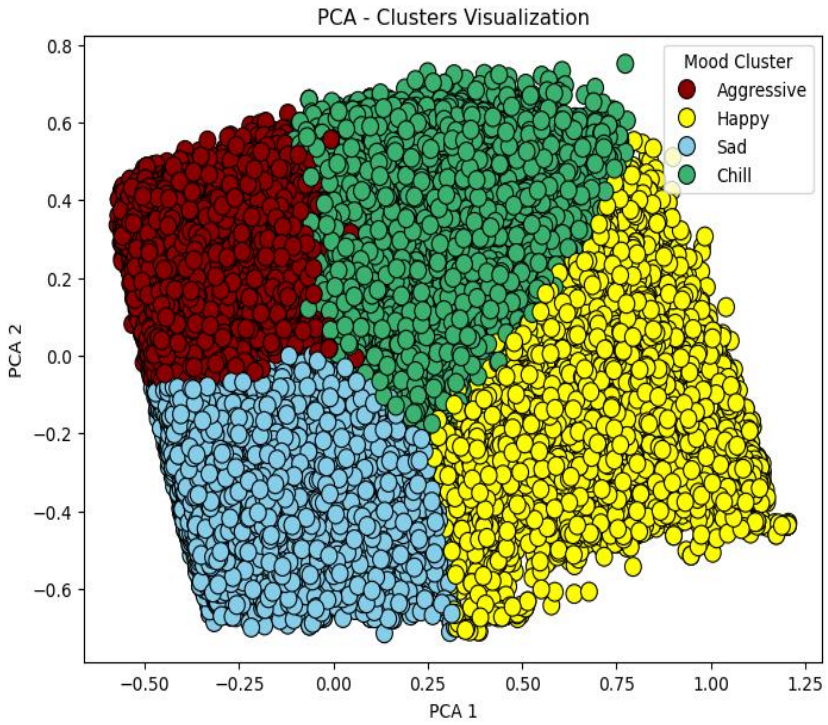

Figure 4.5. PCA-Based Visualization of Mood Clusters

Figure 4.5 displays clear separation between some mood clusters, although slight overlaps exist. The Aggressive group is positioned distinctly, driven by high loudness and energy. The Chill

and Sad clusters are more compact and lower in energy and tempo, while Happy tracks occupy the mid-high energy and valence region. This confirms that mood-specific traits are captured effectively through clustering.

The silhouette score of 0.405 indicates a moderate level of cluster quality — not perfectly distinct but meaningful enough for basic mood categorization. Despite the model's limitations, this approach supports emotional tagging and lays the foundation for mood-aware recommendation systems.

4.5 Recommendation System

To enhance the practical application of the model, two lightweight recommendation approaches were developed using the clustered dataset and audio features.

4.5.1 Mood-Based Recommendation

A simple rule-based recommender was built using the mood labels derived from K-Means clustering. When a user selects a mood category (e.g., Happy, Sad, Chill, or Aggressive), the system returns a random sample of songs from that mood cluster.

This system relies on the clustering performed earlier, where each track was grouped based on six key features: valence, energy, acousticness, danceability, tempo, and loudness. Although basic in design, this tool demonstrates how emotion-driven filtering can be integrated into user interfaces to personalize song discovery.

4.5.2 Similarity-Based Recommendation

A second recommendation system was implemented using cosine similarity. Given a reference song, the model compares it with all other songs based on selected audio features — including danceability, energy, loudness, valence, tempo, acousticness, and instrumentalness — and returns the top 5 most similar tracks.

This method highlights how audio-based feature vectors can be used to create content-based recommendation systems, where recommendations are based on feature similarity.

Chapter 5

Conclusion

5.1 Conclusion

This project set out to explore whether a song's popularity on Spotify could be predicted using its musical and metadata features. The dataset, taken from Kaggle, included over 113,000 tracks with attributes like energy, valence, acousticness, tempo, loudness, and more, along with details such as the track name, artist, genre, and explicit content.

After preprocessing the data and conducting a thorough exploratory and statistical analysis, various machine learning models were tested. Among them, the Random Forest Regressor performed best, achieving an R^2 score of 0.7788 and a mean squared error of 110.95. This indicates that the model was able to explain a substantial portion of the variation in song popularity based on the given features.

Feature importance analysis showed that artist identity played the most dominant role in predicting popularity, followed by audio features like tempo and valence. Additional statistical tests also revealed small but noticeable differences in popularity based on features like explicit content and musical mode.

In the second part of the study, unsupervised learning was used to group songs into four mood-based clusters: Happy, Sad, Chill, and Aggressive. A PCA-based visualization helped illustrate how songs were spread across these emotional categories. Although the silhouette score was 0.405, the clusters were still interpretable and useful for building a basic mood-based recommendation system. Another simple content-based recommender was also created using cosine similarity, allowing songs to be suggested based on audio similarity to a selected track.

