Music Recommendation System

Tools and Technique Lab
Project Presentation

By:

Praneesh Sharma, 21052264
Nayeer Naushad, 21052338
Ronita Chakraborty, 21052697
Antra Amrit, 21052780
Aditya Sharma, 21052472

ABSTRACT

This project report presents the development and evaluation of a music recommendation system utilizing KMeansClustering to identify similar songs based on audio features. The primary objective of the project was to create a robust recommendation system capable of suggesting relevant music tracks to users based on their preferences. The methodology involved collecting a diverse dataset of music audio features, preprocessing the data, and applying KMeansClustering to group similar songs into clusters.

Key findings from the project include the successful implementation of the KMeansClustering algorithm to effectively group songs with similar audio features. The system demonstrated promising results in recommending songs that align with users' preferences, showcasing its potential for personalized music recommendations. Additionally, the project highlighted the importance of feature engineering and data preprocessing techniques in enhancing the clustering accuracy and recommendation quality.

TABLE OF CONTENTS

Introduction

Music recommendation systems have become integral components of modern music streaming platforms, catering to users' diverse musical preferences and enhancing their listening experience. With the exponential growth of digital music libraries, users often face challenges in discovering new songs that resonate with their tastes amid a vast pool of choices. This project addresses the need for an intelligent music recommendation system that leverages machine learning algorithms to deliver personalized song suggestions to users.

Traditional methods of music recommendation, such as collaborative filtering and content-based filtering, have limitations in capturing the intricate nuances of users' music preferences and delivering accurate recommendations. Therefore, there is a pressing need for a more sophisticated recommendation approach that takes into account the audio features of songs to identify similarities and recommend relevant tracks effectively

METHODOLOGY

The methodology employed in developing the music recommendation system adhered to a structured scientific approach. Data preprocessing was conducted using Python's pandas and NumPy libraries, ensuring data integrity and handling missing values. Exploratory data analysis (EDA) techniques were then applied with Matplotlib and Seaborn to visualize the distribution and characteristics of audio features. Feature engineering involved selecting pertinent audio attributes and scaling them appropriately for subsequent analysis. The core technique, KMeans clustering, was implemented using scikit-learn to group similar songs into clusters. Euclidean distance, a metric within KMeans clustering, calculated the similarity between data points, aiding in generating accurate song recommendations.

METHODOLOGY

KMeans clustering is a popular unsupervised machine learning algorithm that partitions data into K clusters by minimizing the within-cluster sum of squares (WCSS). It iteratively assigns data points to the nearest cluster centroid and updates centroids until convergence. Euclidean distance, a fundamental distance metric, calculates the straight-line distance between two points in multidimensional space, serving as a crucial component in the KMeans clustering process. By quantifying the similarity between data points, Euclidean distance aids in effectively grouping songs with similar audio features into clusters, facilitating accurate and personalized music recommendations.

IMPLEMENTATION

Data Preprocessing

The first step was data preprocessing, which included loading the dataset from Kaggle into a pandas DataFrame. The dataset contained audio features of songs such as tempo, key, energy, and acousticness. Challenges arose during data cleaning and handling missing values, which were addressed by applying appropriate data imputation techniques and ensuring data integrity before proceeding to the next steps.

Exploratory Data Analysis (EDA)

EDA was performed using pandas and NumPy to gain insights into the distribution of audio features, identify outliers, and understand the characteristics of the dataset. Visualizations generated using Matplotlib and Seaborn helped in visualizing trends and patterns, aiding in feature selection and clustering decisions.

IMPLEMENTATION

Feature Engineering

Feature engineering involved selecting relevant audio features for clustering and recommendation purposes. Challenges were encountered in determining the optimal feature subset and scaling features appropriately for clustering algorithms. Feature scaling techniques such as Min-Max scaling and Standardization were applied to address these challenges and ensure uniform feature ranges for clustering.

Data Visualization

In the data visualization phase, trends in acousticness and energy were explored to understand their distributions and potential correlations. Matplotlib and Seaborn were used to create histograms, box plots, and violin plots to visualize the distributions of acousticness and energy across songs. These visualizations provided insights into the range and variability of these features, aiding in feature engineering decisions. Additionally, a regression plot (regplot) was created to visualize the relationship between acousticness and energy.

