Music Recommendation System

Stephanie Chen, Deepitha Mysore, Prameela Prakasha Kubsad

UCI School of Information and Computer Science

University of California, Irvine

Irvine, California USA

stephtc1@uci.edu, dmysorec@uci.edu, prameelp@uci.edu

***Abstract*–This paper presents a machine learning-based recommendation system that suggests 10 songs based on an existing song already in the system, or on a new song inputted into the system. Through the use of statistical and data science techniques, this research hopes to achieve an accurate system for song recommendation. This project utilizes the Kaggle dataset "Spotify Tracks Data," created by Maharshi Pandya, and contains a listing of songs, along with features such as danceability, energy, and valence [1]. By modeling the feature distribution with Gaussian Mixture Models (GMM), the system identifies underlying groupings within the music dataset and uses these clusters to assess song similarity. This approach balances interpretability and performance, effectively capturing nuanced patterns in musical attributes to deliver precise recommendations that align with user preferences.**

***Index Terms*–music recommendation system, gaussian mixture models (GMM), machine learning in music, song similarity, feature-based clustering, song feature analysis**

I. INTRODUCTION

Music recommendation systems are commonly used in modern music applications such as Spotify, Apple Music, and YouTube Music. These applications frequently provide their users with customized playlists and even popular features, such as Spotify Wrapped, will look at features in order to deliver statistics on listening history and comparisons to similar listeners. As the volume of available music continues to grow, there has been an increasing demand for personalized, efficient, and accurate recommendation systems. These systems are integral for enhancing the user experience and minimizing the time spent searching for enjoyable music content.

Recommendation systems can be built using a variety of approaches, depending on the type of data available and the desired functionality. Some common methods include collaborative filtering, content-based filtering, deep learning-based approaches, and reinforcement learning. While many existing systems rely on collaborative filtering or popularity-based methods, these approaches have notable limitations. Collaborative filtering can struggle to recommend songs if there is not enough data to look through, thereby reducing its effectiveness in smaller music libraries. It also has an over-reliance on behavioral data.

The importance of creating music recommendation systems lies in enhancing user experience, providing personalized suggestions, promoting music discovery, and fostering market competitiveness among rival music platforms. This project aims to address the gaps of collaborative filtering and popularity-based approaches by clustering songs on the intrinsic properties of music. However, these properties are highly subjective and can vary significantly among listeners, presenting a substantial challenge in leveraging them to create personalized recommendations. Nonetheless, they provide a valuable foundation for the system, bridging the gap between subjective interpretation and data-driven insights.

By leveraging the Gaussian Mixture Model approach to cluster a large song database, the system effectively captures complex, multidimensional song features to identify subtle patterns and relationships within the dataset. Given an input song (either an existing song in the system, or a new song the user will insert into the system), the system computes its likelihood across all clusters and identifies the 10 songs closest to the input in the feature space.

The remainder of this paper is organized as follows: Section 2 describes the approach used to develop the music recommendation system. Section 3 reviews the algorithm used in this paper, and Section 4 discusses the dataset and the process of feature extraction. Section 5 analyzes the results and evaluates the system's performance. Section 6 reviews related work in music recommendation and clustering techniques. Finally, Section 7 concludes the paper with a summary of findings and suggestions for future research directions.

TABLE I: Revised dataset, with new columns, taken from Maharshi Pandya's "Spotify Tracks Data" dataset.

Attribute	Description (Domain)
track_name	title of the track
popularity	popularity of the track (numeric: from 0 to 100, where 100 represents the most popular track)
danceability	how suitable the track is for dancing, based on factors like tempo, rhythm stability, beat strength, and overall regularity (numeric: from 0.0 to 1.0, where 1.0 is most danceable)
energy	perceptual measure of the track's intensity and activity level (numeric: from 0.0 to 1.0, 1.0 being the most high energy)
speechiness	measure of the presence of spoken words in a track (numeric: from 0.0 to 1.0, where 1.0 is exclusively speechlike)
acousticness	confidence measure of whether track is acoustic (numeric: from 0.0 to 1.0, where

	1.0represents high confidence that track is acoustic)
instrumentalness	predicts whether track contains no vocals (numeric: from 0.0 to 1.0, where 1.0 represents no vocal content)
valence	musical positiveness conveyed by a track (numeric: from 0.0 to 1.0, where 100 represents the most positive musicality)
tempo	estimated tempo of the track, measured in beats per minute (BPM)
track_genre	genre to which the track belongs

II. APPROACH

A. Cleaning

TABLE I shows the various factors, along with specific descriptions and explanations used in this paper. This dataset is taken from the kaggle website [1], where the creator of the dataset, Maharshi Pandya, collected and cleaned the data using Spotify's Web API and Python.

