



DS 203 : Programming for Data Science

# *Group Project*

Clustering and Classification of Songs Using MFCC Features

10 November, 2024



# *Team Members*

Aansh Samyani

22B0424

(Anchor)

Danish Siddiqui

22B2104

Vaishnav S Vernekar

22B2107

Akshat Jain

22B2717



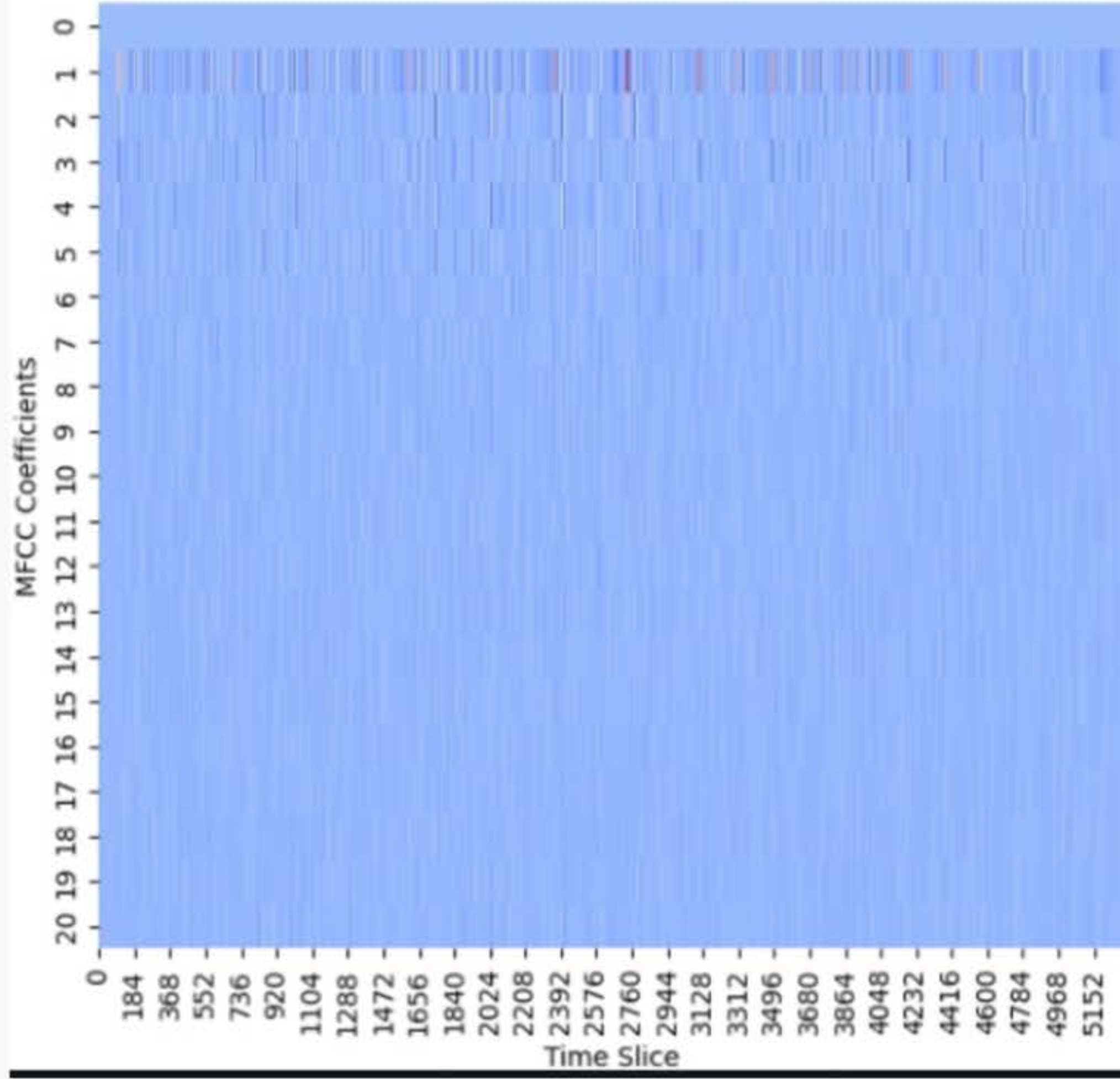


# *Audio to MFCC*

- **Song Selection:** Collected 12 songs from YouTube (2 songs per artist group).
- **Preprocessing:** Converted songs to MP3 and trimmed each to ~2 minutes.
- **Feature Extraction:**
  - **MFCC Calculation:** Generated Mel-Frequency Cepstral Coefficients (MFCC) for each song and saved each in CSV files using python code provided.
  - **Delta and Delta-2 MFCCs:** Calculated MFCC slope and acceleration to study and capture dynamic changes in audio features to differentiate amongst songs .
  - **Mel Spectrogram:** Visualized frequency distribution over time for each song.
- What is the purpose behind?
  - Created labeled data to cluster songs by similarity, enabling us to group new, unlabeled songs based on their closest matches in audio features.

