Minor Exam Report

Anshul Thakur (B21CS85)

March 9, 2025

repository link: https://github.com/dr-ghost/Speech-Understanding-Minor-Exam

1 Question1

1.1 Introduction

In this project, I collected 10 audio samples of my own voice under different conditions (varying text content, pitch, and volume). The primary objectives are to extract and analyze several audio features including amplitude, pitch, frequency, and RMS energy, and to visualize the results using spectrograms.

1.2 Dataset Collection

The dataset consists of 10 audio samples recorded using a standard microphone. Each sample contains different textual content and was recorded under varying conditions of pitch and volume.

1.3 Feature Extraction and Analysis

In this section, I describe the features extracted from the audio signals, the methods used to compute them, and the subsequent analysis.

1.3.1 Amplitude Analysis

Amplitude represents the absolute value of the audio signal and provides information about its dynamic range. The waveform is first loaded and converted to its absolute value to observe the amplitude variations over time.

Method: The function `plot_waveform` computes and plots the amplitude by taking the absolute value of the signal and then plotting it against time.

1.4 Frequency Spectrum Analysis

The frequency spectrum provides insight into the distribution of power across different frequencies. It is computed using periodogram analysis as well as FFT-based methods.

Method: The function `plot_frequency_spectrum` uses `scipy.signal.periodogram` to calculate the power spectral density of the signal, which is then plotted on a logarithmic scale.

1.5 RMS Energy Calculation

RMS (Root Mean Square) energy is a measure of the energy present in the audio signal over time. It is especially useful for identifying segments with higher loudness.

Method: The `plot_rms_energy` function divides the waveform into overlapping frames, calculates the square of the amplitudes, computes the mean over each frame, and then takes the square root to obtain the RMS energy.

2 Comparative Study

A comparative study was performed on the 10 audio samples to observe how changes in text, pitch, and volume affect the audio features:

- **Amplitude:** Variations in volume are clearly visible in the amplitude plots. Samples recorded at a higher volume show larger amplitude values.
- **Pitch:** Changes in pitch are reflected in the pitch contour plots. Samples with varying intonation exhibit different pitch profiles, which can be correlated with the textual content and emotional tone.
- **Frequency Spectrum:** The periodogram analysis reveals differences in frequency content between samples. For example, a sample with a lower pitch tends to have more energy concentrated in lower frequencies.
- **RMS Energy:** The RMS energy plots indicate segments of higher energy corresponding to louder parts of the recording. These variations are useful for segmenting the audio into regions of interest.

The plots of features and spectrograms are in the ipynb files attached with this report

3 Question 2

3.1 Introduction

In this report, I analyze speeches from prominent early 20th century figures(given in the dataset) by extracting key audio features. The aim is to correlate these features with the perceived emotional tone of the speeches, thereby providing insights into the speakers' vocal delivery.

3.2 Methodology

3.2.1 Audio Feature Extraction

I extracted the following traditional speech features from the recordings:

- **Zero-Crossing Rate (ZCR):** Measures the rate at which the audio signal changes its sign within a frame. It provides an indication of the signal's frequency content.
- **Short-Time Energy (STE):** Computes the energy of the signal in short frames by summing the squared amplitudes. Higher energy often corresponds to louder or more dynamic speech segments.
- **Mel-Frequency Cepstral Coefficients (MFCCs):** The first 13 coefficients are extracted to capture the spectral envelope of the speech, which is linked to the shape and configuration of the vocal tract.

3.3 Feature Calculation Details

3.3.1 Zero-Crossing Rate (ZCR)

The ZCR is calculated by dividing the audio signal into overlapping frames and counting the number of times the signal changes its sign, normalized by the frame length. A higher ZCR is typically associated with a more dynamic or energetic signal.

3.3.2 Short-Time Energy (STE)

STE is computed by segmenting the audio signal into short frames, squaring the amplitude values, and then summing these values over each frame. This value is normalized by the frame length. Speech segments with high energy are often perceived as louder or more passionate.

