

# Interpretable Spotify Clustering with Robust Preprocessing, Validation, and Stream Analysis

Meliysa Pasa Bagna Aprilia Said<sup>1</sup>, Kholifatus Sholihah<sup>2</sup>, Khafidlotur Rofiqah<sup>3</sup>, Mula Agung Barata<sup>4</sup>

Teknik Informatika, Universitas Nahdlatul Ulama Sunan Giri Bojonegoro

[bagnasaid4@gmail.com](mailto:bagnasaid4@gmail.com)<sup>1</sup>, [olalap125@gmail.com](mailto:olalap125@gmail.com)<sup>2</sup>, [khafidloturrofiqah@gmail.com](mailto:khafidloturrofiqah@gmail.com)<sup>3</sup>, [mula.ab26@gmail.com](mailto:mula.ab26@gmail.com)<sup>4</sup>

## Article Info

### Article history:

Received 2026-07-03

Revised 2026-09-10

Accepted 2026-10-14

### Keyword:

Audio Features,  
Clustering, K-Means,  
Spotify,  
Unsupervised Learning.

## ABSTRACT

This study identifies patterns in popular Spotify songs based on measurable audio characteristics and evaluates the relationship between the resulting clusters and stream-count tendencies. The Most Streamed Spotify Songs 2023 dataset was used, consisting initially of 953 records and 24 attributes, with 948 songs retained after data cleaning. Eight audio features—bpm, danceability, valence, energy, acousticness, instrumentalness, liveness, and speechiness—were used for clustering, while streams was analyzed only after clustering. Preprocessing included outlier examination and comparison of RobustScaler, StandardScaler, and MinMaxScaler. The number of clusters was evaluated using the Elbow Method and Silhouette Score, with  $k = 2$  selected as the final configuration. The K-Means model achieved a Silhouette Score of 0.9218, a Davies-Bouldin Index of 0.3284, and a Calinski-Harabasz Index of 2857.8019. Cluster 0 contained 930 songs and showed higher danceability, valence, energy, and speechiness with very low instrumentalness, whereas Cluster 1 contained 18 songs and was primarily distinguished by much higher instrumentalness. Sensitivity analysis showed that the imbalanced structure was stable across centroid initializations but strongly dependent on instrumentalness and the selected scaling method. Comparison with Agglomerative Clustering and Gaussian Mixture Model showed that K-Means offered competitive clustering quality with the lowest observed runtime. Although Cluster 0 had a higher mean stream count, the Mann-Whitney U test found no statistically significant difference between clusters ( $p = 0.3495$ ). The final solution is therefore interpreted as an algorithmically stable but feature-dependent separation, while stream differences are treated as descriptive rather than causal.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

## I. INTRODUCTION

The growth of music streaming services has encouraged new analytical approaches to understanding music because song data are now used not only for distribution but also for mapping audio characteristics and listener behavior. Deldjoo, Schedl, and Knees reviewed 55 articles on content-driven music recommendation and showed that music recommendation models have shifted from purely collaborative filtering or content-based approaches toward hybrid models that combine both [1]. This finding indicates that digital music data, including Spotify data, are increasingly relevant for computational analysis.

In the Spotify context, audio features are important because they represent musical characteristics quantitatively. Duman et al. found that dance music data analyzed from Spotify had higher levels of energy, danceability, valence, and loudness than a general music reference dataset, while speechiness, instrumentalness, and acousticness tended to be lower [2]. This shows that audio features can capture differences in song characteristics and are therefore appropriate as a basis for music clustering. The relationship between audio features and song popularity has also been discussed in previous studies. Khan et al. reported that a music popularity classification model using all features achieved an accuracy of 95.15%, while a model using selected features achieved 95.14% [3]. Saragih also showed that song popularity prediction based on

Spotify audio features can be developed using a public Kaggle dataset consisting of 92,755 rows and 20 columns [4]. These findings indicate that audio features are not only descriptive but also informative in explaining song performance tendencies on digital platforms.

The use of Spotify digital traces to identify song characteristics in specific contexts has also developed. Scarratt et al. extracted 225,626 songs from Spotify sleep-themed playlists and found that sleep music tended to be softer, slower, more instrumental, and divisible into six subgroups [5]. In another context, study-related Spotify playlists have also been analyzed to identify the characteristics of music used while studying [6]. These studies demonstrate that Spotify playlist data can be used to capture audio patterns in different music-use contexts.

Several studies closely related to this topic have applied clustering directly to Spotify data. Rohman and Wibowo reported that clustering popular Spotify songs in 2023 using K-Means produced two clusters, consisting of 863 data points in cluster 0 and 90 data points in cluster 1, with a Silhouette Coefficient of 0.81, which was categorized as a strong structure [7]. Musyarofah et al. classified popular Spotify songs from 2010 to 2019 using K-Means and reported that pop dominated at 83.9%, followed by EDM at 3.05%, while boyband and hip hop each accounted for 2.54% [8]. These results indicate that Spotify data can be examined both from the perspective of song characteristics and popularity trends.

In the recommendation domain, Maheshwari explored the use of deep learning for music recommendation on Spotify [9], while Chopade et al. described a deep-learning-based recommendation approach that incorporates user listening behavior and audio features to support personalized song recommendations [10]. These studies show that Spotify is important not only as a distribution platform but also as a rich data source for analyzing patterns in songs and listeners.

More specifically, K-Means remains widely used because of its simplicity and interpretability in grouping numerical data. Shulhiyana and Voutama reported that after exploration, modification, normalization, and modeling stages, K-Means formed three music clusters with a silhouette score of 0.435 [11]. Branchris, Yech, and Wijaya also showed that Spotify song clustering based on valence, energy, and danceability produced three main clusters, namely Happy/Cheerful, Sad/Melancholy, and Energetic/Intense, with a Silhouette Coefficient of 0.30 [12]. Putra et al. noted that in a music recommendation system based on Spotify audio features, an initial dataset of approximately 114,000 songs was processed into about 81,000 valid songs and then sampled into 5,000 songs for experimentation; in that test, K-Means obtained a Silhouette Score of 0.1900 and a Davies-Bouldin Index of 1.4445, performing slightly better than Hierarchical Clustering [13]. These findings show that although clustering quality may vary, K-Means remains relevant for music data based on audio features. However, studies focusing on streaming popularity indicate that streams are important but do not always reflect current momentum. Wibawa and

Valentiya emphasized that total streams represent cumulative popularity more than short-term popularity dynamics [14]. Therefore, grouping songs based on audio-feature similarity is still necessary so that the analysis does not only focus on the number of plays but also on the sound characteristics of highly streamed songs. Based on this background, this study aims to cluster Spotify songs based on selected audio features using the K-Means algorithm and to interpret the relationship between the resulting clusters and stream-count tendencies. In addition, this study evaluates the clustering results using internal validation metrics, compares K-Means with alternative clustering methods, and discusses the practical implications and limitations of the resulting clusters.

However, previous studies on Spotify song clustering have primarily focused on applying clustering algorithms to group songs based on audio characteristics, while limited attention has been given to evaluating the reliability and robustness of the clustering process itself. The effects of preprocessing strategies, feature contribution, initialization stability, and the consistency of clustering results are often not examined comprehensively. Furthermore, many studies interpret clusters only through internal validation metrics without exploring their relationship with popularity-related indicators such as stream counts. Therefore, a systematic and interpretable clustering workflow that evaluates preprocessing choices, clustering stability, algorithm performance, and post-clustering interpretation is still needed. The contribution of this study lies in developing an evaluation-oriented and interpretable clustering workflow that integrates robust preprocessing assessment, feature sensitivity analysis, initialization stability evaluation, alternative clustering comparison, and post-clustering interpretation of stream characteristics.

## II. METHOD

This study employed a quantitative approach using an unsupervised learning technique to group songs based on the similarity of their audio characteristics. The object analyzed was the Most Streamed Spotify Songs 2023 dataset obtained from Kaggle and processed using Google Colab. In general, Spotify datasets contain two important types of information, namely song metadata and numerical audio features. Attributes such as danceability, energy, valence, acousticness, instrumentality, liveness, speechiness, and tempo are relevant for describing measurable audio patterns and have also been used in previous Spotify clustering studies [15].

As shown in Figure 1, this study was conducted through a systematic sequence of stages consisting of dataset collection, data cleaning and feature selection, outlier examination, scaling-method comparison, determination of the number of clusters, K-Means clustering, feature-sensitivity analysis, initialization-stability analysis, internal cluster validation, comparison with alternative clustering methods, PCA and correlation analysis, post-clustering stream analysis, and interpretation of the resulting clusters. This workflow was designed not only to form clusters based on audio-feature

similarity but also to evaluate the robustness, stability, feature dependence, and interpretability of the resulting clustering structure.