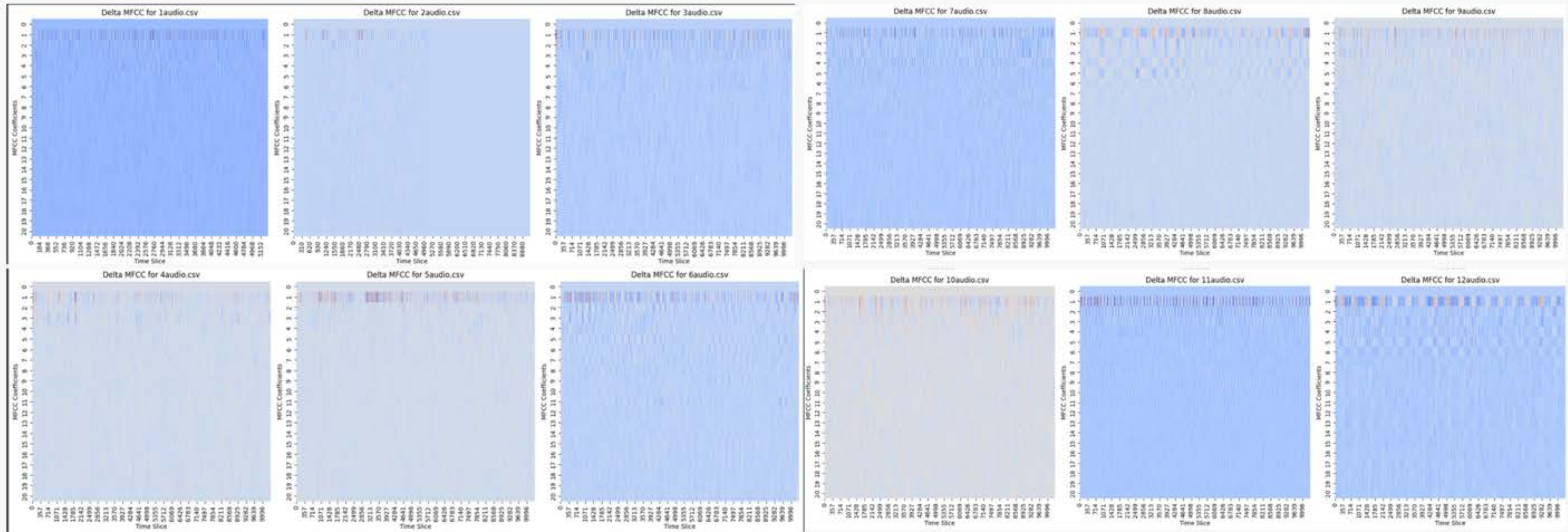


Delta MFCC for 1audio.csv

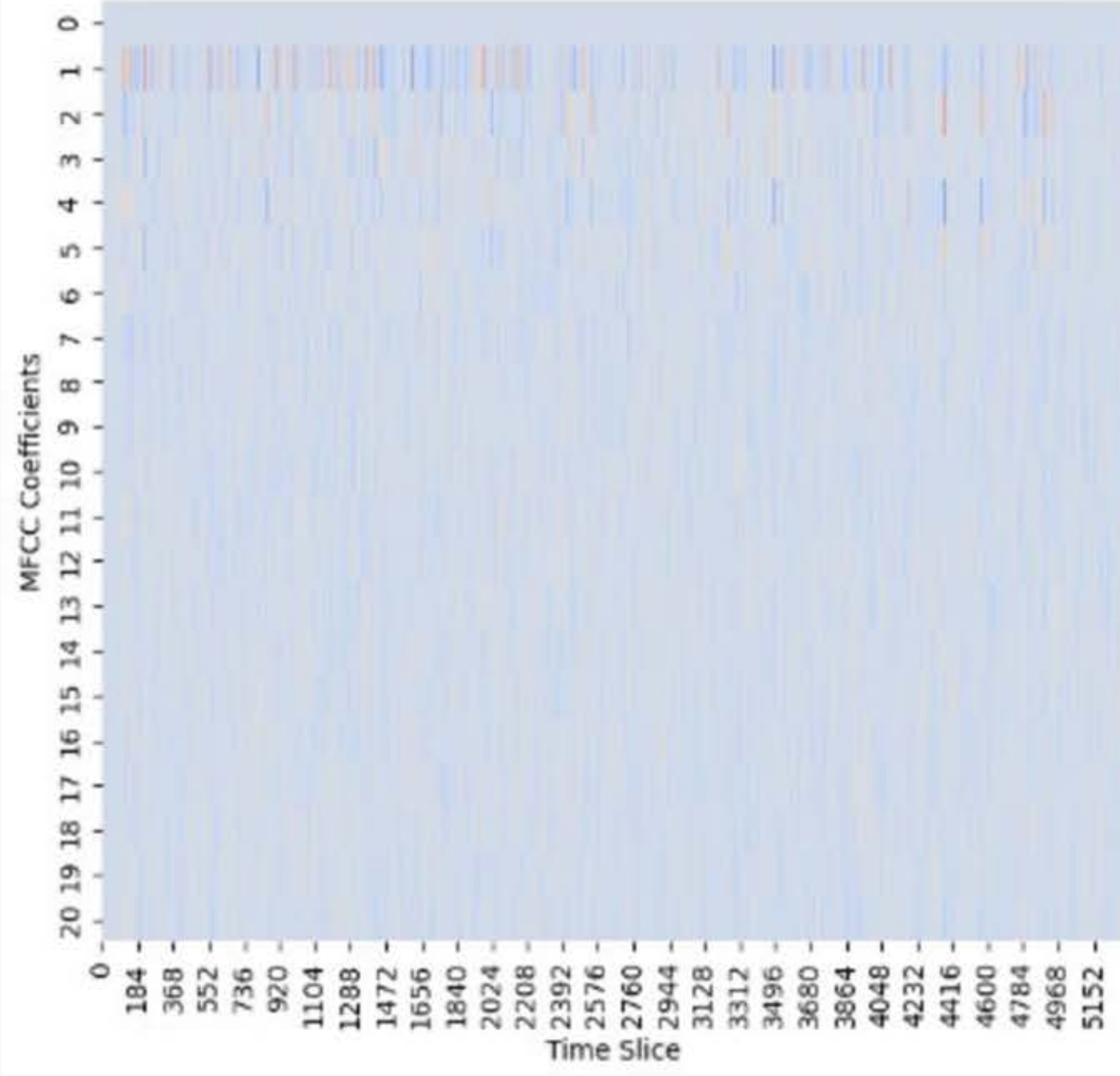




# Delta\_MFCC for 12 audio files

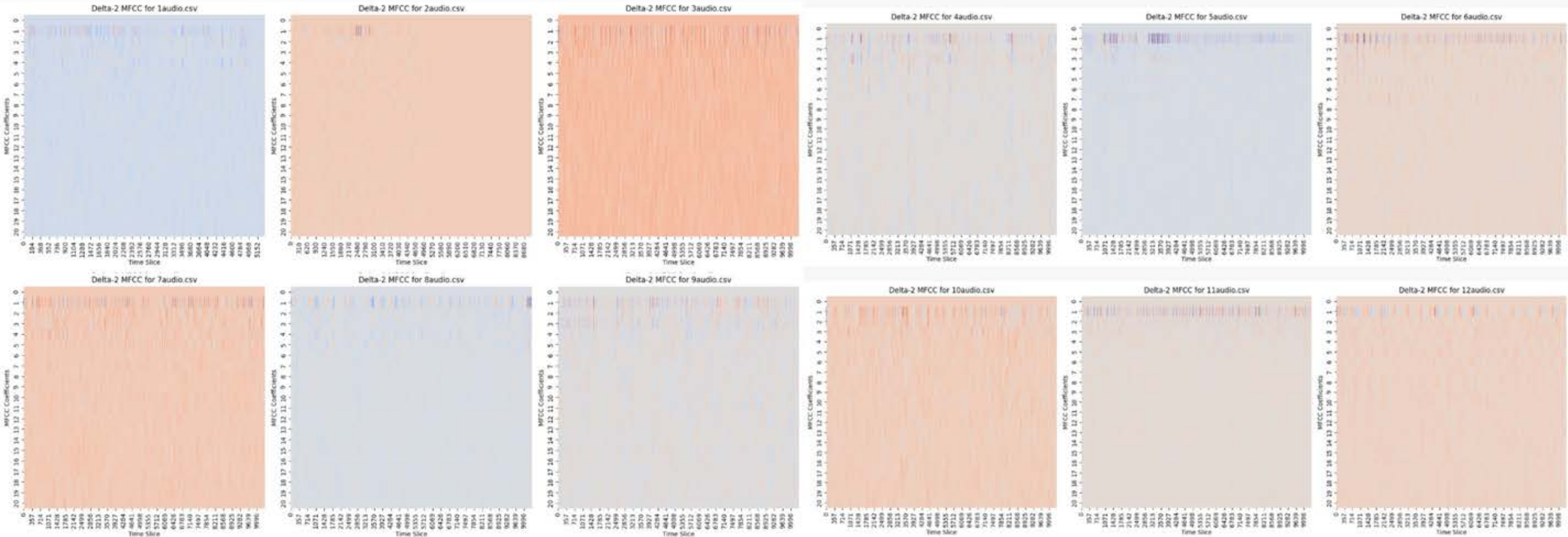


Delta-2 MFCC for 1audio.csv



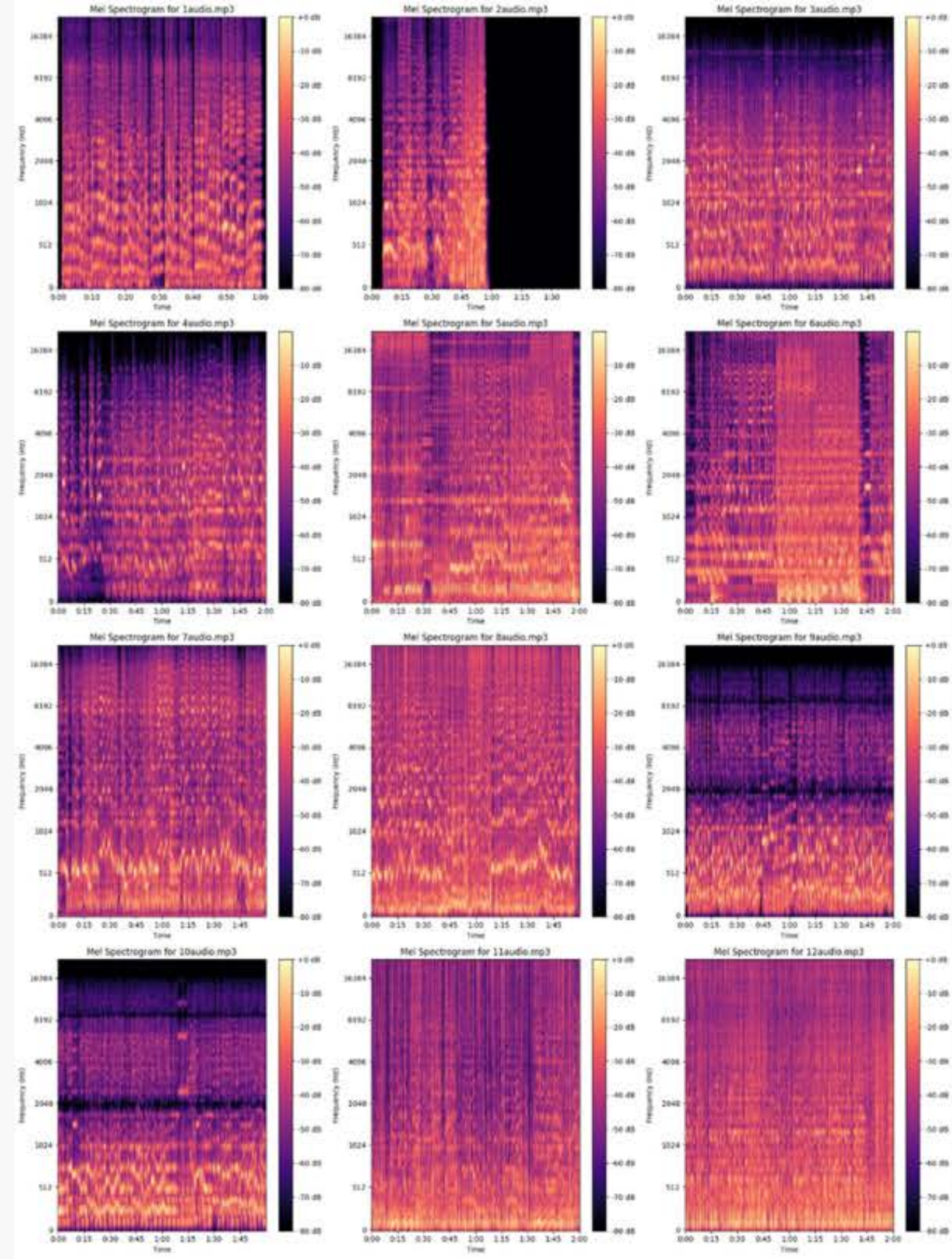


# Delta\_2\_MFCC for 12 audio files





# Mel spectrograms of the 12 audio files





# Executive Summary



**Project Objective** Cluster and classify a set of 116 songs using MFCC (Mel-frequency Cepstral Coefficients) to group according to genres, singers, and specific song categories.

**Dataset** The dataset contains MFCC features extracted from 116 songs, by Asha Bhosale, Kishore Kumar, Michael Jackson, Marathi Bhaav Geet, Marathi Laavni, and the rendition Indian National Anthem.

## *Approach 1: Feature-Rich Windowed Clustering Approach*

We segmented each dataset into 20x100 windows and extracted 9 features per window, creating a 20x900 matrix per song to retain key information. Analyzing 116 songs with KMeans, PCA, Agglomerative Clustering, and DBSCAN, followed by t-SNE visualization, revealed distinct patterns and similarities in the data.

## *Approach 2: Enhanced MFCC Feature Fusion Approach*


Added 8 features to enhance MFCC structure, creating a final 20x1700 feature set. Analyzing 116 songs with KMeans, PCA, Agglomerative Clustering, and DBSCAN, and visualizing with t-SNE, revealed detailed patterns among songs.







## *Approach 1: Windowing of CSV columns*

- For each dataset, we divided the columns using a windowing approach (where window size = number of columns//100), resulting in a final data shape of 20x100. Within each window, we extracted a set of 9 features—such as mean, variance, minimum, maximum, and others - giving a transformed size of 20x900 per song after concatenation.
  - This approach preserved essential information across windows. We then conducted an analysis on 116 songs using KMeans and Agglomerative Clustering. Finally, t-SNE was used to visualize the clustering results, providing a clearer view of data patterns and similarities.
- 