Several additional columns are present in the original dataset, but they were removed for the purposes of this. Specifically, the columns “artists,” “album_name,” “duration_ms,” “explicit,” “key,” “loudness,” “mode,” and “time_signature” were excluded, as they were not considered to be significant features that accurately represent a song’s intrinsic characteristics relevant to the recommendation system.

The categorical columns, “track_name” and “track_genre,” are converted into numerical values, and all values are scaled from a scale of 0.0 to 1.0. Due to the large size of the dataset, the top 50 most popular songs were selected from each genre to include in the system. Additionally, the “popularity” column was removed to prevent it from becoming an overly influential factor in the cluster process. The track names served as the dependent variable (y), while the features “danceability,” “energy,” “speechiness,” “acousticness,” “instrumentalness,” “valence,” “tempo,” and “track_genre” were used as the independent variables (x).

B. Experimentation

K-means clustering was initially implemented for the system, with the assumption that it could group songs based on features such as energy and tempo. However, K-Means struggled to identify meaningful clusters due to its sensitivity to the initial placements of centroids and the varying scales of features. This approach proved ineffective when the data assumed that clusters have the same variance and faces difficulty when clusters overlap. This led to a switch to Gaussian Mixture Models (GMM), which are better equipped to handle the complexity and variability within the dataset.

C. Plotting GMM and Recommendations

After applying the elbow method and GMM to the dataset, the resulting clusters were visualized in a scatterplot to better understand the groupings formed by the model. Each song was assigned a probability distribution across the clusters, with the GMM algorithm identifying the most likely cluster for each song based on its feature set. This allowed for a comprehensive view of how the songs were distributed across the different clusters and facilitated the identification of similar songs for recommendations.

III. ALGORITHM

A Gaussian Mixture Model is a probabilistic model that assumes a data set is generated from a mixture of several Gaussian distributions with unknown parameters. It is widely used for clustering, density estimation, and as a generative model in various machine learning tasks. Unlike K-Means or other clustering algorithms, Gaussian Mixed Models are more effective when performing soft clustering and when working with different variances and orientations.

Properties of GMM include its significant flexibility, as GMMs can represent a wide range of distributions. It also includes soft clustering, in which GMM assigns a probability to each data point for belonging to each cluster. This means that data points can belong to multiple clusters to different degrees. Another property of GMM is its iterative estimation, as the parameters of GMM are typically estimated using the Expectation-Maximization (EM) algorithm. This particular algorithm iterates between the E-step, which computes the probabilities that the data point belongs to a component, and the M-step, which updates the parameters based on the current responsibilities. However, GMM does not guarantee global optimality, so different initializations can lead to different results. Additionally, GMM can model clusters with varying shapes by adjusting the covariance structure.

IV. DATA-ANALYSIS

A. Evaluation Goals

The goal of this project is to recommend 10 songs that are most similar to a given input song based on a set of musical and audio features. The recommendations are generated by clustering the songs into groups with similar characteristics using the Gaussian Mixture Model (GMM) and identifying the closest matches within the same cluster. This approach ensures contextual relevance, with recommendations grounded in both objective data features and their relationships in a clustered space. The analysis focuses on identifying proximity in the feature space by ensuring that the recommended songs are not just in the same cluster but are quantitatively close to the input song based on feature values such as danceability, energy, valence, etc. By using a clustering model combined with distance-based analysis, the system ensures that the song recommendations are both relevant and accurate.

B. Metrics Used for Evaluation

The analysis focused on key musical attributes across over 125 genres. These features were used to create a multidimensional representation, with the categorical variable "track_name" included as a proxy for broader genre classification. Together, these attributes ensured that the clustering captured both quantitative aspects (e.g., tempo) and qualitative aspects (e.g., mood and style) of the songs, leading to a more comprehensive and accurate clustering of the music. To generate 10 recommendations for the input song, the song was first located in the feature space and assigned to its corresponding cluster. The system then identified the top 10 songs within the same cluster that were closest to the input song. Proximity was calculated using pairwise distance, and the 10 songs with the smallest distances to the input song were chosen as the recommendations.