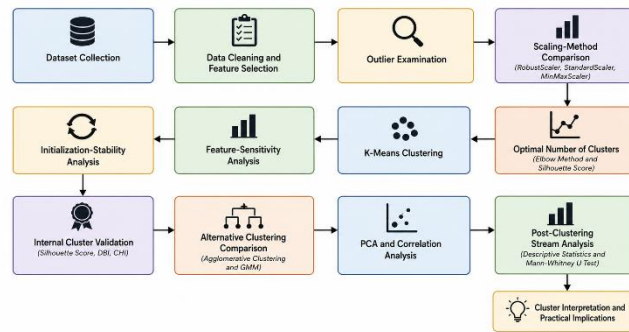


Figure 1. Research Workflow

In the initial stage, the dataset was examined to identify its structure, data quality, and available variables. Data cleaning and feature selection were then performed to retain the numerical audio attributes relevant to the clustering objective. Extreme values were examined using robust statistical considerations, after which RobustScaler, StandardScaler, and MinMaxScaler were compared to evaluate the sensitivity of the clustering results to preprocessing choices. The number of clusters was then evaluated using the Elbow Method and Silhouette Score before the final K-Means model was constructed. To assess the robustness of the resulting structure, feature-sensitivity and initialization-stability analyses were conducted. The clustering solution was subsequently evaluated using the Silhouette Score, Davies-Bouldin Index, and Calinski-Harabasz Index and compared with Agglomerative Clustering and Gaussian Mixture Model. PCA and correlation analysis were used to further examine the structure of the audio-feature space. Finally, stream counts were analyzed after clustering using descriptive statistics and the Mann-Whitney U test, followed by interpretation of the cluster characteristics, limitations, and practical implications.

#### A. Dataset and Data Characteristics

The dataset used in this study was the Most Streamed Spotify Songs 2023 dataset obtained from Kaggle and processed in Google Colab. The dataset contains song metadata, popularity indicators, and numerical audio features that represent song characteristics. Attributes such as danceability, energy, valence, acousticness, instrumentalness, liveness, speechiness, and tempo are relevant for describing audio patterns and have also been used in previous Spotify clustering studies [15].

Table 1 presents a sample of the Most Streamed Spotify Songs 2023 dataset obtained from Kaggle [16]. The table shows several key variables, including data number, track name, artist, streams, bpm, and danceability. This sample is provided to give an initial overview of the dataset structure before data cleaning, feature selection, preprocessing, and clustering are performed. Overall, the initial dataset consisted

of 953 rows and 24 columns covering song metadata, popularity indicators, and audio features.

TABLE I  
MOST STREAMED SPOTIFY SONGS 2023 DATASET

| No  | Track Name                | Artist            | Streams     | BPM | Dance. (%) |
|-----|---------------------------|-------------------|-------------|-----|------------|
| 1   | Seven (feat. Latto)       | Latto, Jung Kook  | 141,381,703 | 125 | 80         |
| 2   | LALA                      | Myke Towers       | 133,716,286 | 92  | 71         |
| 238 | Lavender Haze             | Taylor Swift      | 488,386,797 | 97  | 73         |
| 477 | It's Beginning to Look... | Perry Como et al. | 295,998,468 | 113 | 73         |
| 953 | Alone                     | Burna Boy         | 96,007,391  | 90  | 61         |

Based on the dataset structure, this study focused on numerical attributes that are most relevant for representing song audio characteristics. The variables used to form clusters were bpm, danceability\_%, valence\_%, energy\_%, acousticness\_%, instrumentalness\_%, liveness\_%, and speechiness\_%. These eight audio features were selected because they directly represent measurable musical characteristics. BPM represents tempo, danceability reflects the suitability of a song for movement or dancing, valence represents emotional characteristics, energy reflects musical intensity, acousticness indicates the degree of acoustic properties, instrumentalness represents the dominance of instrumental components, liveness reflects live performance characteristics, and speechiness indicates the presence of spoken-word elements. Other attributes, such as track name, artist name, release date, playlist counts, chart positions, and stream count, were not used as clustering variables because they represent metadata or popularity-related indicators rather than intrinsic audio characteristics. Categorical audio attributes such as key, mode, and time signature were also excluded because the clustering objective focused on continuous numerical audio characteristics. These variables

represent discrete musical attributes that require additional encoding procedures, which may introduce artificial distance relationships between categories. In particular, streams was excluded from the clustering process to avoid forming clusters based directly on popularity; it was used only after clustering to interpret stream-count tendencies across the formed clusters. This separation of variable functions is consistent with feature engineering practices, which emphasize that feature selection must align with the analysis objective so that the model uses only relevant attributes [15]. Kotsiantis et al. also stated that data quality and feature selection accuracy are important factors influencing machine learning results [17].

### B. Dataset Handling

Dataset handling was conducted to ensure that the data used in the analysis were clean and suitable for modeling. The initial examination included checking the data structure, data types, missing values, duplicate records, and the consistency of each column. The examination showed one invalid entry in the streams column; therefore, the column was converted into a numeric type using a coercion approach so that values with invalid formats became NaN and could be handled in the next stage. This approach is consistent with Kotsiantis et al., who described data preprocessing as including data cleaning, transformation, normalization, and feature selection to improve data quality before modeling [17].

TABLE II DATA CLEANING RESULTS

| Stage                         | Number of Records |
|-------------------------------|-------------------|
| Initial dataset               | 953               |
| Invalid stream record removed | 1                 |
| Duplicate records removed     | 4                 |
| Final dataset after cleaning  | 948               |

Table II shows the data cleaning process performed before clustering. The initial dataset consisted of 953 song records. First, the streams column was converted into a numeric format, and one record with an invalid stream value was removed because it could not be used for post-clustering stream interpretation. Second, duplicate records were removed based on the combination of track\_name and artist(s)\_name to avoid repeated songs by the same artist. This process removed four duplicate records. After these steps, 948 song records remained and were used for feature selection, preprocessing, and clustering. No records were removed during the outlier examination stage because the detected extreme values were treated using a robust scaling approach rather than deletion.

Duplicate data were removed based on the combination of track\_name and artist(s)\_name. This combination was selected so that songs with the same title but performed by different artists would not be incorrectly treated as duplicates. After that, column selection was performed so that only attributes relevant to the research objective were retained. Kotsiantis et al. emphasized that processes such as entity

matching, feature cleaning, and outlier handling play an important role in the relationship between data cleaning and machine learning [17]. Therefore, dataset handling in this study was intended not only to clean the data but also to ensure that the variables entering the model truly represented song audio characteristics.

### C. Outlier Examination and Preprocessing

The preprocessing stage was conducted after the data had passed initial cleaning and attribute selection. In this study, outliers were examined using robust statistical considerations based on the Interquartile Range (IQR) and Median Absolute Deviation (MAD). Because the detected extreme values could represent valid musical characteristics, no records were removed. Instead, RobustScaler was applied to reduce their influence on the distance calculations used by K-Means. This stage is important because K-Means is a distance-based algorithm and is therefore sensitive to data scale and extreme values [18], [19]. Kotsiantis et al. showed that data cleaning quality directly affects machine learning results [17]. Based on these considerations, this study applied a robust approach to outlier examination and scaling using Interquartile Range (IQR), Median Absolute Deviation (MAD), and RobustScaler.

#### 1) Interquartile Range (IQR)

Interquartile Range (IQR) was used as an initial basis for outlier examination because it is simple, easy to interpret, and relatively stable for data that are not normally distributed [20]. Rousseeuw and Hubert explained that the interquartile range is a robust measure based on the first and third quartiles and can be used in identifying potential outliers [20]. In this study, IQR was considered appropriate because Spotify audio features, such as instrumentalness\_% or speechiness\_%, may have asymmetric distributions. Nugraha also stated that robust approaches are more suitable when data contain outliers that may interfere with the analysis [18].

Mathematically, IQR is formulated as follows:

$$\text{IQR} = Q_3 - Q_1 \quad (1)$$

where:

$Q_1$  = first quartile,

$Q_3$  = third quartile.

The outlier boundaries are then determined using the following equations:

$$\text{Lower Bound} = Q_1 - 1.5 \times \text{IQR}$$

$$\text{Upper Bound} = Q_3 + 1.5 \times \text{IQR} \quad (2)$$

Values below the lower bound or above the upper bound are indicated as outliers [18], [20]. In this study, IQR was not used to automatically remove data, but rather to identify whether extreme values existed in each audio feature. This approach was selected because the purpose of the study was not to discard as much data as possible, but to understand the data distribution before scaling was performed.