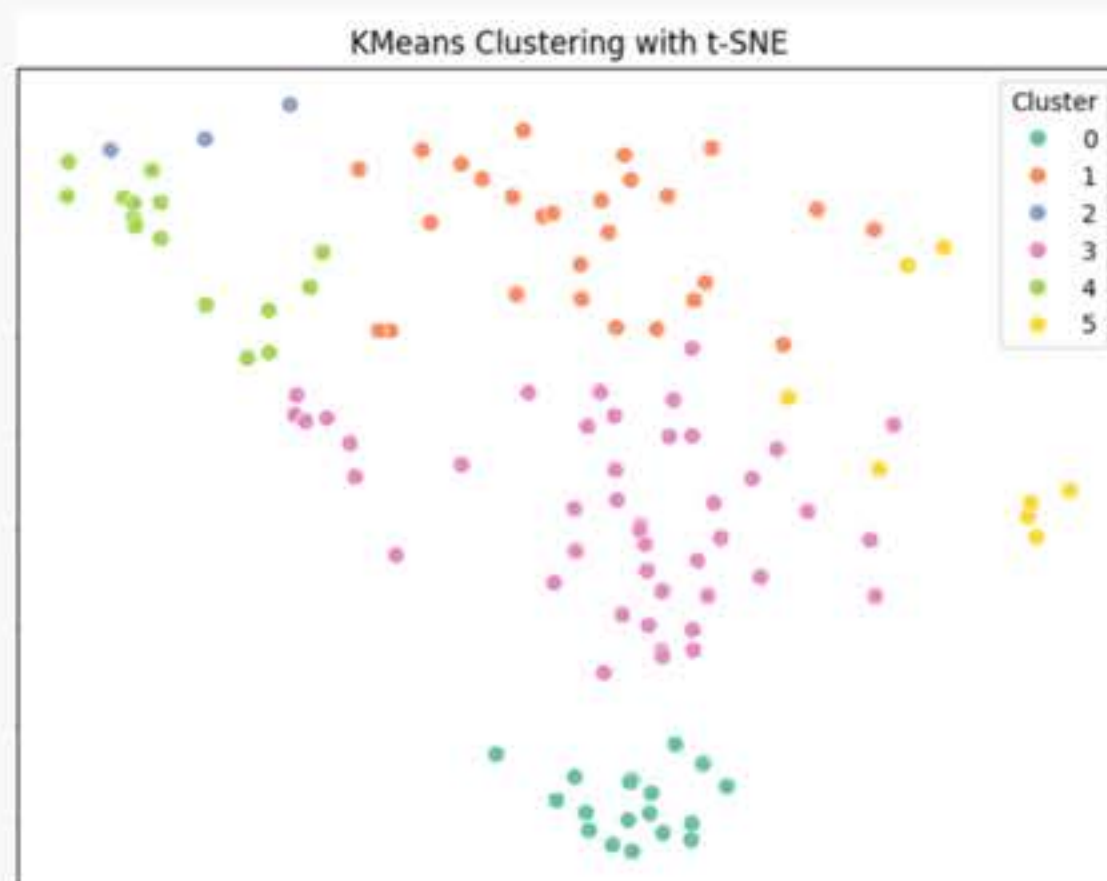
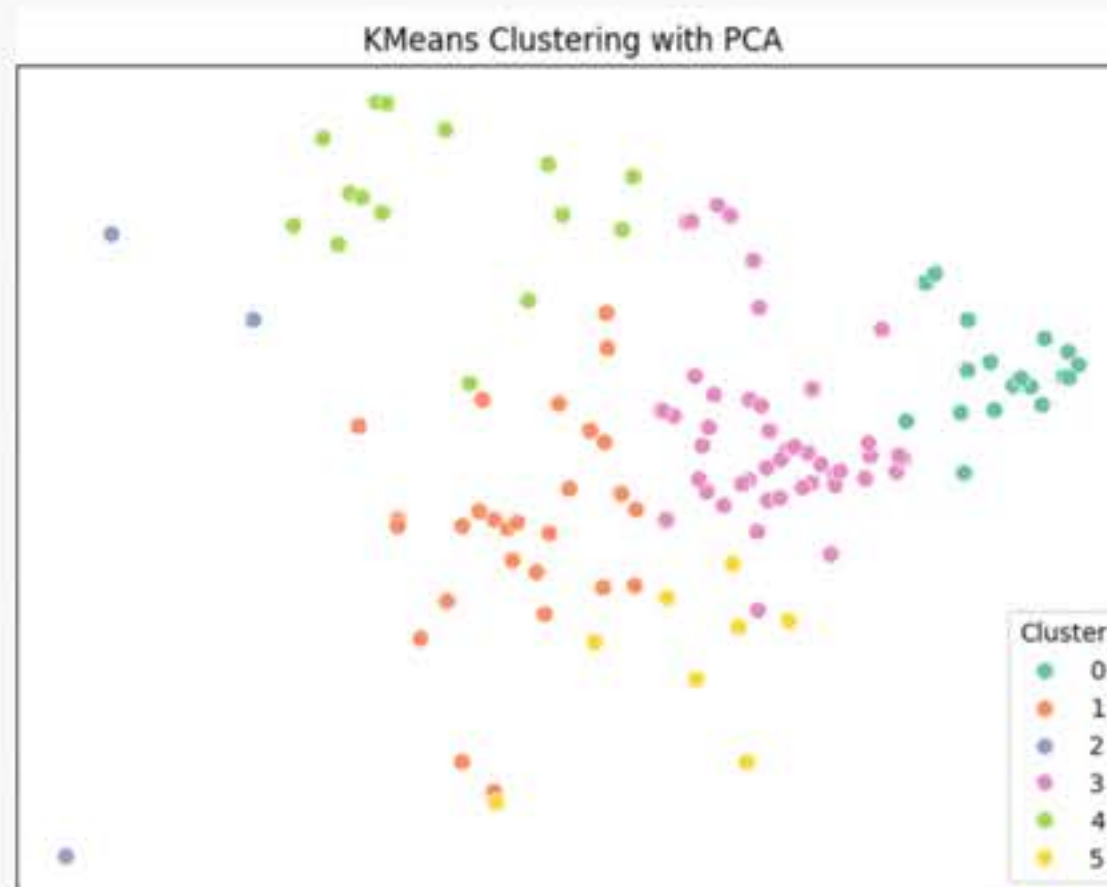


