CSE 343: Machine Learning
Music Recommendation System Using Machine Learning

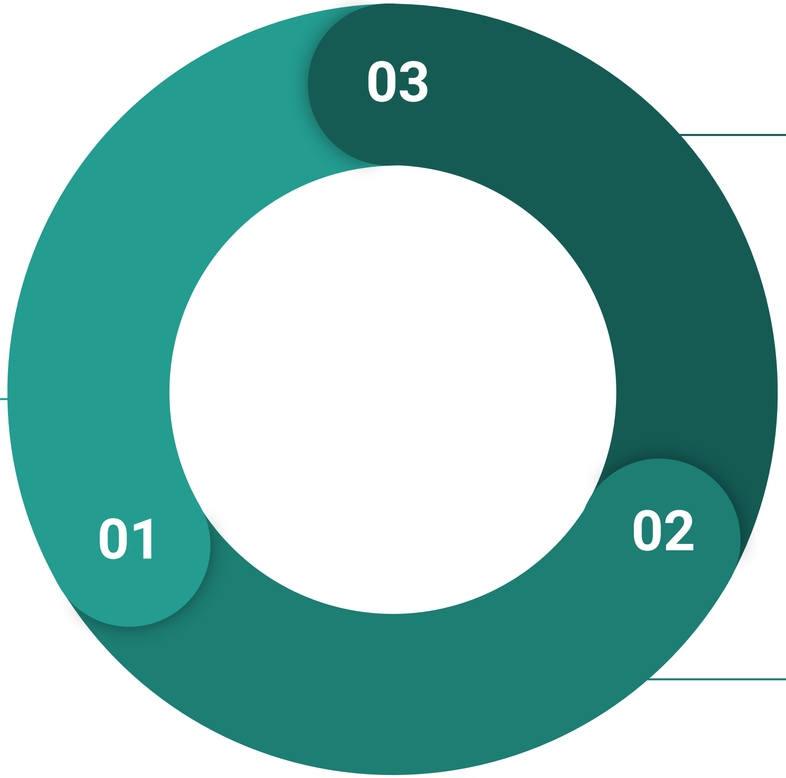
Aniket Kanojia
Ashutosh Gera
Piyush Kumar
Tushar Chandra

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY
DELHI

Motivation

Limited Diversity

We want to create a music recommendation system that stems from our love for music and the desire to continually explore new artists and tones.



Learning

Building a music recommendation system helped us build a thorough understanding of various machine learning algorithms and feature engineering techniques.

Bias

Existing systems are biased towards popular content and mostly keep revolving with same set of songs.

Literature Review

- D. Lin and S. Jayarathna, "Automated Playlist Generation from Personal Music Libraries," 2018 IEEE International Conference on Information Reuse and Integration (IRI), Salt Lake City, UT, USA, 2018, pp. 217-224, doi: 10.1109/IRI.2018.00039.
    - Given an arbitrary collection of music recordings, they aim to automatically sort songs with similar musical qualities into playlists. They utilize various clustering algorithms such as **K-Means, Affinity Propagation and DBSCAN.**
- A Music Recommendation System with a Dynamic K-means Clustering Algorithm
    - They suggest a method for personalized music recommendation services. They propose a **dynamic K-means clustering algorithm** which clusters the pieces in the music list dynamically adapting the number of clusters. They recommend pieces of music based on the clusters. The Existing recommendation systems analyze a user's preference by simply averaging the properties of music in the user's list. So those cannot recommend correctly if a user prefers several genres of music. By this proposed dynamic K-means clustering algorithm, they aimed to recommend pieces of music which are close to user's preference even though he likes several genres.

Literature Review

- Varsha Verma, Ninad Marathe, Parth Sanghavi, Dr. Prashant Nitnaware , "Music Recommendation System Using Machine Learning ", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 7, Issue 6, pp.80-88, November-December-2021
    - They use a sample data set of songs to find correlations between users and songs so that a new song will be recommended to them based on their previous history. They have used **Cosine similarity along with CountVectorizer.**
- M. G. Galety, R. Thiagarajan, R. Sangeetha, L. K. B. Vignesh, S. Arun and R. Krishnamoorthy, "Personalized Music Recommendation model based on Machine Learning," 2022 8th International Conference on Smart Structures and Systems (ICSSS), Chennai, India, 2022, pp. 1-6, doi: 10.1109/ICSSS54381.2022.9782288
    - Used a library of songs to uncover connections across individuals and music so that a hit album might be offered to individuals derived from history. They primarily focused on **Count Vectorizer (CV), and Cosine similarity (CS)** machine learning techniques delivering a complete end-to-end interface which, when a piece of given music, processes it and provide with the suggested tracks.

Dataset Description

- We are using a [Spotify provided dataset]() which contains audio features of over 1, 204, 012 (1.2 million+) songs obtained with the Spotify API (spotipy).

- The original dataset comprised of 24 columns (features). Live and Love being the most frequent words in the titles of song names.

