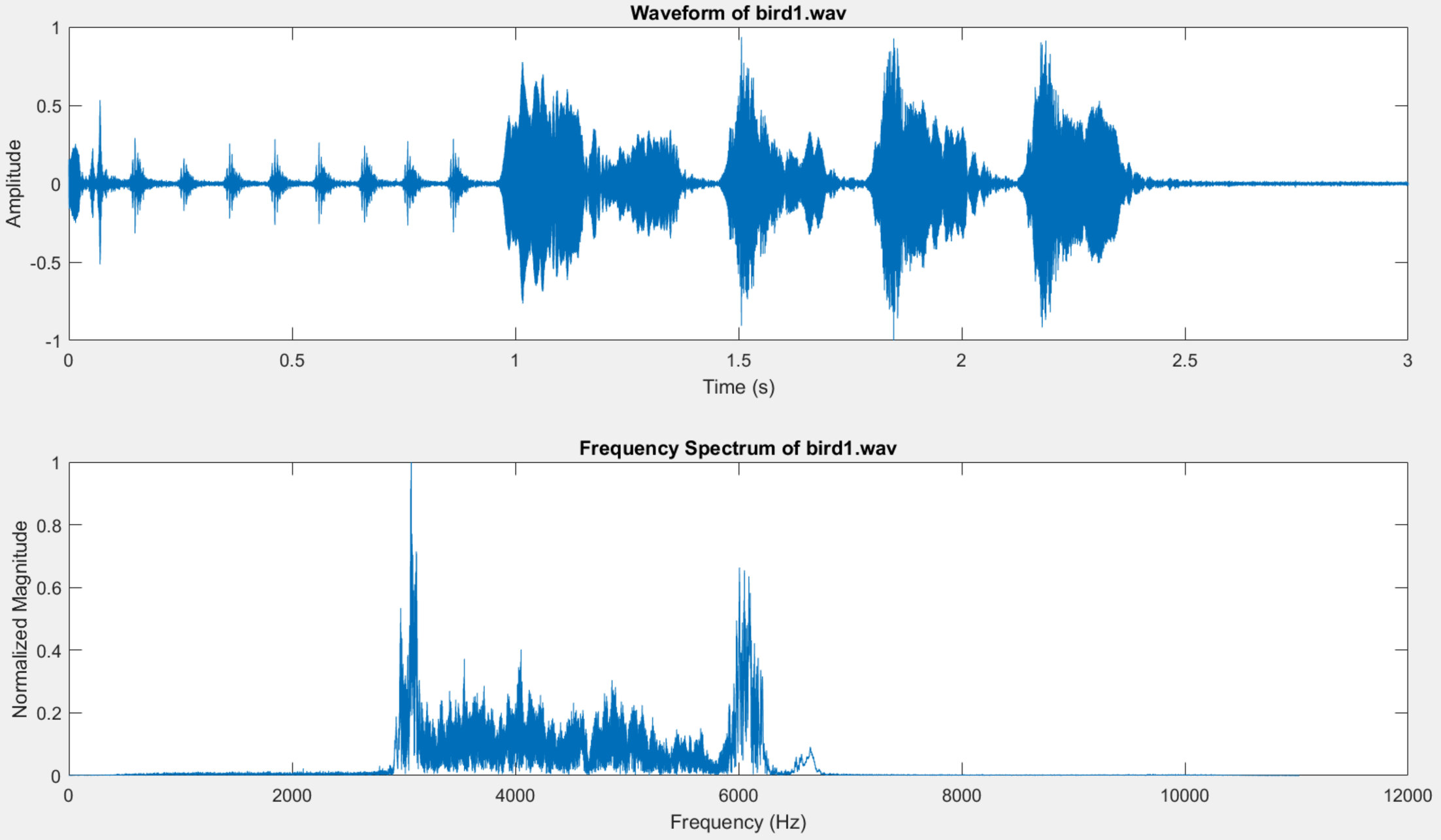
Signal Processing Project

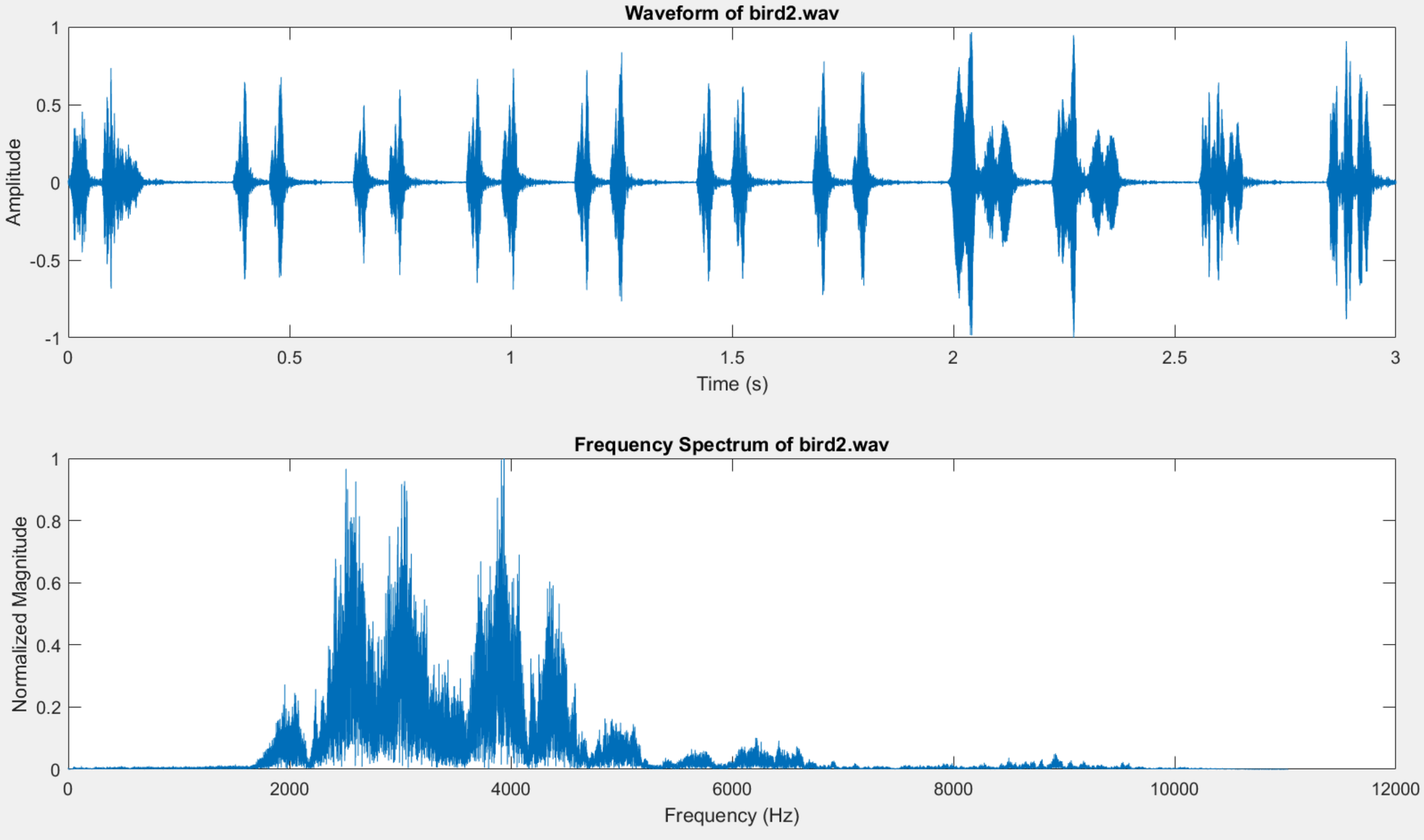
Ritama Sanyal 2023112027
Gandlur Valli 2023102068
Priyanshi Jain 2023112021