IMPLEMENTATION

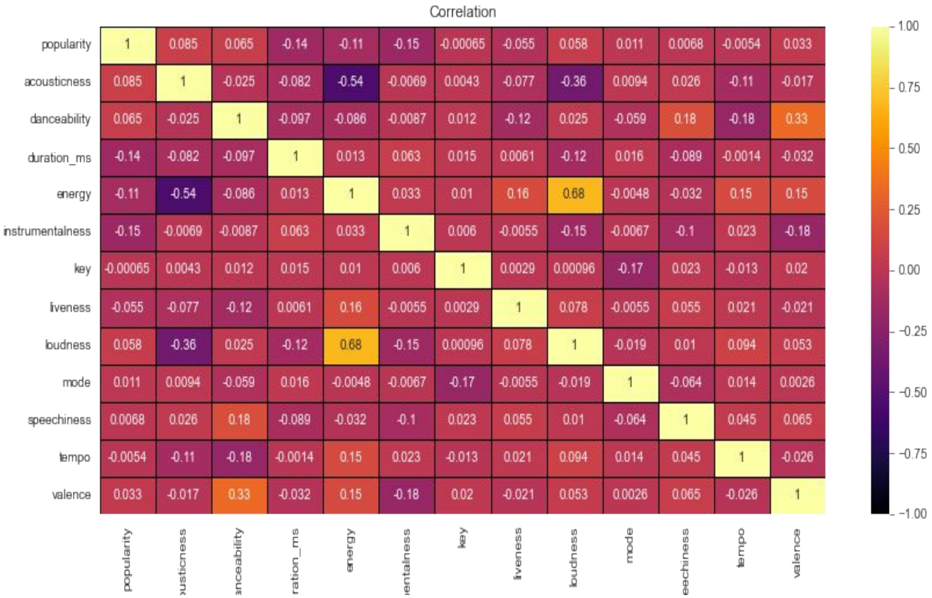
Clustering with KMeans

The KMeans clustering algorithm was implemented using scikit-learn, with the number of clusters determined through iterative experimentation and evaluation. 5 Challenges included selecting an optimal number of clusters and interpreting cluster results. The Elbow Method was employed to identify the optimal number of clusters and evaluate clustering performance. The elbow graph was plotted, and the clusters were visualized using a scatter plot, with the 3 primary colors representing the clusters.

Recommendation Generation

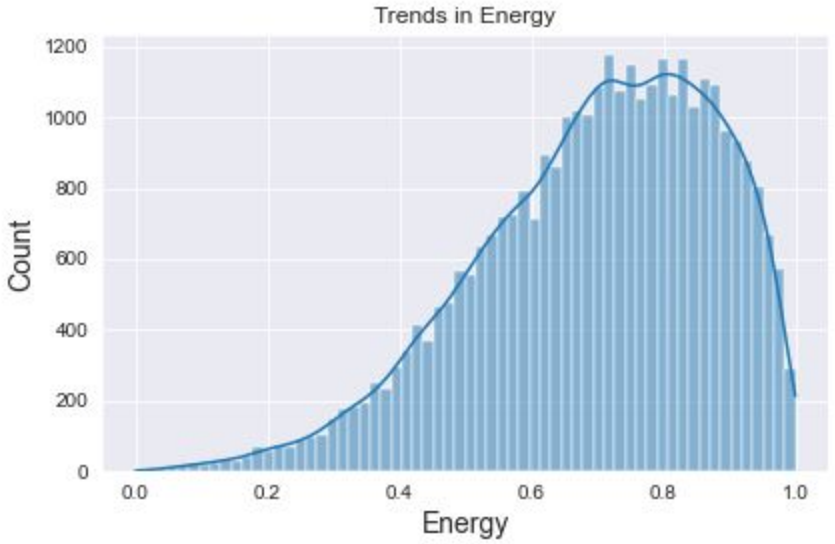
Once clusters were formed, recommendations were generated based on the Euclidean distance between songs and user preferences. Challenges included handling recommendation diversity and ensuring relevant recommendations for diverse user preferences. Solutions involved incorporating diversity constraints andrefining recommendation strategies based on user feedback.