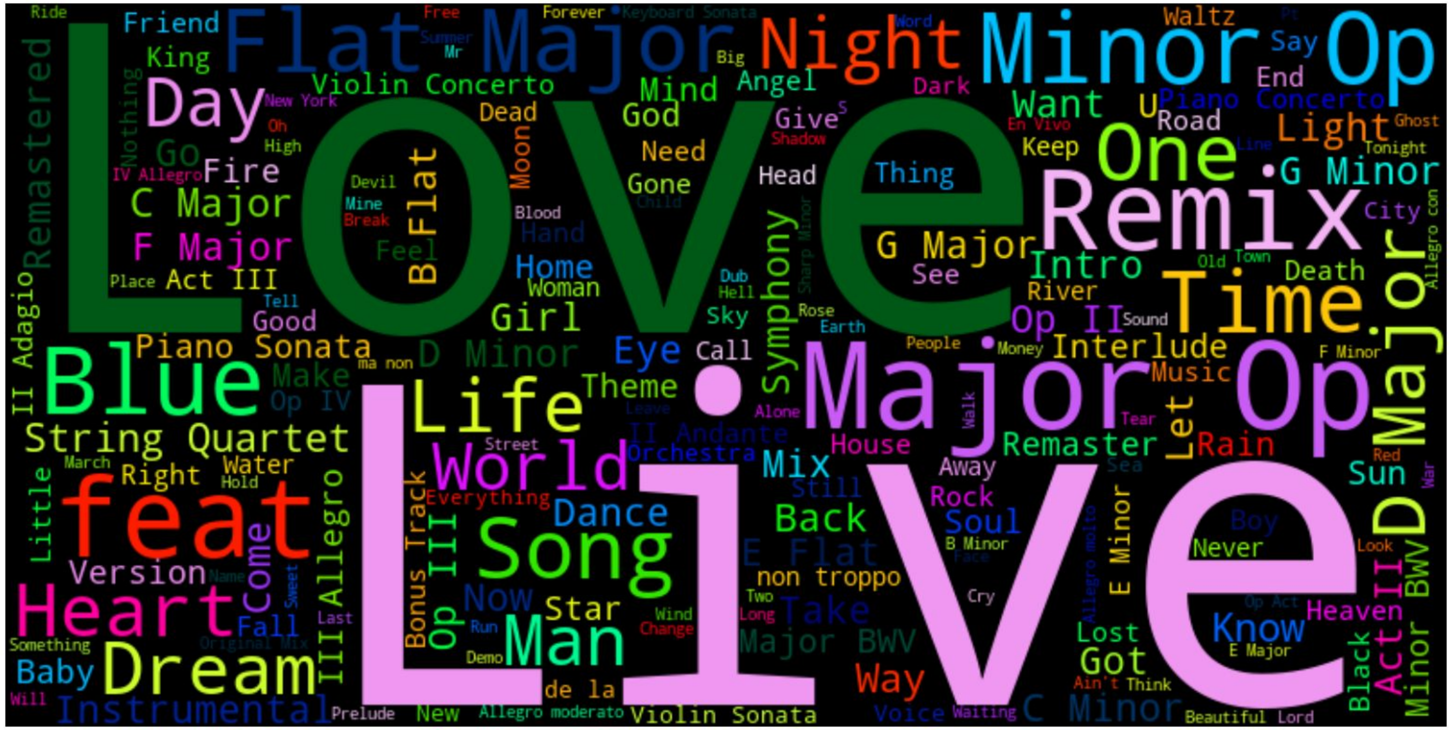


Figure: Word cloud for song titles

Dataset Description

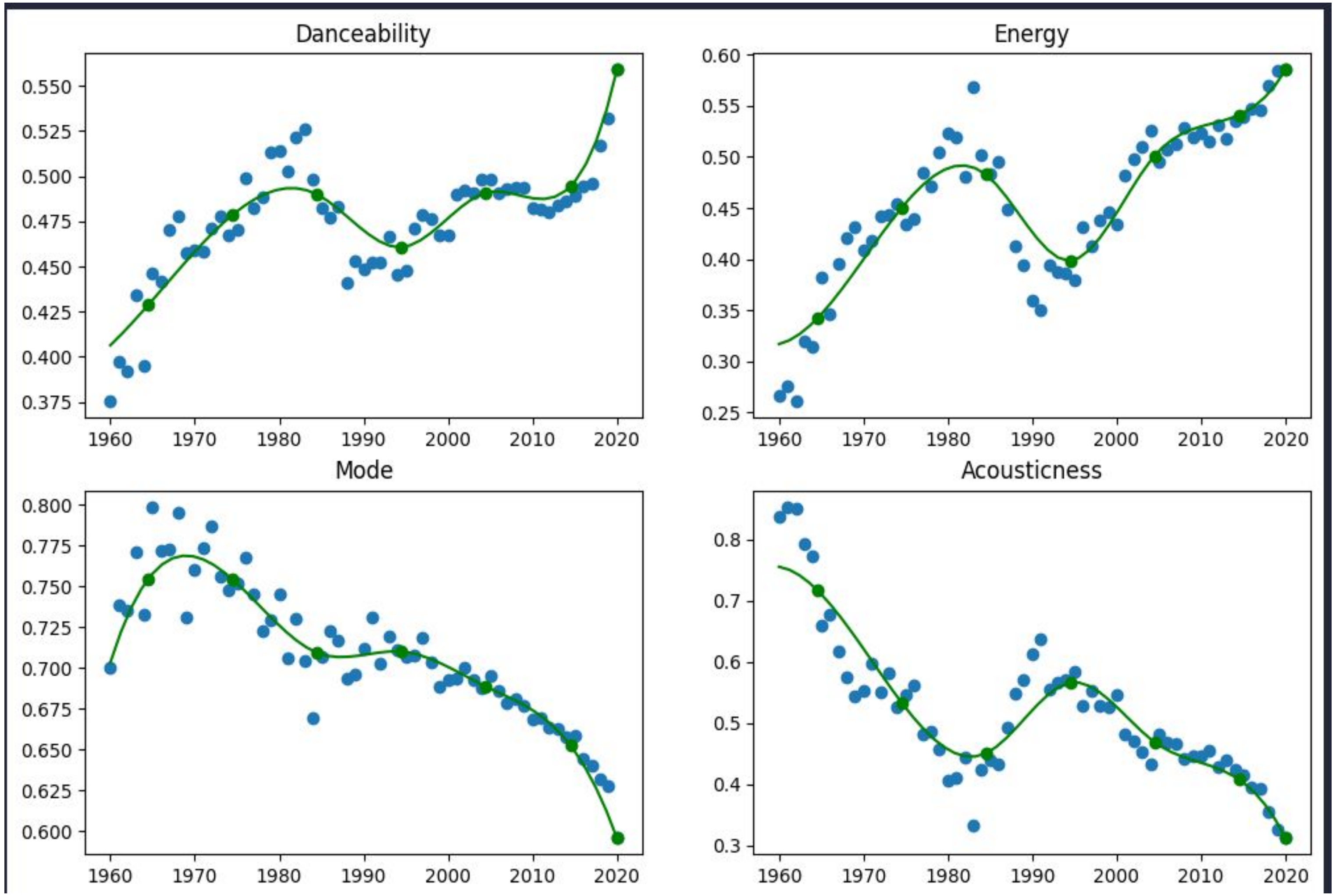
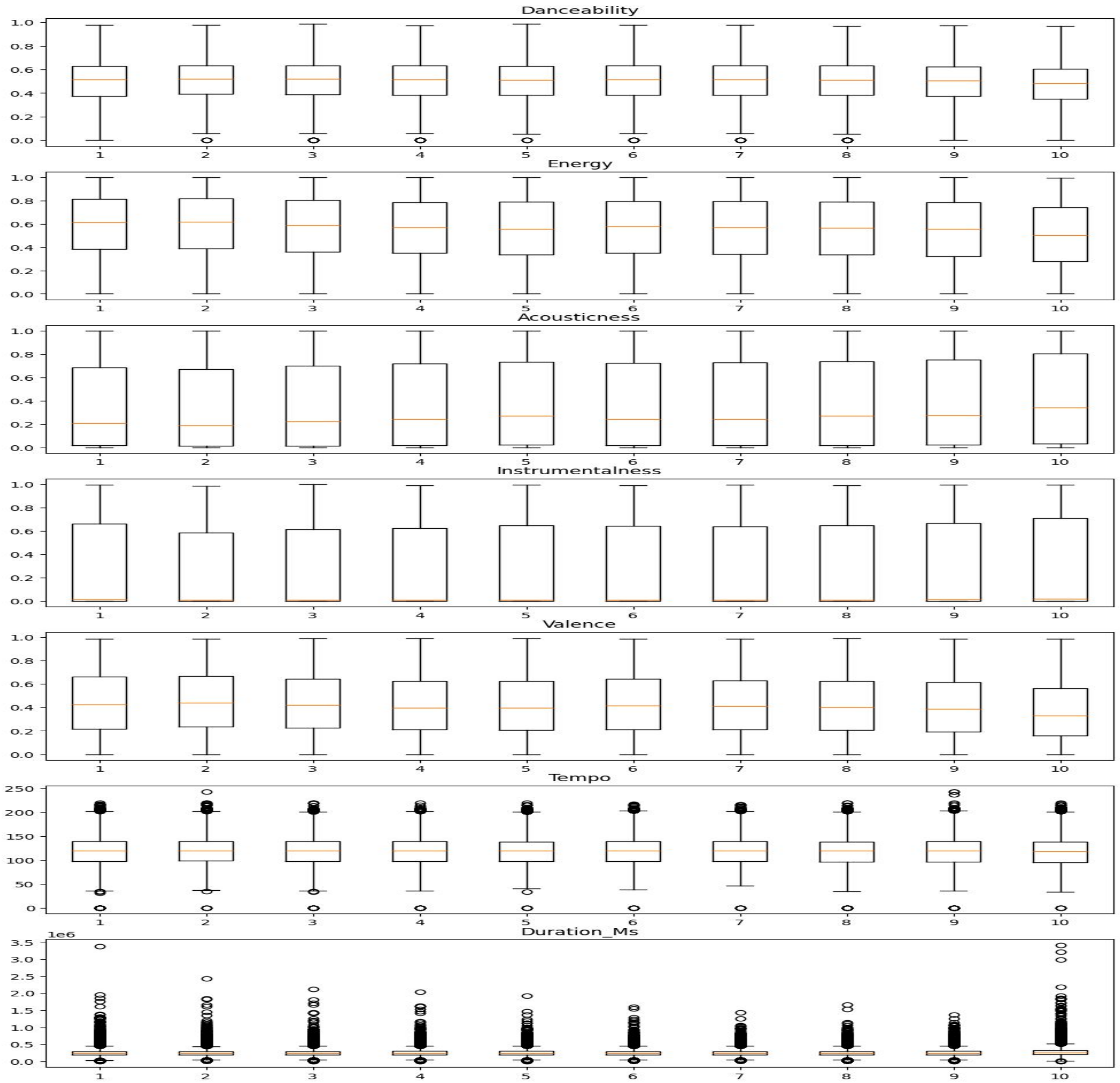


Figure: Variation of Audio features w/ time

Plotting the trends of songs for these attributes from 1960 on. We include mode as an attribute to check for trends in how many songs are in a major key as opposed to a minor key. We can see that overall, **songs increased in danceability and energy, while decreasing in acousticness.** Also, use of minor keys became more common. There was also a reversal in the overall trend during the 80s.

Dataset Description



Focusing on ten-track albums, we can use **boxplots to explore the attributes of songs at different track numbers.** (Some attributes are not plotted because their boxplots are not very informative.)

These plots suggest that songs tend to **decrease in energy from the beginning to the end of an album, while increasing in acousticness.** Relatively long songs may be more common at the end of an album as well.