## 2) Median Absolute Deviation (MAD)

In addition to IQR, Median Absolute Deviation (MAD) was used as a theoretical basis to strengthen the robust approach to outlier examination. MAD was selected because it uses the median as the center point and is therefore more resistant to extreme values than dispersion measures based on the mean [20]. Rousseeuw and Hubert described MAD as a robust measure of scale based on the median of absolute deviations from the median, making it less sensitive to extreme observations than conventional measures based on the mean [20]. In other words, when data do not follow a symmetrical distribution, MAD is more representative for measuring the deviation of values from the data center. Mathematically, MAD is formulated as follows:

$$\text{MAD} = \text{median}(|x_i - \text{median}(x)|) \quad (3)$$

where:

$x_i$  = the  $i$ -th data value,

$\text{median}(x)$  = the median of the data.

The formula shows that MAD is calculated from the median of the absolute deviations of each data value from the overall median. In this study, MAD was not used as the main transformation method but served as a theoretical basis to emphasize that robust outlier examination should rely on the median rather than the mean. The use of MAD also supports the decision to apply a scaling method based on the median and quartiles in the next stage.

## 3) RobustScaler

After outlier examination, this study used RobustScaler as the data scaling method. The main reason for selecting RobustScaler is that it is more resistant to extreme values than StandardScaler or other scaling methods based on the mean and standard deviation [18]. Nugraha stated that RobustScaler uses the median and IQR as the basis for transformation, making it more suitable for data containing outliers [18]. This median- and IQR-based transformation is consistent with robust-statistics principles that reduce sensitivity to extreme observations [20].

RobustScaler was selected because Spotify audio features have different value ranges. For example, bpm has a different scale from danceability\_%, valence\_%, or instrumentalness\_%. If these data were directly entered into K-Means without scaling, features with larger ranges could dominate the distance calculation [19], [20]. Because K-Means works based on distance similarity between data points, the data must be transformed into a more comparable scale [20]. In this context, RobustScaler was considered

appropriate because it is less affected by extreme values that may appear in several audio features.

Mathematically, RobustScaler transformation is formulated as follows:

$$x_{\text{scaled}} = (x - \text{median}) / \text{IQR} \quad (4)$$

where:

$x$  = original value,

median = variable median,

IQR = interquartile range of the variable.

The formula shows that each variable is centered using the median and scaled using the IQR. In this way, the data become more comparable without being overly distorted by extreme values [18], [20]. Therefore, RobustScaler was selected because it is suitable for Spotify datasets containing numerical features with possible asymmetric distributions and outliers in several audio attributes.

Overall, IQR, MAD, and RobustScaler were selected based on the same consideration, namely the need to handle numerical data that may contain extreme values without losing too much information. IQR was used to identify outliers exploratively [18], [20]. MAD was used as a theoretical basis to support a robust median-based approach [20], while RobustScaler was used as the main preprocessing method because it is suitable for preparing the data before processing with the K-Means algorithm [18], [19].

## 4) Comparison of Scaling Methods

To assess the sensitivity of K-Means clustering to the scaling method, three preprocessing techniques were compared: RobustScaler, StandardScaler, and MinMaxScaler. Each scaling method was applied separately to the same eight audio features. K-Means clustering was then performed for  $k$  values ranging from 2 to 10 using the same initialization settings. The resulting solutions were evaluated using the Silhouette Score, Davies-Bouldin Index, Calinski-Harabasz Index, inertia, and cluster-size distribution. The final scaling method was selected by considering both internal validation performance and the balance of the resulting cluster structure.

## 5) Feature Sensitivity Analysis

To examine whether the resulting cluster structure was jointly driven by the selected audio features or dominated by a single variable, a feature-sensitivity analysis was conducted. First, the K-Means model using all eight audio features was compared with a model excluding instrumentalness. Second, a leave-one-feature-out analysis was performed by removing each feature separately while retaining the same RobustScaler,  $k = 2$ ,  $n_{\text{init}} = 20$ , and  $\text{random\_state} = 42$  settings. The resulting solutions were evaluated using the Silhouette Score, Davies-Bouldin Index, Calinski-Harabasz Index, cluster-size distribution, Principal Component

Analysis, and Adjusted Rand Index. The Adjusted Rand Index was used to measure the consistency of cluster membership between the original model and each reduced-feature model.

#### D. K-Means Algorithm

After data cleaning, outlier examination, and scaling, the next stage was grouping the data using the K-Means algorithm. This algorithm was selected because it is suitable for numerical data and can form groups based on the similarity between objects. Wani explained that K-Means is a centroid-based clustering method that remains widely used because it is simple, efficient, and relatively easy to interpret [19]. Ali et al. also emphasized that K-Means is still one of the most commonly used clustering algorithms because it can handle large datasets with relatively fast computation [21].

In this study, K-Means was selected because all variables used were numerical, namely eight audio features that had been scaled. This condition makes K-Means more suitable than methods designed for categorical or mixed-type data. In addition, the objective of this study was to form song groups based on similar audio characteristics; therefore, a centroid-based approach was considered appropriate because it can represent the central characteristics of each cluster [19].

Mathematically, K-Means works by minimizing the sum of squared distances between each data point and the centroid of its assigned cluster. The K-Means objective function is formulated as follows:

$$J = \sum_{i=1}^k \sum_{x \in C_i} \|x - \mu_i\|^2 \quad (5)$$

where:

$k$  = number of clusters,

$C_i$  = set of data points in the  $i$ -th cluster,

$x$  = data point,

$\mu_i$  = centroid of the  $i$ -th cluster.

A smaller objective function value indicates that cluster members are closer to their respective centroids, meaning that the formed clusters are more compact [19].

Operationally, the K-Means algorithm in this study consisted of the following steps:

1. determining the number of clusters;
2. selecting the initial centroids;
3. calculating the distance from each data point to all centroids;
4. assigning each data point to the nearest cluster;
5. recalculating the new centroid of each cluster;
6. repeating the process until the centroids become stable or no data points change clusters.

Because K-Means results can be influenced by centroid initialization, the final model was configured using `k-means++` initialization with `n_init` = 20 and `random_state` = 42. The `n_init` parameter indicates that the initialization process was repeated 20 times, and the solution with the

lowest inertia value was retained. The fixed random state was applied to ensure reproducibility of the clustering results. These settings were selected because K-Means performance can be sensitive to the initial centroid configuration [19].

To evaluate whether the clustering solution depended on a particular centroid initialization, an initialization-stability analysis was conducted. K-Means was repeatedly fitted using 30 random seeds, ranging from 0 to 29, and five `n_init` settings consisting of 1, 5, 10, 20, and 50. All experiments used the same eight audio features, RobustScaler,  $k = 2$ , and the Lloyd algorithm, resulting in 150 clustering runs. Each solution was evaluated using the Silhouette Score, Davies-Bouldin Index, Calinski-Harabasz Index, inertia, cluster-size distribution, and Adjusted Rand Index. The Adjusted Rand Index was calculated relative to the reference model using `random_state` = 42 and `n_init` = 20 to evaluate the consistency of cluster membership.

#### E. Determination of the Number of Clusters

The K-Means algorithm requires the number of clusters to be determined before clustering is performed. Therefore, this study used the Elbow Method and Silhouette Score to determine the most appropriate number of clusters. Ali et al. explained that selecting the number of clusters is one of the main issues in K-Means because clustering quality is strongly influenced by the accuracy of the selected value [21]. Junthopas and Wongoutong also emphasized the importance of determining an appropriate number of clusters before applying K-Means because the selected value of  $k$  directly influences the resulting clustering structure [22].

In the Elbow Method, the observed indicator is the Within-Cluster Sum of Squares (WCSS), also called inertia. WCSS measures the total variation within clusters and is formulated as follows:

$$WCSS = \sum_{j=1}^k \sum_{x_i \in C_j} \|x_i - \mu_j\|^2 \quad (6)$$

where:

$k$  = number of clusters,

$C_j$  = set of data points in the  $j$ -th cluster,

$x_i$  = data point,

$\mu_j$  = centroid of the  $j$ -th cluster.

WCSS decreases as the number of clusters increases, but the decrease is not always significant. The point at which the decrease begins to flatten is called the elbow point, and this point is considered a candidate for the optimal number of clusters [21].

In addition to the Elbow Method, this study used Silhouette Score to assess cluster separation quality. Kossakov et al. stated that this metric is effective for quantitatively comparing clustering performance [23]. Ekemeyong Awong and Zielinska also used the Silhouette Coefficient as a cluster-quality measure for evaluating clustering solutions [24]. Mathematically, the silhouette value

for the  $i$ -th data point is formulated as follows:

$$s(i) = \frac{b(i)-a(i)}{\max(a(i),b(i))} \quad (7)$$

where:

$a(i)$  = average distance from the  $i$ -th data point to other data points in the same cluster,

$b(i)$  = minimum average distance from the  $i$ -th data point to another cluster.