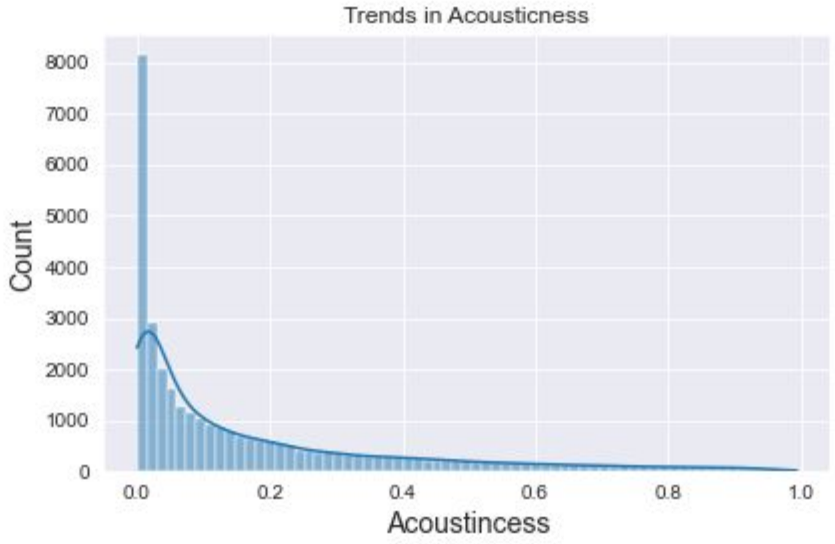
RESULTS AND ANALYSIS



From the heatmap, it is evident that the acousticness and energy are highly correlated with a value of 0.54 Hence, we chose to consider only these 2 parameters while offering recommendations.