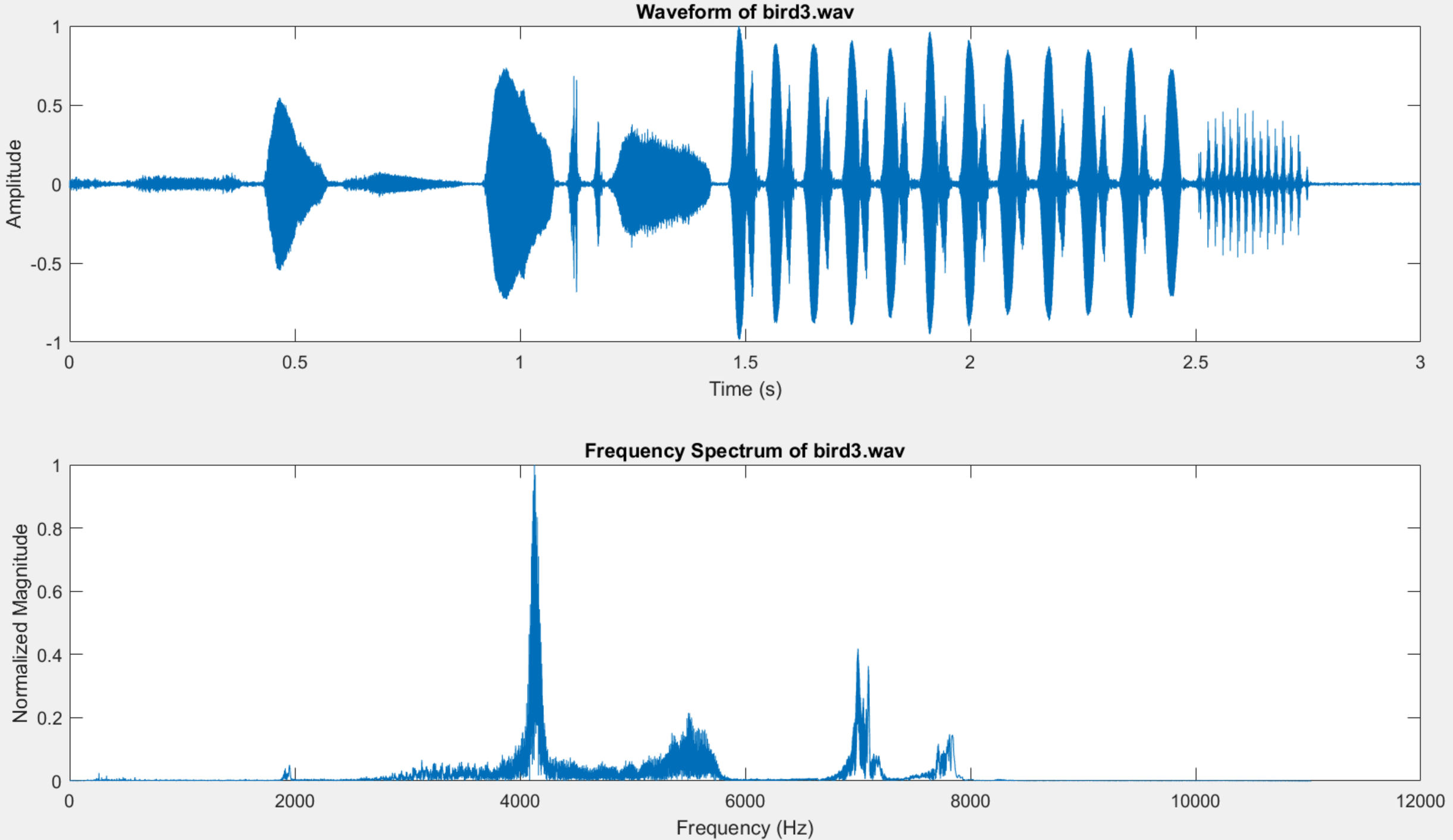
PART 1
Bird Recognition

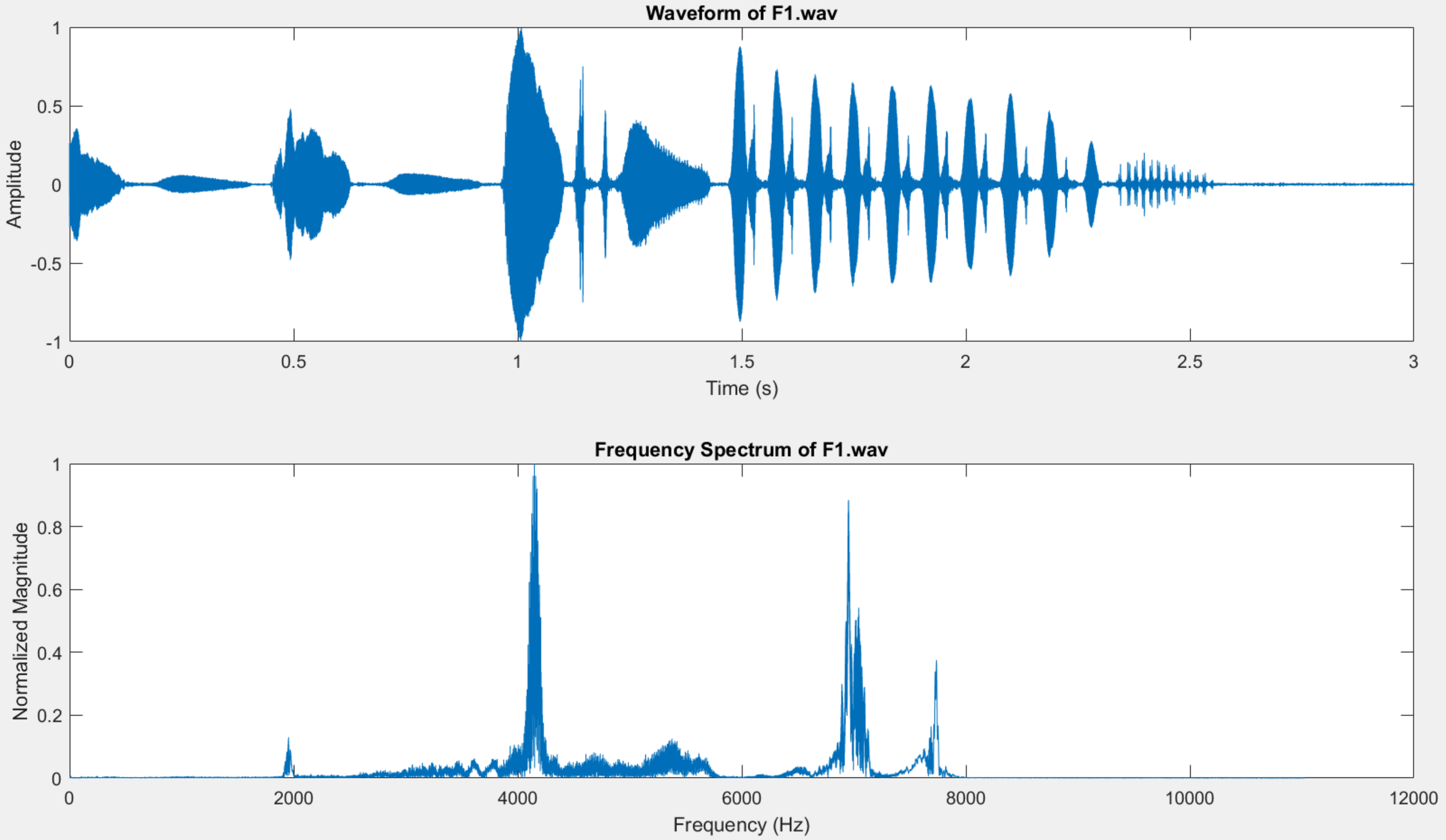
Result

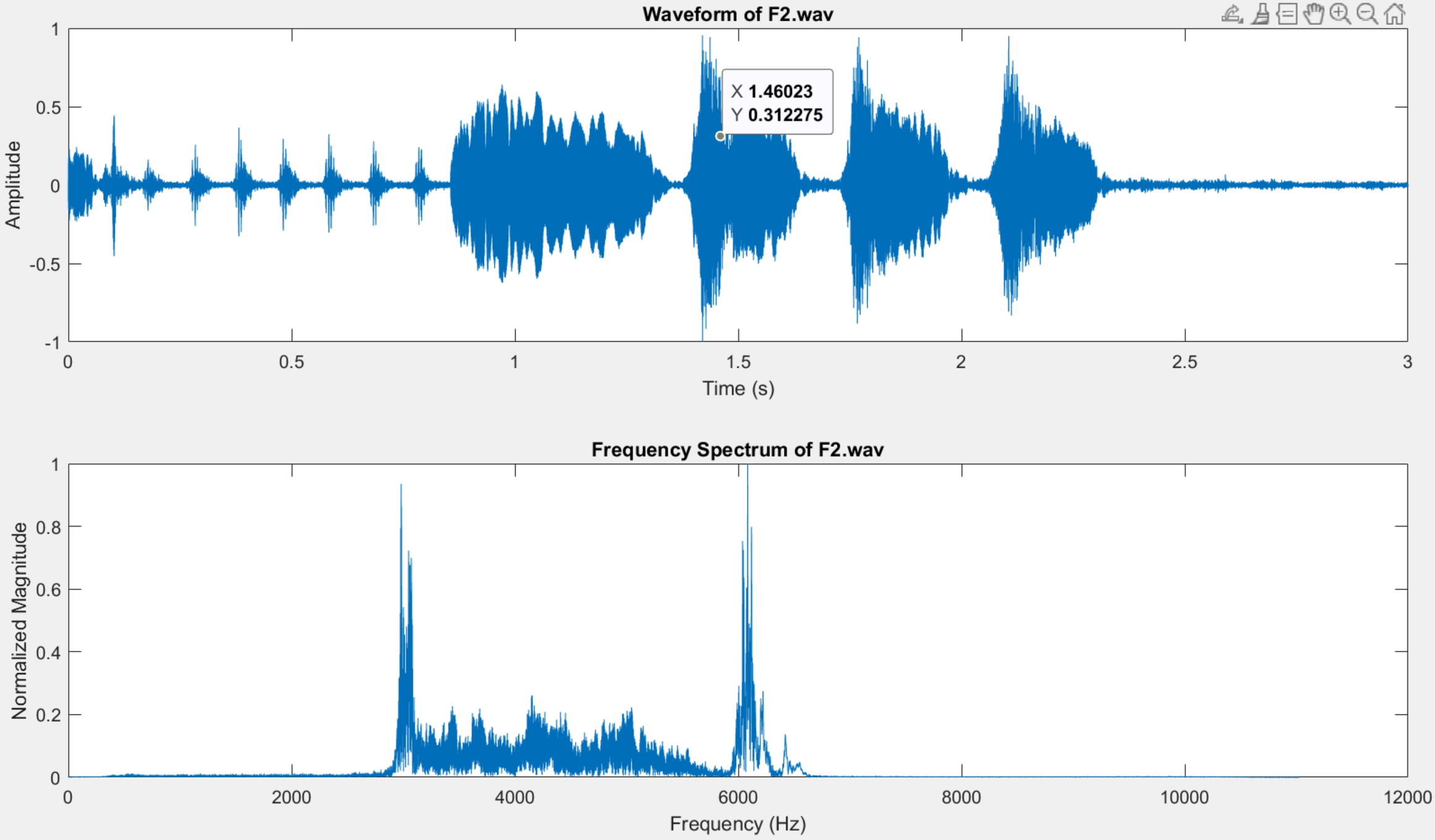
```
File F1.wav matches with Bird 3
File F2.wav matches with Bird 1
File F3.wav matches with Bird 2
File F4.wav matches with Bird 3
File F5.wav matches with Bird 1
File F6.wav matches with Bird 3
File F7.wav matches with Bird 1
File F8.wav matches with Bird 2
```

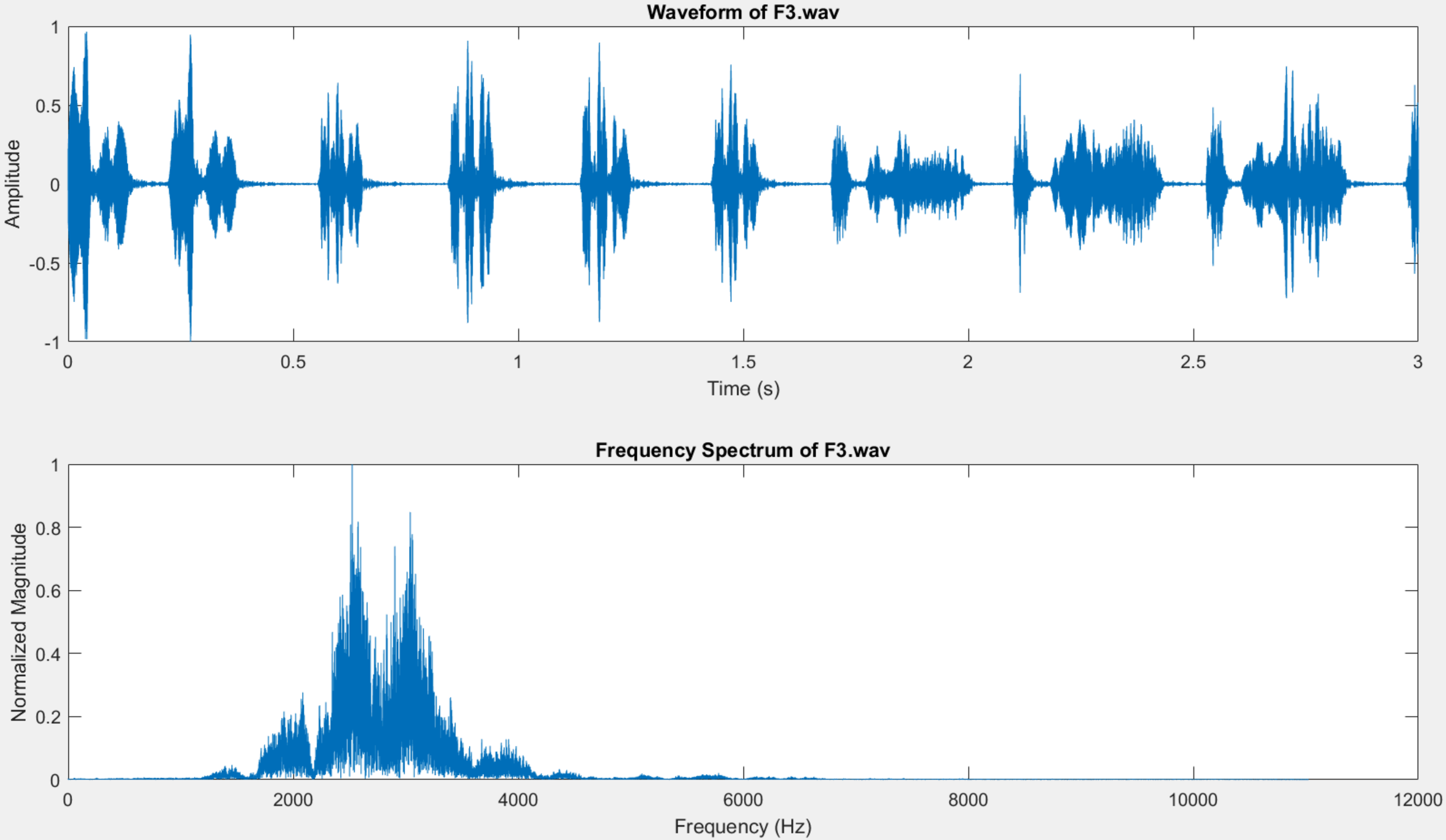

Waveform of bird1.wav
Amplitude
Time (s)
1
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
3
Frequency Spectrum of bird1.wav
Normalized Magnitude
Frequency (Hz)
0.8
0.6
0.4
0.2
2000
4000
6000
8000
10000
12000

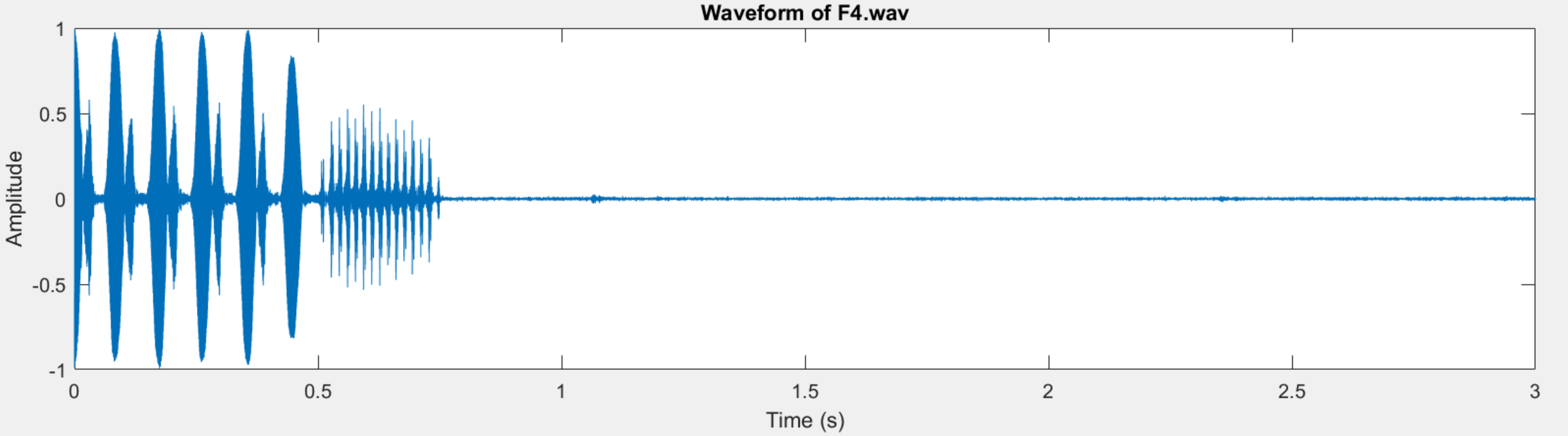

Waveform of bird2.wav
Amplitude
Time (s)
1
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
3
Frequency Spectrum of bird2.wav
Normalized Magnitude
Frequency (Hz)
0.8
0.6
0.4
0.2
2000
4000
6000
8000
10000
12000