The silhouette value ranges from -1 to 1. A value closer to 1 indicates better clustering quality because data points are closer to their own cluster than to other clusters [23], [24].

In this study, several  $k$  values were tested and the final selection was based on the combined results of the Elbow Method and Silhouette Score. Based on the evaluation, the best number of clusters used in the final model was two because it produced the best separation quality compared with other alternatives.

#### F. Final Evaluation

After the number of clusters was determined and the K-Means model was run, the next step was to evaluate the quality of the clustering results. This study used three internal metrics: Silhouette Score, Davies-Bouldin Index (DBI), and Calinski-Harabasz Index (CHI). Ikotun et al. explained that cluster validity indices are used to assess the quality of clusters in unlabeled data through measures of within-cluster compactness and between-cluster separation [25]. Kossakov et al. also showed that these metrics are relevant for quantitatively comparing clustering results [23].

Davies-Bouldin Index measures the average similarity between one cluster and the most similar other cluster. Mathematically, DBI is formulated as follows:

$$DBI = \frac{1}{k} \sum_{i=1}^k \max_{j \neq i} \frac{S_i + S_j}{M_{ij}} \quad (8)$$

where:

$k$  = number of clusters,

$S_i$  = dispersion of the  $i$ -th cluster,

$S_j$  = dispersion of the  $j$ -th cluster,

$M_{ij}$  = distance between the centroids of the  $i$ -th and  $j$ -th clusters.

A lower DBI value indicates better clustering quality because clusters are more separated and more compact [23], [25]. Meanwhile, the Calinski-Harabasz Index measures the ratio of between-cluster dispersion to within-cluster dispersion. CHI is formulated as follows:

$$CHI = \frac{Bk}{Wk} \times \frac{n-k}{k-1} \quad (9)$$

where:

$B_k$  = between-cluster dispersion,

$W_k$  = within-cluster dispersion,

$n$  = number of data points,

$k$  = number of clusters.

A higher CHI value indicates a better cluster structure [24], [25]. In this study, the final model evaluation produced a Silhouette Score of 0.9218, a Davies-Bouldin Index of 0.3284, and a Calinski-Harabasz Index of 2857.8019. These values indicate that the model produced compact clusters with clear separation between clusters. This evaluation approach is consistent with previous clustering studies that combine K-Means with internal validation methods such as the Elbow Method and Silhouette Score [23], [25].

Based on these stages, the final result of this study was not only the formation of two Spotify song clusters but also a post-clustering interpretation of the streams variable to identify which song group tended to have higher playback counts. Thus, the evaluation did not stop at cluster structure quality but continued with the interpretation of audio characteristics in each formed cluster.

### III. RESULTS AND DISCUSSION

In this study, clustering was performed on Spotify songs based on the similarity of audio features using the K-Means algorithm. After data cleaning, feature selection, outlier examination, and scaling using RobustScaler, 948 songs were retained for modeling. The variables used to form clusters consisted of eight audio features: bpm, danceability\_%, valence\_%, energy\_%, acousticness\_%, instrumentalness\_%, liveness\_%, and speechiness\_%. The streams variable was not used as a clustering variable but was used in the interpretation stage to examine playback tendencies in each cluster.

#### A. Sensitivity Analysis of Scaling Methods

The scaling-method sensitivity analysis showed that  $k = 2$  produced the highest Silhouette Score across the three preprocessing methods. RobustScaler achieved a Silhouette Score of 0.9218, a Davies-Bouldin Index of 0.3284, and a Calinski-Harabasz Index of 2857.8019. In comparison, StandardScaler produced a Silhouette Score of 0.2271, a Davies-Bouldin Index of 1.8914, and a Calinski-Harabasz Index of 190.3357, whereas MinMaxScaler produced corresponding values of 0.2691, 1.4858, and 281.9338.

Although RobustScaler produced substantially better internal validation values, it also generated a highly imbalanced cluster distribution of 930 and 18 songs. StandardScaler and MinMaxScaler produced more balanced cluster distributions of 689 and 259 songs and 727 and 221 songs, respectively. Therefore, the scaling methods produced substantially different cluster structures even though all three methods selected  $k = 2$  according to the highest Silhouette

Score. The detailed comparison of the three scaling methods at  $k = 2$  is presented in Table III.

TABLE III COMPARISON OF SCALING METHODS AT  $k = 2$ 

| Scaling method | Silhouette | DBI    | CHI       | Cluster sizes |
|----------------|------------|--------|-----------|---------------|
| RobustScaler   | 0.9218     | 0.3284 | 2857.8019 | 930 / 18      |
| StandardScaler | 0.2271     | 1.8914 | 190.3357  | 689 / 259     |
| MinMaxScaler   | 0.2691     | 1.4858 | 281.9338  | 727 / 221     |

RobustScaler was retained for the primary analysis because it produced the strongest internal separation and was consistent with the outlier-aware preprocessing objective. However, its high validation values were not interpreted independently of the extreme cluster-size imbalance. Additional feature-sensitivity and initialization-stability analyses were therefore conducted to determine the source and robustness of the resulting structure.

### B. Feature Sensitivity Analysis

The feature-sensitivity analysis revealed that the original two-cluster structure was strongly dominated by instrumentality. Using all eight audio features, K-Means produced a Silhouette Score of 0.9218 and a highly imbalanced cluster distribution of 930 and 18 songs. After instrumentality was excluded, the Silhouette Score decreased to 0.3853, while the cluster distribution changed to 784 and 164 songs. The Adjusted Rand Index between the two solutions was  $-0.0272$ , indicating that the cluster memberships were essentially inconsistent. The detailed results are presented in Table IV.

TABLE IV FEATURE-SENSITIVITY ANALYSIS

| Scenario                 | Silhouette | DBI    | CHI       | Cluster sizes | ARI       |
|--------------------------|------------|--------|-----------|---------------|-----------|
| All eight audio features | 0.9218     | 0.3284 | 2857.8019 | 930 / 18      | 1.0000    |
| Without instrumentality  | 0.3853     | 1.1325 | 401.0459  | 784 / 164     | $-0.0272$ |

The dominance of instrumentality was also supported by PCA. In the full-feature model, the first principal component explained 92.95% of the variance, while the first two principal components jointly explained 95.61%. After instrumentality was excluded, the variance explained by the first principal component decreased to 37.84%, while the first two components jointly explained 58.16%. This substantial reduction indicates that the strong low-dimensional separation in the original model was largely associated with instrumentality.

Furthermore, the leave-one-feature-out analysis showed that removing bpm, danceability, valence, energy, acousticness, liveness, or speechiness did not change the original 930–18 cluster membership, with an Adjusted Rand Index of 1.0000. Removing instrumentality was the only condition that fundamentally changed the clustering solution.

These findings demonstrate that the high Silhouette Score obtained using RobustScaler was primarily associated with the separation of a small group of songs having substantially higher instrumentality values. The clustering structure was therefore not jointly determined to the same extent by all eight audio features. The Adjusted Rand Index of  $-0.0272$  obtained after removing instrumentality indicates that the reduced-feature solution showed essentially no agreement with the original cluster membership beyond chance. Consequently, the 930–18 distribution should not be interpreted as a broad or balanced segmentation of popular Spotify songs. Instead, it represents a feature-dominant structure in which K-Means consistently isolates a small high-instrumentality subgroup from the majority of songs with very low instrumentality.

This result also demonstrates that internal validation metrics must be interpreted together with feature contribution and cluster-size distribution. A high Silhouette Score does not automatically indicate that all selected variables contribute equally or that the resulting clusters represent equally meaningful musical or market segments. In the present dataset, the high separation quality primarily reflects the distinctive position of the small high-instrumentality subgroup in the RobustScaler-transformed feature space.

### C. Initialization Stability Analysis

The initialization-stability analysis showed that the clustering solution was fully consistent across the tested configurations. A total of 150 K-Means runs were conducted using 30 random seeds and five  $n\_init$  settings consisting of 1, 5, 10, 20, and 50. All experiments produced the same cluster distribution of 930 and 18 songs. The Silhouette Score remained 0.9218, the Davies-Bouldin Index remained 0.3284, the Calinski-Harabasz Index remained 2857.8019, and the inertia values were identical across all tested configurations. The summarized initialization-stability results are presented in Table V.

