Assignment 5

Mahdi Tabatabaei - Heliya Shakeri

Biomedical Signal and Image Processing Lab

Nastaran Nouri

November 16, 2024

Contents

Frequency Band Limitation and Noise Reduction of Signal

In this section, we explore methods to limit the frequency band of an ECG signal and reduce noise by applying various filtering techniques. The data used in this experiment is from a healthy volunteer, sampled at 250 Hz and saved in `normal.mat` with two columns representing time (in seconds) and the ECG signal (in volts).

Question 1

Analyze a typical 10-second segment from the clean ECG data in terms of frequency content using the `pwelch` function. Similarly, select a 10-second segment from the noisy data and examine its frequency content as well. Plot the power spectrum in dB for both segments, paying attention to the frequency content of the noisy and clean signals.

```
%% Section 1
clc; close all;

load('normal.mat');

% Parameters
Fs = 250;
t = normal(:,1);
t_total = length(sig) / Fs;
sig = normal(:,2);
sig2 = randn(1,1000);

%% Q1.1
% Time segments for clean and noisy parts
t_clean_start = 5;
t_clean_end = 15;
t_noisy_start = 250;
t_noisy_end = 260;

% Clean and noisy signal segments
sig_clean = sig(t_clean_start*Fs+1 : t_clean_end*Fs);
sig_noisy = sig(t_noisy_start*Fs+1 : t_noisy_end*Fs);

% Clean signal segment
figure;
subplot(2, 2, 1);
plot(t(t_clean_start*Fs+1 : t_clean_end*Fs), sig_clean);
title('Clean Signal Segment');
xlabel('Time (s)');
ylabel('Amplitude (V)');

% Noisy signal segment
subplot(2, 2, 2);
plot(t(t_noisy_start*Fs+1 : t_noisy_end*Fs), sig_noisy);
title('Noisy Signal Segment');
xlabel('Time (s)');
ylabel('Amplitude (V)');

% PSD for clean signal segment
subplot(2, 2, 3);
window = gausswin(128);
noverlap = 64;
nfft = 128;
```

```
    [p,f] = pwelch(sig_clean', window, noverlap, nfft, Fs);
    plot(f,db(p))
    title('Power Spectrum of Clean Signal Segment');
    xlabel('Frequency (Hz)');
    ylabel('Power/Frequency (dB/Hz)');
    xlim([0,120]);

    % PSD for noisy signal segment
    subplot(2, 2, 4);
    [p, f] = pwelch(sig_noisy', window, noverlap, nfft, Fs);
    plot(f,db(p))
    title('Power Spectrum of Noisy Signal Segment');
    xlabel('Frequency (Hz)');
    ylabel('Power/Frequency (dB/Hz)');
    xlim([0,120]);
```

Source Code 1: Frequency Analysis of Clean and Noisy ECG Segments

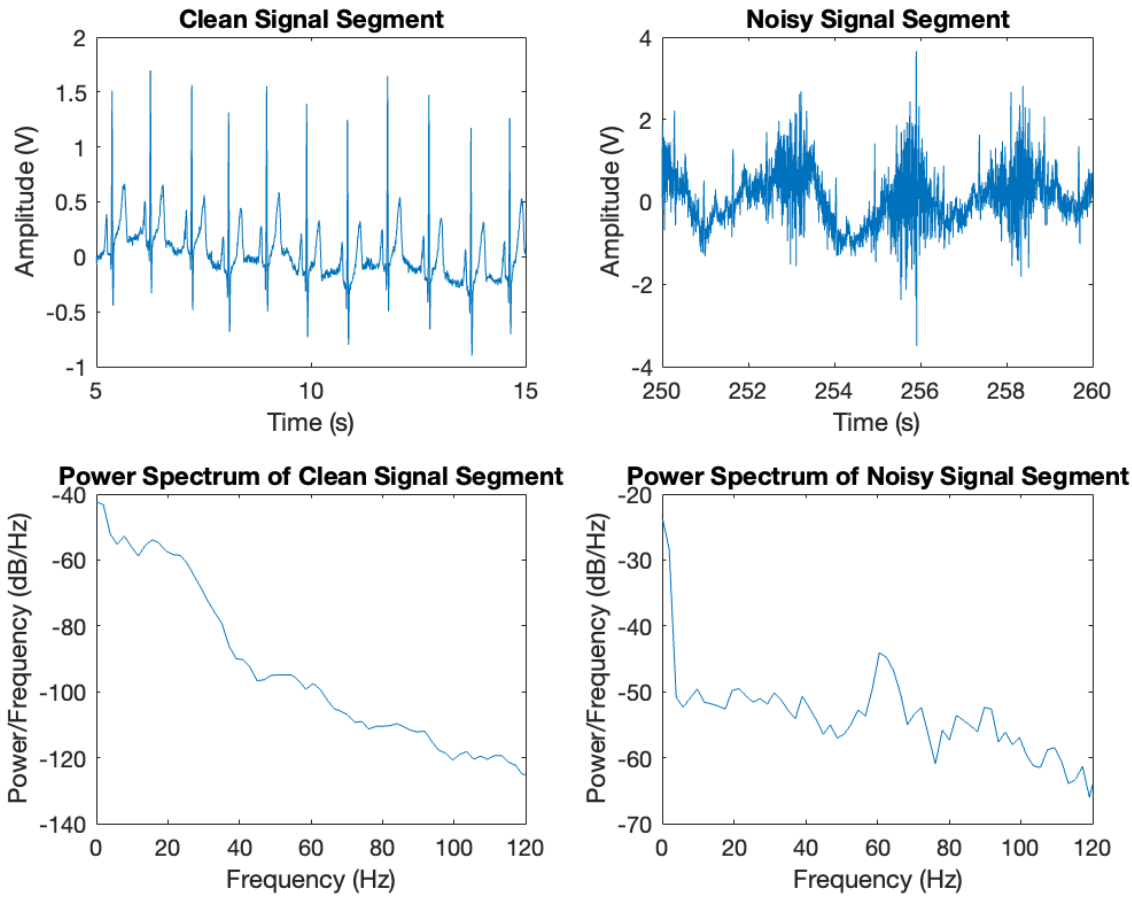

Figure 1: Power Spectral Density of Signals

Soloution

Upon analyzing the noisy segment of the signal, it appears that EMG artifacts have affected the ECG signal. Additionally, observing the frequency domain, we can see that the noisy segment contains more energy in the higher frequencies.