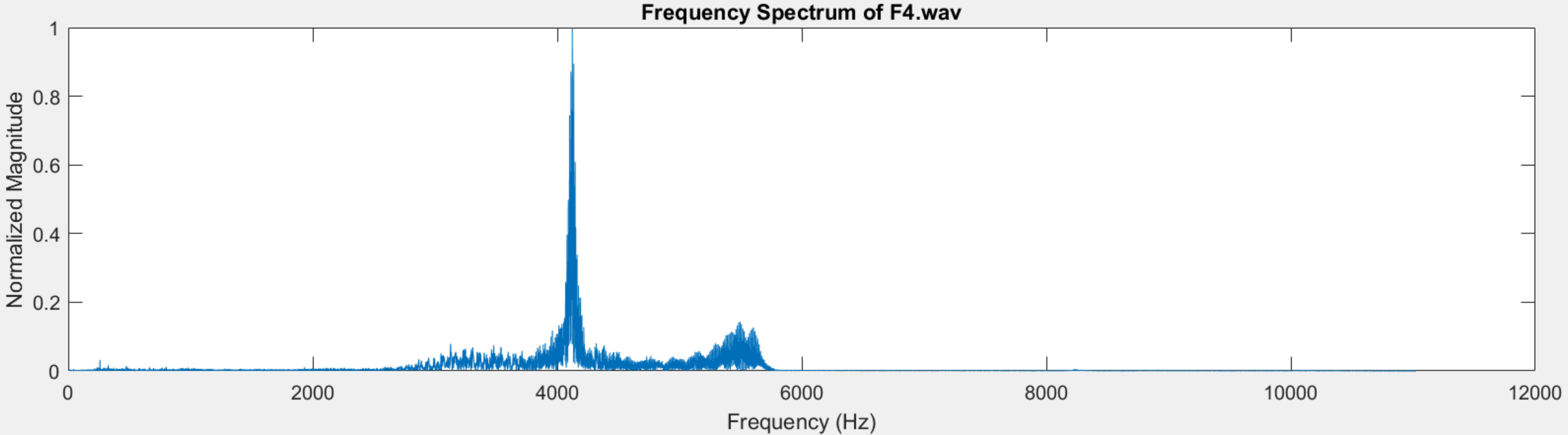

Waveform of bird3.wav
Amplitude
Time (s)
1
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
3
Frequency Spectrum of bird3.wav
Normalized Magnitude
Frequency (Hz)
1
0.8
0.6
0.4
0.2
0
0
2000
4000
6000
8000
10000
12000

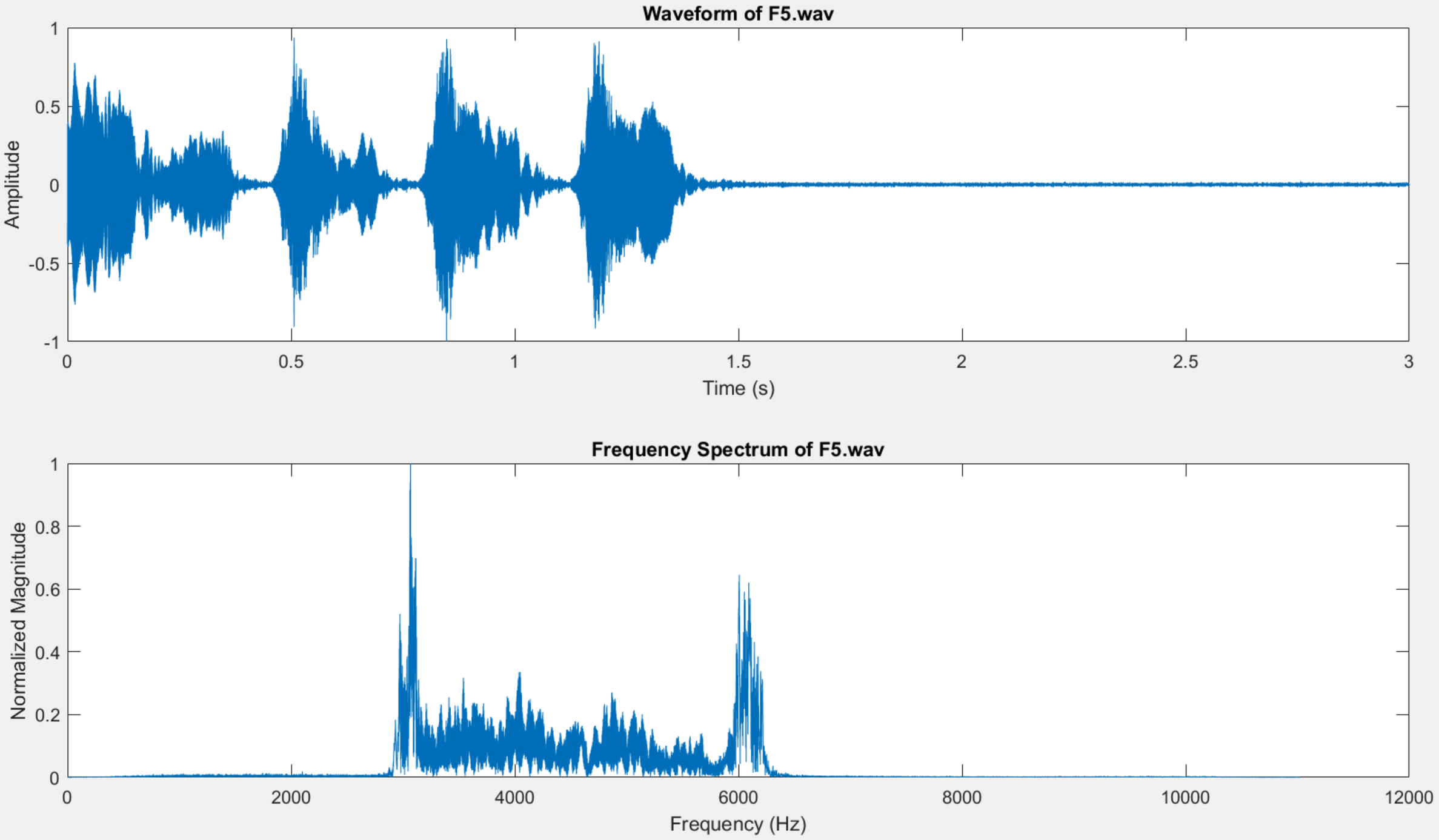

Waveform of F1.wav
Amplitude
Time (s)
1
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
3
Frequency Spectrum of F1.wav
Normalized Magnitude
Frequency (Hz)
1
0.8
0.6
0.4
0.2
0
0
2000
4000
6000
8000
10000
12000

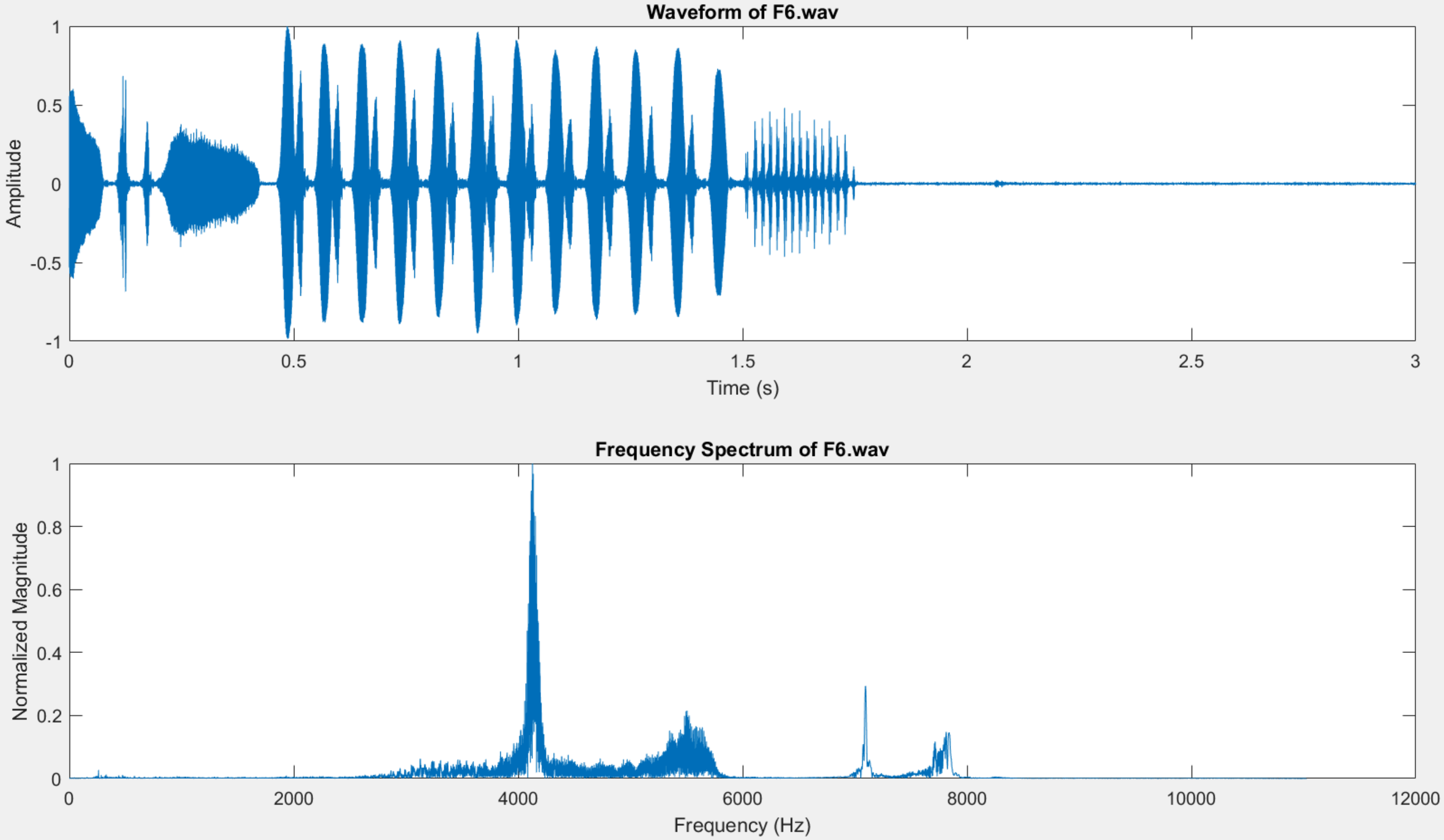

Waveform of F2.wav
X 1.46023
Y 0.312275
Amplitude
Time (s)
Frequency Spectrum of F2.wav
Normalized Magnitude
Frequency (Hz)

