ML PROJECT

Song Recommendation System

1. Objective

The objective of this project is to develop a content-based Song Recommendation System using machine learning techniques. The system aims to recommend songs to users based on their preferences by analyzing audio features such as danceability, energy, tempo, and valence. The model learns the similarity between songs based on their audio characteristics and suggests similar tracks to users.

2. Dataset Used

The dataset used in this project is sourced from Kaggle:

🔗 Spotify Tracks Dataset by Maharshi Pandya:

https://www.kaggle.com/datasets/maharshipandya/-spotify-tracks-dataset

This dataset contains metadata and audio features of various tracks from Spotify, including numerical attributes that describe the nature and feel of each song. It is well-suited for building a content-based recommendation system using machine learning techniques.

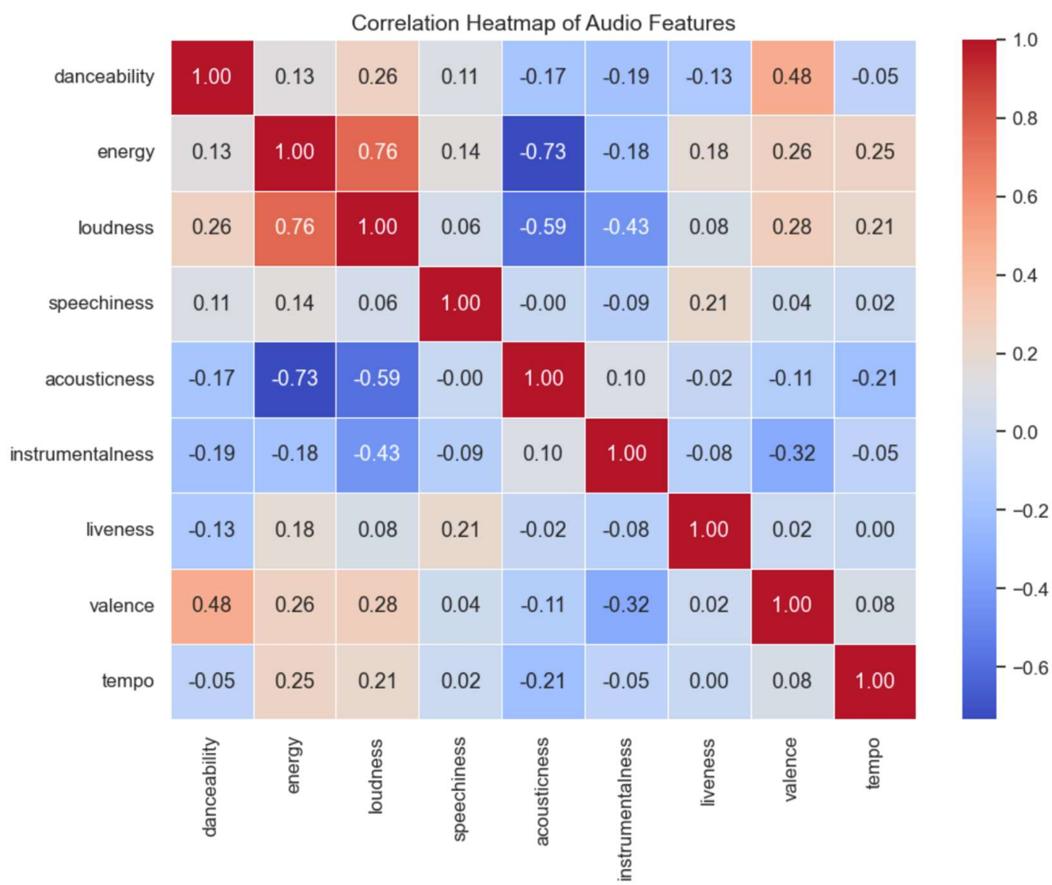
Key attributes in the dataset include:

- Danceability: How suitable a track is for dancing
- Energy: Intensity and activity level of the song
- Loudness: Average decibel level
- Speechiness: Presence of spoken words (e.g., rap, podcasts)
- Acousticness: Likelihood of a song being acoustic
- Instrumentalness: Probability of a track having no vocals
- Liveness: Presence of a live audience
- Valence: Positivity conveyed by the track
- Tempo: Beats per minute

Preprocessing Steps:

- Removed non-numeric columns like song names, track IDs, and artist names to focus on audio-based features.