## *Feature Engineering - Statistical Features*

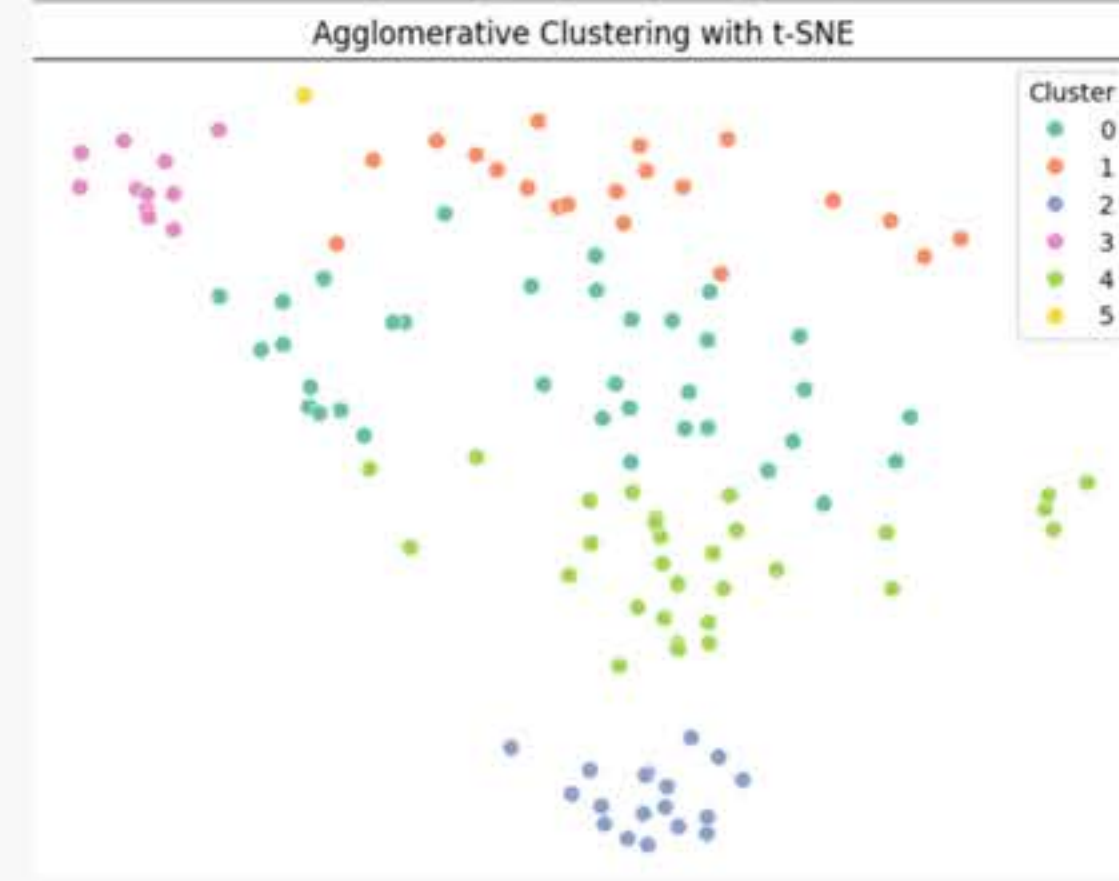
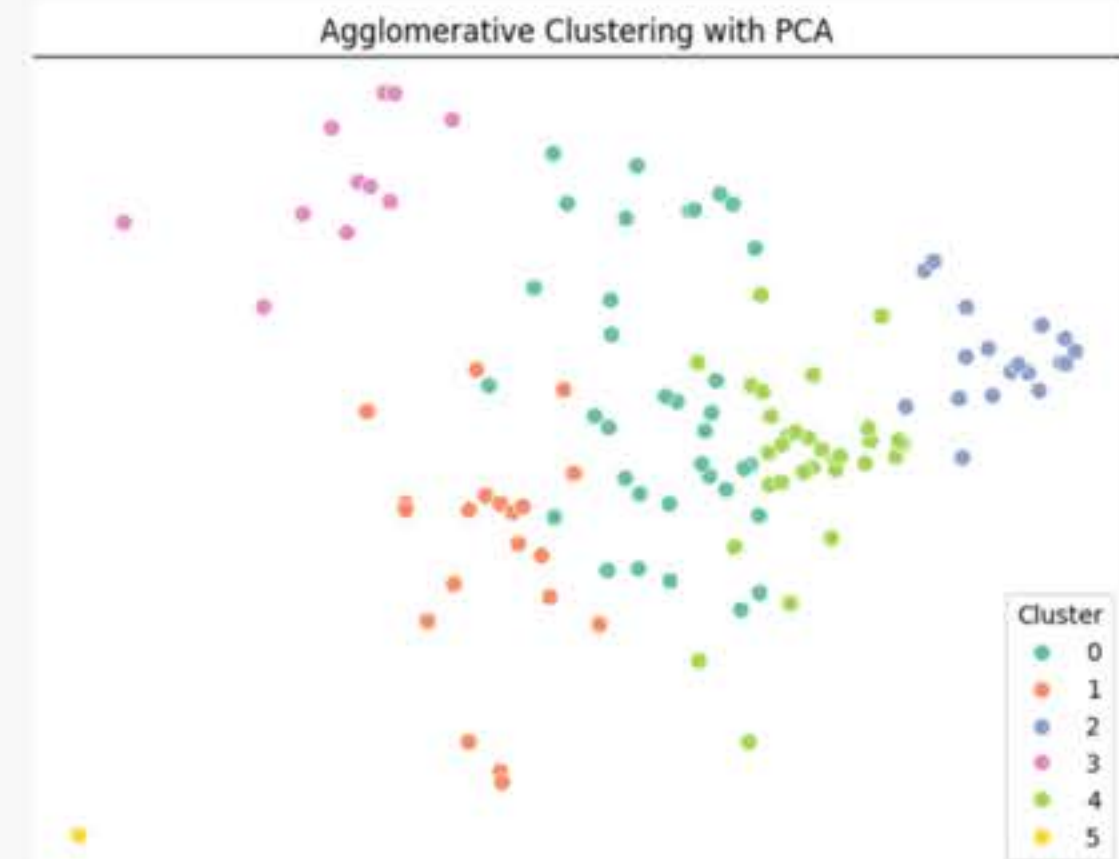
- **Objective:** Capture general statistics from each windowed segment.
- Features Extracted:
  - Mean, Variance, Minimum, Maximum, Median, Sum, and Peak-to-Peak Distance (P2P) for each windowed segment.
  - Skewness and Kurtosis were also computed to capture the distribution shape within each segment.
- Implementation:
  - Functions were created to calculate each feature per segment, then combined into a feature array.
  - Output Shape: Feature array for each song after Approach 1 has dimensions aligned with each segment's key statistics.



# K-Means Clustering



# Agglomerative Clustering





---



---

## *Approach 2: Enhanced MFCC Feature Fusion Approach*

- We introduced 8 additional features—such as spectral energy, dominant frequency, 25th and 75th percentiles, signal entropy, zero crossing rate, root mean square, and variance of absolute differences—to better capture the structural characteristics of the MFCCs.
- We then combined these features with those from our initial approach, resulting in a final vector of size 20x1700. This enriched feature set was analyzed across 116 songs using KMeans, PCA, Agglomerative Clustering, and DBSCAN, with t-SNE applied to visualize clustering results. This approach provided a detailed view of data patterns and structural similarities between songs.





# Overview of the 8 additional features



## Spectral Energy

$$E = \sum_{i=1}^N |X_i|^2$$

Spectral energy represents the total energy of the signal across frequencies, capturing the intensity of the audio. Higher energy indicates a louder or more powerful sound.

## Dominant Frequency

$$f_{\text{dominant}} = \operatorname{argmax}(|X(f)|)$$

The dominant frequency is the frequency with the highest amplitude in the signal. It helps in identifying the main pitch or characteristic frequency of the sound.

## 5th Percentile and 75th Percentile

$$P_{25} = \operatorname{percentile}(X, 25) \quad P_{75} = \operatorname{percentile}(X, 75)$$

These values represent the 25th and 75th percentiles of the signal's frequency distribution, giving insight into the spread and central tendency of the spectral components.

## Signal Entropy

$$H = - \sum_i p_i \log(p_i)$$

Entropy measures the randomness or complexity of the signal. Higher entropy indicates a more complex or diverse frequency content, while lower entropy suggests simpler, more uniform frequencies.







## Zero Crossing Rate (ZCR)

$$ZCR = \frac{1}{N-1} \sum_{i=1}^{N-1} \mathbf{1}_{\{X[i] \cdot X[i+1] < 0\}}$$

ZCR measures the rate at which the signal changes sign. It's a common feature for identifying percussive or noisier parts of the audio.

## Root Mean Square (RMS)

$$\text{RMS} = \sqrt{\frac{1}{N} \sum_{i=1}^N X_i^2}$$

RMS is a measure of the signal's magnitude, reflecting its average power. Higher RMS values indicate higher energy and intensity in the signal.

## Variance of Absolute Differences

$$\text{Variance} = \frac{1}{N-1} \sum_{i=1}^{N-1} (|X[i+1] - X[i]| - \mu)^2$$

This feature captures the variability in the signal's amplitude changes. Higher variance suggests more dynamic changes in the signal, useful for capturing structural complexity in audio.





---



## *Why Approach 2?*

It captures audio-specific characteristics essential for classifying songs:

- **Rich Audio Features:** Approach 2 uses features like spectral energy, dominant frequency, and zero-crossing rate that reveal key patterns in music data beyond basic statistics.
- **Temporal and Frequency Precision:** Features such as RMS and variance of absolute differences capture subtle changes in signal strength and structure over time, which are crucial for audio analysis.
- **Higher Classification Accuracy:** These audio-focused features better distinguish between music classes, leading to improved classification performance usually for musical signals.