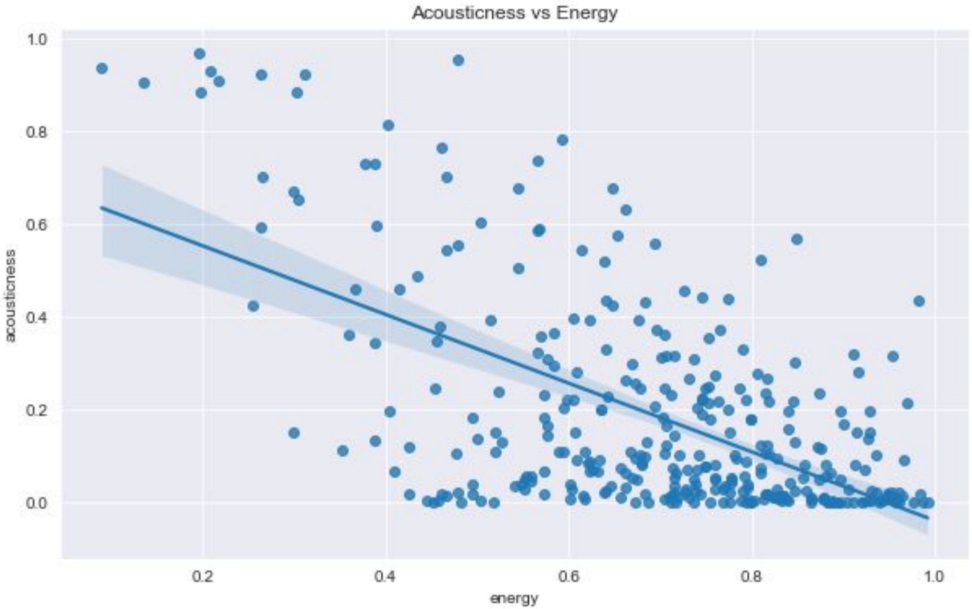
RESULTS AND ANALYSIS





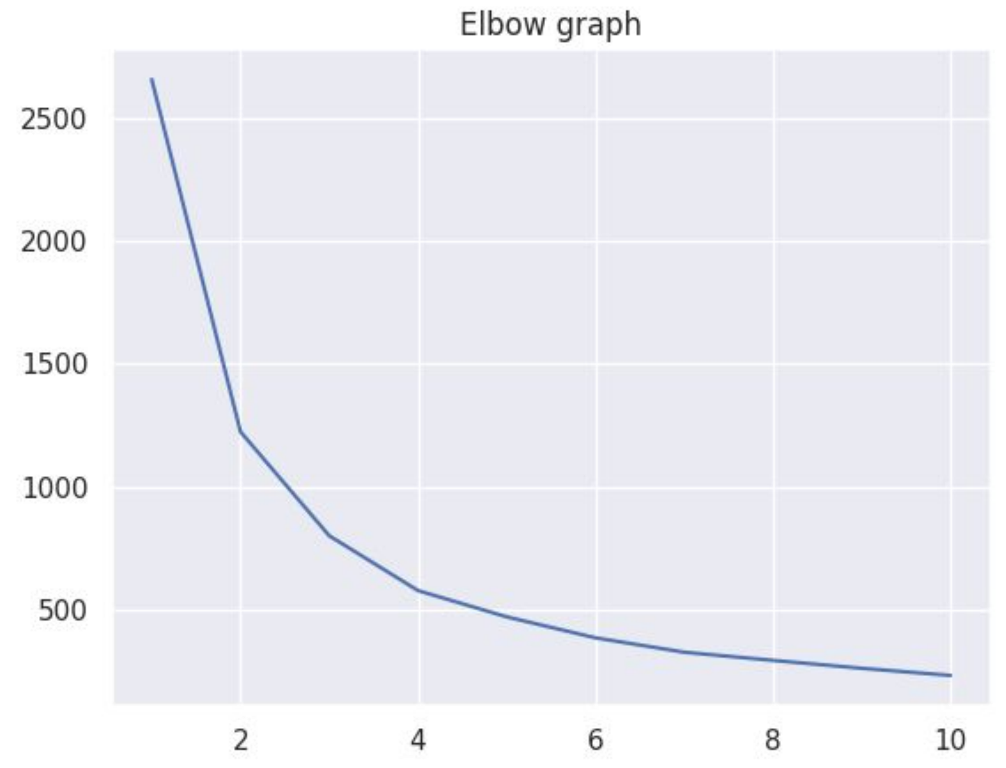
The above graphs show the trends in the acousticness and energy of the music in the dataset. We can see that the majority of the music has low acousticness. Meanwhile, energy is distributed almost equally in past 0.6 to 1.0

RESULTS AND ANALYSIS



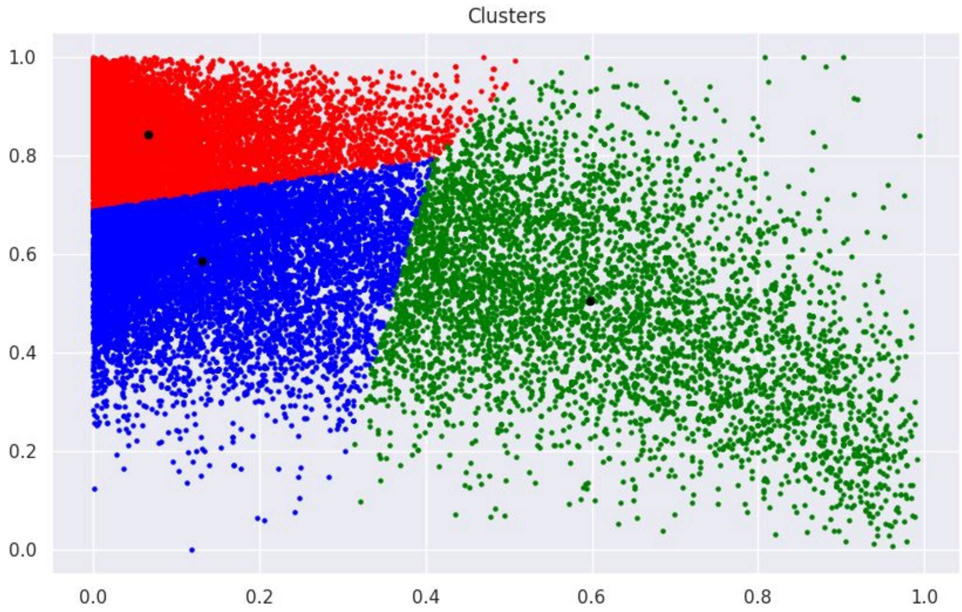
This graph shows the relation between acousticness and energy. As seen, the acousticness is inversely proportional to energy, ie, acousticness decreases with increase in energy of a song