3.3.3 Mel-Frequency Cepstral Coefficients (MFCCs)

MFCCs are obtained by:

1. Applying a Fourier transform to convert the time-domain signal to the frequency domain.
2. Mapping the powers of the spectrum onto the mel scale using a set of triangular filter banks.
3. Taking the logarithm of the mel spectrum.
4. Computing the Discrete Cosine Transform (DCT) to decorrelate the filter bank coefficients.

The first 13 coefficients are retained as they effectively represent the shape of the vocal tract. (I did use the torchaudio library specifically the torchaudio.transforms.MFCC transform to compute the MFCC features).

3.4 Comparative Analysis

I selected these two audio clips the analysis:

1. **Mahatma Gandhi's speech:** A speech characterized by a measured, steady delivery.
2. **PM Shri Narendra Modi's speech:** A speech marked by dynamic and higher energy.

3.4.1 Energy of the Signal

The analysis of Short-Time Energy (STE) reveals that the passionate speech exhibits higher energy levels (PM Shri Narendra Modi), indicative of stronger emphasis and louder segments. In contrast, the calm speech maintains lower, more consistent energy levels (Mahatma Gandhi). Figure 1 and 2 shows the STE comparison.

3.5 Frequency Content

The Zero-Crossing Rate (ZCR) comparison (see Figure 3 and 4) indicates that Mahatma Gandhi's speech had more fluctuations than PM Shri Narendra Modi's Speech.