TABLE V K-MEANS INITIALIZATION STABILITY RESULTS

| n_init | Number of runs | Mean Silhouette | SD Silhouette | Mean ARI | Cluster sizes |
|--------|----------------|-----------------|---------------|----------|---------------|
| 1      | 30             | 0.9218          | 0.0000        | 1.0000   | 930 / 18      |
| 5      | 30             | 0.9218          | 0.0000        | 1.0000   | 930 / 18      |
| 10     | 30             | 0.9218          | 0.0000        | 1.0000   | 930 / 18      |
| 20     | 30             | 0.9218          | 0.0000        | 1.0000   | 930 / 18      |
| 50     | 30             | 0.9218          | 0.0000        | 1.0000   | 930 / 18      |

The Adjusted Rand Index relative to the reference model was 1.0000 in every experiment, indicating identical cluster membership regardless of centroid initialization. Even when n\_init was set to 1, all 30 random seeds produced the same clustering solution. Therefore, the final result was not an accidental consequence of selecting random\_state = 42 or n\_init = 20.

The initialization-stability analysis confirms that the 930–18 clustering solution was not caused by a particular random seed or an accidental centroid configuration. However, stability across initializations does not by itself demonstrate that the clusters represent two naturally balanced or equally generalizable musical segments. When interpreted together with the feature-sensitivity and scaling-method analyses, the results indicate that K-Means consistently identifies a small high-instrumentalness subgroup under RobustScaler preprocessing. Therefore, the resulting solution is considered algorithmically stable but feature-dependent.

#### D. Determination of the Optimal Number of Clusters

The first stage in the results section was determining the most appropriate number of clusters. This was done using two approaches: the Elbow Method and Silhouette Score.

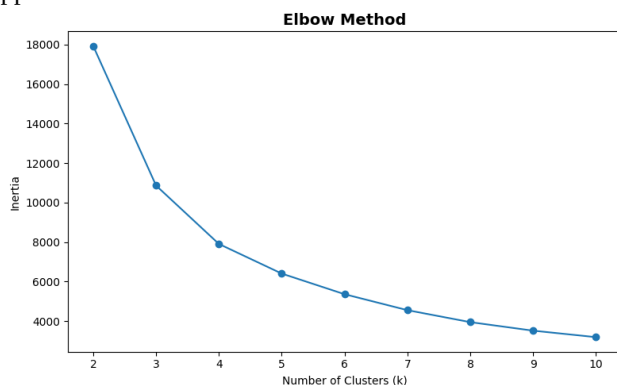


Figure 2. Elbow Method

The use of both approaches aimed to ensure that the selected number of clusters was not based only on the decrease in inertia but also considered the quality of separation between clusters. As shown in Figure 2, the inertia value decreased sharply from k = 2 to k = 4 and then tended to flatten at higher k values.

This pattern indicates that a relatively small number of clusters was sufficient to capture the main structure of the data. However, the Elbow Method graph did not provide a

very clear single elbow point, so the result needed to be confirmed using the Silhouette Score.

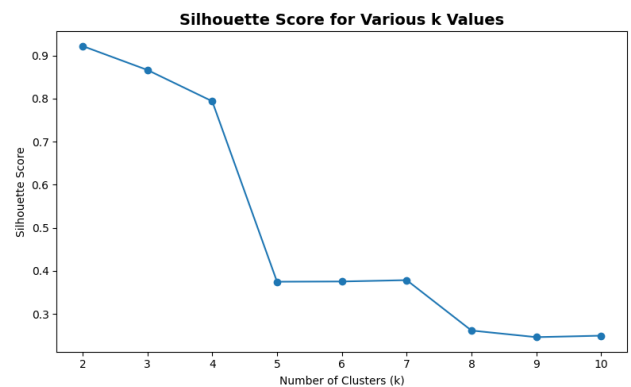


Figure 3. Silhouette Score for Various k Values

Figure 3 shows that the highest Silhouette Score was obtained at k = 2, with a value of 0.9218. The score then decreased to 0.8663 at k = 3 and continued to decline at subsequent k values. This condition indicates that the best separation occurred when the data were divided into two clusters because each object had high similarity to its own cluster and sufficient distance from other clusters. A higher Silhouette Score indicates better separation quality and cluster compactness.

Therefore, the value of 0.9218 at k = 2 indicates that the cluster structure in this study was very good. The decreasing silhouette values from k = 3 to k = 10 show that adding more clusters reduced the overall separation quality. Thus, the selection of k = 2 was based not only on the highest Silhouette Score but also on the comparison with the Elbow Method. Although the Elbow Method did not indicate a distinct elbow point, the Silhouette Score provided a clearer quantitative criterion for evaluating cluster separation. Therefore, k = 2 was selected as the final clustering configuration because it produced the best separation quality among the tested alternatives. The detailed Silhouette Score values are presented in Table VI.

Based on these results, k = 2 was selected as the final clustering configuration because it produced the highest Silhouette Score among the tested alternatives. The Elbow Method showed a progressive reduction in inertia at lower k values but did not reveal a single unambiguous elbow point. Therefore, the Elbow Method was used as supporting visual evidence rather than as the primary decision criterion. The final selection of k = 2 was primarily supported by the Silhouette Score, which provided a clearer quantitative indication of cluster separation quality.

TABLE VI  
SILHOUETTE SCORE VALUES FOR VARIOUS NUMBERS OF CLUSTERS

| Number of Clusters (k) | Silhouette Score |
|------------------------|------------------|
| 2                      | 0.9218           |
| 3                      | 0.8663           |

| Number of Clusters (k) | Silhouette Score |
|------------------------|------------------|
| 4                      | 0.7937           |
| 5                      | 0.3748           |
| 6                      | 0.3753           |
| 7                      | 0.3785           |
| 8                      | 0.2615           |
| 9                      | 0.2462           |
| 10                     | 0.2496           |

#### E. Clustering Results and Cluster Characteristics

After the number of clusters was set to two, the clustering process was performed using K-Means on the eight scaled audio features. The results showed that out of 948 songs, 930 songs were assigned to Cluster 0, while 18 songs were assigned to Cluster 1. The highly unequal cluster sizes indicate that the model primarily separated a small high-instrumentalness group from the dominant low-instrumentalness group. Therefore, the two clusters should not be interpreted as balanced segments jointly determined by all selected audio features. To understand the characteristics of each cluster, the mean value of each audio feature was calculated for each cluster. The results were then visualized using a heatmap.

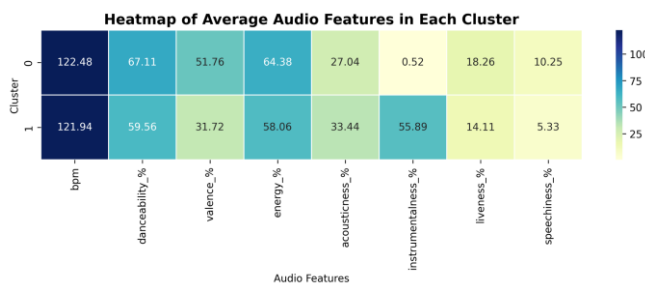


Figure 4. Heatmap of Average Audio Features in Each Cluster

As shown in Figure 4, there are clear differences in audio characteristics between the two clusters. The mean values of the audio features in each cluster are presented in Table VII.

TABLE VII  
AVERAGE AUDIO FEATURES IN EACH CLUSTER

| Audio Features       | Cluster 0 | Cluster 1 |
|----------------------|-----------|-----------|
| BPM                  | 122.48    | 121.94    |
| Danceability (%)     | 67.11     | 59.56     |
| Valence (%)          | 51.76     | 31.72     |
| Energy (%)           | 64.38     | 58.06     |
| Acousticness (%)     | 27.04     | 33.44     |
| Instrumentalness (%) | 0.52      | 55.89     |
| Liveness (%)         | 18.26     | 14.11     |
| Speechiness (%)      | 10.25     | 5.33      |

Table VII shows a clear difference in audio characteristics between Cluster 0 and Cluster 1. Cluster 0 has higher average values of danceability, valence, energy, and speechiness than Cluster 1, while its instrumentalness value is very low. This

indicates that songs in Cluster 0 tend to be more rhythmic, energetic, expressive, and vocally dominant. In contrast, Cluster 1 has a much higher instrumentalness value, accompanied by lower valence, energy, danceability, and speechiness values. This indicates that songs in Cluster 1 tend to be more instrumental, less vocally dominant, and relatively calmer. Therefore, Table VII confirms that the main difference between the two clusters lies in the level of instrumental dominance and overall audio intensity.

From a music information retrieval perspective, instrumentalness is not merely a descriptive label but an audio-content feature that represents the probability that a track contains no vocal content [26]. In the present study, Cluster 1 had an average instrumentalness value of 55.89%, compared with only 0.52% in Cluster 0. Cluster 1 also exhibited lower speechiness, danceability, valence, and energy. The combination of substantially higher instrumentalness and lower speechiness indicates that Cluster 1 represents a more instrumental-oriented and less vocal-dominant audio profile. In contrast, Cluster 0 is characterized by very low instrumentalness together with higher danceability, valence, energy, and speechiness, indicating a more vocal-oriented, rhythmic, and energetic profile. This interpretation is consistent with previous Spotify-based research showing that dance-oriented music tends to exhibit higher energy, danceability, and valence together with lower instrumentalness [2].