Question 2

Design a bandpass filter for frequency limitation of the signal. The filter should remove baseline fluctuations and reduce high-frequency noise while preserving energy within the main frequency range of the signal. Avoid sharp transition bands when selecting the filter's lower and upper cutoff frequencies. Use the `plot` function instead of `stem` to display the impulse response of the filter.

```matlab
%% Q1.2
n = size(normal, 1);

% Frequency limits
low_freq = 1 * n / Fs;
high_freq = 90 * n / Fs;

% FFT of the signal and isolate positive frequencies
normal_fft = abs(fftshift(fft(normal(:, 2))));
normal_half = normal_fft(n/2+1:n);

% Power of the signal
normal_pwr = sum(normal_half(low_freq:n/2).^2);

% Bandstop filter to attenuate frequencies between 1 and 90 Hz
[b, a] = butter(3, [1/(Fs/2) 40/(Fs/2)], 'stop');
normal_filtered = filter(b, a, normal_half(low_freq:n/2));

% Power of the filtered signal
filter_pwr = sum(normal_filtered.^2);

% Frequency response of the filter
figure;
subplot(2, 1, 1);
freqz(b, a, 50);
title("Frequency Response of Bandstop Filter");
grid minor;

% Impulse response of the filter
subplot(2, 1, 2);
impz(b, a, 50);
title("Impulse Response of Bandstop Filter");
grid minor;
```

Source Code 2: Bandpass Filter Design and Impulse Response

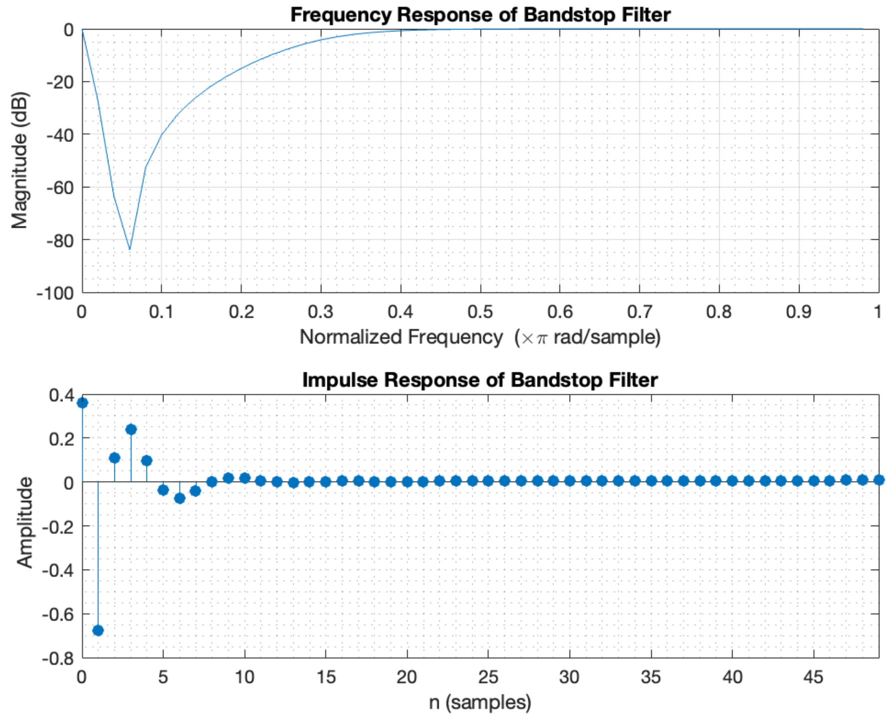

Figure 2: Frequency and Impulse Responses of the Designed Bandstop Filter

Soloution

To design a filter with a linear phase response and avoid phase distortion, we use an FIR filter. This filter is designed using the `filterDesigner` tool in MATLAB.

Question 3

Filter the designed segments (each 10-second segment) of the clean and noisy ECG data, and show how much noise is removed by the filter by plotting the power spectrum of the filtered signal.

```
%% Q1.3
% Filter the clean and noisy segments
clean_filtered = filter(b, a, clean_signal);
noisy_filtered = filter(b, a, noisy_signal);

figure;
% PSD of filtered clean signal
subplot(2, 1, 1);
[p, f] = pwelch(clean_filtered', [], [], [], Fs);
plot(f,db(p))
title("Clean ECG Signal (Filtered)");
grid minor;
xlim([0 120]);
% PSD of filtered noisy signal
subplot(2, 1, 2);
[p, f] = pwelch(noisy_filtered', [], [], [], Fs);
plot(f,db(p))
title("Noisy Signal (Filtered)");
grid minor;
xlim([0 120]);
```