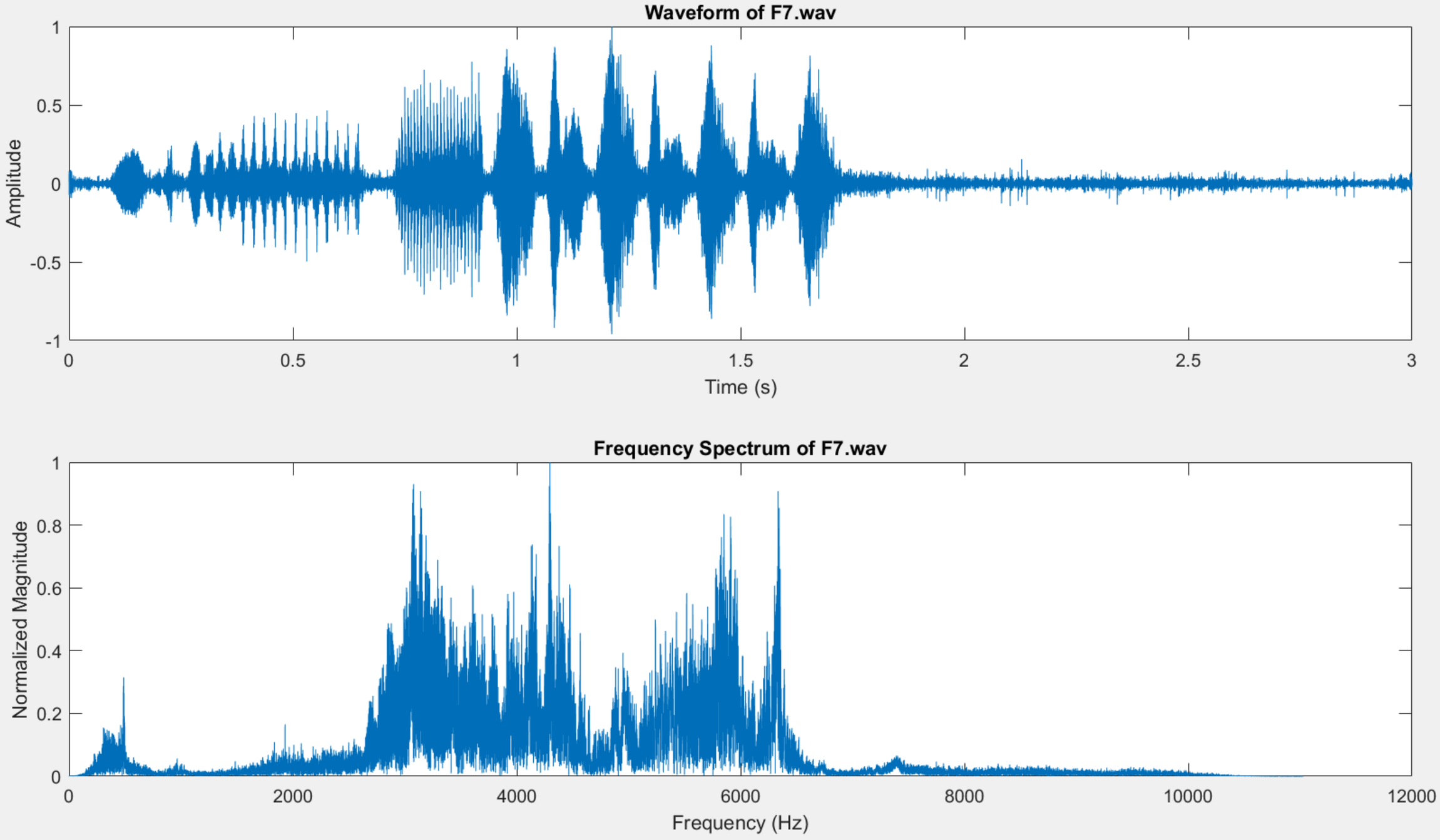

Waveform of F3.wav
Amplitude
Time (s)
1
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
3
Frequency Spectrum of F3.wav
Normalized Magnitude
Frequency (Hz)
1
0.8
0.6
0.4
0.2
0
0
2000
4000
6000
8000
10000
12000

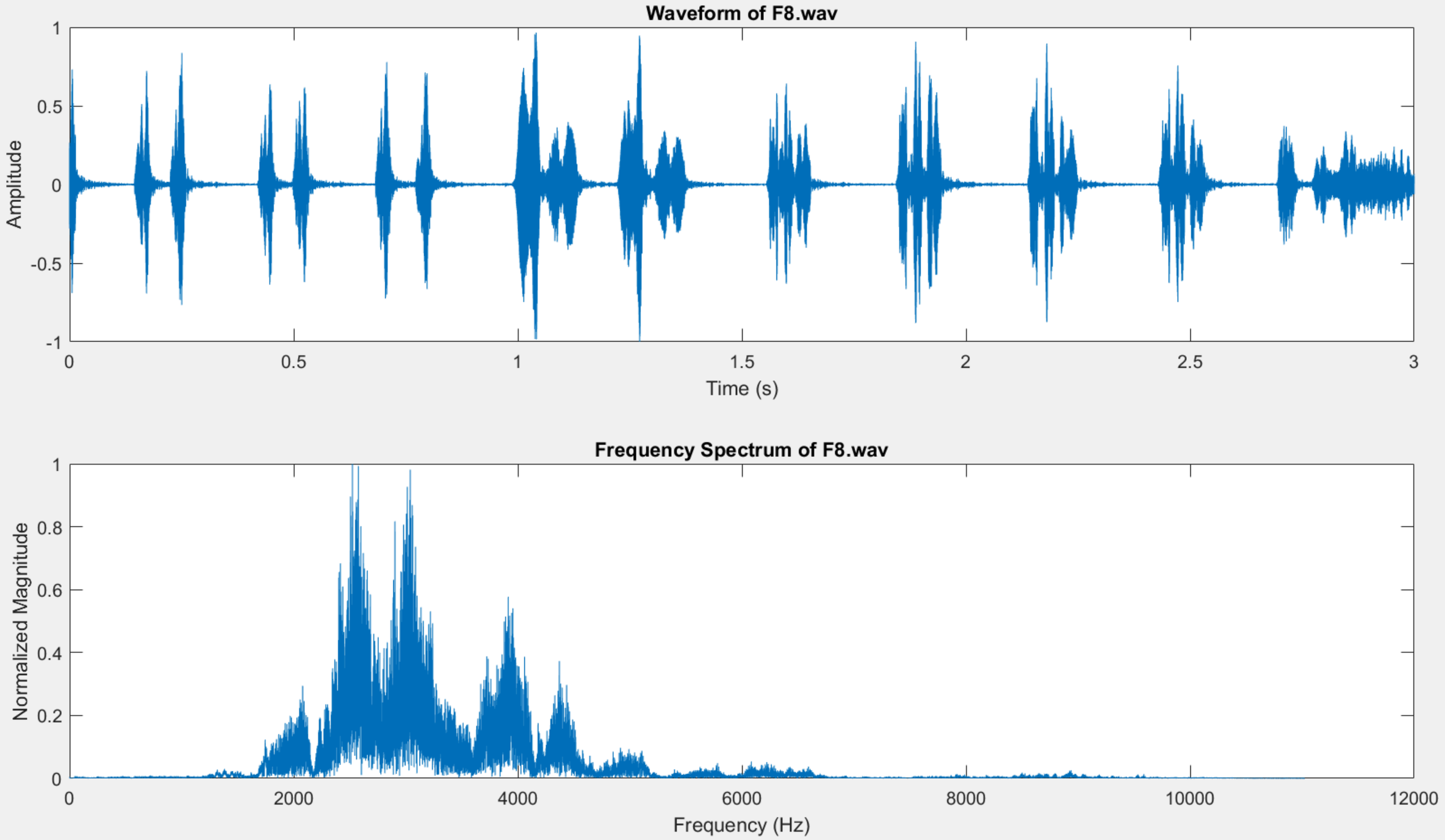

Waveform of F4.wav
Amplitude
Time (s)
1
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
3


Frequency Spectrum of F4.wav
Normalized Magnitude
Frequency (Hz)
1
0.8
0.6
0.4
0.2
0
0
2000
4000
6000
8000
10000
12000


Waveform of F5.wav
Amplitude
Time (s)
1
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
3
Frequency Spectrum of F5.wav
Normalized Magnitude
Frequency (Hz)
1
0.8
0.6
0.4
0.2
0
0
2000
4000
6000
8000
10000
12000


Waveform of F6.wav
Amplitude
Time (s)
1
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
3
Frequency Spectrum of F6.wav
Normalized Magnitude
Frequency (Hz)
1
0.8
0.6
0.4
0.2
0
0
2000
4000
6000
8000
10000
12000


Waveform of F7.wav
Amplitude
Time (s)
1
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
3
Frequency Spectrum of F7.wav
Normalized Magnitude
Frequency (Hz)
1
0.8
0.6
0.4
0.2
0
0
2000
4000
6000
8000
10000
12000


Waveform of F8.wav
Amplitude
Time (s)
1
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
3
Frequency Spectrum of F8.wav
Normalized Magnitude
Frequency (Hz)
0.8
0.6
0.4
0.2
2000
4000
6000
8000
10000
12000

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

The computeSpectrum function processes a time-domain signal to calculate its normalized magnitude spectrum in the frequency domain. It first applies the Fast Fourier Transform (FFT) to transform the signal, taking the absolute value to extract the magnitudes of the frequency components. Since real-valued signals produce symmetric FFT outputs, only the positive frequencies (first half of the spectrum) are retained. Finally, the spectrum is normalized by dividing by the maximum value, ensuring the output is scaled between 0 and 1 for consistent comparison. This function is useful for analyzing the frequency content of signals in a concise and standardized form.

```
function spectrum = computeSpectrum(signal)
    % compute the magnitude spectrum of the signal
    spectrum = abs(fft(signal));
    spectrum = spectrum(1:floor(length(spectrum)/2)); % take positive frequencies only
    spectrum = spectrum / max(spectrum); % normalize
end
```

PART 2
Heart Rate Estimation

Task-1: Find and Plot HR as a function of time

- We find the distance between the 2 peaks obtained, multiply with sampling time.
- This basically samples the signal. Now, this function is called RR value.
- Standard heart rate is defined as 60/RR_intervals.
- After that, we just plot the heart rate against time.

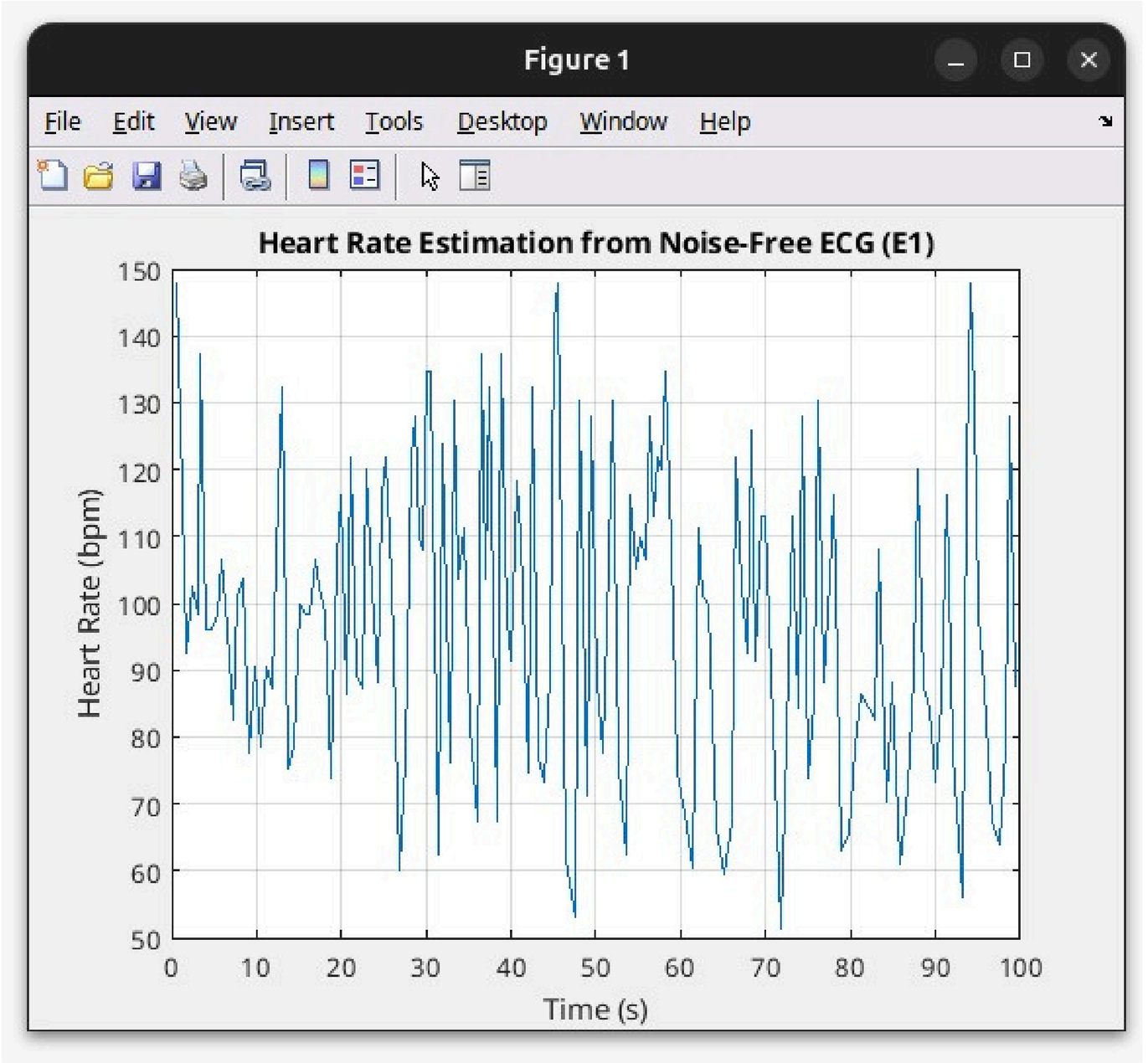


Figure 1
File Edit View Insert Tools Desktop Window Help
Heart Rate Estimation from Noise-Free ECG (E1)
Heart Rate (bpm)
150
140
130
120
110
100
90
80
70
60
50
0 10 20 30 40 50 60 70 80 90 100
Time (s)

Task-2: E2 and E3 plot for HR vs time

- First, we need to remove noise. For that, we use a Butterworth filter. This is because, Butterworth filters avoid ripples or oscillations. They also provide smooth transition from pass-band to stop band. Since typical ECG signals have frequency range from 0.5Hz - 50Hz, we choose a cutoff frequency of 50Hz.
- After applying the filter, we use same method as that of task 1.