Figure: Variation of audio features across songs of an album

Pre-Processing

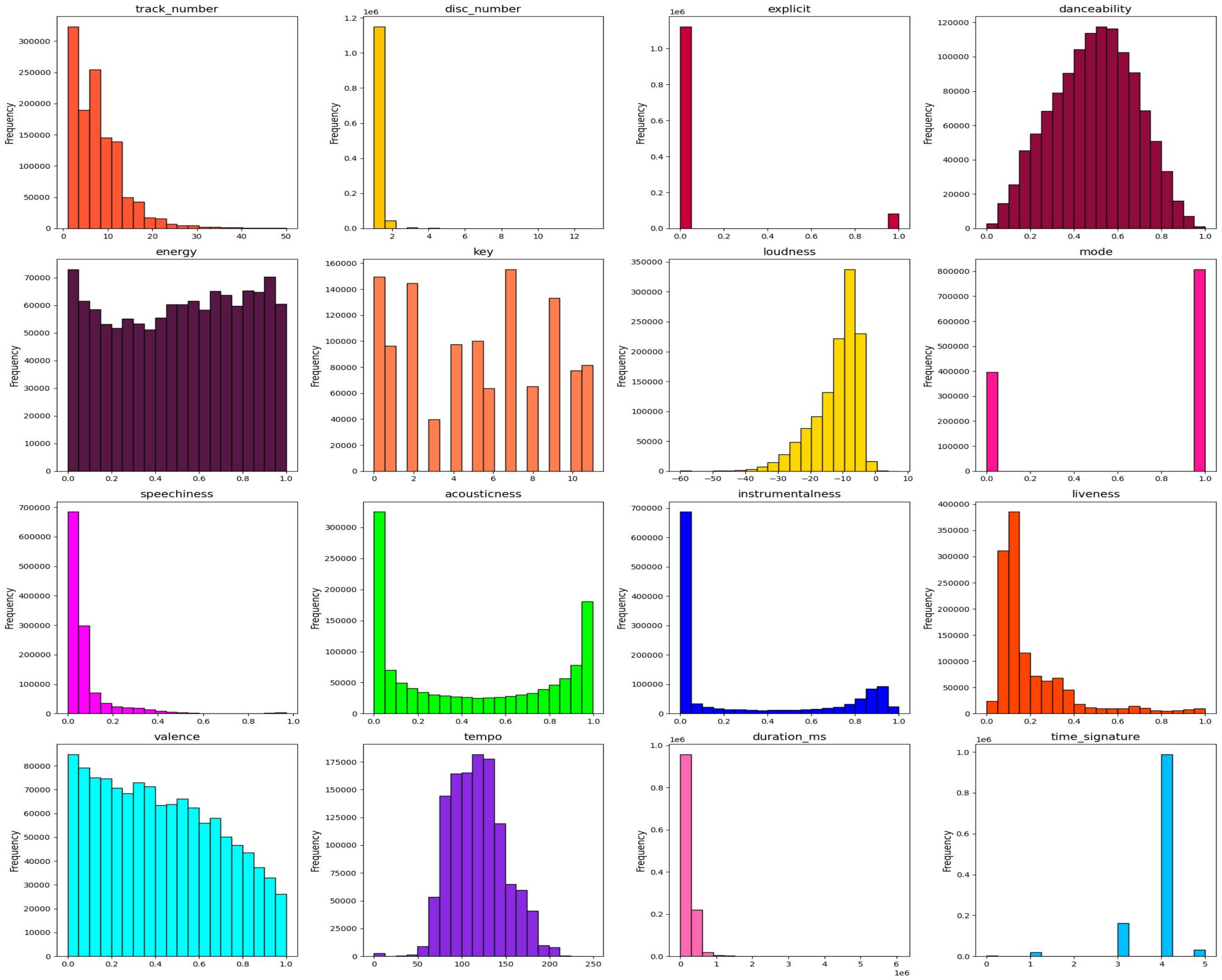


Figure 2: HISTOGRAMS OF ALL FEATURES

- During preprocessing, we excluded certain numeric features due to their limited variability across the dataset. Visualizations like Histograms helped in decision.