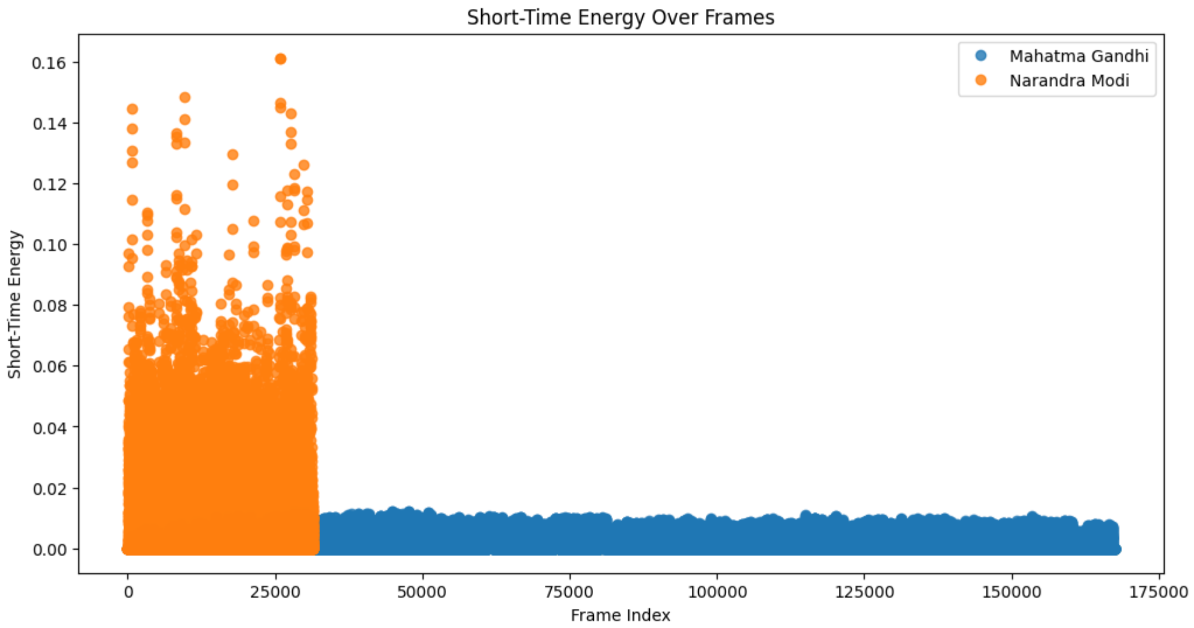

Figure 1: STE Plot

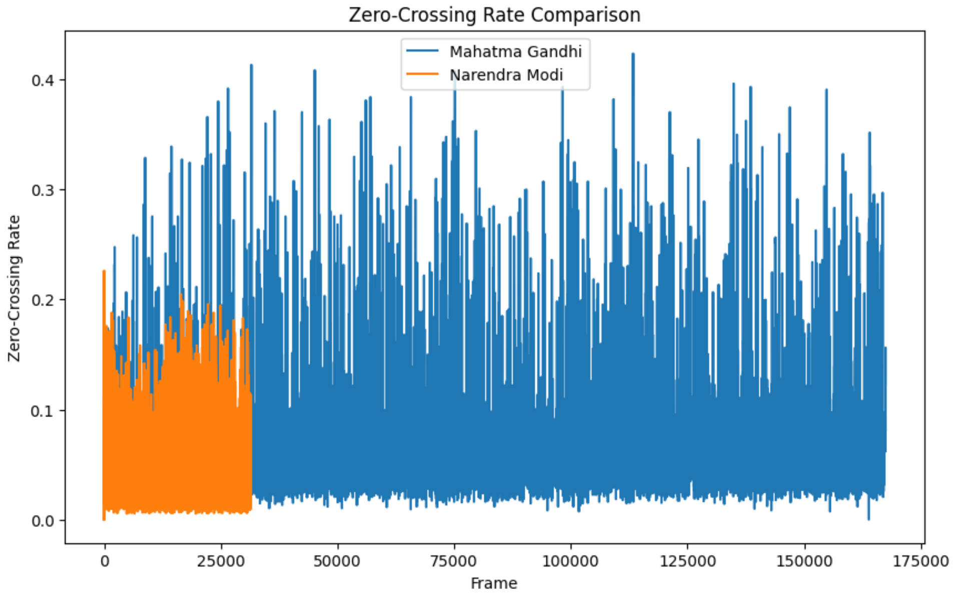

Figure 3: ZCR Plot

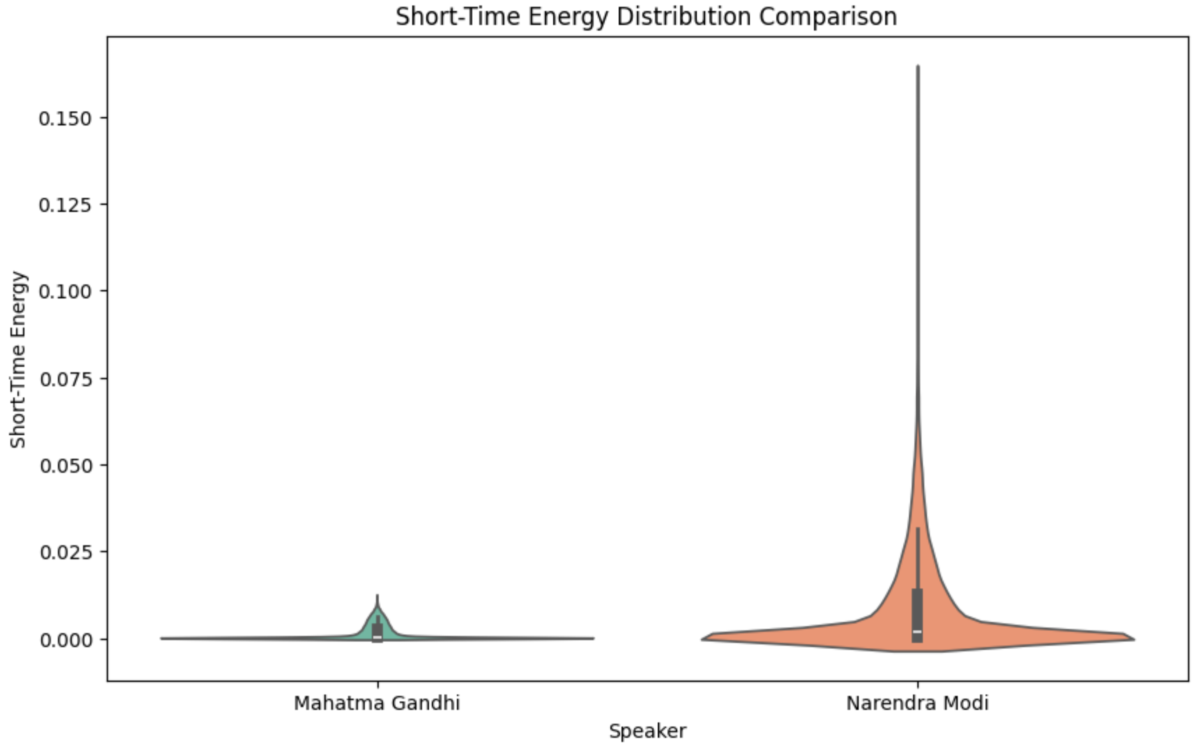

Figure 2: STE Plot Violin

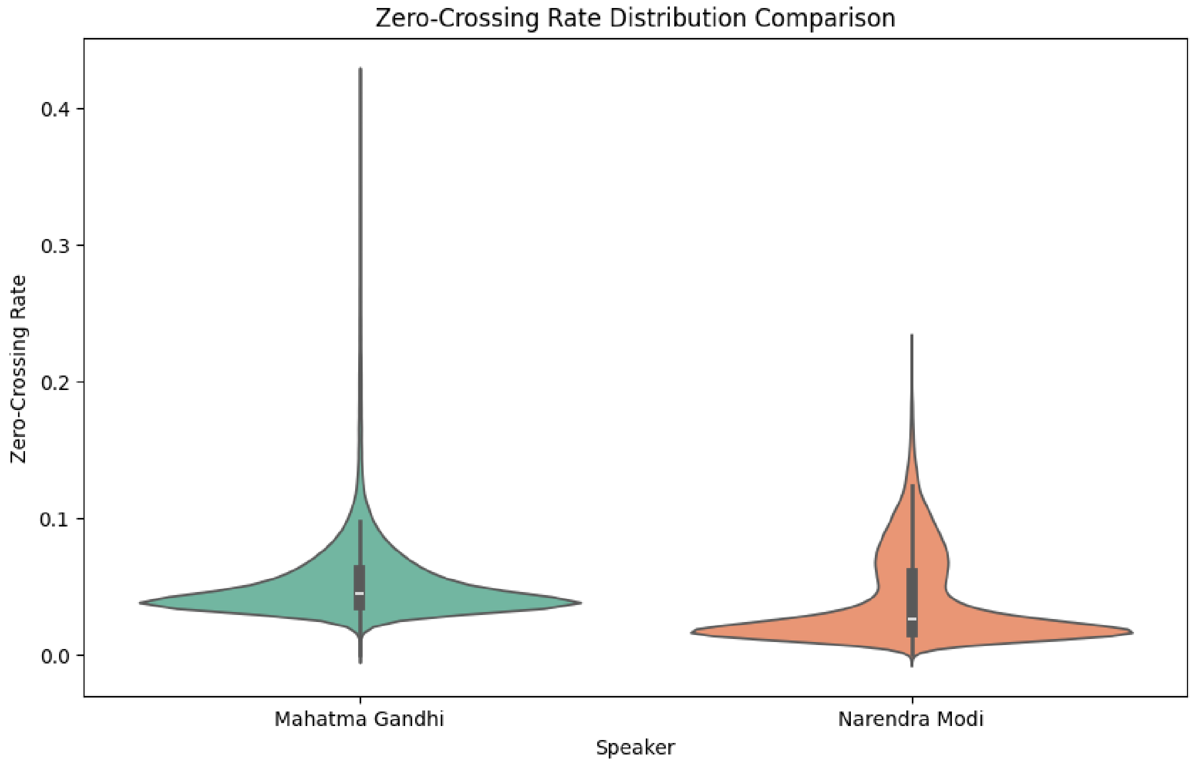

Figure 4: ZCR Plot Violin

3.6 Vocal Tract Characteristics

The MFCC analysis, particularly the first 13 coefficients, provides insights into the spectral envelope related to the vocal tract shape. The differences observed in the MFCC plots (Figure 5 and 6)