Source Code 3: Evaluation of Filtered Signal Segments

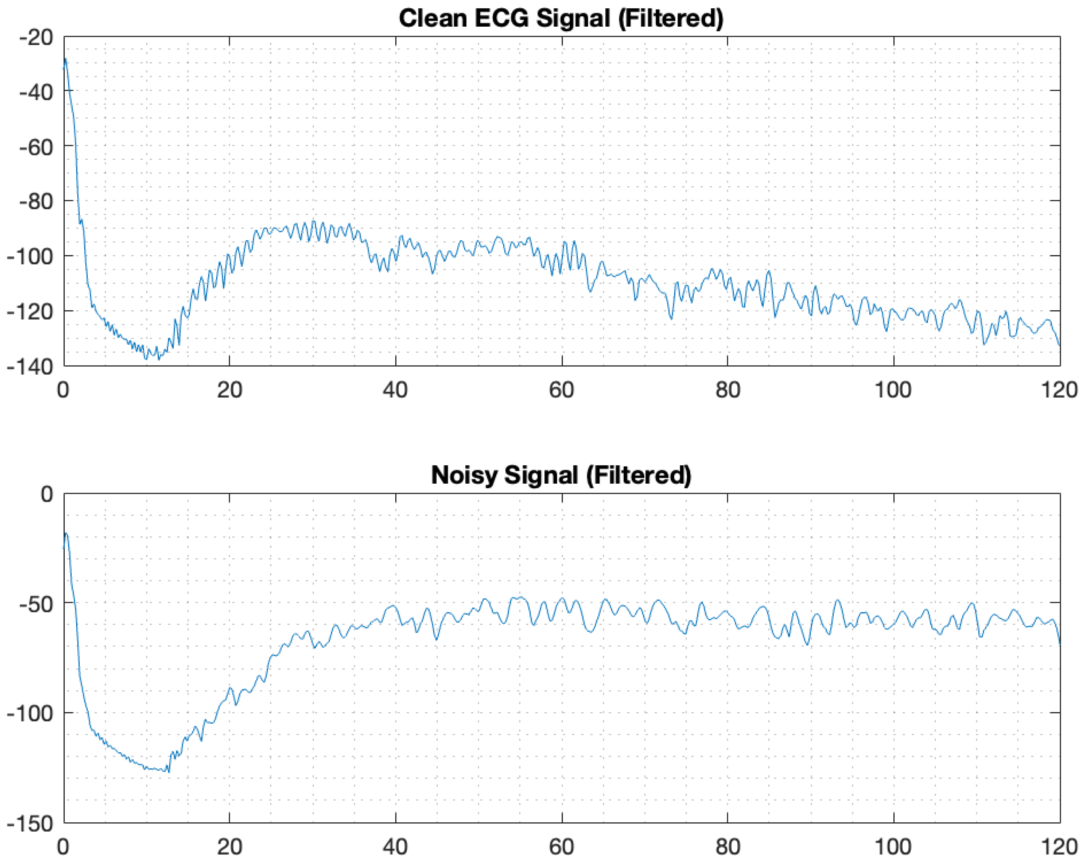

Figure 3: Power Spectral Density of Filtered Clean and Noisy ECG Signals

Soloution

Based on the plots, we observe that the designed filter performs effectively in removing noise from the signal. Both filtered outputs display similar patterns, indicating that the filter successfully attenuated unwanted noise components.

Detection of Cardiac Arrhythmias

In this section, a system is designed for detecting abnormal cardiac arrhythmias. Non-normal ECG segments are taken from the MIT-BIH arrhythmia database, recorded in a hospital setting. The dataset contains 10-minute recordings sampled at 250 Hz with 12-bit quantization.

Question 1

To design the detection system, certain criteria must be established to distinguish between normal and abnormal ECGs. Select a few files from the list below and load them in MATLAB to read and plot the ECG signals. Identify which segment corresponds to normal or abnormal (arrhythmic) signals for each pair and analyze their frequency content using the 'pwelch' function. Compare the normal and arrhythmic segments and note observations for identifying arrhythmias.

- n_422.mat - episode of ventricular fibrillation
- n_424.mat - episode of ventricular fibrillation
- n_426.mat - ventricular fibrillation and low frequency noise
- n_429.mat - ventricular flutter (2 episodes) and ventricular tachycardia
- n_430.mat - ventricular flutter and ventricular fibrillation
- n_421.mat - normal sinus rhythm with noise
- n_423.mat - atrial fibrillation and noise

```
%% 2-1
load("n_422.mat");
fs=250;

normal_1=n_422(8210:10710);
abnormal_1=n_422(11442:13942);

normal_2=n_422(1:2500);
abnormal_2=n_422(57211:59711);

t_normal_1=0:1/fs:length(normal_1)/fs-1/fs;
t_abnormal_1=0:1/fs:length(abnormal_1)/fs-1/fs;

t_normal_2=0:1/fs:length(normal_2)/fs-1/fs;
t_abnormal_2=0:1/fs:length(abnormal_2)/fs-1/fs;

figure
subplot(4,1,1)
pwelch(normal_1)
title("normal 1")
subplot(4,1,2)
pwelch(abnormal_1)
title("abnormal 1")
subplot(4,1,3)
pwelch(normal_2)
title("normal 2")
subplot(4,1,4)
pwelch(abnormal_2)
```

```
title("abnormal 2")
```

Source Code 4: Classification Criteria for Normal and Abnormal ECG

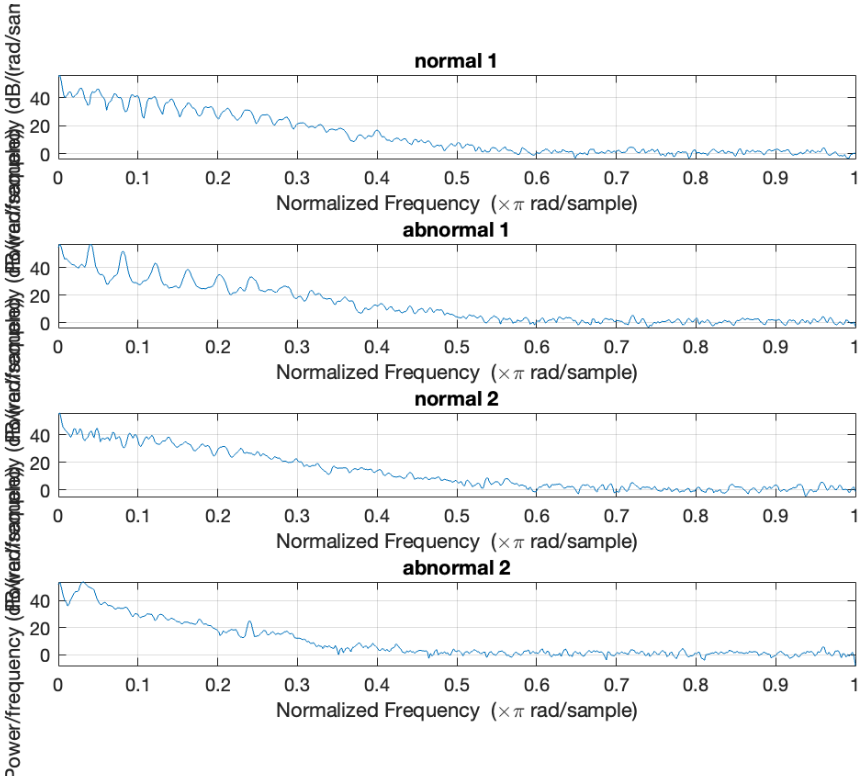

Figure 4: Power Spectral Density of Normal and Abnormal Signals

Soloution

The abnormal ECG signal has more energy at higher frequencies.