- Handled missing or null values to ensure clean data input for model training.
- Applied Min-Max Scaling to normalize all feature values to the [0, 1] range.
- Created visualizations like correlation heatmaps, pair plots, and histograms to understand feature relationships and distribution.



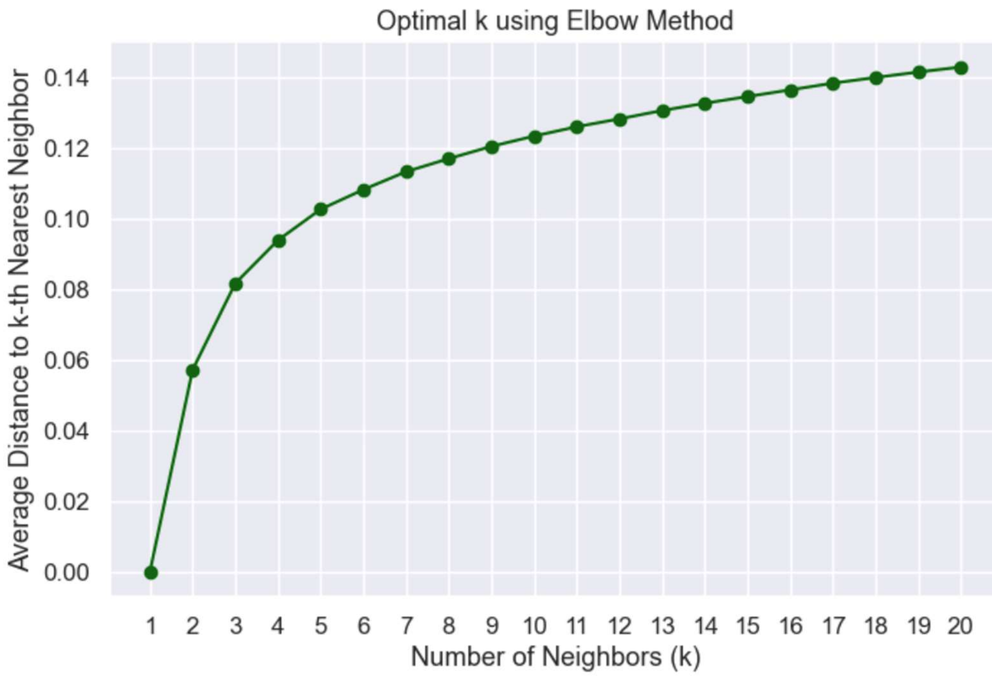
3. Model Chosen

The core model used in this project is the K-Nearest Neighbors (KNN) algorithm. KNN is a simple, yet effective unsupervised learning algorithm that finds the most similar songs based on feature vectors.

K-Nearest Neighbors with Cosine Similarity:

- Once clusters are formed, KNN is used within the same cluster to recommend songs similar to a user's selected track. Cosine similarity helps in measuring the angle between feature vectors, focusing on feature distribution rather than magnitude — making it ideal for music recommendations where scale can vary but the pattern matters more.

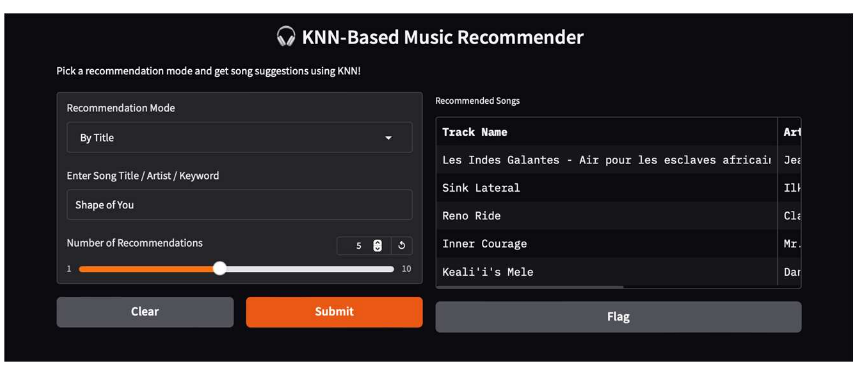
The optimal number of neighbors (K) is determined using the **Elbow Method**, analyzing the average distance to the Kth nearest neighbor across the dataset.