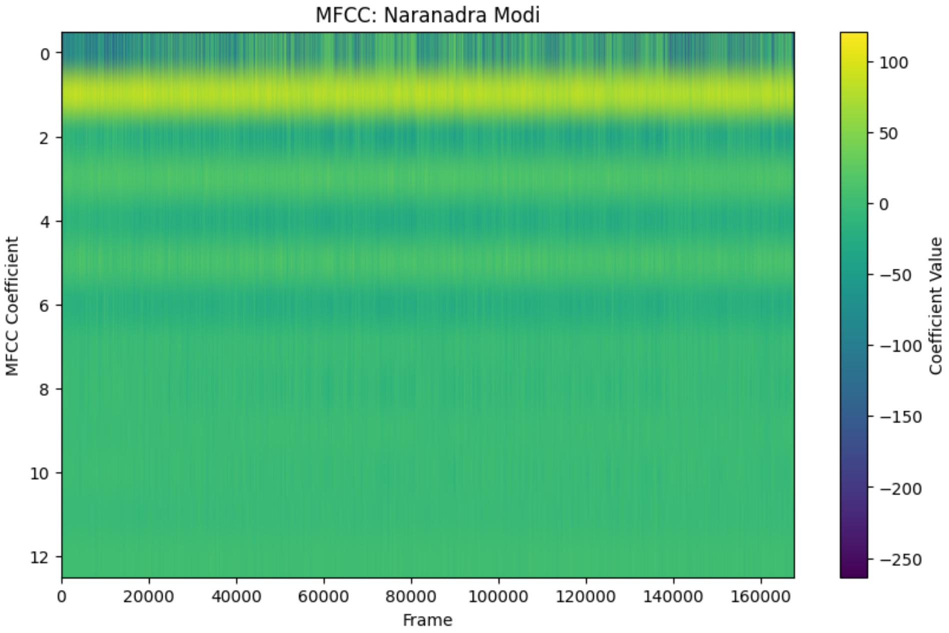

Figure 5: MFCC

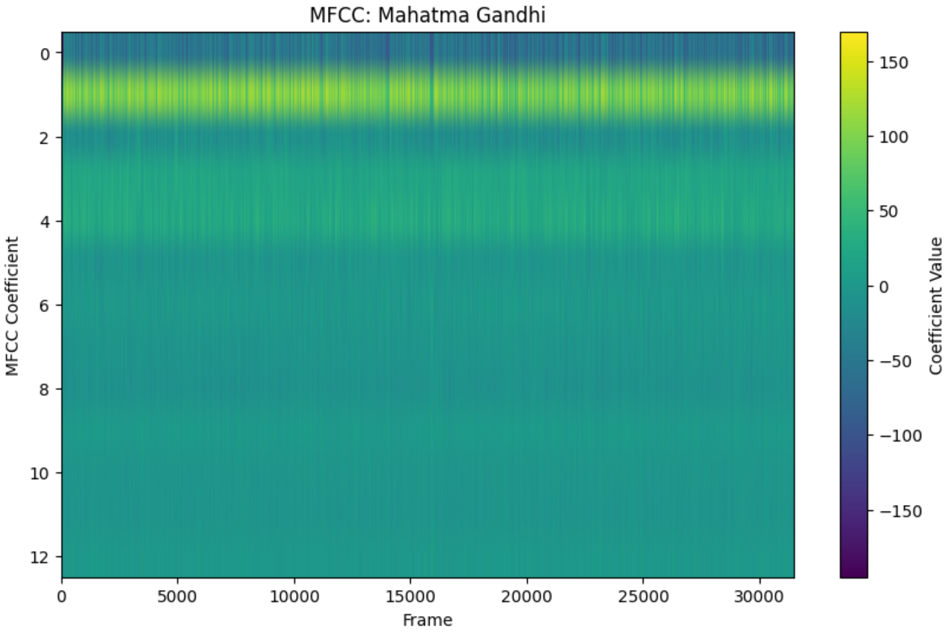

Figure 6: MFCC

3.7 Limitations and Improvements

3.7.1 Limitations

While traditional features (ZCR, STE, MFCCs) offer valuable insights, there are several limitations when applied to historical recordings:

- **Noise Sensitivity:** Background noise and recording artifacts can distort the feature values.
- **Limited Emotional Cues:** These features do not capture prosodic or contextual nuances that may be critical for accurate emotion detection.
- **Fixed Window Analysis:** The use of fixed-length windows may not adapt well to varying speech dynamics.