Pre-Processing

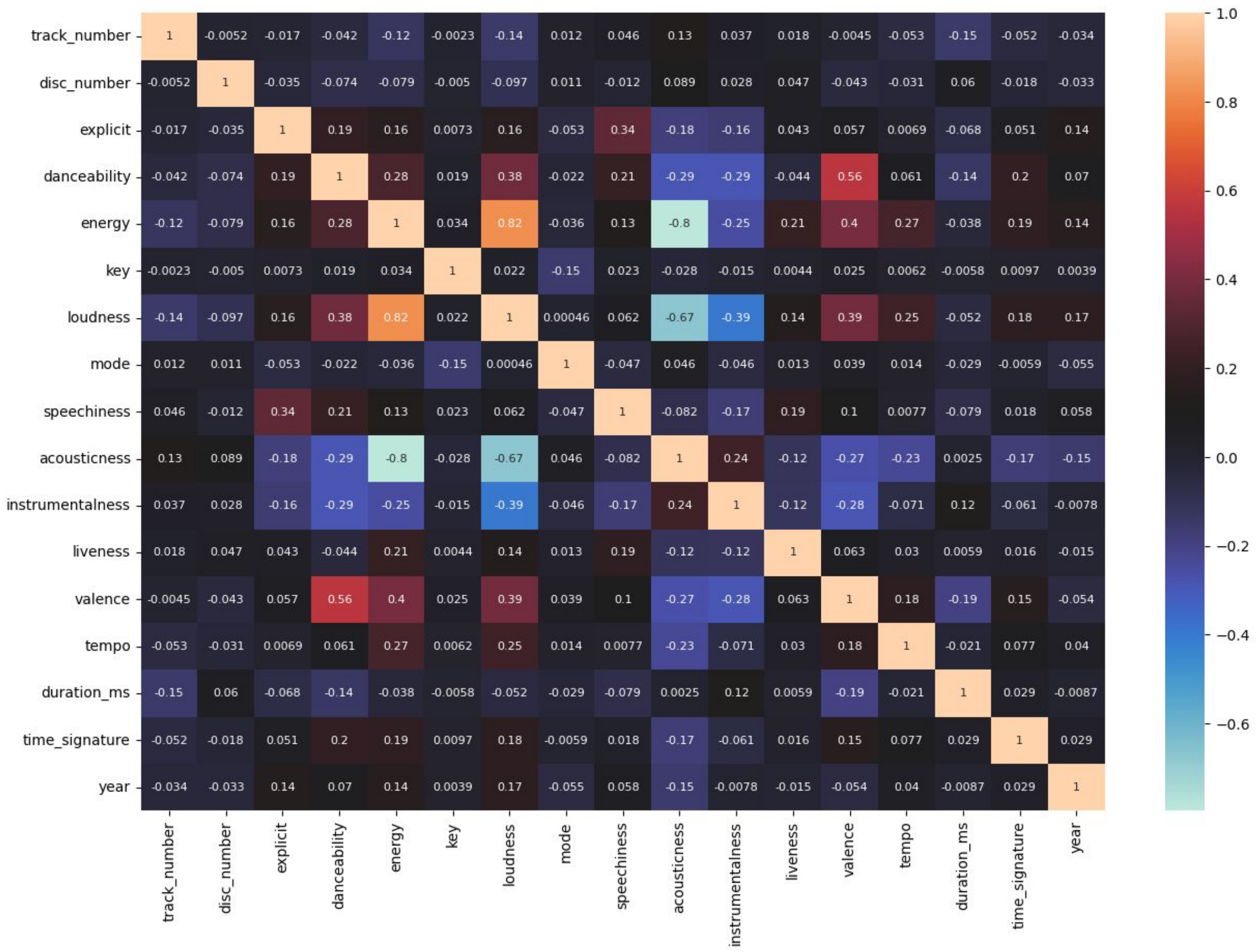
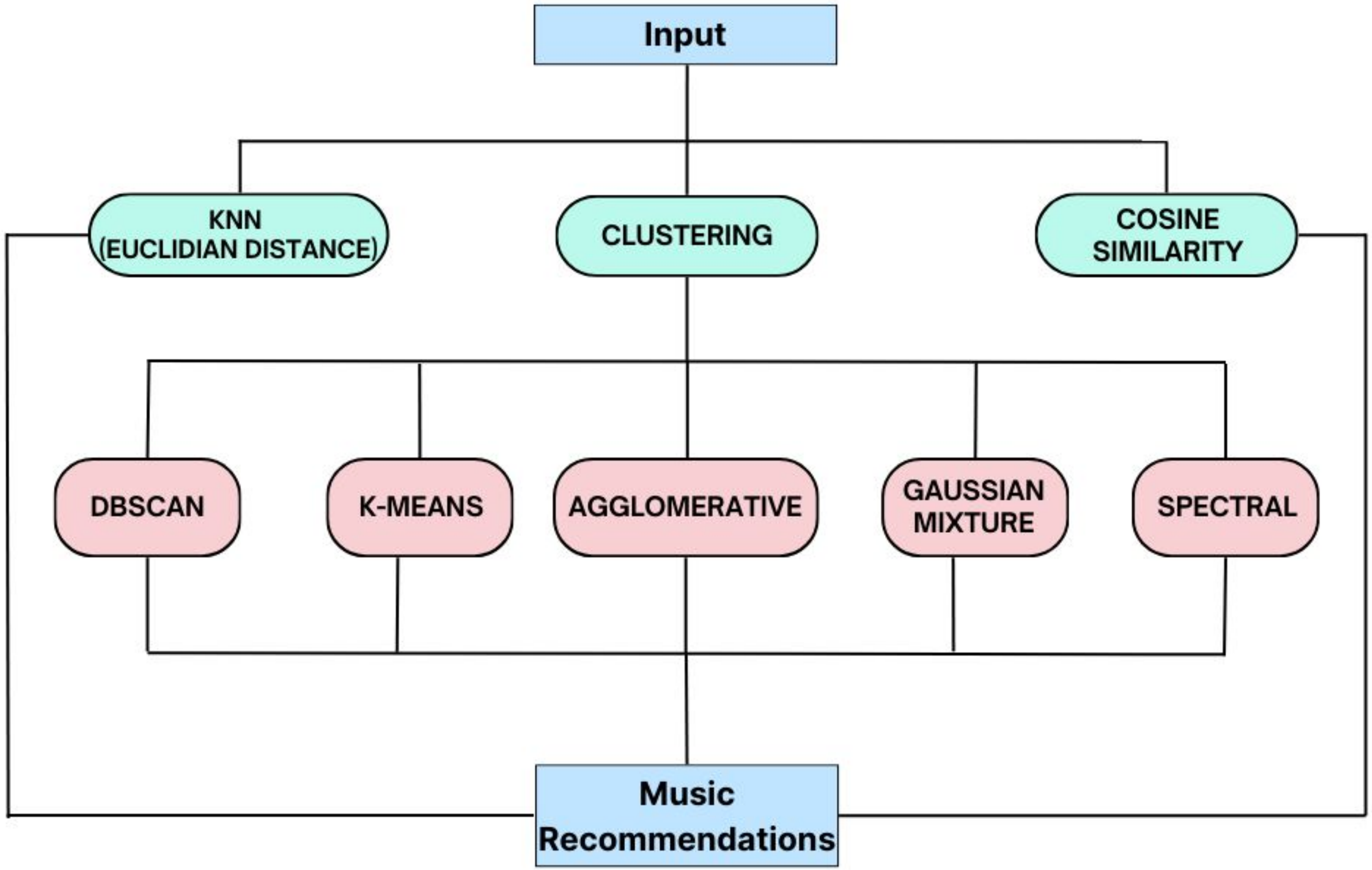


Figure 3: CORRELATION HEATMAPS OF ALL FEATURES

- Additionally, correlation heatmaps were plotted which helped to eliminate highly correlated features.
- Finally, we scaled numeric features and excluded categorical features.
- Initially, our dataset had 24 features. However, after analyzing and preprocessing it using visualization tools such as histograms and correlation matrices, we narrowed it down to 16 features.

Methodology



Methodology

Cluster Similarity

Songs are represented in an N-dimensional space based on their features. By leveraging clustering algorithms, songs with similar attributes are grouped together.