5.2 Limitations

While this study achieved meaningful insights into music popularity prediction and mood-based clustering, several limitations were encountered across different stages of the analysis. These limitations stem from the scope and quality of the dataset, modeling constraints, statistical assumptions, and the simplicity of the clustering and recommendation components.

Acknowledging these challenges is important for understanding the boundaries of the study's conclusions and identifying opportunities for further improvement.

5.2.1 Data Limitations and Feature Relationships

The dataset used in this study, sourced from Kaggle, provided a wide range of audio features and metadata but excluded real-time user interaction data such as skip rates, listening history, playlist placements, and user engagement — all of which are available through the Spotify API and could have enriched the model's predictive ability. Moreover, the popularity variable was highly imbalanced, with the majority of songs falling into the lower popularity range, which may have biased the learning process. Important external factors such as artist reputation, lyrical content, or visual elements that influence a track's success were also absent. Additionally, high-dimensional genre labels made genre-based analysis more complex. From a correlation perspective, strong positive correlation between energy and loudness and a negative correlation between energy and acousticness were observed, but many other features showed weak or ambiguous relationships, limiting direct interpretability.

5.2.2 Model Constraints and Prediction Limitations

Although the Random Forest model achieved a relatively strong R^2 value, the residual and predicted vs. actual plots revealed heteroscedasticity — particularly at higher popularity values — indicating that the model struggled to maintain consistent error across the entire range. This suggests that prediction reliability declined for tracks with very high or very low popularity. The limited non-linear relationships between features and the target may also have reduced the performance of simpler models, requiring more sophisticated tuning and feature transformations. Despite tuning several tree-based models, the unexplained variance suggests that key predictive signals may be missing from the data.

5.2.3 Clustering Limitations

The mood-based clustering using K-Means, although useful, had limitations in terms of cluster separation and interpretability. The silhouette score of 0.405 indicated only moderate distinction between the clusters, and some overlap was evident in the PCA visualization. The mood labels (Aggressive, Chill, Happy, Sad) were manually assigned based on average feature values, which adds a subjective layer to interpretation. Additionally, the K-Means algorithm

assumes spherical clusters and equal variance, which may not fully align with the structure of musical mood data.

5.2.4 Recommendation System Constraints

The recommendation systems built — one based on mood and the other on feature similarity — were simple and illustrative but lacked personalization. The mood-based recommender depends solely on predefined clusters and doesn't account for individual user preferences or listening history. Similarly, the cosine similarity approach recommends tracks based only on audio feature proximity, without integrating behavioral signals or contextual factors. In real-world applications, more sophisticated systems would benefit from hybrid filtering methods that combine both content and collaborative data for improved accuracy and user relevance.

5.3 Future Scope

This study provides a foundation for analyzing music data through machine learning, but there is still considerable room for growth. One of the most promising directions is to expand the dataset by integrating additional information available through the Spotify API — such as playlist placements, user engagement metrics (like skips or saves), and artist-level popularity. These real-world indicators could help improve model accuracy, especially for capturing what makes certain tracks go viral or perform exceptionally well.

Future work can also explore more advanced modeling techniques, including deep learning or ensemble methods that may better handle the complex, non-linear relationships between features and popularity. Improving the feature set — such as incorporating lyrical content, release timing, or even album artwork — could bring in new predictive signals.

The clustering process could be enhanced by trying other unsupervised methods like DBSCAN or Gaussian Mixture Models, which might offer better separation than K-Means. Additionally, more nuanced mood categories could be explored with input from actual listeners.

Lastly, the recommendation system could be developed into a more interactive tool by combining content-based filtering with collaborative data. With access to user behavior and preferences, a hybrid recommender could deliver personalized results and serve as a valuable feature for streaming platforms.

References

1. Kaggle. (n.d.). Spotify Tracks Dataset. Retrieved from: https://www.kaggle.com/datasets

2. Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

3. Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. Computing in Science & Engineering, 9(3), 90–95.

4. Waskom, M. L. (2021). Seaborn: Statistical Data Visualization. Journal of Open Source Software, 6(60), 3021.

5. McKinney, W. (2010). Data Structures for Statistical Computing in Python. Proceedings of the 9th Python in Science Conference, 51–56.

6. Spotify Developer Platform. (n.d.). Spotify Web API Documentation. Retrieved from: https://developer.spotify.com/documentation/web-api/

7. Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd ed.). Springer.

8. Jolliffe, I. T. (2002). Principal Component Analysis. Springer Series in Statistics.

9. Mann, H. B., & Whitney, D. R. (1947). On a Test of Whether One of Two Random Variables is Stochastically Larger Than the Other. Annals of Mathematical Statistics, 18(1), 50–60.

10. Shapiro, S. S., & Wilk, M. B. (1965). An Analysis of Variance Test for Normality (Complete Samples). Biometrika, 52(3/4), 591–611.