---



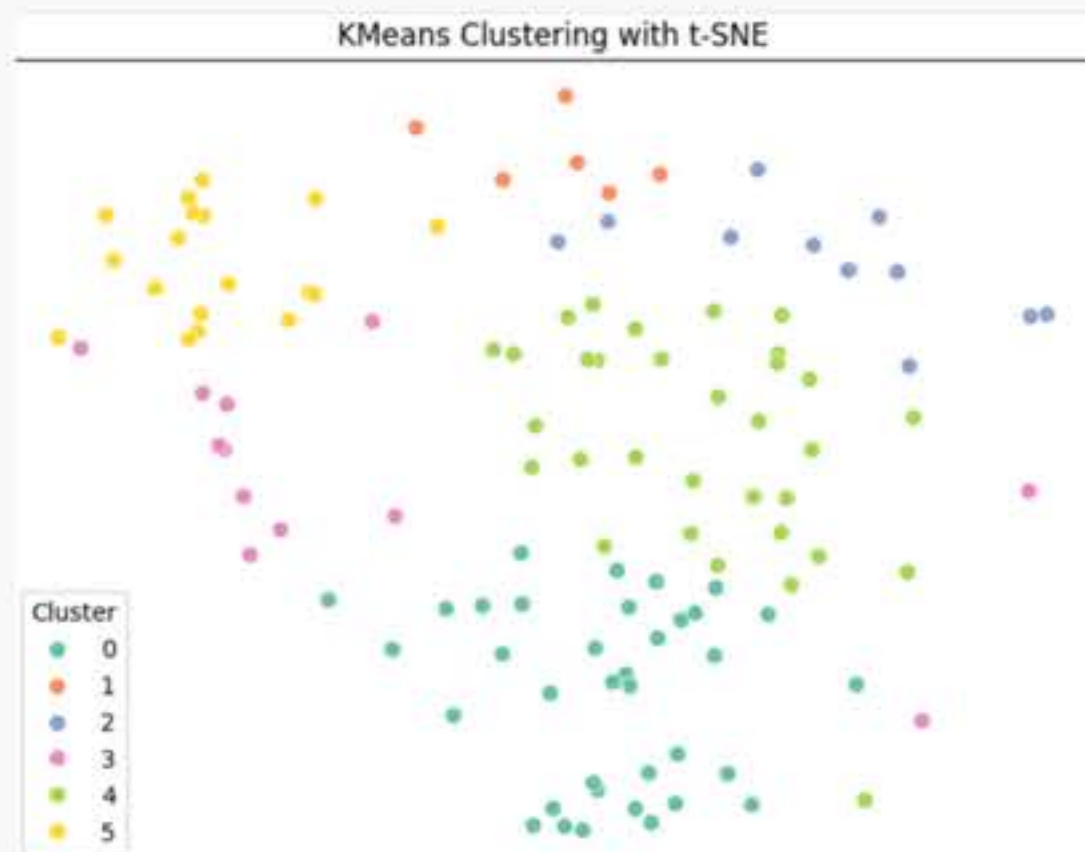
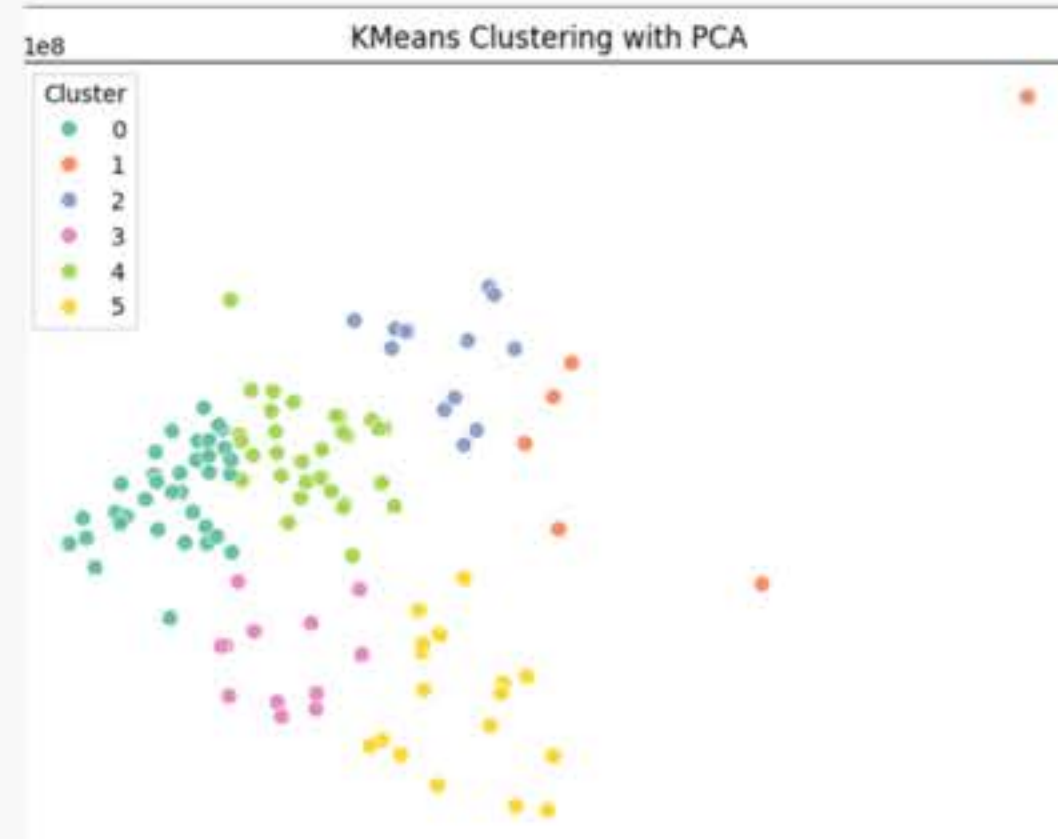
---