Nevertheless, the clusters should not be interpreted as specific genres or definitive musical categories. The analysis did not include genre labels, lyrical content, detailed timbral descriptors, harmonic structure, or listening-context information. Therefore, the terms instrumental-oriented and vocal-oriented are used only to describe the measurable audio-feature profiles identified in this dataset. The feature-sensitivity analysis further showed that instrumentalness was the principal variable separating the small minority subgroup from the dominant cluster. From an interpretable MIR perspective, the resulting structure therefore represents a distinctive audio-content pattern rather than two comprehensive genre or market segments.

#### F. Relationship Between Clusters and Streams

After the characteristics of each cluster were identified, the next stage was to analyze the relationship between the formed clusters and the number of streams.

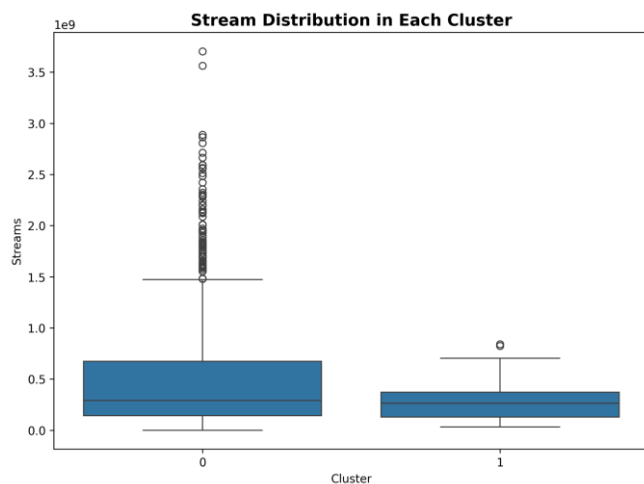


Figure 5. Stream Distribution in Each Cluster

This analysis aimed to identify which song group tended to have higher playback counts. As shown in Figure 5, the stream distribution in Cluster 0 appears wider than in Cluster 1 because Cluster 0 contains more songs and includes several songs with very high stream counts. However, this visual difference must be interpreted carefully because the number of songs in the two clusters is highly imbalanced.

The summary statistics are presented in Table VIII. These statistics were used as descriptive information, while statistical testing was conducted to examine whether the difference in stream counts between clusters was significant.

TABLE VIII.  
STREAM STATISTICS IN EACH CLUSTER

| Statistic       | Cluster 0     | Cluster 1   |
|-----------------|---------------|-------------|
| Number of songs | 930           | 18          |
| Mean streams    | 517,638,300   | 326,963,900 |
| Median streams  | 289,165,138.5 | 263,770,968 |
| Minimum streams | 2,762         | 31,959,571  |
| Maximum streams | 3,703,895,000 | 841,749,500 |

Table VIII shows that Cluster 0 contains far more songs, namely 930 songs, compared with Cluster 1, which contains only 18 songs. In terms of descriptive statistics, Cluster 0 has a higher average number of streams, namely 517,638,300, while Cluster 1 has an average of 326,963,900 streams. The median streams value in Cluster 0 is also slightly higher than that in Cluster 1. However, these values should not be interpreted as evidence that audio characteristics directly determine popularity because stream counts may be affected by many external factors beyond audio features.

To further examine whether the difference in stream counts between clusters was statistically significant, the Mann-Whitney U test was conducted. This test was selected because the number of songs in each cluster was highly imbalanced and stream counts generally showed a skewed distribution. The test produced a Mann-Whitney U statistic of 9447.0 and a p-value of 0.3495. Since the p-value is greater

than 0.05, the difference in stream counts between Cluster 0 and Cluster 1 is not statistically significant.

Therefore, the comparison of stream counts should be interpreted as a descriptive tendency within the analyzed dataset rather than as evidence of a statistically significant or causal relationship. Other factors, such as artist popularity, playlist placement, release timing, promotional strategy, and social media trends, may also influence the number of streams but were not included in the clustering model.

#### G. Cluster Separation Visualization Using PCA

To provide a visual overview of the separation between the two formed clusters, the clustering results were visualized in two dimensions using Principal Component Analysis (PCA).

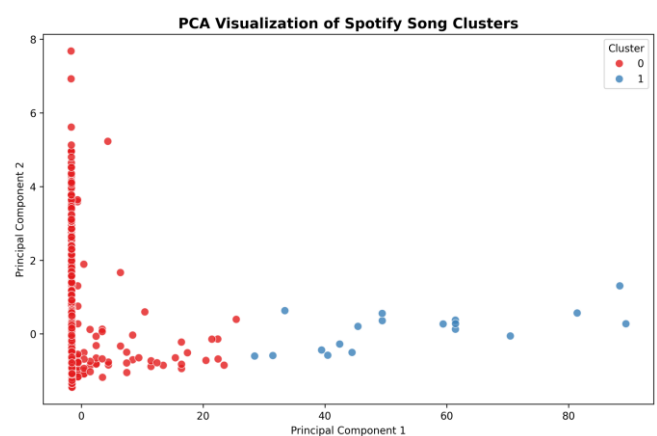


Figure 6. Song Cluster Visualization Using PCA

Figure 6 shows that the two clusters are separated quite clearly in the two-dimensional PCA space. The first principal component explained 92.95% of the data variance, while the second principal component explained 2.66%. Together, the two components explained 95.61% of the total variance. This indicates that the PCA visualization was able to represent most of the information contained in the selected audio features.

Most data points are concentrated in one large group, namely Cluster 0, while Cluster 1 appears as a small group that is relatively separated, especially along the first principal component axis. This visualization reinforces the finding that the dataset is dominated by one major audio-feature pattern and a smaller distinctive group. The visual separation shown by PCA is also consistent with the high Silhouette Score of 0.9218, although the imbalance in cluster size must still be considered when interpreting the result.

#### H. Cluster Size Imbalance and Feature-Dominant Structure

The final K-Means solution assigned 930 songs to Cluster 0 and only 18 songs to Cluster 1, indicating a highly imbalanced cluster-size distribution. Unequal cluster sizes do not automatically indicate clustering failure because unsupervised algorithms are not designed to produce equally

sized groups. Nevertheless, the magnitude of the imbalance requires additional examination before the clusters can be interpreted as meaningful song segments.

The initialization-stability analysis showed that the same 930–18 distribution was obtained across 150 combinations of random seeds and *n\_init* values. This finding confirms that the result was not caused by unstable centroid initialization. However, the scaling-method comparison showed that StandardScaler and MinMaxScaler produced substantially different and more balanced cluster distributions. Furthermore, excluding instrumentality changed the cluster distribution to 784 and 164 songs and produced an Adjusted Rand Index of  $-0.0272$  relative to the original solution.

These findings indicate that the 930–18 structure is stable under repeated K-Means initialization but sensitive to preprocessing choices and strongly dominated by instrumentality. Cluster 1 should therefore be interpreted as a small and distinctive high-instrumentality subgroup rather than as a second segment of equal importance to Cluster 0. The clustering result represents a dominant majority audio pattern accompanied by a specific minority pattern, rather than two balanced musical or market categories.

This interpretation also provides a more cautious explanation of the high Silhouette Score. The score reflects strong geometric separation in the RobustScaler-transformed feature space, particularly along the instrumentality dimension, but does not imply that all eight audio features contributed equally to cluster formation.

#### I. Correlation Analysis of Audio Features

A correlation analysis was conducted to examine the relationships among the selected audio features and to identify possible multicollinearity before interpreting the cluster results. Figure 7 shows that most correlations among the selected audio features were low to moderate.

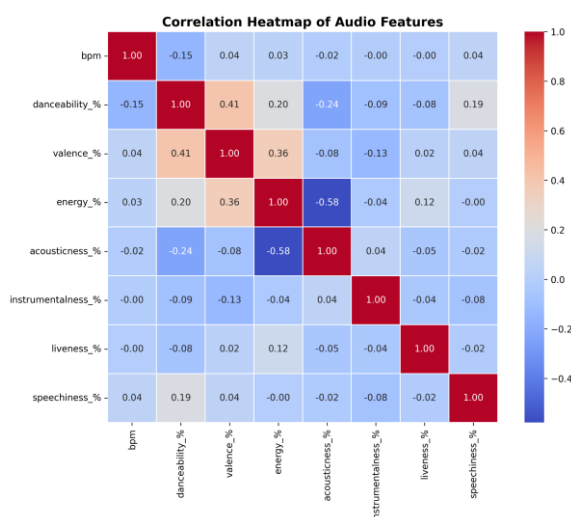


Figure 7. Correlation Heatmap of Audio Features

The strongest correlation was observed between energy\_% and acousticness\_% with a negative correlation of

$-0.58$ , indicating that songs with higher energy tended to have lower acousticness. Danceability\_% and valence\_% showed a moderate positive correlation of 0.41, while valence\_% and energy\_% showed a positive correlation of 0.36. No correlation exceeded 0.80, indicating that the selected features did not exhibit severe multicollinearity. Therefore, all eight features were retained because they represent different aspects of song audio characteristics.