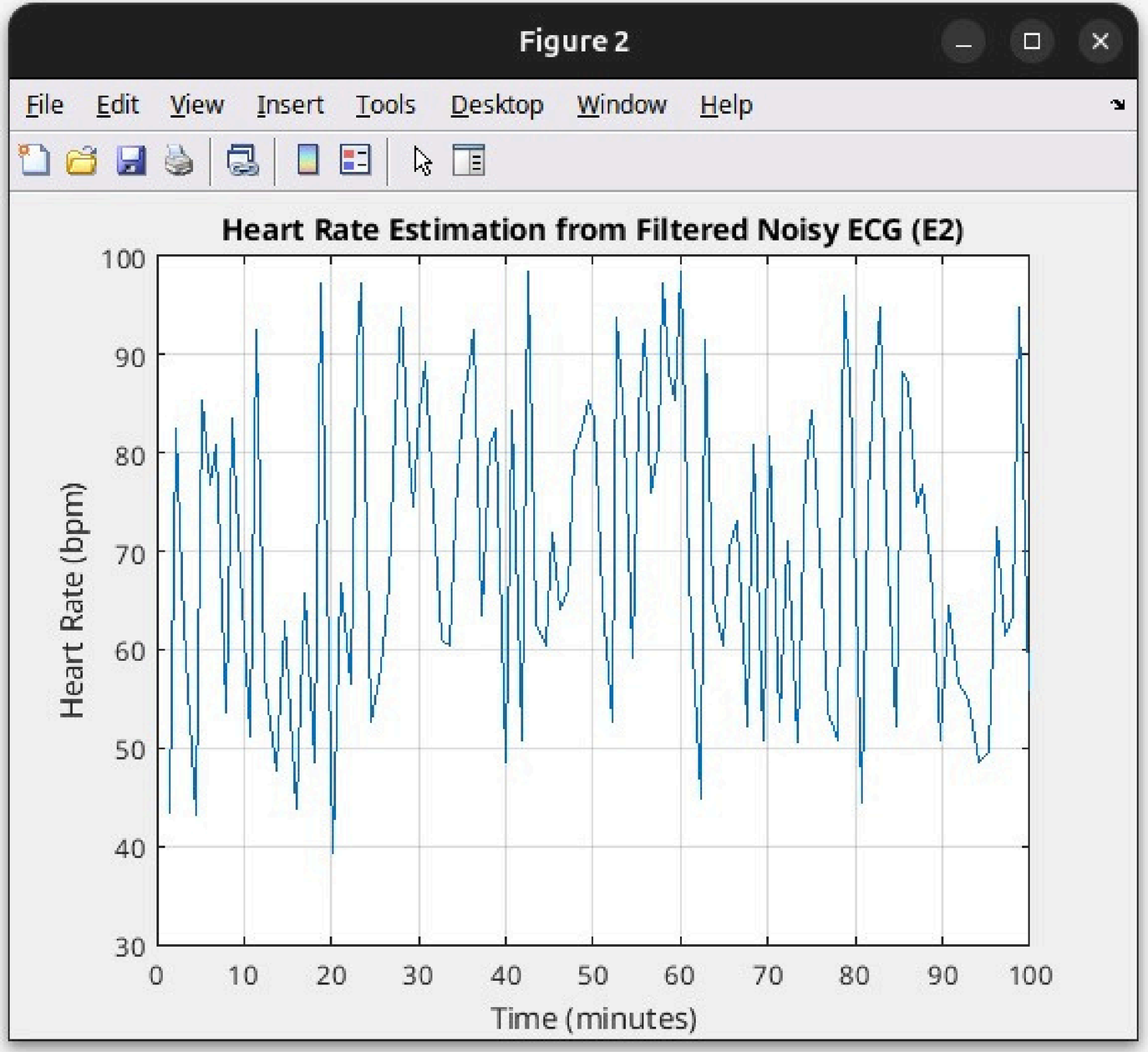


Figure 2
File Edit View Insert Tools Desktop Window Help
Heart Rate Estimation from Filtered Noisy ECG (E2)
Heart Rate (bpm)
100
90
80
70
60
50
40
30
0 10 20 30 40 50 60 70 80 90 100
Time (minutes)

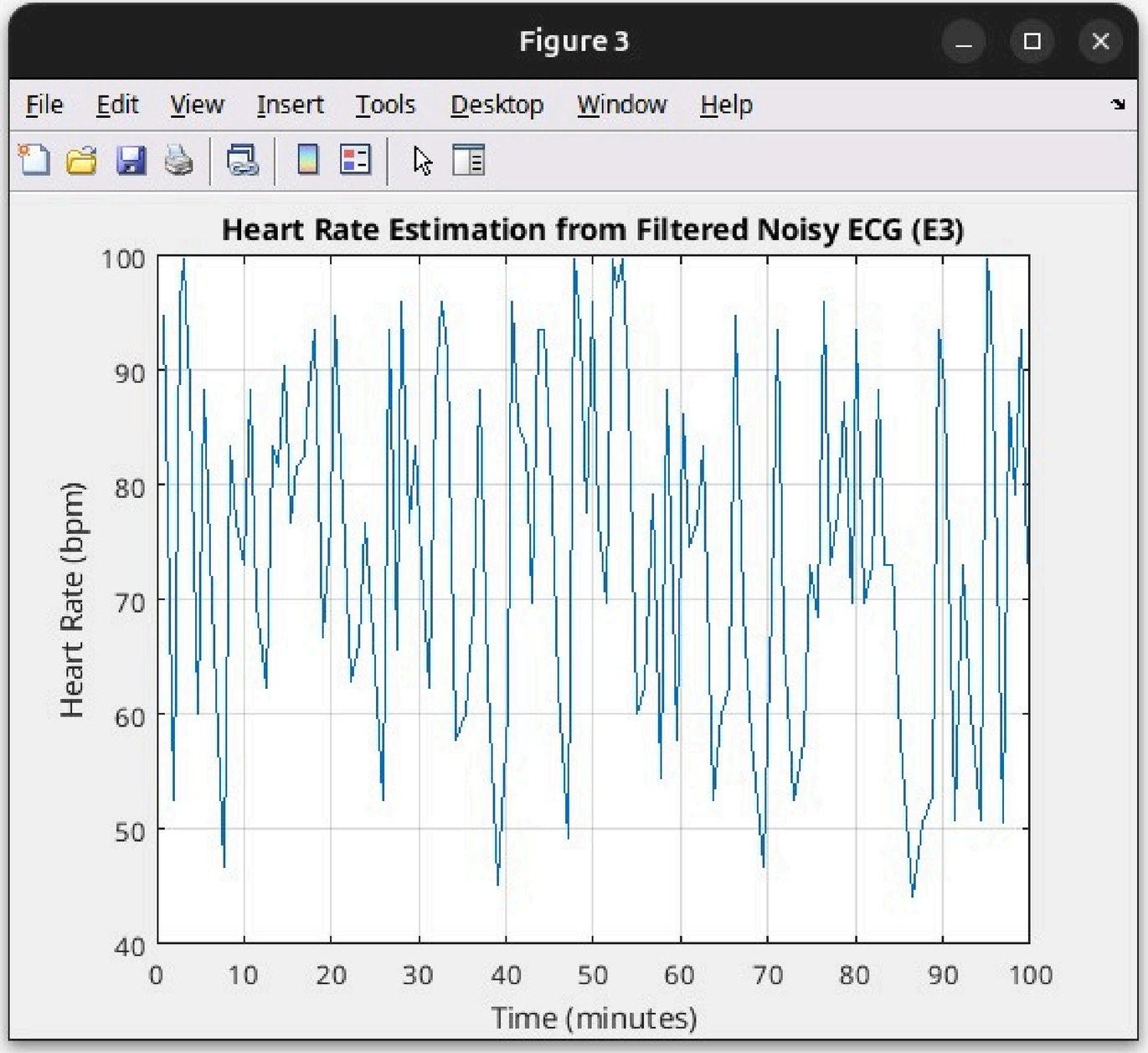


Figure 3
File Edit View Insert Tools Desktop Window Help
Heart Rate Estimation from Filtered Noisy ECG (E3)
Heart Rate (bpm)
100
90
80
70
60
50
40
0 10 20 30 40 50 60 70 80 90 100
Time (minutes)

PART 3
Loudness Segmentation

Task-1: Finding the louder words using start and end times

- First we loaded the audio files one by one in order to figure out a common threshold value for loudness.
- This included sampling the audio based on start and end times by multiplying with their sampling frequencies.
- After that, we extracted the audio into different word segments as per the time intervals and calculated root mean square values for each segment.
- Then we just compared each RMS value with the threshold and displayed the outputs accordingly.

Audio File 1

```
>> q3a2
word: wow
start time: 0.475139, end time: 1.093177, RMS loudness: 0.1955

word: i
start time: 1.093177, end time: 1.295022, RMS loudness: 0.0642

word: won
start time: 1.295022, end time: 1.551347, RMS loudness: 0.0940

word: it
start time: 1.551347, end time: 1.972006, RMS loudness: 0.0396

LOUD WORDS:
wow
>>
```

Audio File 2

```
>> q3a2
word: he
start time: 0.502840, end time: 0.726325, RMS loudness: 0.0680

word: always
start time: 0.726325, end time: 1.471274, RMS loudness: 0.1358

word: manages
start time: 1.555081, end time: 2.163455, RMS loudness: 0.0514

word: to
start time: 2.532826, end time: 2.638360, RMS loudness: 0.0659

word: find
start time: 2.638360, end time: 3.032562, RMS loudness: 0.0982

word: the
start time: 3.032562, end time: 3.209488, RMS loudness: 0.0677

word: best
start time: 3.209488, end time: 3.721640, RMS loudness: 0.1190

word: deals
start time: 3.721640, end time: 4.227584, RMS loudness: 0.0302

LOUD WORDS:
always
>>
```

Audio File 3

```
>> q3a2
word: i
start time: 0.452592, end time: 0.646911, RMS loudness: 0.0801

word: cant
start time: 0.646911, end time: 1.003572, RMS loudness: 0.1786

word: believe
start time: 1.003572, end time: 1.340556, RMS loudness: 0.0799

word: we
start time: 1.340556, end time: 1.426647, RMS loudness: 0.0687

word: are
start time: 1.426647, end time: 1.542255, RMS loudness: 0.1049

word: actually
start time: 1.542255, end time: 2.088316, RMS loudness: 0.1493

word: going
start time: 2.088316, end time: 2.540908, RMS loudness: 0.0717

word: to
start time: 2.540908, end time: 2.789341, RMS loudness: 0.0294

word: paros
start time: 2.789341, end time: 3.360000, RMS loudness: 0.1298

LOUD WORDS:
cant
actually
paros
>>
```

Audio File 4

```
>> q3a2
word: i
start time: 0.505534, end time: 0.724802, RMS loudness: 0.0496

word: cant
start time: 0.770483, end time: 1.269927, RMS loudness: 0.1615

word: believe
start time: 1.269927, end time: 1.839414, RMS loudness: 0.0444

word: we
start time: 1.882050, end time: 2.067818, RMS loudness: 0.0377

word: are
start time: 2.067818, end time: 2.262723, RMS loudness: 0.0716

word: actually
start time: 2.262723, end time: 2.865710, RMS loudness: 0.0387

word: going
start time: 2.865710, end time: 3.142840, RMS loudness: 0.0437

word: to
start time: 3.142840, end time: 3.328609, RMS loudness: 0.0124

word: paris
start time: 3.362108, end time: 4.022958, RMS loudness: 0.1419

LOUD WORDS:
cant
paris
>>
```