## *Feature Engineering - Spectral and Signal Features*

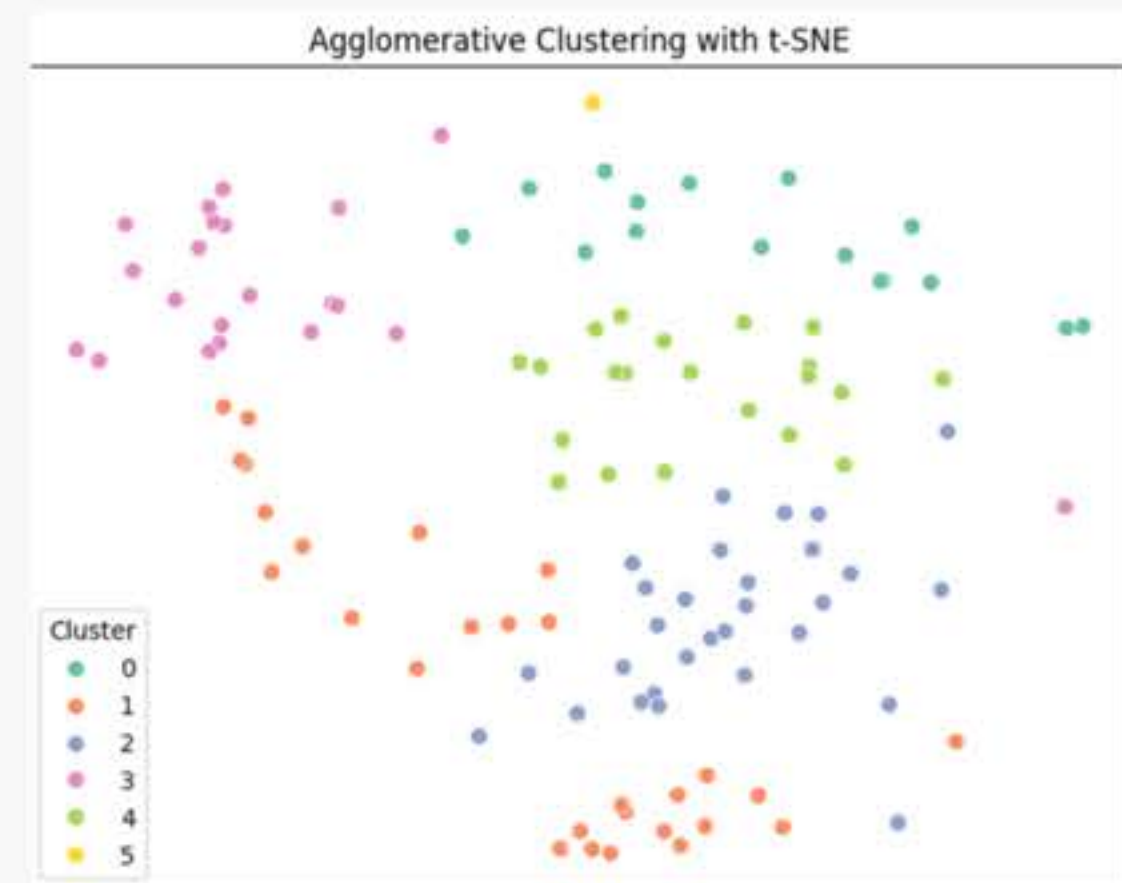
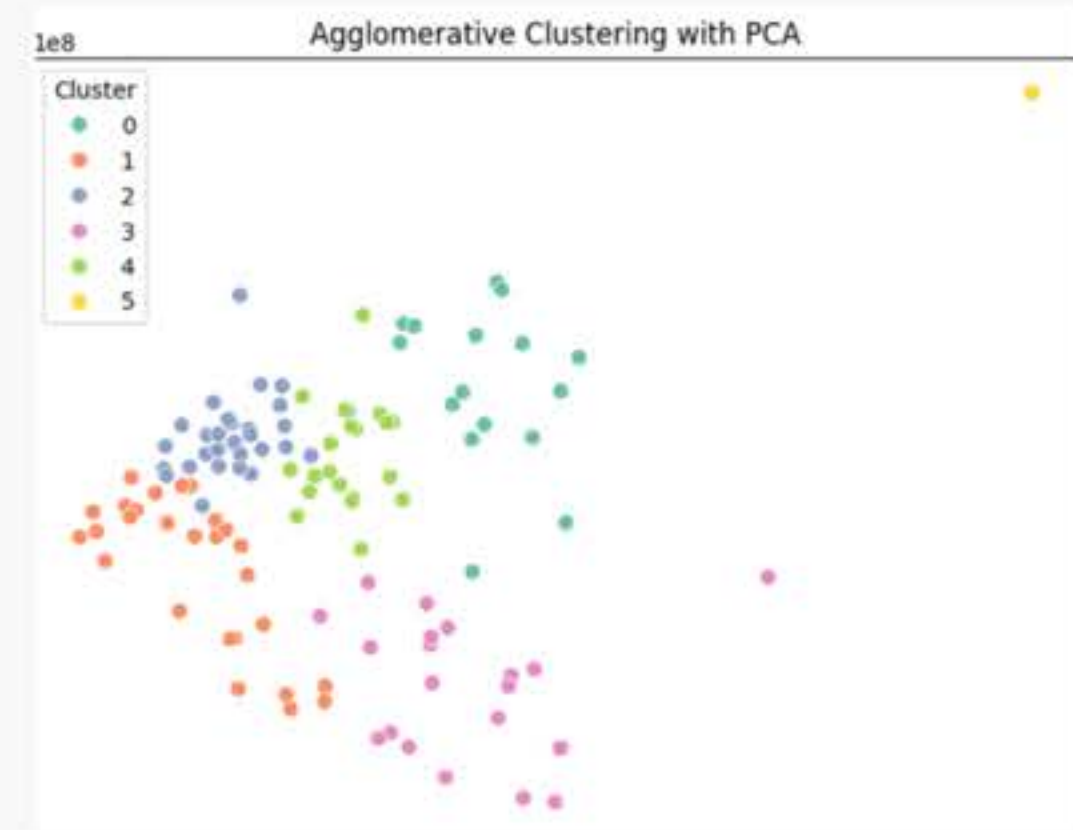
- **Objective:** Capture frequency-based and signal characteristics for better classification.
- Features Extracted:
  - Spectral Energy and Dominant Frequency: Captures energy concentration and most common frequency within each segment.
  - Percentiles (25th and 75th): Used to describe distribution of signal amplitudes.
  - Signal Entropy: Measures randomness within each segment.
  - Zero Crossing Rate: Frequency of signal changes from positive to negative.
  - Root Mean Square (RMS) and Variance of Absolute Differences: Measures signal strength and variability.
- Implementation:
  - Calculated each feature per segment and combined into a spectral feature array.
  - Output Shape: Aligned for each segment's spectral and signal characteristics.



# K-Means Clustering



# Agglomerative Clustering







## *FINAL APPROACH*

To prevent the temporal loss or shift in song information caused by variable-sized windows, we implemented fixed-sized windows. Although this approach resulted in feature vectors of unequal lengths, we addressed this by calculating distances between vectors of different sizes using methods such as FastDTW, Wavelet transform, and Cosine Similarity. These techniques allowed us to accurately measure distances and maintain the temporal integrity of the song data.





## *FastDTW (Fast Dynamic Time Warping)*

FastDTW calculates the similarity or alignment between two sequences by finding the optimal path that minimizes the distance, even if the sequences vary in speed or length. This is particularly useful for comparing audio features where slight shifts in time (such as variations in tempo) should not impact the similarity score drastically. FastDTW is faster and efficient than standard DTW, therefore practical for large datasets.

## *Wavelet Transform*

$$W_x(a, b) = \int_{-\infty}^{\infty} x(t) \psi \left( \frac{t - b}{a} \right) dt$$

The Wavelet Transform analyzes a signal at multiple scales and translations, providing a multi-resolution view. This is helpful for capturing both high-frequency (detailed) and low-frequency (coarse) patterns in the audio. Unlike Fourier Transform, which only provides frequency information, the Wavelet Transform retains time-localization, making it ideal for analyzing non-stationary signals like audio.