Question 2

Identify the differences between normal and arrhythmic segments in both time and frequency domains. Use appropriate plots to compare these differences.

```
% 2-2
figure
subplot(2,1,1)
plot(t_normal_1,normal_1);
xlabel('time(second)')
title("normal 1")
subplot(2,1,2)
plot(t_abnormal_1,abnormal_1);
xlabel('time(second)')
title("abnormal 1")

figure
subplot(2,1,1)
plot(t_normal_2,normal_2);
xlabel('time(second)')
title("normal 2")
subplot(2,1,2)
```

```
    plot(t_abnormal_2,abnormal_2);
    xlabel('time(second)')
    title("abnormal 2")
```

Source Code 5: Differences between Normal and Arrhythmic Segments

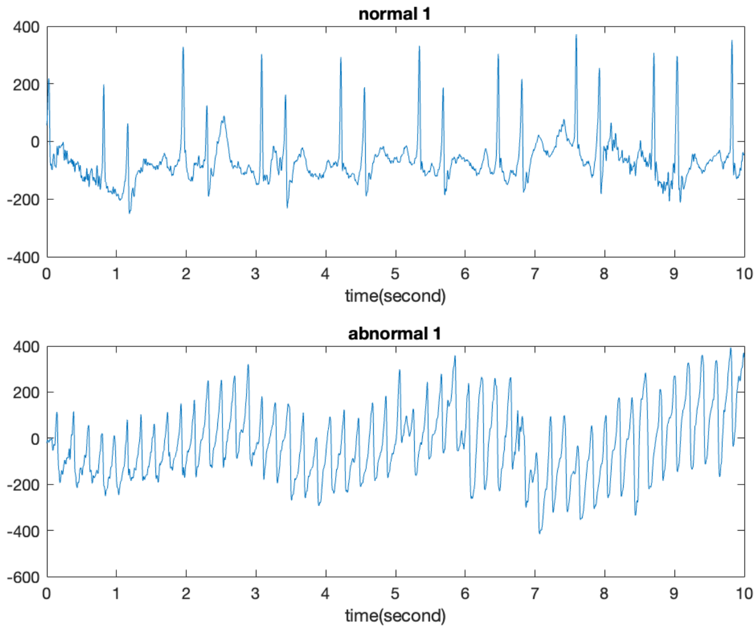

Figure 5: Segmentation of first Normal and Abnormal Signals

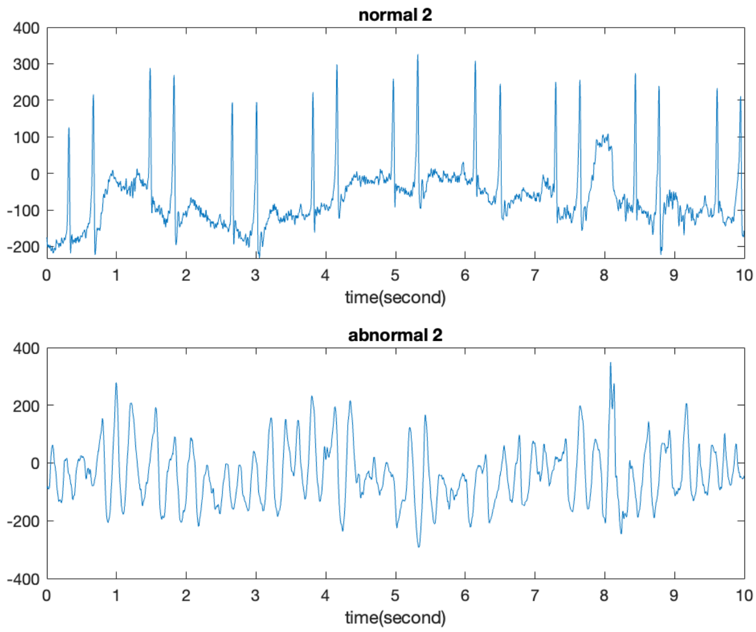

Figure 6: Segmentation of second Normal and Abnormal Signals

Soloution

In the arrhythmic time domain, there is no distinct regular pattern or rhythm. In the frequency domain, there is also higher energy at the high frequencies in the arrhythmic segments.

Question 3

Label each signal segment using predefined windows for different time frames and assign real labels to evaluate performance. Use 5-second windows with a 50% overlap and label each window as per the provided standards.

Assign labels for each window as follows:

1. 1: Normal
2. 2: VFIB (Ventricular Fibrillation)
3. 3: VT (Ventricular Tachycardia)
4. 4: Noise
5. 0: None

```
%% 2-3
labels=zeros(59,11);
for i=1:59
    t_start=(i-1)*5*250;
    t_end=((i-1)*5+10)*250;
    labels(i,2)=t_end;
    if((t_start>=1 && t_end<10711))
        labels(i,1)=1;
    elseif((t_start>=10711 && t_end<11211))
        labels(i,1)= 3;
    elseif((t_start>=11211 && t_end<11441))
        labels(i,1)= 1;
    elseif((t_start>=11442 && t_end<59711))
        labels(i,1)= 3;
    elseif((t_start>=59711 && t_end<61288))
        labels(i,1)= 4;
    elseif((t_start>=61289))
        labels(i,1)= 2;
    else
        labels(i,1)= 0;
    end
end
```