Cosine Similarity

Each song is represented by a series of features. The similarity between the given song and other songs in the database is determined using the cosine similarity metric.

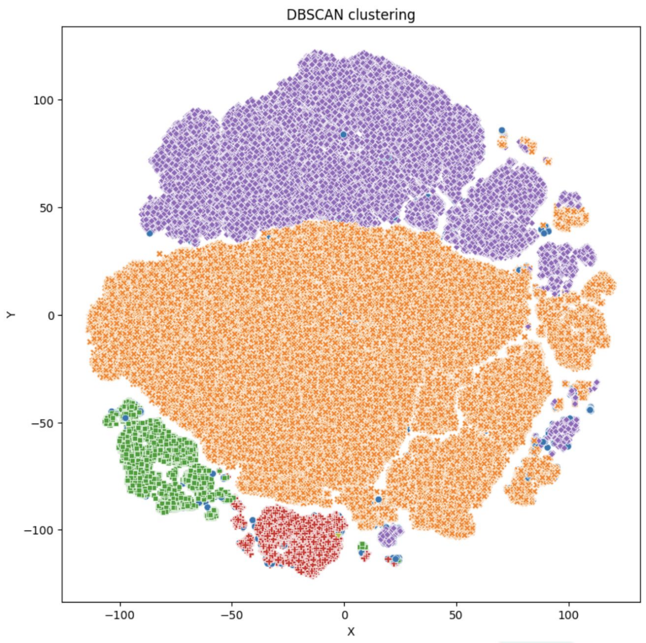
Formula Used: Similarity = (A.B) / (||A||.||B||)

Methodology

Now let's focus models which doesn't require 'k' numbers of clusters.

1. **Density-Based Spatial Clustering of Applications with Noise (DBSCAN)**

- DBSCAN works on the two major inputs i.e epsilon and minimum points.
- here, Epsilon(ε) sets the neighbor-search radius for clustering whereas minimum samples define the points required to be present around the point.
- If the number of points within this radius exceeds a predetermined threshold (minimum samples), the point is labeled as a core point.
- Any point not classified as a core point or part of a cluster, yet within the vicinity of a cluster, is marked as a border point.
- Points that do not fall into any cluster and do not have enough neighboring points within the defined radius are labeled as noise/outliers.
- DBSCAN is adept at handling irregularly shaped clusters and doesn't require the number of clusters to be predefined
- Total of 4 clusters are formed on the basis of density at specific values of epsilon and minimum samples. (Value of epsilon used here is 0.5 and the minimum samples is 15).

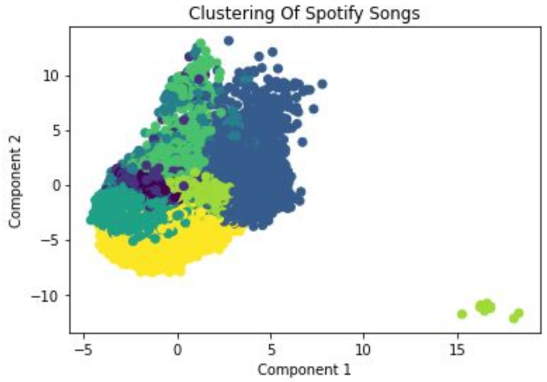


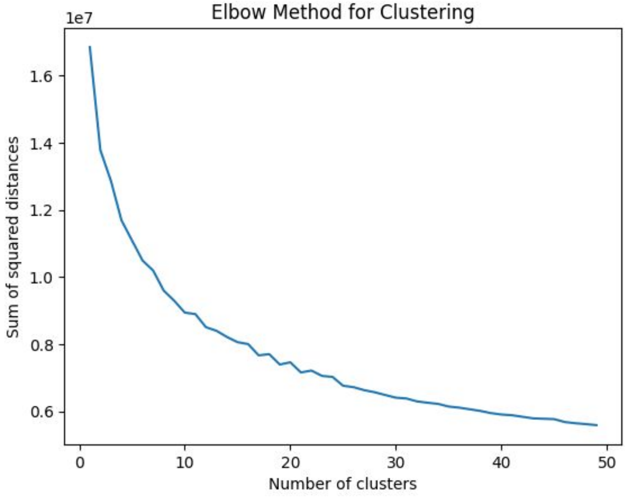
Methodology

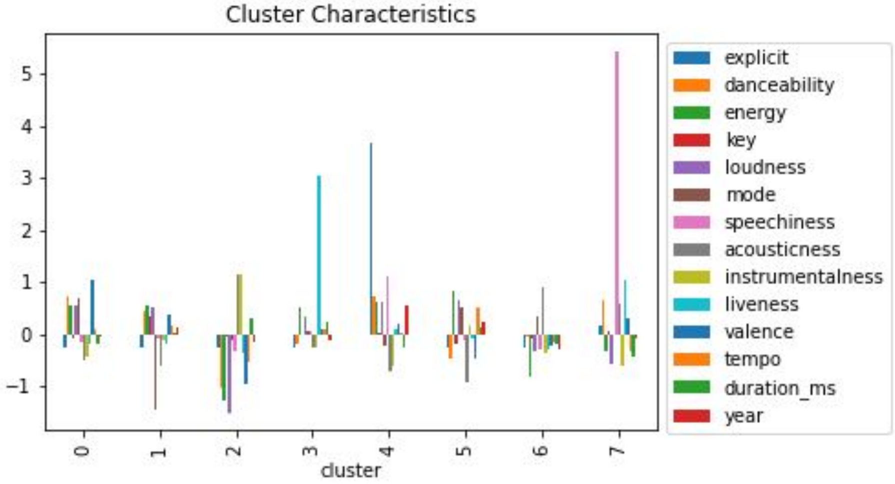
Models with k-number of clusters:

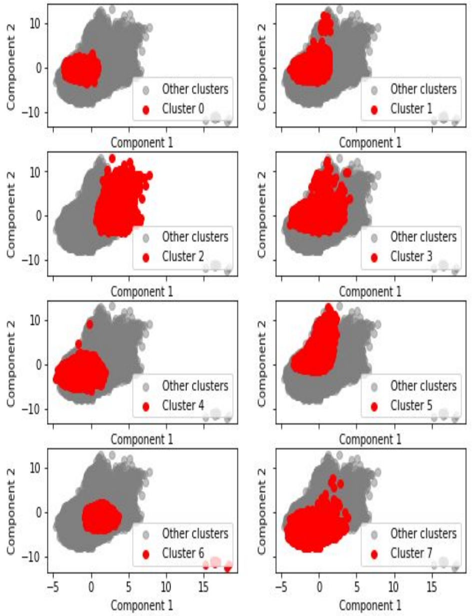
2. K-Means Clustering

- A centroid-based clustering method.
- Divides data into a predefined number of clusters (k).
- number of clusters = 8
- Silhouette score: 0.125









Methodology

3. Agglomerative Clustering

- A hierarchical clustering method.
- Starts with each data point as its cluster and iteratively merges the closest clusters.
- Forms a hierarchy of clusters.
- The number of clusters is determined by cutting the dendrogram at a certain level.

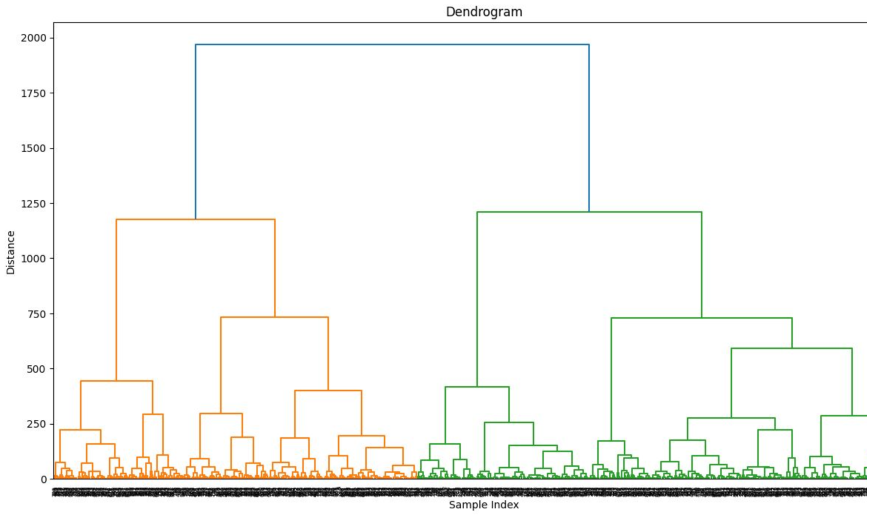
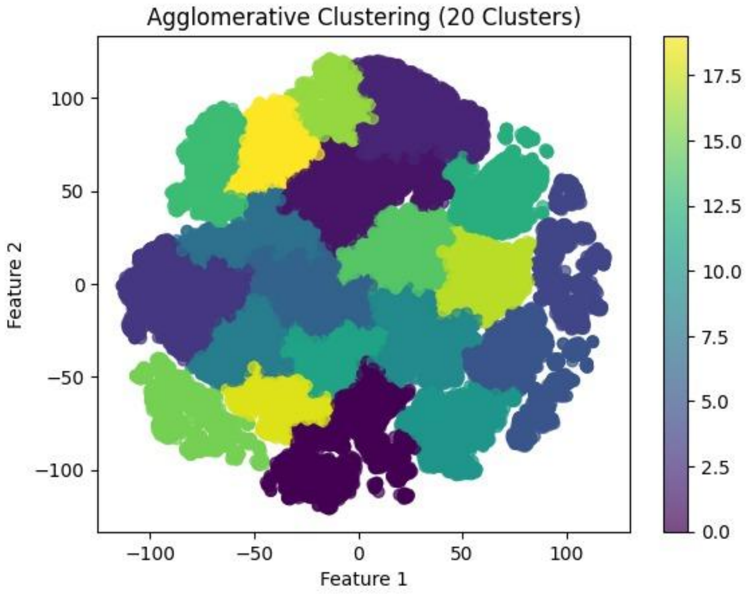


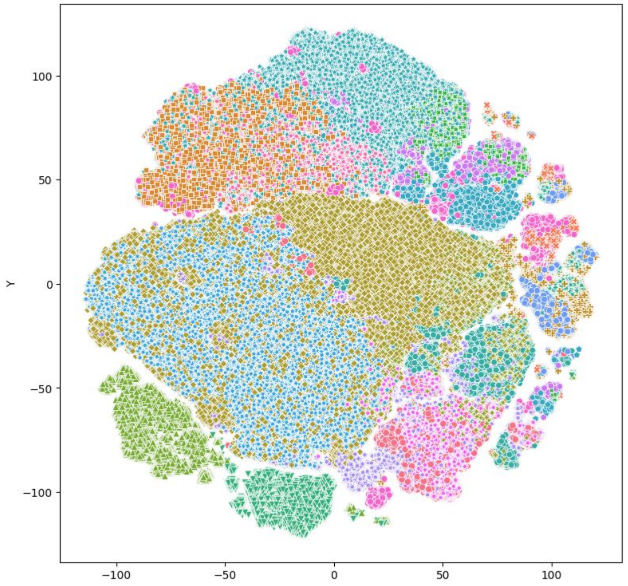
Fig: Dendrogram of data points (n=1000)



Methodology

4. Gaussian Mixture:

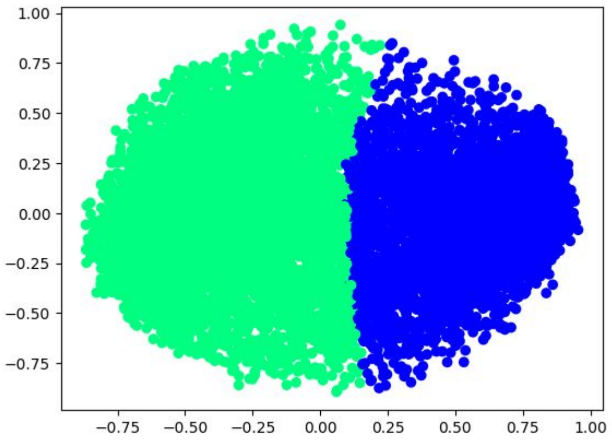
- A probabilistic model for clustering.
- Assumes that data points are generated from a mixture of Gaussian distributions.
- It estimates the parameters (means, variances, and mixing coefficients) of these Gaussians.
- Useful for modeling clusters that may have different shapes and orientations.