C. Results

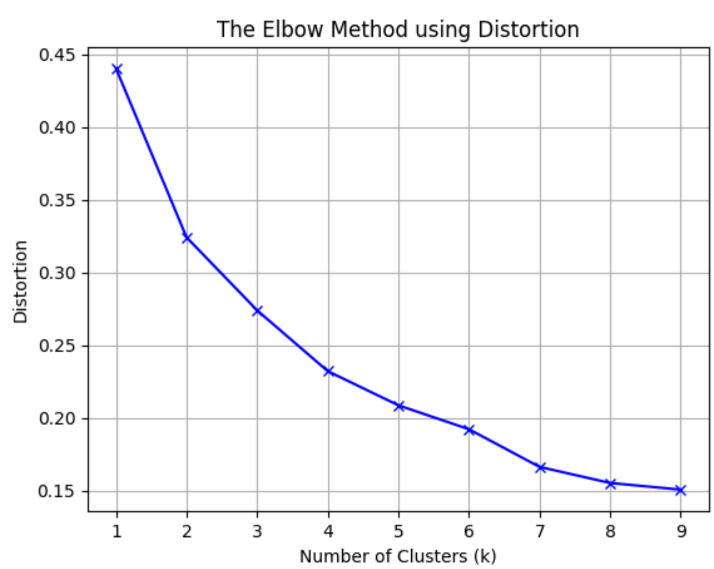

Fig. 1: Graph of Elbow Method Using Distortion

Fig. 1 illustrates the Elbow Method applied to determine the optimal number of clusters (k) for the K-Means clustering algorithm. The X-axis represents the number of clusters (k), while the Y-axis indicates the distortion, calculated as the average squared Euclidean distance between each data point and its nearest cluster center. The plot shows a sharp decrease in distortion as K increases, followed by a gradual flattening of the curve.

The "elbow point" marks the value of k where the distortion no longer significantly decreases with additional clusters. This suggests that adding more clusters beyond this point yields diminishing returns. In this case, the elbow point appears at k=4, indicating that 4 clusters are optimal for this dataset when using K-Means. This insight ensures an appropriate balance between model complexity and clustering performance.

Fig. 2: Graph of GMM Clustering Results with an Existing Song

Fig. 2 presents the clustering results obtained using the Gaussian Mixture Model (GMM). The scatter plot depicts the data points in two dimensions, with colors representing the 6 clusters identified by the GMM algorithm. Each cluster is probabilistically assigned based on the likelihood of a data point belonging to that cluster, allowing for overlapping and more flexible boundaries compared to K-Means. Unlike K-Means, GMM does not assume spherical clusters but instead models each cluster as a multivariate Gaussian distribution, which can better capture complex cluster shapes. The distinct color regions in the plot demonstrate the GMM’s ability to segment the data into meaningful groups. This visualization highlights the GMM's capability to handle soft clustering, where each point has a degree of membership in multiple clusters, making it suitable for datasets with overlapping or non-spherical clusters.

Fig. 3: Graph of GMM Clustering Results with a New Song

Fig. 3 shows the clustering results obtained using the Gaussian Mixture Model (GMM) when a new feature is added to the input dataset. The image is similar to Fig. 2 as one can observe distinct clusters formed based on the updated feature set. However, the addition of the new feature enhances the separability of the clusters, making them more defined compared to Fig. 2. This improvement indicates that the newly introduced feature contributes significantly to the clustering process by providing additional dimensions for differentiation. The clustering still demonstrates the probabilistic nature of GMM, with some overlap between clusters, but the overall structure suggests improved accuracy in grouping similar data points. This highlights the robustness of the GMM in adapting to changes in the feature space while maintaining the integrity of the clustering results.

V. EVALUATION AND DISCUSSION

The implemented music recommendation system demonstrates strong performance due to thoughtful preprocessing and algorithm selection. Normalization via Min-Max Scaling ensures that all features, such as tempo and danceability, are scaled to the same range, preventing any single feature from disproportionately influencing clustering. This is especially critical for algorithms like K-Means and Gaussian Mixture Models (GMM), which are sensitive to feature scales. However, if the dataset contains significant outliers, Min-Max Scaling can skew the results, potentially reducing the quality of clusters. An alternative scaling method, such as StandardScaler or robust scaling, could alleviate this issue.