3.7.2 Potential Improvements

To overcome these challenges, one potential improvement is the integration of deep learning methods for emotion recognition

- **Deep Neural Networks:** Models such as Convolutional Neural Networks (CNNs) or Recurrent Neural Networks (RNNs) or ResNet Architectures can be trained on a large corpus of annotated speech data to learn robust feature representations.

4 Question 3

4.1 Introduction

In this task I was tasked to develop a classification system to classify the vowel sounds (`/a/`, `/e/`, `/i/`, `/o/`, `/u/`) using traditional speech processing techniques. Specifically my tasks were:

- Extracting key acoustic features, including the formant frequencies (F1, F2, F3) and the fundamental frequency (F0),
- Visualizing the vowel space using an F1-F2 plot,
- Implementing a classification algorithm (KNN) based on these features, and
- Evaluating performance using confusion matrices and accuracy metrics.

The dataset comprised of speech samples from adult males and females (that was provided in the question), and the data is split into training and testing sets with an 80:20 ratio (random state = 45).

4.2 Methodology

4.2.1 Audio Pre-processing

I first loaded and preprocessed Audio files. The speech signal is segmented into frames using techniques such as framing and windowing. I have written different windowing functions such as rectangular windowing hanning windowing and hamming windowing but I have used hanning window technique throughout the question.

4.2.2 Formant Frequency Extraction Using LPC

I employed Linear Predictive Coding (LPC) to model the spectral envelope of the speech signal. LPC assumes that a speech signal $x[n]$ can be approximated by a linear combination of its past samples:

$$x[n] \approx -\sum_{k=1}^{p} a_k \, x[n-k] + e[n],$$

where a_k are the LPC coefficients, p is the model order, and $e[n]$ is the prediction error.

The LPC polynomial is defined as:

$$A(z) = 1 - \sum_{k=1}^{p} a_k \, z^{-k}.$$

By finding the roots of $A(z)$ and considering only those with positive imaginary parts, the system estimates the resonant frequencies (formants). In this implementation, the first three formants (F1, F2, F3) are extracted.

4.2.3 Fundamental Frequency (F0) Extraction

The fundamental frequency $F0$ is estimated using an autocorrelation method. The signal is first centered by subtracting its mean, and its autocorrelation is computed. The peak in the autocorrelation function within a plausible pitch range (50–500 Hz) is used to estimate $F0$ as:

$$F0 = \frac{\text{Sampling Rate}}{\text{Peak Lag}}.$$

4.3 Vowel Space Visualization

An F1-F2 plot 7 is generated to visualize the vowel space. In this plot, each vowel is represented by a point where the x-axis corresponds to F1 and the y-axis corresponds to F2.

4.4 Classification System

4.4.1 Feature Vector Construction

For each audio sample, I constructed a feature vector by concatenating the extracted features: $[F0, F1, F2, F3]$. The feature vectors from multiple frames are flattened into a single vector for each sample.

4.4.2 k-Nearest Neighbors (KNN) Classifier

I implemented a KNN classifier from scratch. The classifier stores the training feature vectors and their corresponding labels. For each test sample, Euclidean distances to all training samples are computed. The k nearest neighbors are identified, and the most frequent class among them is assigned to the test sample.

4.4.3 Training and Evaluation

The dataset is split into training and testing sets in an 80:20 ratio (random state = 45). The classifier is trained on the training set and then evaluated on the test set. Performance is quantified using metrics such as accuracy and the F1 score, and a confusion matrix is generated to illustrate the classification results.