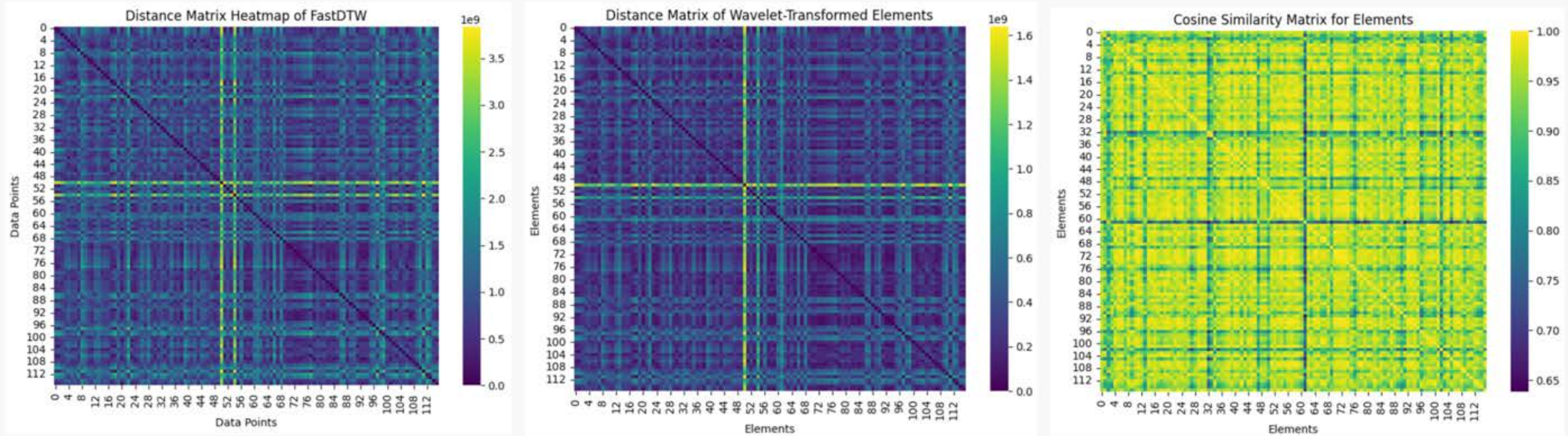
## *Cosine Similarity*

$$\text{Cosine Similarity} = \frac{\sum_{i=1}^N A_i \cdot B_i}{\sqrt{\sum_{i=1}^N A_i^2} \cdot \sqrt{\sum_{i=1}^N B_i^2}}$$

Cosine Similarity measures the angle between two vectors, giving a similarity score between -1 and 1. A score closer to 1 indicates that the vectors (representing songs in this case) are similar in direction, hence more alike in feature space. Cosine Similarity is commonly used in high-dimensional data analysis, as it effectively captures the relative orientation of feature vectors, which is useful for comparing song features



## Heatmaps for different distance metrics

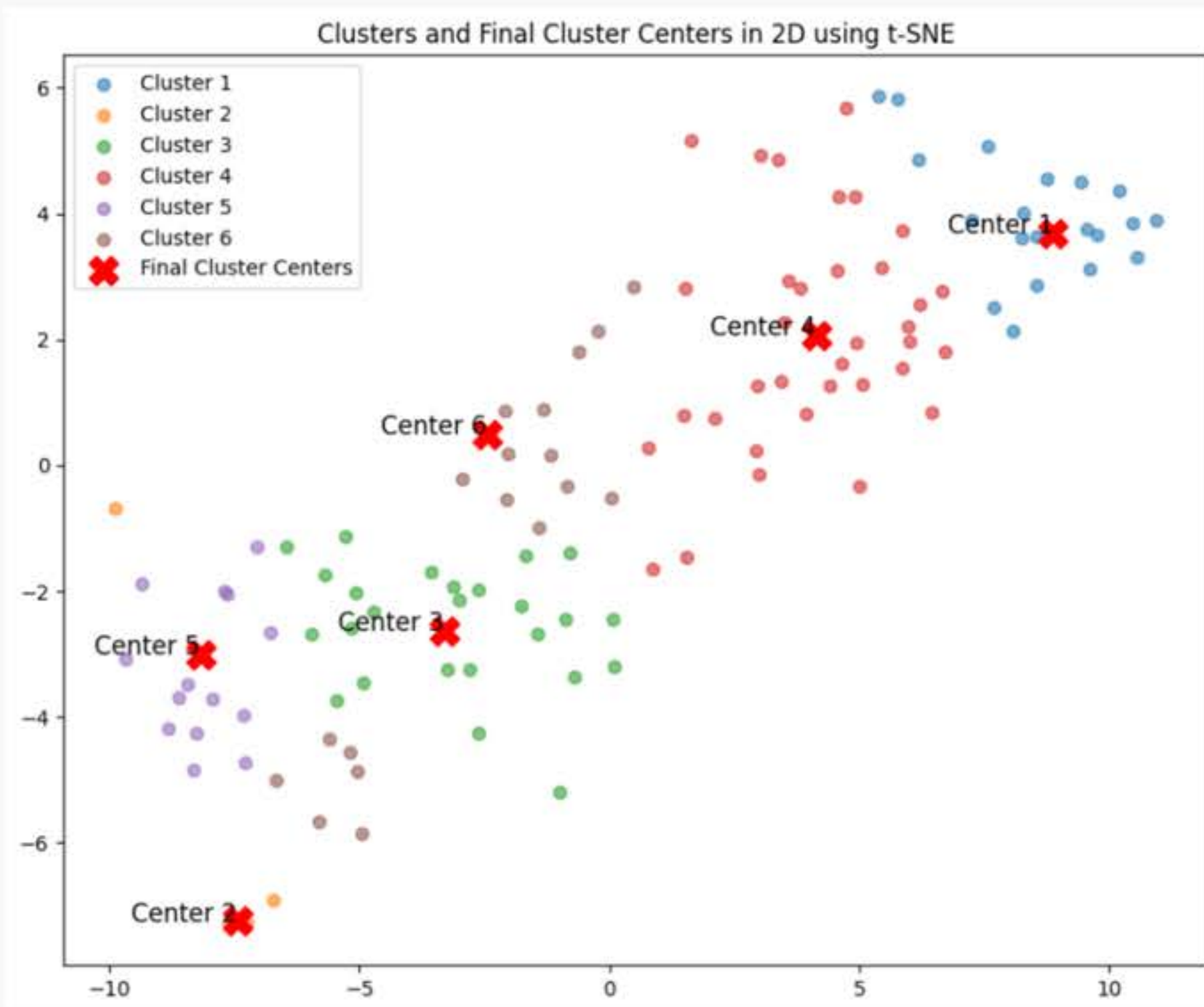


*Final clustering metrics for all the approaches we experimented with:*

Approach	Silhouette Score	Davies-Bouldin Index	Calinski-Harabasz Index
Approach 1	0.18	2.8	50.43
Approach 2	0.24	1.74	101.56
Final Approach	0.34	1.23	152.6



## *Final Approach - Marked Cluster Centers (Scrapped Data/ Labelled Data)*



*This achieves the maximum of the following classification metrics:*

- 1) Silhouette Score: 0.34*
- 2) Davies Bouldin Index: 1.23*
- 3) Calinski-Harabasz Index: 152.6*



---



---

## *Answers to the given questions*

Q1): Analyze MFCC files to organize the 115 files into groups broadly corresponding to those listed .


Here is the link to csv file containing MFCC to category mapping

<https://drive.google.com/file/d/1lavDkbzEfPWeUmvFzUUE2Z2FvKpVnSUJ/view>

Q2): Identify at least 3 files containing the National Anthem.

File number 1, 2 and 104 corresponds to National Anthem

Q3): Identify at least 3 files (each) containing solo songs by Asha Bhosale, Kishor Kumar, and Michael Jackson

- Asha Bhosle: 7, 8 and 47
  - Kishore Kumar: 9, 10 and 113
  - Michael Jackson: 11, 12 and 30
- 





*Thank you*