Source Code 6: Labeling of Signal Windows

Question 4

For classifying the VFIB and Normal windows in the n_422.mat signal, examine frequency-based features within these windows. Use different features (e.g., bandpower, medfreq, meanfreq) to analyze and select distinguishing characteristics.

```
%% 2-4
d = designfilt('bandpassiir', 'FilterOrder', 2, 'HalfPowerFrequency1', 10,
 'HalfPowerFrequency2', 30, 'DesignMethod', 'butter', 'SampleRate', fs);
filtered_signal=filtfilt(d, n_422);
```

```
    for i=1:59
        if(labels(i,1)==1 || labels(i,1)==2)
            signal = n_422(labels(i,2)-10*250+1:labels(i,2));
            filtered_signal=filtfilt(d, signal);
            labels(i,3)=meanfreq(signal);
            labels(i,4)=medfreq(signal);
            labels(i,5)=sum(filtered_signal.^2,'all');
        end
    end

    normal_features=labels(2:7,:);
    VFIB_features=labels(51:59,:);
```

Source Code 7: Frequency-Based Feature Extraction and Classification

Question 5

Calculate morphological (time-domain shape) and statistical features for the VFIB and Normal windows. Select various features for all windows, including max, min, peak-to-peak, number of zero crossings, and variance. Plot the histograms of features for both VFIB and Normal windows and compare them to evaluate feature separability.

```
%% 2-5
figure
histogram(normal_features(:,3).','FaceColor', 'blue','BinWidth',0.005)
hold on
histogram(VFIB_features(:,3),'FaceColor', 'red','BinWidth',0.005)
legend("normal","VFIB")
title("mean frequency")
hold off

figure
histogram(normal_features(:,4).','FaceColor', 'blue','BinWidth',0.005)
hold on
histogram(VFIB_features(:,4),'FaceColor', 'red','BinWidth',0.005)
legend("normal","VFIB")
title("median frequency")
hold off

figure
histogram(normal_features(:,5).','FaceColor', 'blue','BinWidth',10^5)
hold on
histogram(VFIB_features(:,5),'FaceColor', 'red','BinWidth',10^5)
legend("normal","VFIB")
title("band power")
hold off
```