Audio File 5

```
>> q3a2
word: i
start time: 0.513851, end time: 0.838009, RMS loudness: 0.0528

word: told
start time: 0.979678, end time: 1.366266, RMS loudness: 0.2288

word: you
start time: 1.366266, end time: 1.846501, RMS loudness: 0.0463

word: this
start time: 1.846501, end time: 2.105827, RMS loudness: 0.0246

word: would
start time: 2.105827, end time: 2.264305, RMS loudness: 0.0458

word: happen
start time: 2.264305, end time: 2.895813, RMS loudness: 0.0267

LOUD WORDS:
told
>>
```

Audio File 6

```
>> q3a2
word: i
start time: 0.457101, end time: 0.721523, RMS loudness: 0.0981

word: told
start time: 0.721523, end time: 1.004392, RMS loudness: 0.1668

word: you
start time: 1.004392, end time: 1.184773, RMS loudness: 0.0937

word: this
start time: 1.184773, end time: 1.539385, RMS loudness: 0.0431

word: would
start time: 1.539385, end time: 1.781259, RMS loudness: 0.1515

word: happen
start time: 1.781259, end time: 2.357247, RMS loudness: 0.0541

LOUD WORDS:
told
would
>>
```

Audio File 7

```
>> q3a2
word: i
start time: 0.449975, end time: 0.556333, RMS loudness: 0.0405

word: didnt
start time: 0.556333, end time: 0.744504, RMS loudness: 0.1321

word: say
start time: 0.744504, end time: 1.026761, RMS loudness: 0.0708

word: he
start time: 1.186297, end time: 1.362197, RMS loudness: 0.1498

word: stole
start time: 1.362197, end time: 1.703768, RMS loudness: 0.0650

word: the
start time: 1.703768, end time: 1.818307, RMS loudness: 0.0301

word: money
start time: 1.818307, end time: 2.125108, RMS loudness: 0.0313

LOUD WORDS:
didnt
he
>>
```

Audio File 8

```
>> q3a2
word: i
start time: 0.458962, end time: 0.615380, RMS loudness: 0.0638

word: didnt
start time: 0.615380, end time: 0.965261, RMS loudness: 0.1140

word: say
start time: 0.965261, end time: 1.218411, RMS loudness: 0.0449

word: he
start time: 1.218411, end time: 1.323375, RMS loudness: 0.0404

word: stole
start time: 1.323375, end time: 1.578583, RMS loudness: 0.0497

word: the
start time: 1.578583, end time: 1.681489, RMS loudness: 0.0577

word: money
start time: 1.681489, end time: 2.033429, RMS loudness: 0.0355

LOUD WORDS:
didnt
>>
```

Audio File 9

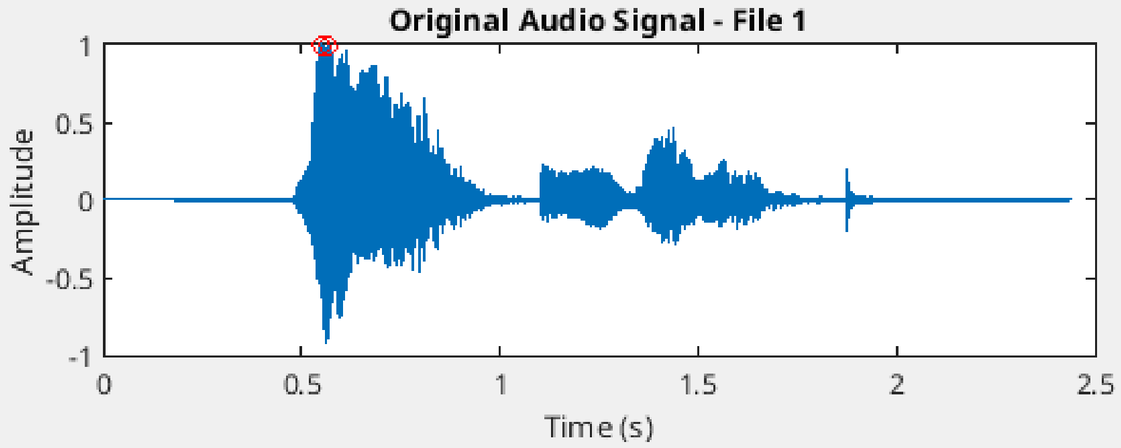
```
>> q3a2
word: i
start time: 0.463641, end time: 0.555976, RMS loudness: 0.0393

word: didnt
start time: 0.555976, end time: 0.685638, RMS loudness: 0.0988

word: say
start time: 0.685638, end time: 0.919423, RMS loudness: 0.0643

word: he
start time: 0.919423, end time: 0.999971, RMS loudness: 0.0871

word: stole
start time: 0.999971, end time: 1.701326, RMS loudness: 0.1417

word: the
start time: 1.701326, end time: 1.795626, RMS loudness: 0.0979

word: money
start time: 1.795626, end time: 2.208187, RMS loudness: 0.0301

LOUD WORDS:
stole
```

Task-2: Finding the louder words without using start and end times

- First we loaded the audio files and then stored the audio and their sampling frequency.
- Then we normalized the audio signals and took fft to remove high frequency components, if there are any and then took ifft to get the signals back to time domain and then normalized these as well.
- After that, we defined a window length (we took an arbitrary value at 200 and then to improve accuracy, changed it to 250) and defined a step size around half the window size for accuracy. This was used to calculate the number of windows.
- Then we initialized the energy vector and ran a for loop to calculate energies for each window and then normalized it and used the moving average to make the energy smoother. Then we took mean of the energies and figured out a threshold for energy.
- We assumed an initial value for threshold and then adjusted it accordingly, looking at the plots till we obtained the desired peaks.
- We then detected peaks in energy using the energy values and threshold and plotted the original signal and rms energies along with peaks to verify their functionality.

Figure 23

File Edit View Insert Tools Desktop Window Help



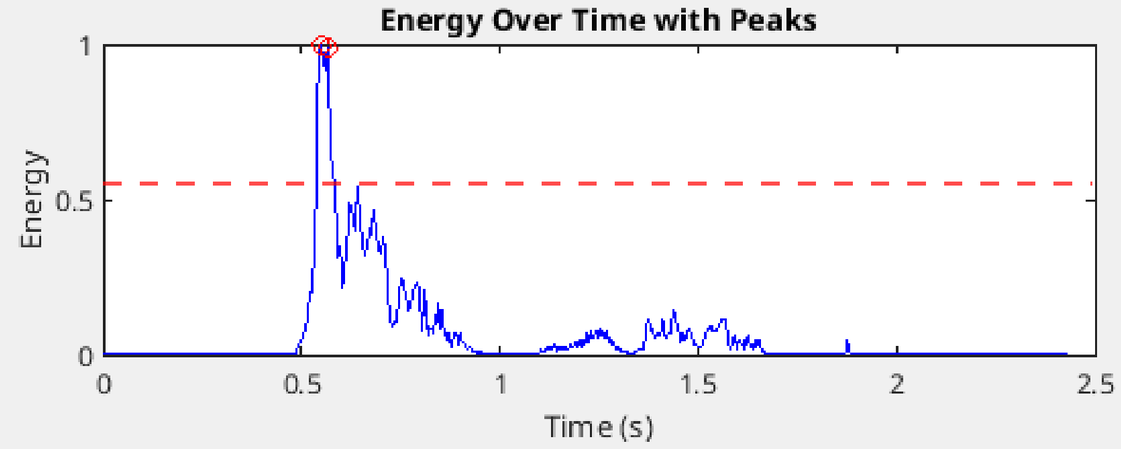
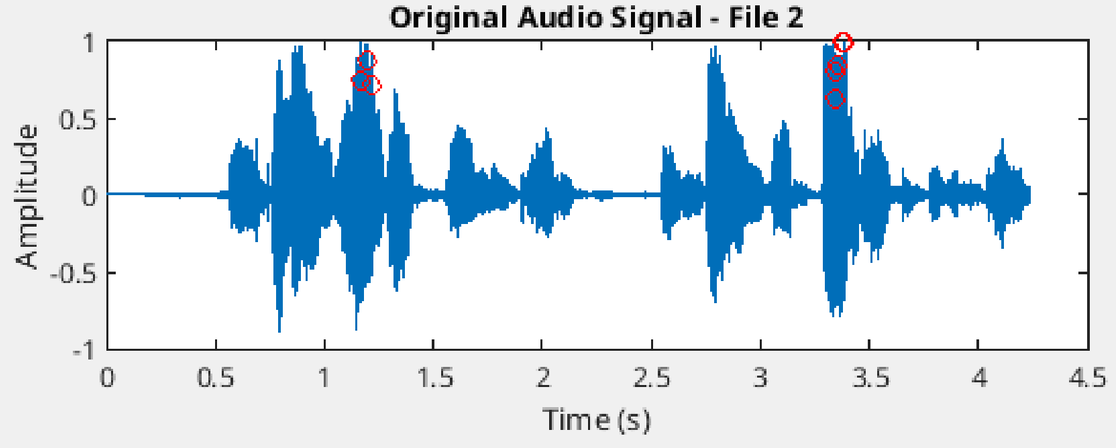


Figure 24

File Edit View Insert Tools Desktop Window Help



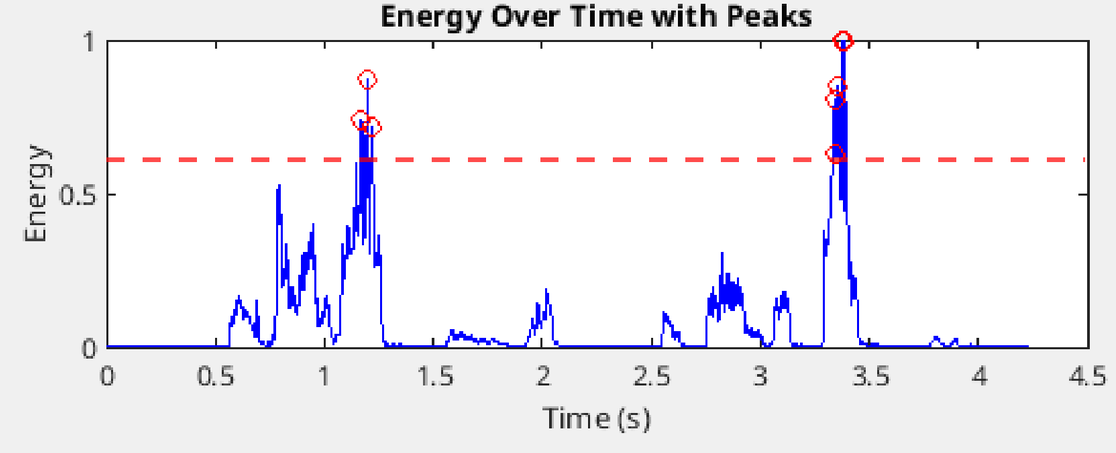
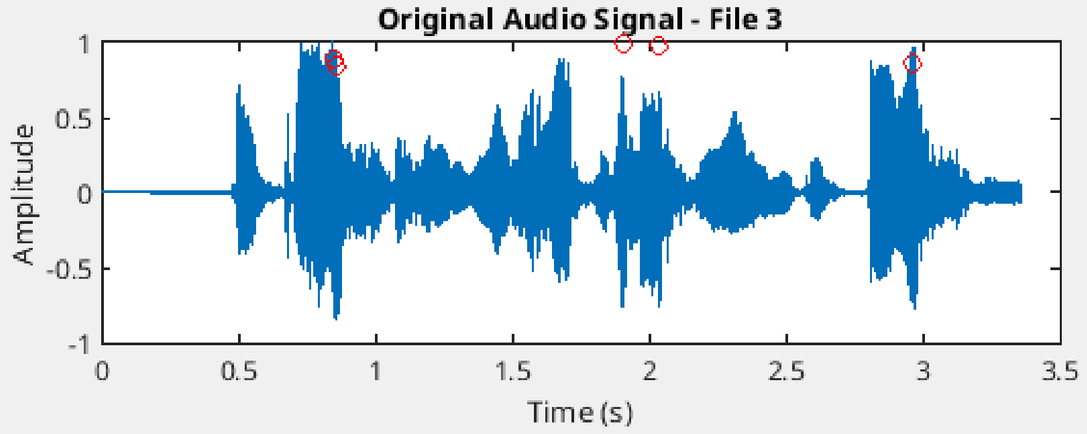