The choice of features, including popularity, danceability, and energy, aligns well with the goal of identifying clusters that represent meaningful song groupings. These features capture the essence of a song's appeal and genre characteristics. However, the inclusion of irrelevant or redundant features could introduce noise, degrading the quality of clustering. Feature selection techniques or dimensionality reduction methods like Principal Component Analysis (PCA) might further enhance clustering performance. The elbow method used to determine the optimal number of clusters ensures that the model balances complexity and interpretability.

Both K-Means and GMM are employed for clustering, each with its strengths and limitations. K-Means performs well on datasets with well-separated, spherical clusters but struggles with overlapping or irregularly shaped clusters. GMM, on the other hand, is more flexible, modeling clusters as Gaussian distributions that vary in shape and size. This flexibility allows GMM to capture more nuanced relationships between songs. However, GMM is computationally intensive, sensitive to initialization, and may struggle with high-dimensional or non-Gaussian data distributions.

The system's effectiveness is largely dependent on the structure of the data and the specific use case. K-Means excels when the dataset contains clusters that are roughly spherical, well-separated, and of similar sizes. Its computational efficiency and scalability make it suitable for large datasets. However, when the clusters have overlapping features or irregular shapes, the GMM algorithm is preferable due to its ability to handle soft clustering. GMM assigns probabilities to each data point, allowing for nuanced groupings that better reflect real-world variability in song features.

The recommendation logic, which identifies similar songs within the same cluster using pairwise distances, enhances the system's relevance. By recommending songs based on proximity within feature space, the system ensures that suggestions align closely with the characteristics of the input song. This is particularly beneficial in contexts where users are looking for songs that share specific attributes, such as danceability or tempo. The elbow method further strengthens the system by providing a systematic approach to selecting the optimal number of clusters, ensuring a balance between model complexity and performance.

Despite its strengths, the program may fail to perform optimally in certain scenarios. K-Means struggles with non-spherical clusters or datasets where clusters have varying densities or sizes. For example, if the dataset includes overlapping genres with shared characteristics, K-Means may incorrectly group them into the same cluster. Similarly, GMM, while more flexible, can suffer from poor initialization, leading to suboptimal clustering. It is also prone to overfitting when the number of Gaussian components is too high, reducing its generalization ability.

The recommendation logic may also falter if the clusters are not well-defined or if the dataset lacks sufficient diversity. For instance, in genres with a small number of songs or limited feature variability, the recommendations may appear repetitive or fail to capture nuanced user preferences. The reliance on pairwise distances could further exacerbate this issue if the distances fail to reflect meaningful similarities between songs.

Several improvements could enhance the performance and robustness of the system. First, employing more robust scaling techniques, such as StandardScaler, could mitigate the impact of outliers. Additionally, applying dimensionality reduction methods like PCA could reduce noise and improve clustering by focusing on the most relevant features.

For the clustering algorithms, initializing K-Means with methods like K-Means++ or using ensemble clustering could provide more consistent results. Similarly, Bayesian Gaussian Mixture Models (BGMM) could offer a more robust alternative to GMM by automatically determining the optimal number of components.

Lastly, the recommendation logic could be improved by introducing weighted distances or hybrid approaches that combine clustering results with collaborative filtering techniques. This would enhance the system's ability to generate personalized recommendations, particularly in edge cases where clusters overlap or lack diversity.

VI. RELATED WORK

The paper, "A Music Recommendation System Based on Music and User Grouping," by Chen and Chen presents a personalized music recommendation system that integrates both music content analysis and user behavior [2]. Their approach involves analyzing polyphonic MIDI music files to extract six features from a representative track of each piece, facilitating music grouping. Additionally, they examine user access histories to develop profiles of user interests and behaviors, enabling user grouping. The system employs content-based, collaborative, and statistics-based recommendation methods, considering users' preferences for specific music groups and their affiliations with particular user groups. Chen and Chen's model excels in comprehensive feature extraction and this detailed representation of each music piece significantly enhances the accuracy of music grouping. Furthermore, the model's hybrid recommendation approach enables the system to leverage a variety of data sources, potentially improving the quality and relevance of the recommendations.