Source Code 8: Morphological and Statistical Feature Calculation

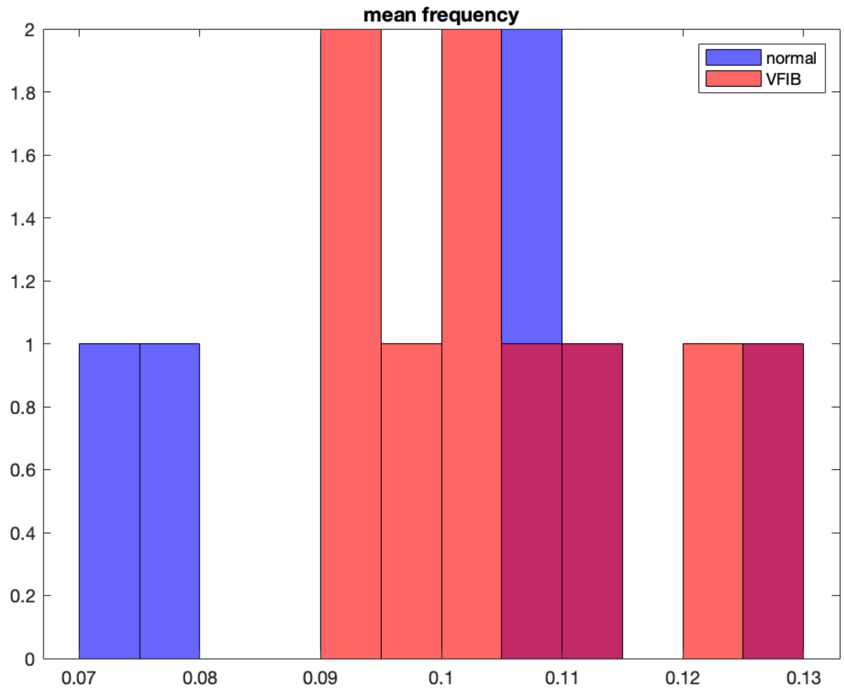

Figure 7: Mean Frequency Comparison for Normal and VFIB Segments

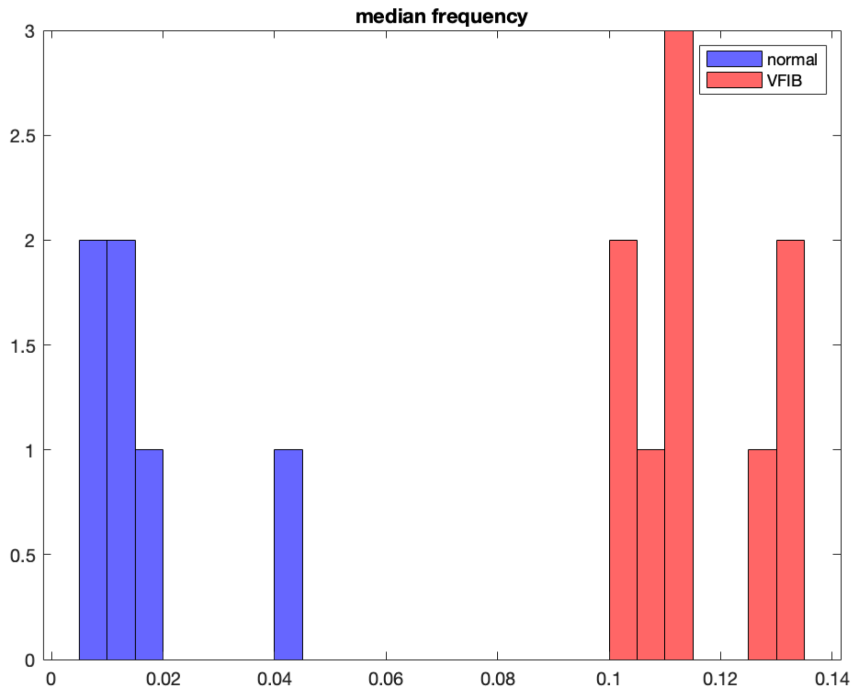

Figure 8: Median Frequency Comparison for Normal and VFIB Segments

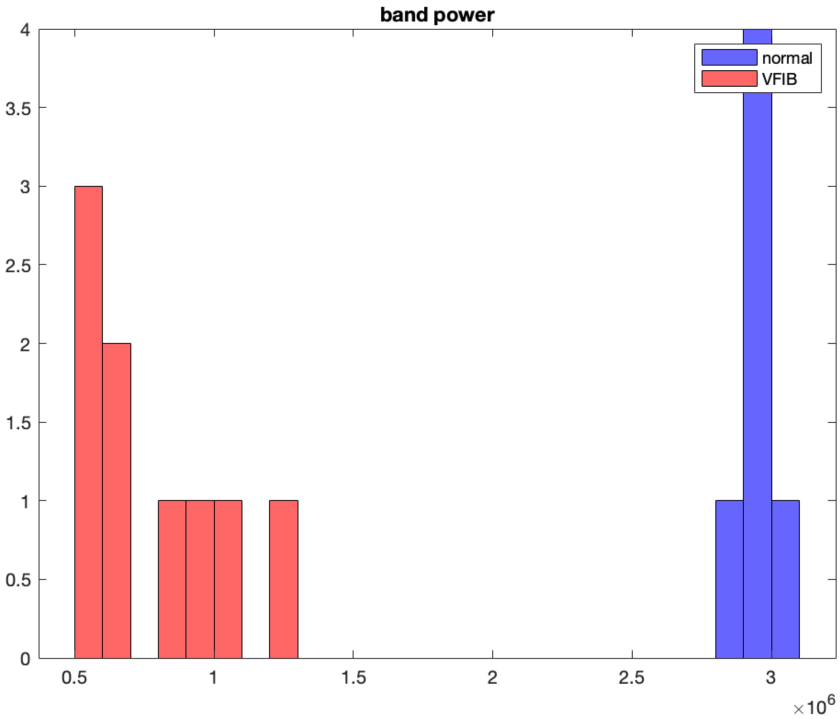

Figure 9: Delta Power in 10-30 Hz Band Comparison for Normal and VFIB Segments

Soloution

Observations from the feature comparison:

- **Criterion 1**: Mean frequency is not suitable for accurate classification.
- **Criterion 2**: Median frequency is suitable for classification, with a threshold set at approximately 0.07.
- **Criterion 3**: Power in the 10 to 30 Hz band is suitable for classification, with a threshold set at approximately 2×10^6.

Question 6

Considering the results of previous part, choose two features and, for each one, complete the function `va_detect.m`. Obtain and review the `alarm` vector (output detected by the feature). Windowing in the function `va_detect.m` is similar to the implementation in question 2, and the `alarm` vector is similar to the actual labels in question 3.

```
    [alarm_bandpower,t_bandpower] = va_detect(n_422 , fs, 'bansdpower');
    [alarm_med,t_med] = va_detect(n_422 ,fs ,'med');
```

Source Code 9: Histogram Comparison of VFIB and Normal Features

Question 7

For each of the features in previous question, form a *Confusion* matrix (two classes, considering `VFIB` as positive and `Normal` as negative) and calculate the metrics *Sensitivity*, *Specificity*, and *Accuracy*. For the windows that are labeled neither `VFIB` nor `Normal`, what was the diagnostic outcome?

```
%% 2-7
targets = labels((labels(:,1) == 1 | labels(:,1) == 2) , 1) == 2;
outputs_med = alarm_med((labels(:,1) == 1 | labels(:,1) == 2)) == 1;
outputs_bp = alarm_bandpower((labels(:,1) == 1 | labels(:,1) == 2)) == 1;

cm1 = confusionmat(targets, outputs_bp);
confusionchart(cm1)

cm2 = confusionmat(targets, outputs_med);
confusionchart(cm2)
```