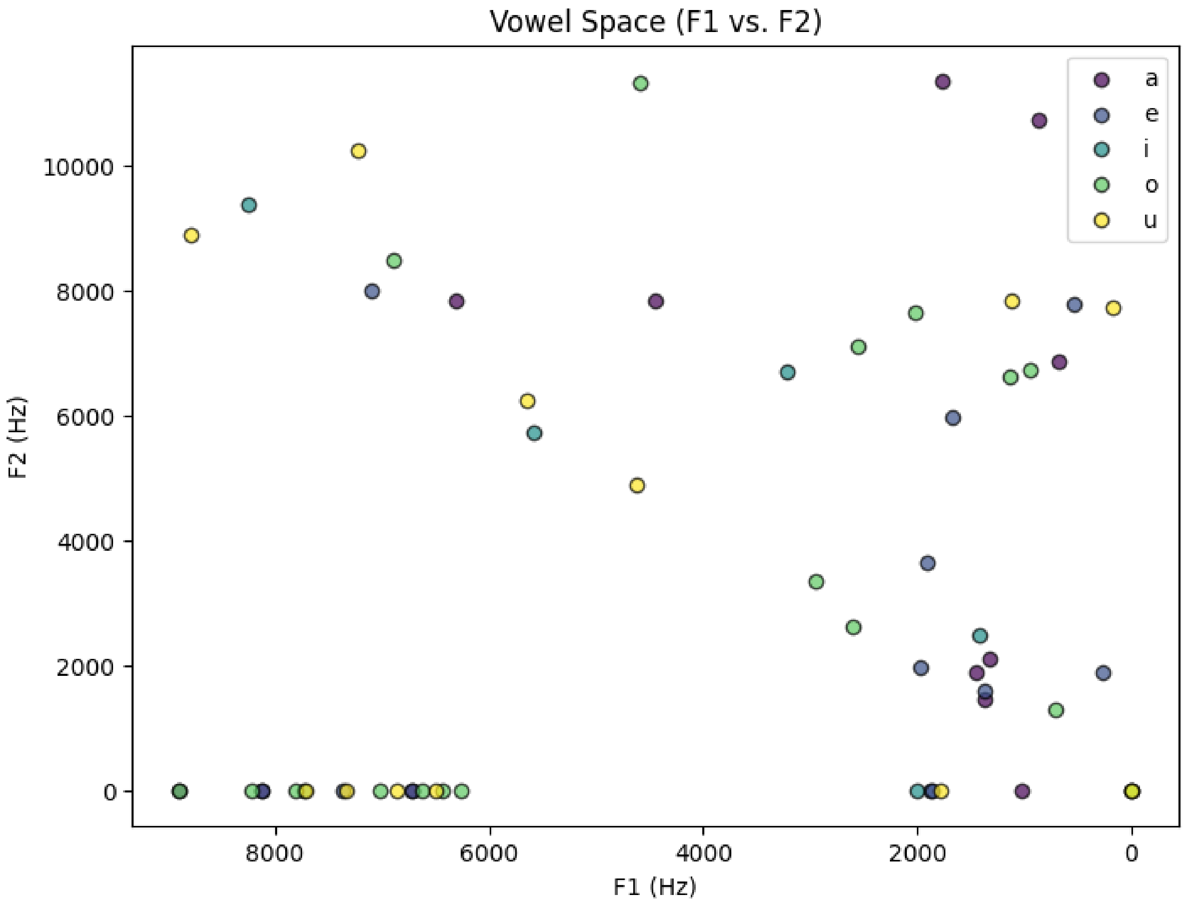

Figure 7: Vowel Space

4.5 Results and Analysis

4.5.1 Classification Performance

For k = 3, the KNN classifier achieved an accuracy of **17%** on the test set. The confusion matrix (see Figure 8) highlights areas where the classifier performs well and where misclassifications occur. Here the classifier is getting, `/o/` and `/a/`) confused.