A key limitation of the model lies in its reliance on the MIDI format, which may restrict its applicability since not all music is available in this format. This could limit the diversity of recommendations. Additionally, the system's dependence on user data introduces challenges, as the effectiveness of user grouping hinges on the availability and accuracy of user access histories. Sparse or noisy data could reduce the system's overall performance.

Paul and Kundu's paper, "A Survey of Music Recommendation Systems and a Novel Approach for a Hybrid Music Recommendation System", provides a detailed exploration of existing music recommendation techniques [3]. It discusses the limitations of current systems, including content-based, collaborative, and emotion-based methods, while proposing a hybrid model to address identified challenges. The authors aim to improve recommendation accuracy and user satisfaction by leveraging a combination of techniques. Paul and Kundu's model analyzes the strengths and weaknesses of current methods in music recommendation systems, and the authors provide valuable insights into the current landscape of music recommendation technologies. While the proposed hybrid system offers a theoretical framework for improvement, the paper lacks empirical validation through implementation and testing. This absence of practical evaluation makes it difficult to assess the real-world effectiveness and scalability of the proposed system. Additionally, the model does not extensively consider the integration of advanced machine learning techniques, such as deep learning, which have shown promise in recent music recommendation research.

When compared to the models proposed by Chen and Chen, and Paul and Kundu, our Gaussian Mixture Model (GMM)-based approach stands out for its flexibility and efficiency. Unlike Chen and Chen's reliance on the MIDI format, which limits the diversity of recommendations, our model works with a broader range of features such as danceability, genre, and liveliness, ensuring applicability across various music types. While Chen and Chen incorporate user behavior for personalization, their dependency on user access histories introduces vulnerabilities when data is sparse or noisy. Our system overcomes this limitation by focusing solely on the intrinsic attributes of songs, offering consistent and reliable recommendations regardless of user-specific data availability.

Paul and Kundu’s hybrid model offers theoretical improvements but lacks empirical validation and the integration of advanced machine learning techniques. In contrast, our GMM-based system provides a scalable, data-driven solution that effectively captures natural groupings in music features. Its probabilistic clustering not only delivers precise recommendations but also offers insights into the similarities between tracks. Additionally, our model avoids the complexity and resource requirements of deep learning-based systems, making it more accessible and efficient. By balancing accuracy, scalability, and simplicity, our system presents a robust framework that stands as a practical and compelling alternative to the compared models, with potential for further enhancement.

VII. CONCLUSION

Recommendation systems have become an integral part of modern technology, shaping user experiences across various domains. From e-commerce platforms like Amazon suggesting products to music streaming services like Spotify personalizing playlists, these systems utilize advanced models such as collaborative filtering, content-based filtering, and deep learning to deliver tailored recommendations. Learning about recommendation systems is crucial as they exemplify the practical application of machine learning and data science, addressing real-world challenges like information overload and user personalization. Their implementation demonstrates how data-driven insights can be leveraged to improve decision-making, optimize user experiences, and drive business outcomes. As technology continues to evolve, mastering recommendation systems equips individuals and organizations with the tools to innovate and thrive in a data-centric world.

Implementing Gaussian Mixture Models (GMMs) in our music recommendation system offers significant advantages by excelling in unsupervised learning, enabling the discovery of natural groupings in music features without requiring labeled data. GMMs assume data points are generated from a mixture of Gaussian distributions, providing flexibility to model diverse cluster shapes and complex data distributions. This adaptability allows GMMs to capture the underlying structure of music features, identifying distinct clusters that represent various musical styles or characteristics. By effectively grouping similar songs, GMMs enhance the relevance and personalization of recommendations, making them ideal for uncovering patterns and improving the system's clustering process. Additionally, their probabilistic nature provides insights into the likelihood of a song belonging to multiple clusters, accommodating the nuanced overlap between musical styles. This robust approach ensures the system can handle varying feature distributions, further refining the accuracy and quality of music recommendations.

References

[1] Pandya, M. (2022, October 22). *Spotify Tracks Database*. Kaggle.

[2] Chen, H. C., & Chen, A. L. (2005). A music recommendation system based on music and user grouping. *Journal of Intelligent Information Systems*, 24, 113-132.

[3] Paul, D., & Kundu, S. (2020). A Survey of Music Recommendation Systems with a Proposed Music Recommendation System. *Advances in Intelligent Systems and Computing* 937, 279–285.