Source Code 10: Confusion Matrix and Performance Metrics Calculation

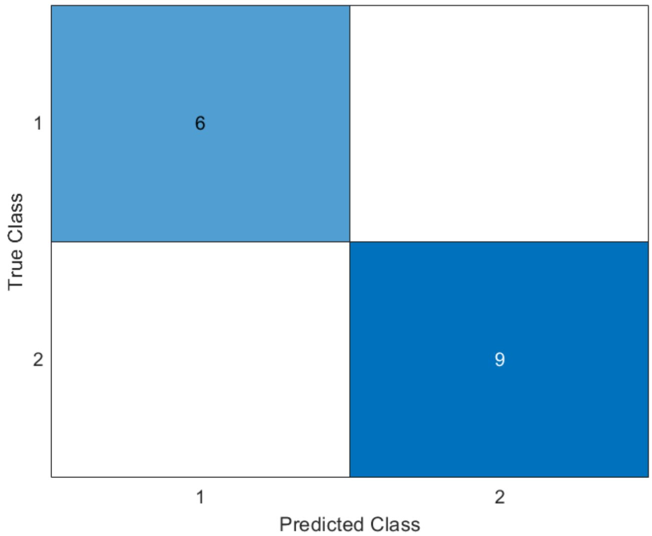

Figure 10: Two class classification based on Band Power confusion matrix

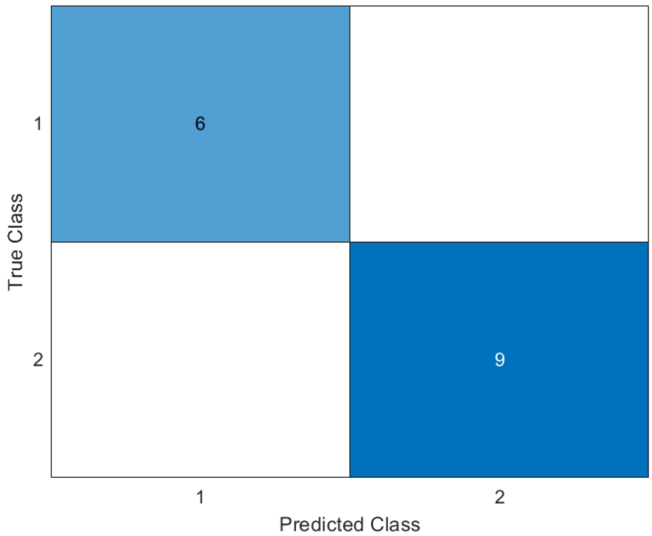

Figure 11: Two class classification based on Median Frequency confusion matrix

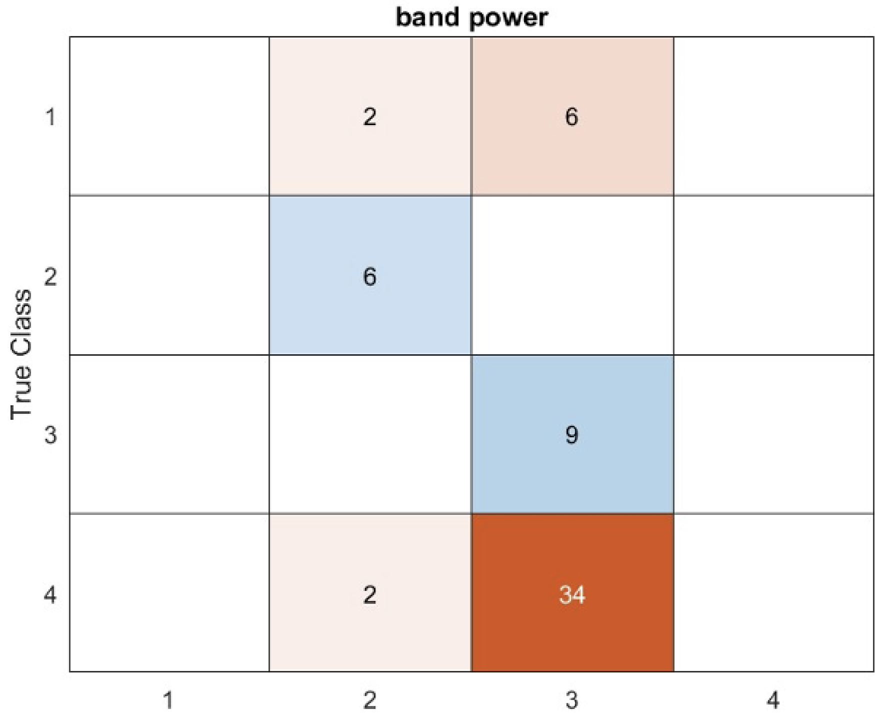

Figure 12: 4 class classification based on Band Power confusion matrix

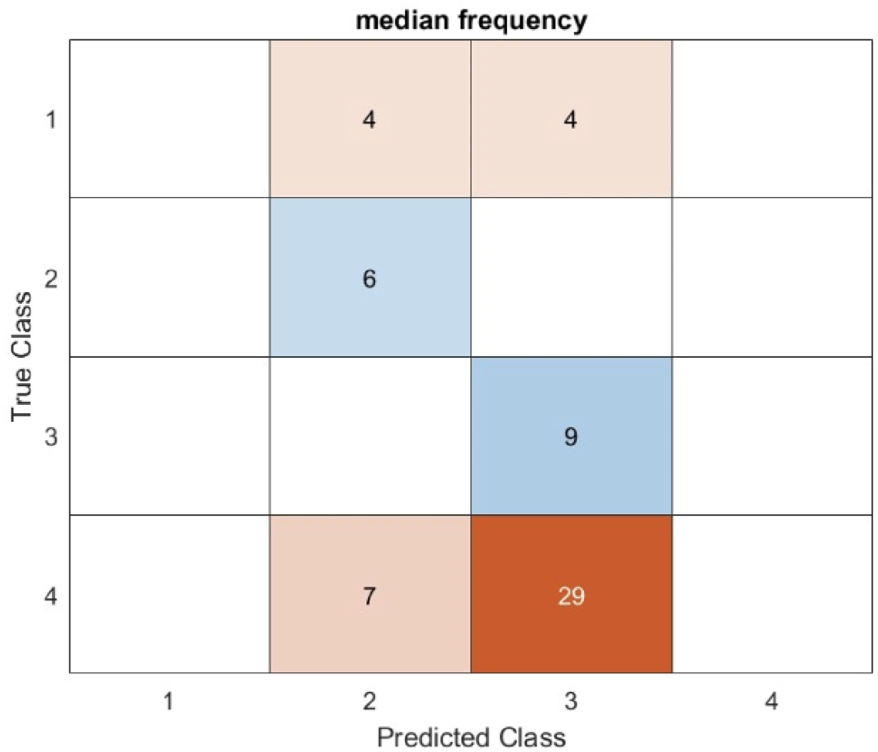

Figure 13: 4 class classification based on Median Frequency confusion matrix

Soloution

Two-Class Classification Based on Band Power and Median Frequency

- **Accuracy** $= \frac{6+9}{6+9+0+0} = 1.0$
- **Sensitivity** $= \frac{6}{6+0} = 1.0$
- **Specificity** $= \frac{9}{9+0} = 1.0$

four-Class Classification Based on Band Power:

The data from the "none" group has been classified into the Normal and VFIB classes with a ratio of 2 to 6.
The data from the "VT" group has been classified into the Normal and VFIB classes with a ratio of 2 to 34.

four-Class Classification Based on Median Frequency:

The data from the "none" group has been classified into the Normal and VFIB classes with a ratio of 4 to 4.
The data from the "VT" group has been classified into the Normal and VFIB classes with a ratio of 7 to 29.