- Value of k = 4

Once a user inputs a song name:

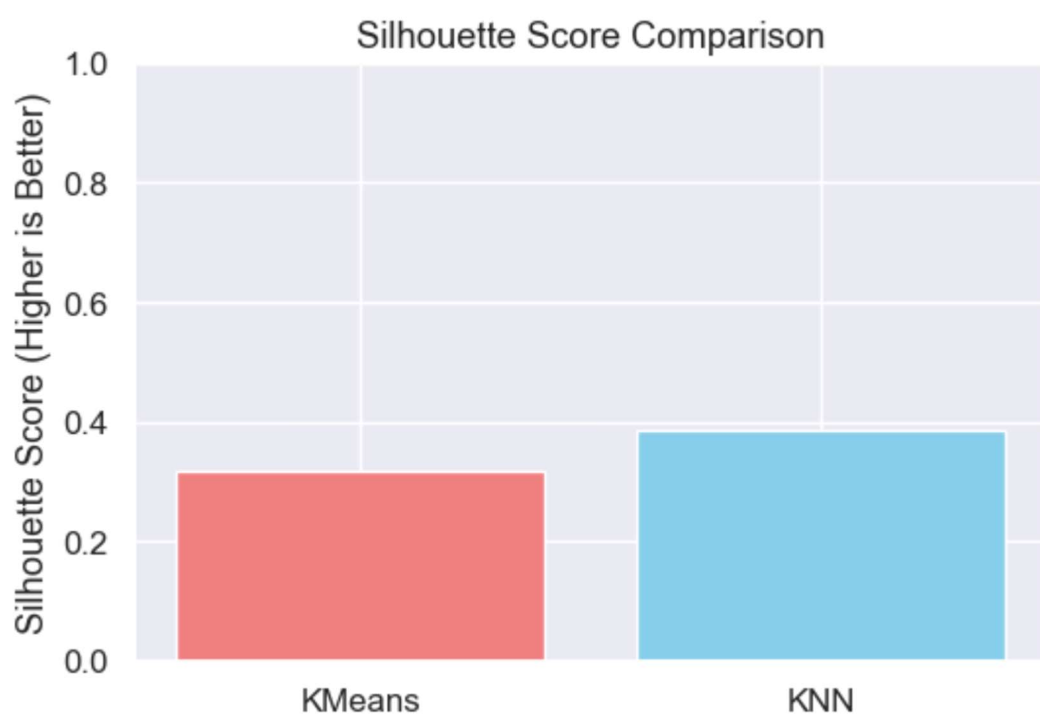
- The model locates that song in the dataset.
- It retrieves the K most similar songs from the feature space.
- These recommendations are returned as a ranked list, representing songs most similar in audio profile to the user's selection.



4. Performance Metrics

Since this is a content-based recommendation system using KNN, traditional classification or regression metrics like accuracy or F1-score are not applicable. Instead, performance is evaluated using metrics such as:
- User satisfaction based on how relevant the recommendations are.
- Visual inspection of feature similarity and correlation.
- Consistency and variety in the recommended song list.
- Silhouette Score (for Clustering)

- Measures how well each song fits into its assigned cluster compared to other clusters
- **Silhouette Score ranges:**
  - **+1**: well-clustered and distinct
  - **0**: overlapping or indistinct
  - **-1**: incorrect clustering



As seen, KNN outperforms KMeans in terms of silhouette score. This indicates that the KNN-based recommendation system groups songs more cohesively, resulting in better and more relevant music recommendations.

5. Challenges & Learnings

Challenges:

- Managing a large and complex dataset with diverse audio features.
- Preprocessing and normalization of features for fair distance computation.
- Balancing the variety and accuracy of recommendations.
- Choosing the right number of neighbors (k) to ensure meaningful results.

Learnings:

- Improved understanding of unsupervised learning and distance-based algorithms.
- Hands-on experience with real-world datasets and preprocessing techniques.
- Knowledge of audio features and their role in music recommendation.
- Enhanced skills in Python libraries like pandas, scikit-learn, seaborn, and matplotlib.