Figure 25

File Edit View Insert Tools Desktop Window Help



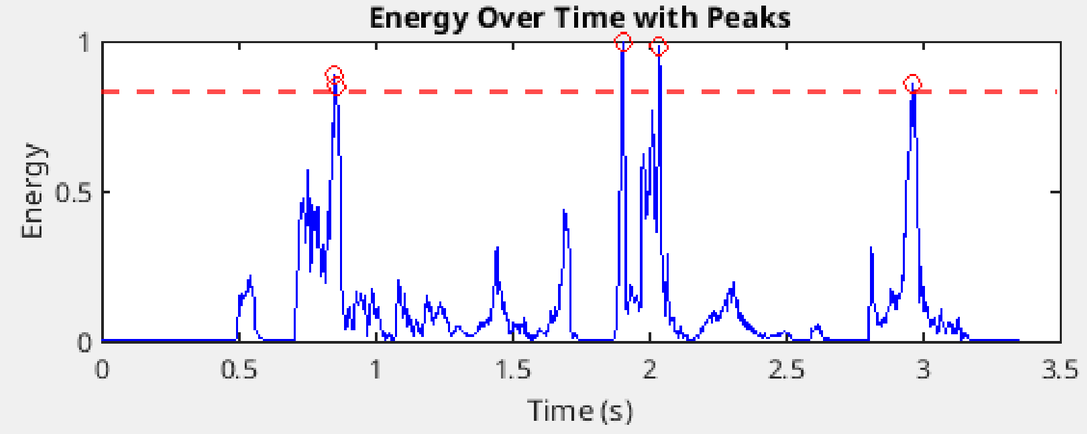
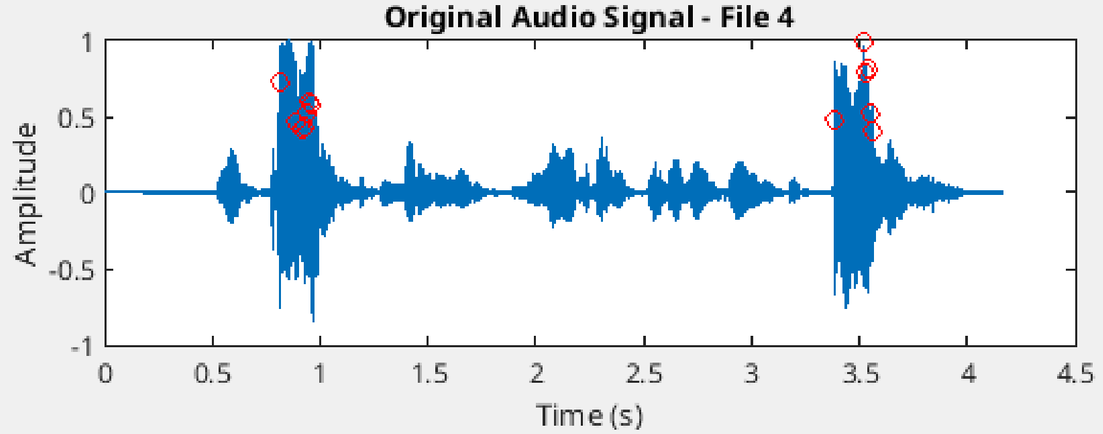
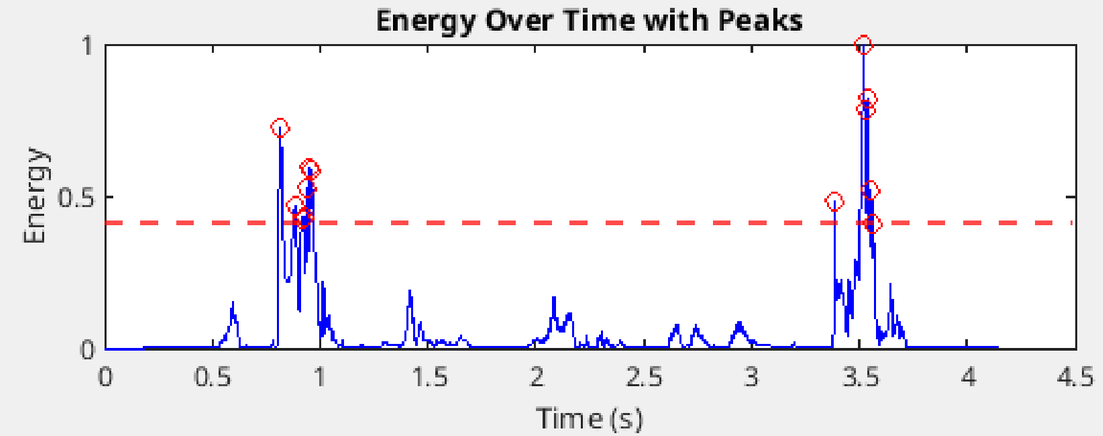


Figure 26

File Edit View Insert Tools Desktop Window Help





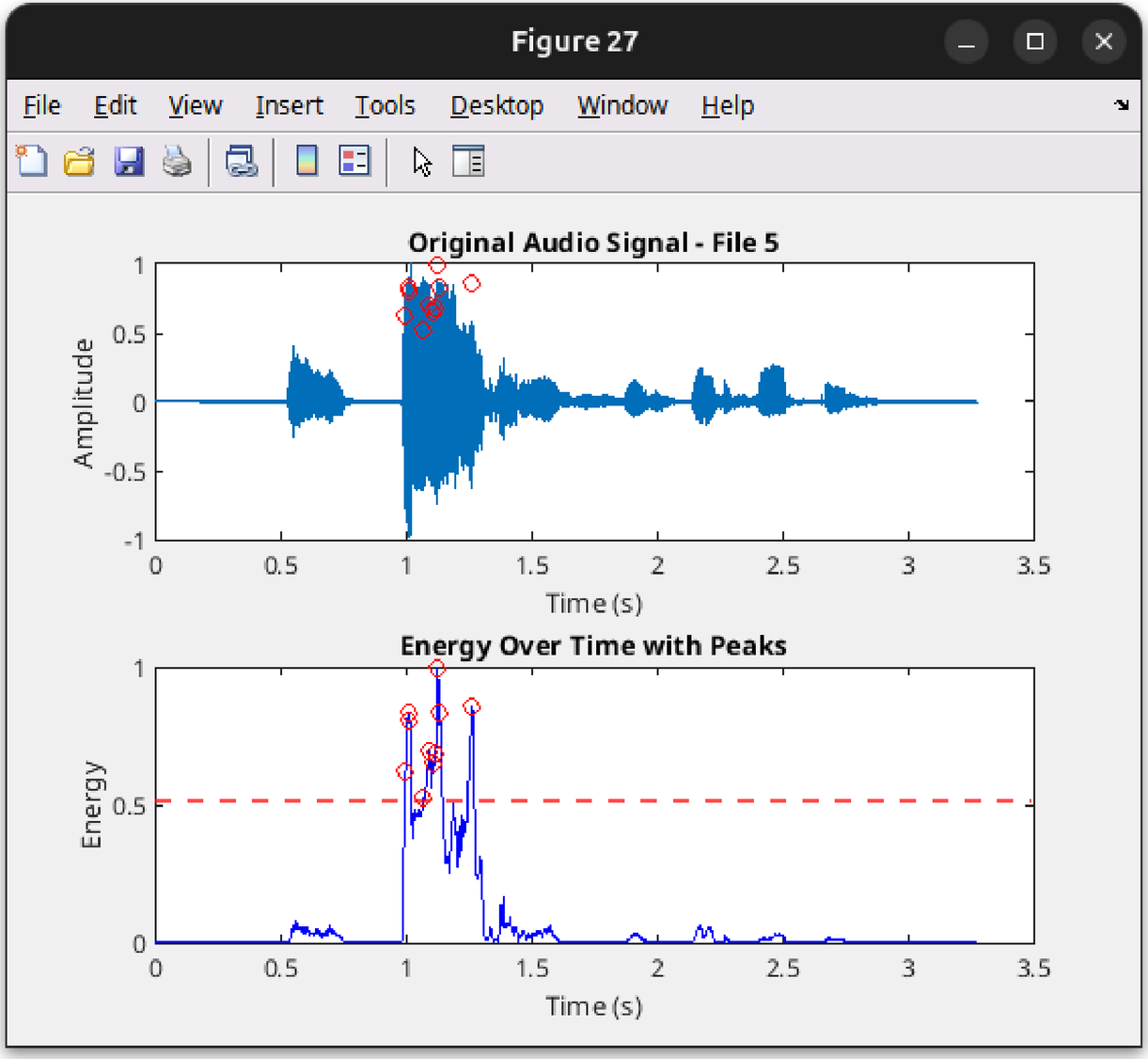


Figure 27
File Edit View Insert Tools Desktop Window Help
Original Audio Signal - File 5
Amplitude
Time (s)
Energy Over Time with Peaks
Energy
Time (s)

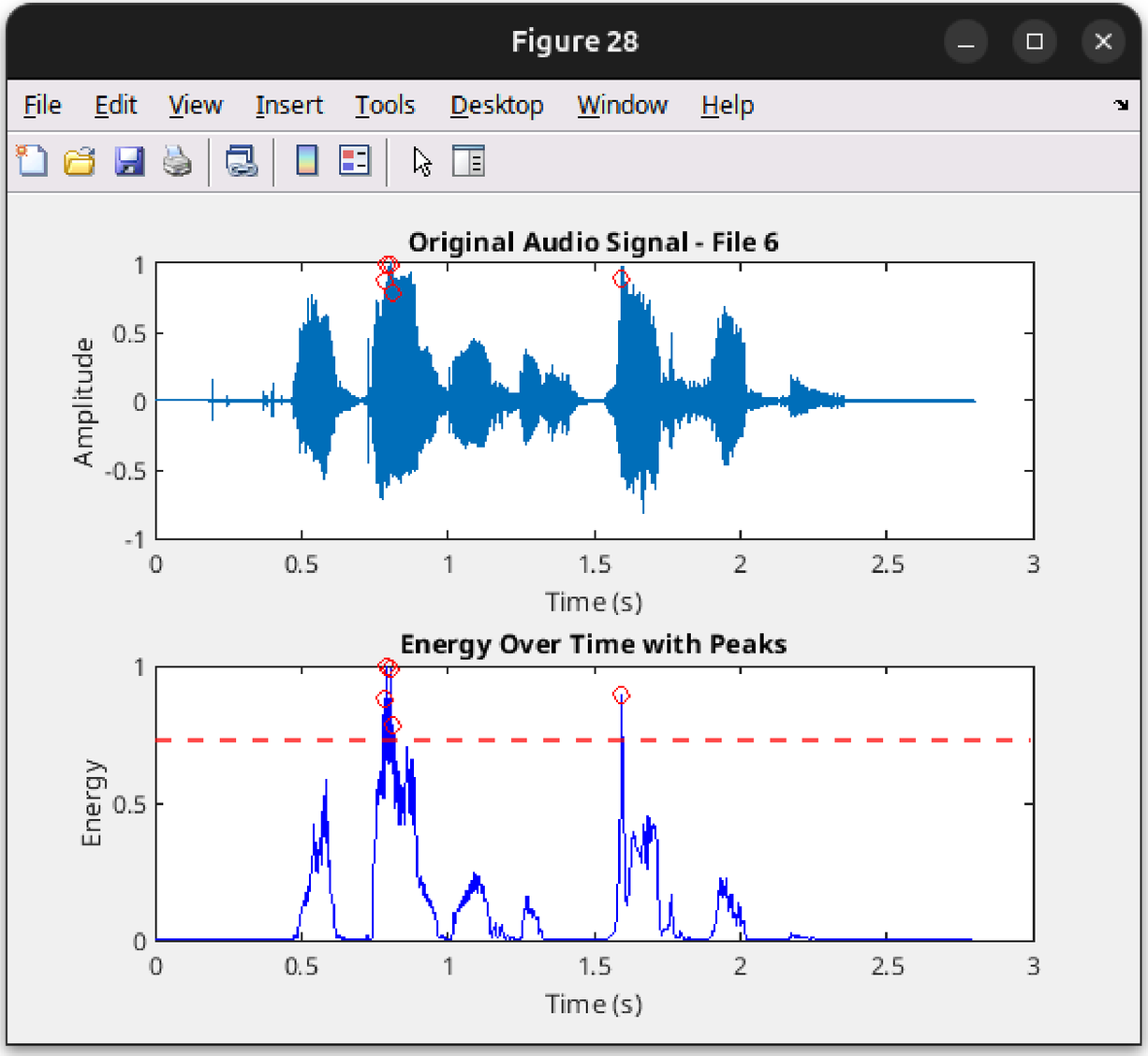


Figure 28
File Edit View Insert Tools Desktop Window Help
Original Audio Signal - File 6
Amplitude
Time (s)
Energy Over Time with Peaks
Energy
Time (s)