#### J. Final Clustering Evaluation

The quality of the clustering results in this study was evaluated using three internal metrics: Silhouette Score, Davies-Bouldin Index, and Calinski-Harabasz Index. Based on the evaluation, the following values were obtained:

- Silhouette Score = 0.9218
- Davies-Bouldin Index = 0.3284
- Calinski-Harabasz Index = 2857.8019

The Silhouette Score, which is close to 1, indicates that objects within the same cluster have high similarity and are well separated from objects in other clusters. The relatively small Davies-Bouldin Index indicates that the similarity between clusters is low, so the separation between clusters can be considered good. Meanwhile, the high Calinski-Harabasz Index indicates that the ratio of between-cluster dispersion to within-cluster dispersion is large, meaning that the resulting cluster structure is good.

Based on the three internal validation metrics, the K-Means model with  $k = 2$  produced a compact and clearly separated structure in the RobustScaler-transformed feature space. However, these results should not be interpreted solely on the basis of the high Silhouette Score, low Davies-Bouldin Index, and high Calinski-Harabasz Index. The highly imbalanced cluster-size distribution and the feature-sensitivity analysis showed that the separation was primarily driven by instrumentality. Therefore, the final solution is interpreted as a geometrically strong but feature-dominant clustering structure rather than a balanced multidimensional segmentation of popular Spotify songs. To further assess whether K-Means remained an appropriate method for the analyzed dataset, its clustering performance and computational efficiency were compared with alternative clustering methods.

#### K. Comparison With Alternative Clustering Methods and Computational Efficiency Analysis

To strengthen the validation of the clustering results and evaluate computational efficiency, K-Means was compared with two alternative clustering methods, namely Agglomerative Clustering and Gaussian Mixture Model (GMM). The comparison was conducted using the same scaled audio features and the same number of clusters,  $k = 2$ . The evaluation considered both clustering quality metrics and computational performance.

TABLE IX.  
COMPARISON OF CLUSTERING METHODS AND  
COMPUTATIONAL EFFICIENCY

| Method        | Clusters | Runtime (s) | Silhouette | DBI    | CHI       |
|---------------|----------|-------------|------------|--------|-----------|
| K-Means       | 2        | 0.0361      | 0.9218     | 0.3284 | 2857.8019 |
| Agglomerative | 2        | 0.0589      | 0.9253     | 0.2766 | 2627.9847 |
| GMM           | 2        | 0.0467      | 0.7240     | 1.1313 | 455.1142  |

The comparison showed that Agglomerative Clustering achieved slightly better Silhouette and Davies-Bouldin Index values than K-Means, with Silhouette Scores of 0.9253 and 0.9218 and Davies-Bouldin Index values of 0.2766 and 0.3284, respectively. However, K-Means obtained the highest Calinski-Harabasz Index (2857.8019), indicating stronger between-cluster separation relative to within-cluster dispersion. Meanwhile, Gaussian Mixture Model produced lower clustering performance across all validation metrics.

In terms of computational efficiency, K-Means achieved the shortest execution time of 0.0361 seconds, followed by Gaussian Mixture Model at 0.0467 seconds and Agglomerative Clustering at 0.0589 seconds. These results indicate that K-Means provided competitive clustering quality with lower computational cost on the analyzed dataset. Accordingly, K-Means was retained as the primary method because it combined strong internal validation performance, relatively low execution time, and straightforward centroid-based interpretation.

However, the computational results should be interpreted within the scale of the present experiment. The dataset contained only 948 songs, and no experiments were conducted on progressively larger Spotify catalogs. Therefore, the reported runtime values demonstrate computational efficiency only for the analyzed dataset and should not be interpreted as evidence of large-scale scalability. In larger music catalogs, computational cost may increase with the number of observations, features, initialization runs, and clustering configurations. Future research should therefore evaluate runtime, memory consumption, and clustering stability across substantially larger datasets to determine whether the observed computational advantage of K-Means remains consistent at scale.

#### L. Overall Discussion and Practical Implications

The clustering results demonstrate that Spotify songs in the Most Streamed Spotify Songs 2023 dataset can be grouped into two main clusters based on audio-feature similarity. The first cluster, which is the dominant group, is characterized by higher danceability, energy, valence, and speechiness values, as well as very low instrumentality. These characteristics describe songs that are more rhythmic, lively, vocal, and energetic. The second cluster has the opposite characteristics, especially high instrumentality, so it can be interpreted as a group of more instrumental songs with less vocal prominence. However, the very large difference in cluster size indicates that the dataset is

dominated by one major audio-feature pattern and a small distinctive group rather than two balanced groups. The highly imbalanced distribution of 930 and 18 songs requires the resulting clusters to be interpreted cautiously. The initialization-stability experiments demonstrated that the small cluster was consistently detected regardless of centroid initialization. Nevertheless, the scaling-method comparison and feature-sensitivity analysis showed that the cluster structure was sensitive to preprocessing choices and was primarily driven by instrumentality. Therefore, Cluster 1 is better understood as an algorithmically stable minority subgroup with distinctive instrumental characteristics rather than as an equally prominent alternative segment of popular Spotify songs. This finding also demonstrates the importance of evaluating initialization stability, feature contribution, cluster-size distribution, and preprocessing sensitivity rather than relying exclusively on internal cluster-validation scores.

When linked to the number of streams, Cluster 0 had a higher average number of streams than Cluster 1. Nevertheless, the Mann-Whitney U test showed that the difference was not statistically significant. Therefore, the relationship between cluster membership and stream counts should be understood as a descriptive tendency only. The findings do not prove that more rhythmic, energetic, or vocal songs necessarily achieve higher streams, because song popularity can also be influenced by external factors such as artist popularity, promotional strategy, playlist placement, release timing, and social media trends.

Methodologically, these results show that the combination of transparent data cleaning, audio feature selection, outlier-aware preprocessing using RobustScaler, determination of the number of clusters using the Elbow Method and Silhouette Score, internal validation metrics, and comparison with alternative clustering methods can produce an interpretable clustering workflow. The limitation of this study is that the analysis used only one Spotify 2023 dataset; therefore, the results cannot be generalized to all Spotify songs or to other years without further validation using larger and more recent datasets.

Compared with clustering approaches that primarily report the selected algorithm and internal validation scores, the workflow applied in this study provides additional interpretive checkpoints throughout the analytical process. Preprocessing sensitivity was examined by comparing three scaling methods, feature dependence was evaluated through leave-one-feature-out analysis, initialization stability was assessed across 150 K-Means configurations, and the final solution was compared with alternative clustering algorithms. PCA and correlation analysis were further used to examine the structure of the audio-feature space, while stream counts were analyzed separately after clustering to avoid directly incorporating popularity into cluster formation. These complementary analyses make the workflow more transparent by showing not only the final clustering result but also how preprocessing choices, individual features,

initialization settings, and algorithm selection influence its interpretation.

From a practical perspective, the resulting clusters can support music analytics applications, such as playlist curation, recommendation system development, and market analysis based on audio characteristics. By identifying groups of songs with similar audio patterns, streaming platforms can obtain additional insights beyond popularity indicators alone and develop more interpretable approaches for organizing and analyzing music catalogs.

#### IV. CONCLUSION

Based on the results, the K-Means algorithm was able to cluster Spotify songs based on audio-feature similarity. The optimal number of clusters was determined as  $k = 2$  based on the highest Silhouette Score and comparison with alternative cluster configurations. Cluster 0 was characterized by relatively higher danceability, energy, valence, and speechiness values compared with Cluster 1, while Cluster 1 exhibited substantially higher instrumentality. Although the final solution produced a highly imbalanced distribution of 930 and 18 songs, the initialization-stability analysis showed that the cluster-membership structure remained consistent across the tested random seeds and  $n\_init$  configurations. However, the feature-sensitivity and scaling-method comparisons revealed that the separation was primarily driven by instrumentality and was sensitive to the selected preprocessing method. Therefore, the resulting clusters should be interpreted as an algorithmically stable but feature-dependent majority-minority structure rather than as two balanced multidimensional song segments. The analysis showed that Cluster 0 had a higher average number of streams than Cluster 1; however, the Mann-Whitney U test indicated that the difference was not statistically significant.