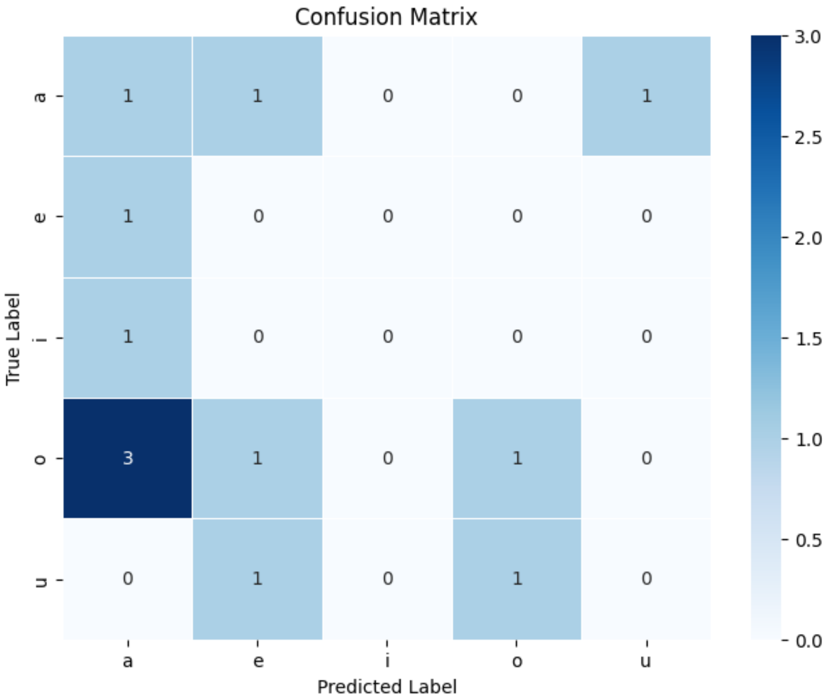

Figure 8: Confusion Matrix for Vowel Classification (k = 3)

For k = 20, the KNN classifier achieved an accuracy of **58.4%** on the test set. The confusion matrix (see Figure 9).

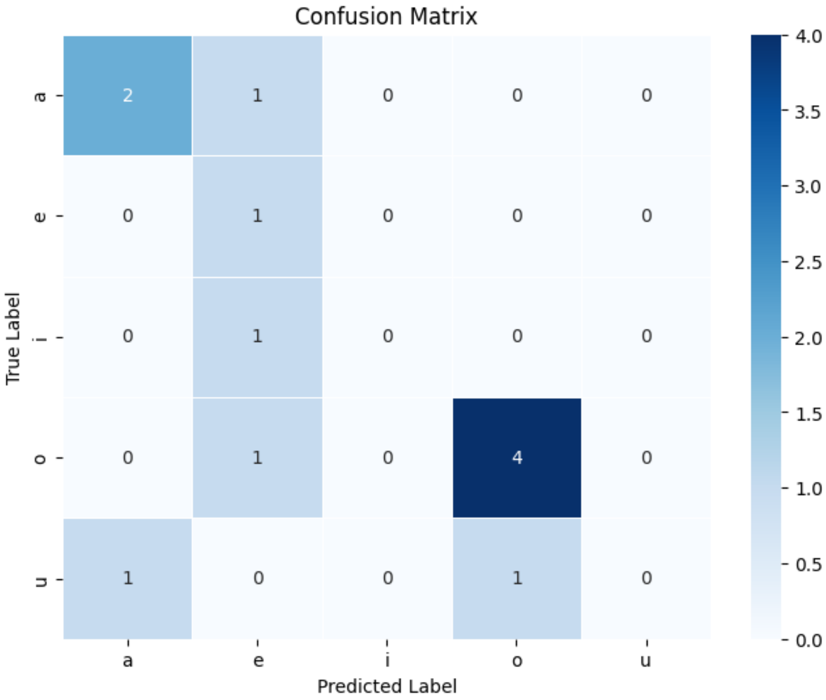

Figure 9: Confusion Matrix for Vowel Classification (k = 20)

4.6 Analysis and Reflection

4.6.1 Comparison with Theoretical Expectations

The experimental results generally align with theoretical vowel formant patterns. Although I couldn't see a clear decision boundary in the F1-F2 vowel space (that is probably due to the dataset).

4.6.2 Sources of Error

Potential sources of error include:

- Variability in speaker characteristics and recording conditions,
- Noise and distortions in the audio signal,
- Limitations of extracting features from a single or a few frames.
- Limited data available.
- weak classifier.

4.6.3 Relation to Historical Speech Recognition Systems

Historically, early speech recognition systems relied heavily on formant-based features, such as those extracted via LPC. Although modern systems often use more complex features (e.g., MFCCs) and deep learning techniques, the methods used in this project reflect the foundational approaches that paved the way for current technologies.

4.6.4 Potential Improvements

Two suggested improvements are:

1. **Multi-Frame Averaging:** Instead of extracting features from a single frame, averaging features over multiple steady-state frames could reduce variability.
2. **Additional Features:** Incorporating supplementary features such as MFCCs or dynamic (delta) features could improve the robustness and accuracy of the classification system.