RESULTS AND ANALYSIS



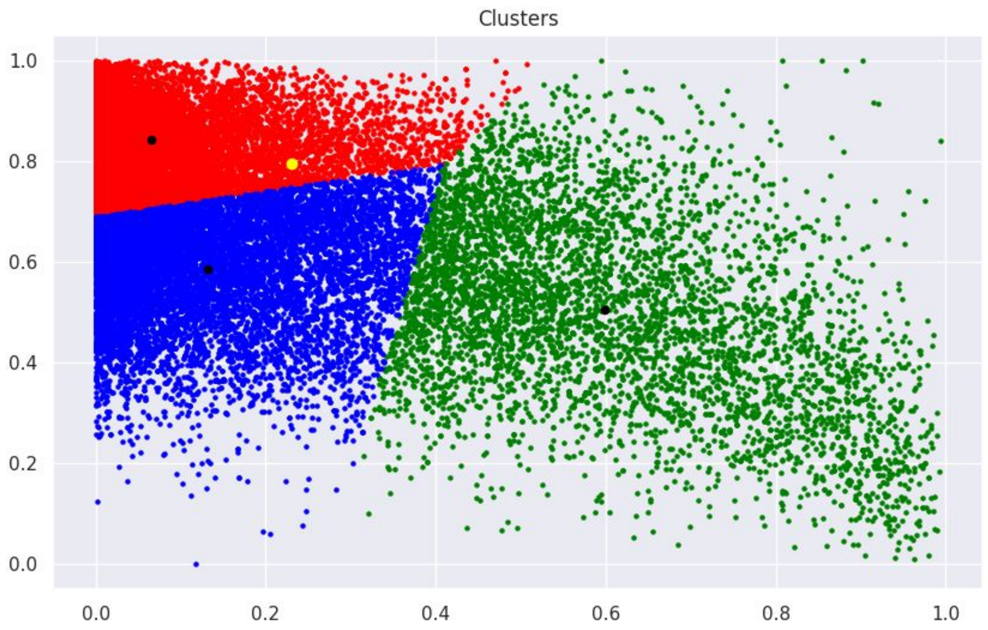
The graph above shows the Elbow Graph. An elbow graph is a visual representation used in clustering analysis to identify the optimal number of clusters. It plots the number of clusters against the within-cluster sum of squares (WCSS). We have gotten an ideal graph when the number of clusters was set to 3.

RESULTS AND ANALYSIS



This scatterplot shows all the datapoints from the dataset. Each point represents features of a particular song. The points are distributed into 3 clusters based on the output of the KMeansClusterning algorithm. The black points represent the centroids of the clusters

RESULTS AND ANALYSIS



This the final graph. The yellow point represents the song input by the user. The program finds the cluster the song belongs to, and then finds the 5 nearest songs based on Euclidean distance.

CONCLUSION

In conclusion, this project has made significant strides in the development and evaluation of a music recommendation system leveraging KMeans clustering, with a focus on personalized song suggestions based on audio features. The main findings from the project underscore the efficacy of KMeans clustering in effectively grouping similar songs, thereby enabling the system to provide accurate recommendations aligned with users' unique music preferences.

Looking ahead, future work could focus on several areas for improvement and expansion. Enhancing recommendation diversity to cater to a broader range of user preferences, integrating user feedback mechanisms for continuous learning and refinement, and exploring advanced clustering algorithms or deep learning models are avenues worth exploring. Additionally, expanding the dataset with a more extensive collection of music genres, incorporating contextual factors such as user mood or activity, and exploring hybrid recommendation approaches could further enhance the system's accuracy and relevance.

THANK YOU