Therefore, the stream comparison should be interpreted as a descriptive tendency within the analyzed dataset rather than as evidence of a causal or statistically significant relationship between audio characteristics and song popularity. The contribution of this study lies in providing a systematic and interpretable clustering workflow that combines transparent data cleaning, outlier-aware preprocessing using RobustScaler, internal cluster validation, comparison with alternative clustering methods, and cautious post-clustering interpretation using stream counts.

Future research may use larger and more recent Spotify datasets, incorporate external popularity factors such as artist popularity, playlist placement, release timing, and social media trends, and compare additional clustering algorithms to validate the consistency of the findings.

#### REFERENCES

- [1] Y. Deldjoo, M. Schedl, and P. Knees, "Content-driven music recommendation: Evolution, state of the art, and challenges," *Comput. Sci. Rev.*, vol. 51, Art. no. 100618, 2024, doi: 10.1016/j.cosrev.2024.100618.
- [2] D. Duman, P. Neto, A. Mavrolampados, P. Toivainen, and G. Luck, "Music we move to: Spotify audio features and reasons for listening," *PLoS ONE*, vol. 17, no. 9, Art. no. e0275228, 2022, doi: 10.1371/journal.pone.0275228.
- [3] F. Khan, I. Tarimer, H. S. Alwageed, B. C. Karadağ, M. Fayaz, A. B. Abdusalomov, and Y.-I. Cho, "Effect of Feature Selection on the Accuracy of Music Popularity Classification Using Machine Learning Algorithms," *Electronics*, vol. 11, no. 21, Art. no. 3518, 2022, doi: 10.3390/electronics11213518.
- [4] H. S. Saragih, "Predicting song popularity based on Spotify's audio features: insights from the Indonesian streaming users," *J. Manag. Anal.*, vol. 10, no. 4, pp. 693–709, 2023, doi: 10.1080/23270012.2023.2239824.
- [5] R. J. Scarratt, O. A. Heggli, P. Vuust, and K. V. Jespersen, "The audio features of sleep music: Universal and subgroup characteristics," *PLoS ONE*, vol. 18, no. 1, Art. no. e0278813, 2023, doi: 10.1371/journal.pone.0278813.
- [6] R. J. Scarratt, O. A. Heggli, P. Vuust, and M. Sadakata, "Music that is used while studying and music that is used for sleep share similar musical features, genres and subgroups," *Sci. Rep.*, vol. 13, no. 1, p. 4735, 2023, doi: 10.1038/s41598-023-31692-8.
- [7] N. Rohman and A. Wibowo, "Clustering Of Popular Spotify Songs In 2023 Using K-Means Method And Silhouette Coefficient," *J. Pilar Nusa Mandiri*, vol. 20, no. 1, pp. 18–24, 2024, doi: 10.33480/pilar.v20i1.4937.
- [8] U. L. Musyarofah, S. N. Alima, and D. S. Y. Kartika, "KLASIFIKASI TOP 50 SPOTIFY TAHUN 2010-2019 MENGGUNAKAN METODE K-MEANS CLUSTERING," *Pros. Semin. Nas. Teknol. dan Sist. Inf.*, vol. 2, no. 1, pp. 215–220, 2022, doi: 10.33005/sitasi.v2i1.300.
- [9] C. Maheshwari, "Music Recommendation on Spotify using Deep Learning," *arXiv*:2312.10079, 2023.
- [10] P. B. Chopade, N. B. Chormale, P. P. Shinde, V. S. Shitole, S. S. Bere, and D. B. Hanchate, "Music Recommendation System on Spotify Using Deep Learning," *Int. J. Sci. Res. Eng. Manag. (IJSREM)*, vol. 8, no. 11, pp. 1–7, 2024, doi: 10.55041/IJSREM38868.
- [11] S. Shulhiyana and A. Voutama, "Penerapan Metode K-Means Clustering untuk Pengelompokan Musik Berdasarkan Karakteristik Audio," *J. Nas. Ilmu Komput.*, vol. 6, no. 2, pp. 59–70, 2025, doi: 10.47747/jurnalnik.v6i2.2751.
- [12] Branchris, K. A. Yech, and A. Wijaya, "Klasterisasi Lagu pada Dataset Spotify Berdasarkan Fitur Audio Menggunakan Algoritma K-Means," *Sinergi: J. Ilm. Multidisiplin*, vol. 2, no. 1, pp. 118–129, 2026, doi: 10.66914/kr91zt57.
- [13] F. R. Putra, S. M. Sinulingga, A. D. Rachmawanto, M. J. Yonathan, V. Wijaya, and V. Tundjungsari, "Perbandingan Metode K-Means dan Hierarchical Clustering pada Rekomendasi Musik Berbasis Audio Spotify Features Sistem," *Informatics Comput. Eng. J.*, vol. 6, no. 1, pp. 77–85, 2026, doi: 10.31294/icej.v6i1.11895.
- [14] A. F. Wibawa and J. Valentiya, "Analisis Pengelompokan Lagu Terpopuler Spotify Menggunakan Algoritma K-Means Berdasarkan Popularitas," *ROUTERS: J. Sist. Teknol. Inf.*, vol. 4, no. 1, pp. 53–63, 2026, doi: 10.25181/rt.v5i1.4713.
- [15] M. I. Firmansyah, R. S. Rohman, and E. Marsusanti, "Penerapan Algoritma Klastering K-Means Untuk Fitur Atribut Pada Layanan Streaming Musik Spotify," *Indonesian J. Comput. Sci.*, vol. 2, no. 2, pp. 58–66, 2023, doi: 10.31294/ijcs.v2i2.2465.
- [16] A. Tarek, "Most Streamed Spotify Songs 2023," Kaggle. [Online]. Available: <https://www.kaggle.com/code/adhamtarek147/most-streamed-spotify-songs-2023>
- [17] S. B. Kotsiantis, D. Kanellopoulos, and P. E. Pintelas, "Data Preprocessing for Supervised Learning," *Int. J. Comput. Sci.*, vol. 1, no. 1, pp. 111–117, 2006.
- [18] W. Nugraha, "Teknik Scaling Menggunakan Robust Scaler Untuk Mengatasi Outlier Data Pada Model Prediksi Serangan Jantung," *Techno. com*, vol. 23, no. 2, pp. 319–327, 2024.
- [19] A. A. Wani, "Comprehensive analysis of clustering algorithms: exploring limitations and innovative solutions," *PeerJ Comput. Sci.*, vol. 10, Art. no. e2286, 2024, doi: 10.7717/peerj-cs.2286.
- [20] P. J. Rousseeuw and M. Hubert, "Robust statistics for outlier

- detection,” *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 1, no. 1, pp. 73–79, 2011.
- [21] I. Ali, A. U. Rehman, D. M. Khan, Z. Khan, M. Shafiq, and J.-G. Choi, “Model Selection Using K-Means Clustering Algorithm for the Symmetrical Segmentation of Remote Sensing Datasets,” *Symmetry*, vol. 14, no. 6, Art. no. 1149, 2022, doi: 10.3390/sym14061149.
- [22] W. Junthopas and C. Wongoutong, “Pre-Determining the Optimal Number of Clusters for k-Means Clustering Using the Parameters Package in R and Distance Metrics,” *Appl. Sci.*, vol. 15, no. 21, Art. no. 11372, 2025, doi: 10.3390/app152111372.
- [23] M. Kossakov, A. Mukasheva, G. Balbayev, S. Seidazimov, D. Mukammejanova, and M. Sydybayeva, “Quantitative Comparison of Machine Learning Clustering Methods for Tuberculosis Data Analysis,” *Eng. Proc.*, vol. 60, no. 1, Art. no. 20, 2024, doi: 10.3390/engproc2024060020.
- [24] L. E. Ekemeyong Awong and T. Zielinska, “Comparative Analysis of the Clustering Quality in Self-Organizing Maps for Human Posture Classification,” *Sensors*, vol. 23, no. 18, Art. no. 7925, 2023, doi: 10.3390/s23187925.
- [25] A. M. Ikotun, F. Habyarimana, and A. E. Ezugwu, “Cluster validity indices for automatic clustering: A comprehensive review,” *Heliyon*, vol. 11, no. 2, Art. no. e41953, 2025, doi: 10.1016/j.heliyon.2025.e41953.
- [26] E. Zangerle, M. Pichl, and M. Schedl, “User Models for Culture-Aware Music Recommendation: Fusing Acoustic and Cultural Cues,” *Trans. Int. Soc. Music Inf. Retr.*, vol. 3, no. 1, pp. 1–16, 2020, doi: 10.5334/tismir.37.