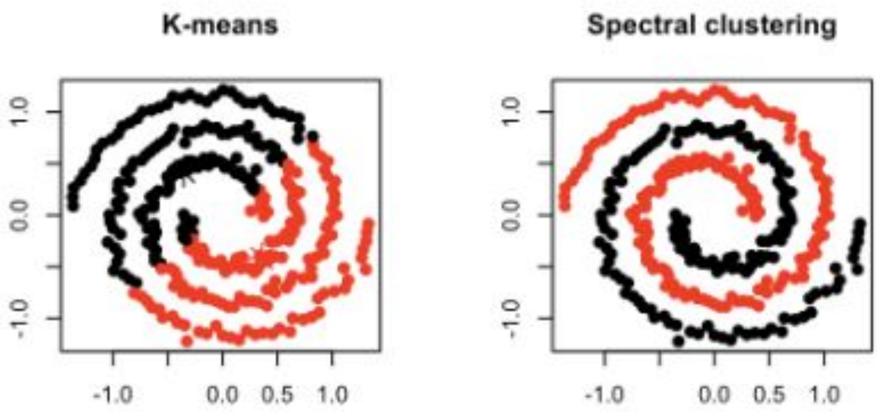


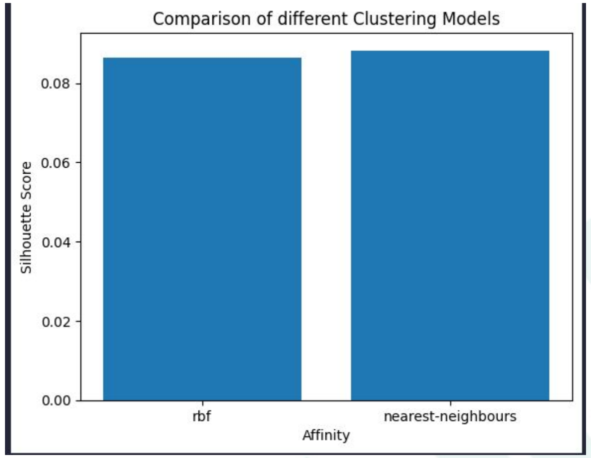
Methodology

5. Spectral Clustering (SC) :

- In SC, data points are treated as nodes of a graph. Thus, spectral clustering is a graph partitioning problem. The nodes are then mapped to a low-dimensional space that can be easily segregated to form clusters. No assumption is made about the shape/form of the clusters. SC is very computationally extensive so we performed it on a small dataset comprising of 10,000 random songs and got 2 clusters using **nearest neighbours as affinity.**
- Affinity metric determines how close, or similar, two points our in our space.







Results

NAME = Testify

ALBUM = The Battle Of Los Angeles

ARTISTS = Rage Against The Machine

EXPLICIT = FALSE

DANCEABILITY=0.47

LOUDNESS=-5.399

SPEECHINESS=0.0727

ACOUSTICNESS=0.0261

LIVENESS=0.356

VALENCE=0.503

TEMPO=117.906

DURATION_MS=210133

TIME SIGNATURE=4.0

YEAR=1999

INPUT

RECOMMENDED SONGS/OUTPUT (By Cosine Similarity)

Testify

Who I Am

Trisha Please Come Come (Live)

What Keeps You Up At Night

Your Love - Radio Edit

If You Could Hold Your Woman

W.B

Indigo Aerial

t

Veillée spatiale

Results

K MEANS	GAUSSIAN MIXTURE MODEL	DBSCAN	AGGLOMERATIVE
Testify	El Camino de la Noche	If You Could Hold Your Woman	Calm like a Bomb
Veillée spatiale	Minimal Slee	Osaka	Born as Ghosts
The Ride	Tombei The Mist	Testify	Voice of the voiceless
Seekir	Indigo Aerial	This Side of Paradise	New Millenium homes
Low Tide	String Quartet No. 4, Op. 37: I. Allegro molto energico	1-0-0	War within a breath
What's Wrong With You	String Quartet No. 4, Op. 37: IV. Allegro	Back In Love	Settle for nothing
Águia não come mosca	Verklärte Nacht, Op. 4: Sehr langsam	And I Love Her So	Wake up
(It's Good) To Be Free - Live for Teenage Cancer Trust	6 Kleine Klavierstucke, Op. 19 (arr. H. Guittart): No. 1. Liecht, zart	The West Mabou Reel/Up Downey /Calu Finlay/All The Rage	Freedom
Punks In The Beerlight	String Trio, Op. 45: Part III	Young - KO:YU Remix	Getaway Car
Appropriate Dipstick	Verklärte Nacht, Op. 4: Sehr ruhig	Veillée spatiale	Osaka

Contributions

Contributions:

- **Aniket Kanojia** (Feature Extraction, Recommendation algorithm: Gaussian Mixture Model, Report and Presentation)
- **Ashutosh Gera** (Data PreProcessing, Feature Extraction, K means clustering, Spectral Clustering, Report and Presentation)
- **Tushar Chandra** (Agglomerative Clustering, Visualization, Presentation)
- **Piyush Kumar** (DBSCAN, Improving results, Presentation)

Thank you!