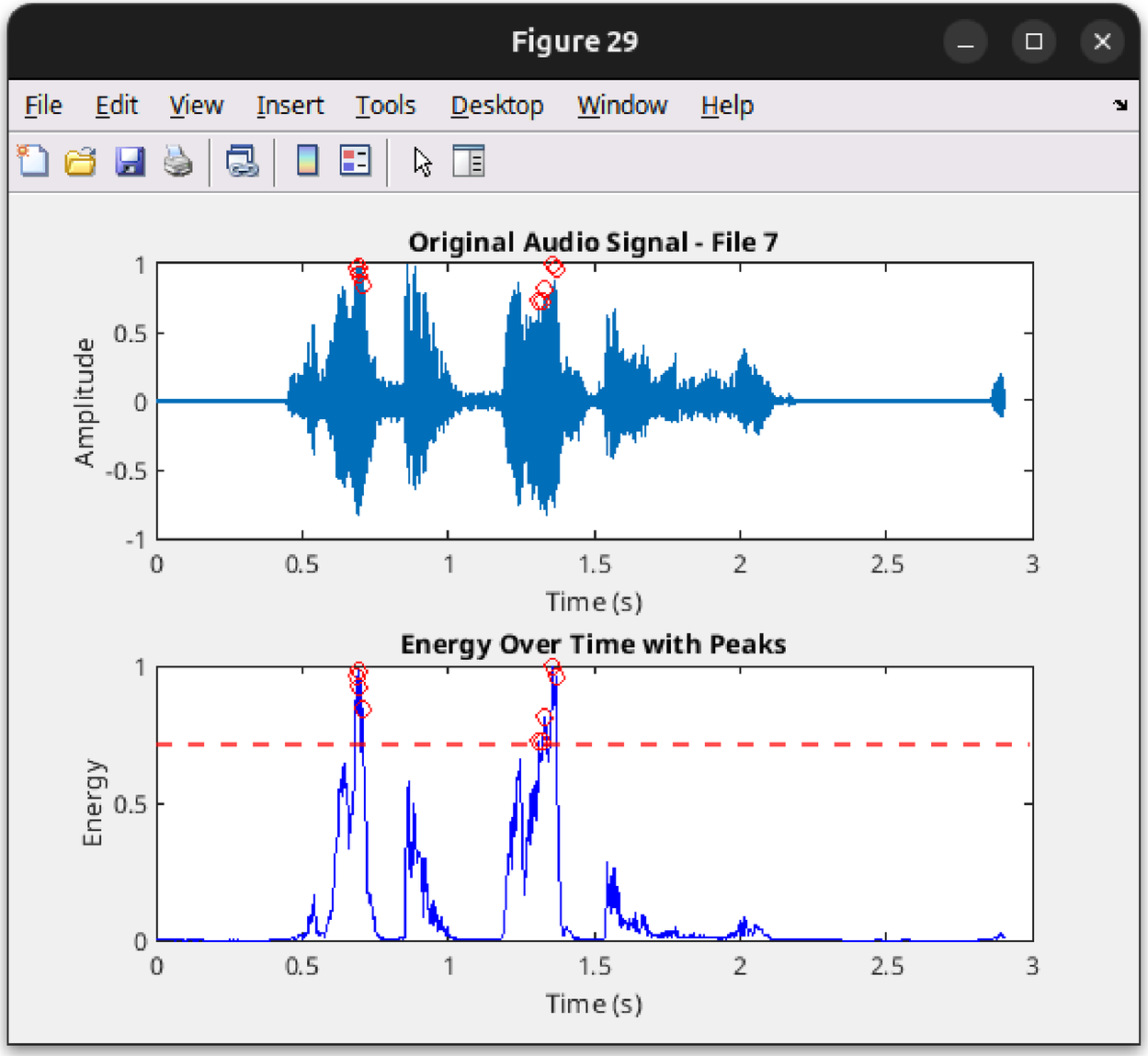


Figure 29
File Edit View Insert Tools Desktop Window Help
Original Audio Signal - File 7
Amplitude
Time (s)
Energy Over Time with Peaks
Energy
Time (s)

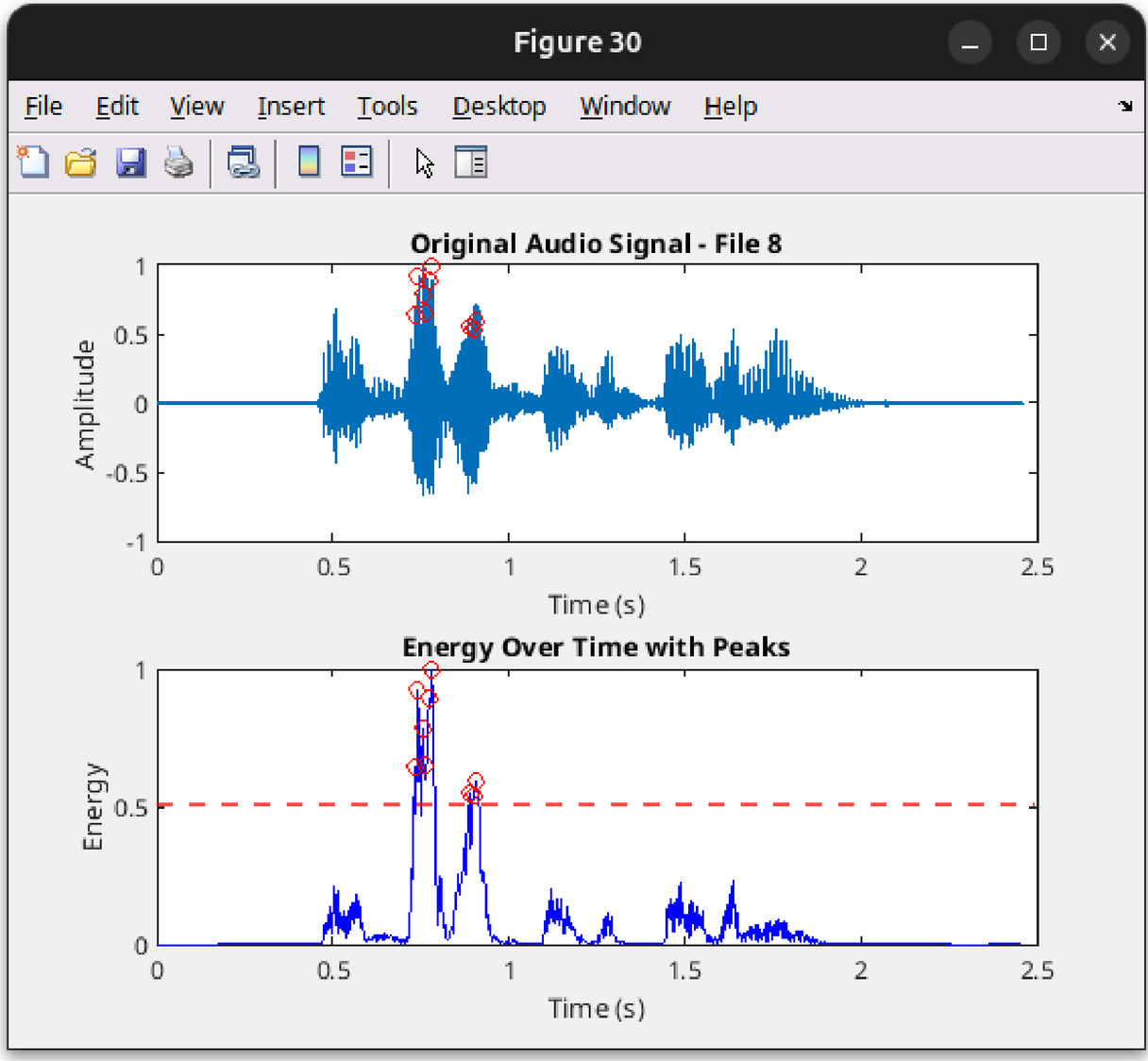


Figure 30
File Edit View Insert Tools Desktop Window Help
Original Audio Signal - File 8
Amplitude
Time (s)
Energy Over Time with Peaks
Energy
Time (s)

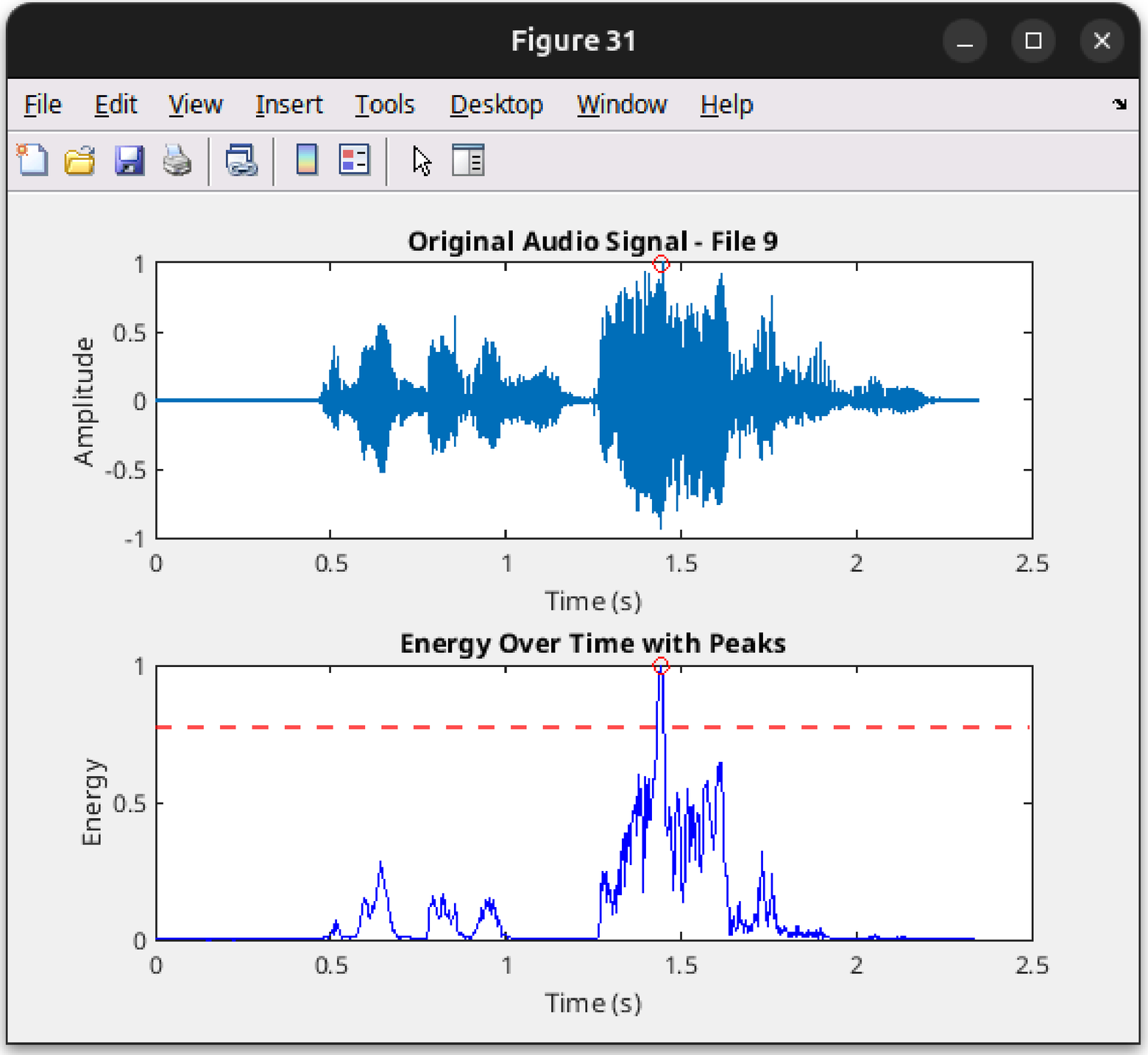


Figure 31
File Edit View Insert Tools Desktop Window Help
Original Audio Signal - File 9
Amplitude
1
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
Time (s)
Energy Over Time with Peaks
Energy
1
0.5
0
0
0.5
1
1.5
2
2.5
Time (s)

Thank You