Question 8

In this section, we aim to evaluate the effectiveness of morphological (shape-temporal) and statistical features in classifying `VFIB` and `normal` windows. We examine various morphological and statistical features and select a few to compute for all data windows. You may use the following features:

- Maximum amplitude (`max`)
- Minimum amplitude (`min`)
- Peak-to-peak amplitude (`peak-to-peak`)
- Use the `findpeaks` command to determine local maxima and minima and use it to compute the mean R-peak amplitude.
- Number of zero crossings
- Variance of amplitudes (`var`)

```
    for i=1:59
        if(labels(i,1)==1 || labels(i,1)==2)
            signal = n_422(labels(i,2)-10*250+1:labels(i,2));
            labels(i,6)=min(signal);
            labels(i,7)=max(signal);
            labels(i,8)=labels(i,7)-labels(i,6);
            labels(i,9)=mean(findpeaks(signal),'all');
            labels(i,10)=sum(signal==0);
            labels(i,11)=var(signal);
        end
```

```
    end

    normal_features=labels(2:7,[6:11]);
    VFIB_features=labels(51:59,[6:11]);
```

Question 9

For each of the computed features in previous question, plot the histogram (using the `hist` command) of the feature for the two classes, `VFIB` and `Normal`, and compare them. Is it possible to separate the two classes using this feature? Select a threshold for each feature to classify the two classes.

```
    figure

    histogram(normal_features(:,1),'FaceColor', 'blue','BinWidth',20)
    hold on
    histogram(VFIB_features(:,1),'FaceColor', 'red','BinWidth',20)
    legend("normal","VFIB")
    title("min amplitude")
    hold off

    figure

    histogram(normal_features(:,2),'FaceColor', 'blue','BinWidth',30)
    hold on
    histogram(VFIB_features(:,2),'FaceColor', 'red','BinWidth',30)
    legend("normal","VFIB")
    title("max amplitude")
    hold off

    figure

    histogram(normal_features(:,3).','FaceColor', 'blue','BinWidth',30)
    hold on
    histogram(VFIB_features(:,3),'FaceColor', 'red','BinWidth',30)
    legend("normal","VFIB")
    title("peak to peak")
    hold off

    figure

    histogram(normal_features(:,4).','FaceColor', 'blue','BinWidth',8)
    hold on
    histogram(VFIB_features(:,4),'FaceColor', 'red','BinWidth',8)
    legend("normal","VFIB")
    title("mean R")
    hold off

    figure

    histogram(normal_features(:,5).','FaceColor', 'blue','BinWidth',3)
    hold on
    histogram(VFIB_features(:,5),'FaceColor', 'red','BinWidth',3)
    legend("normal","VFIB")
    title("zero crossing")
    hold off

    figure
```

```
    histogram(normal_features(:,6).','FaceColor', 'blue','BinWidth',10^3)
    hold on
    histogram(VFIB_features(:,6),'FaceColor', 'red','BinWidth',10^3)
    legend("normal","VFIB")
    title("var")
    hold off
```

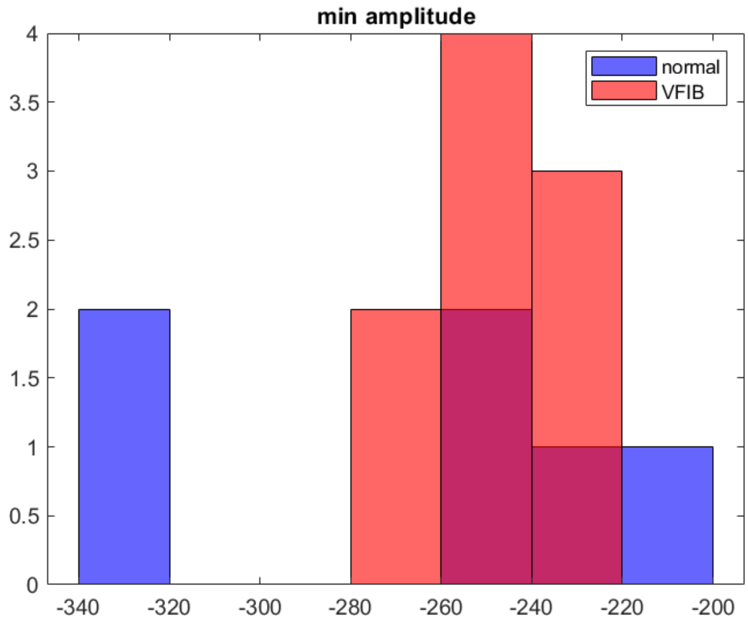

Figure 14: Min Amplitude Comparison for Normal and VFIB Segments

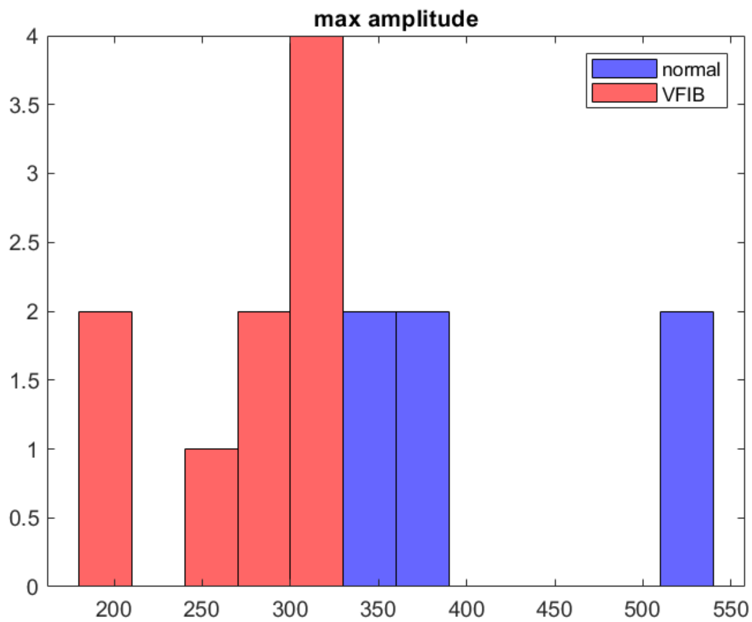

Figure 15: Max Amplitude Comparison for Normal and VFIB Segments

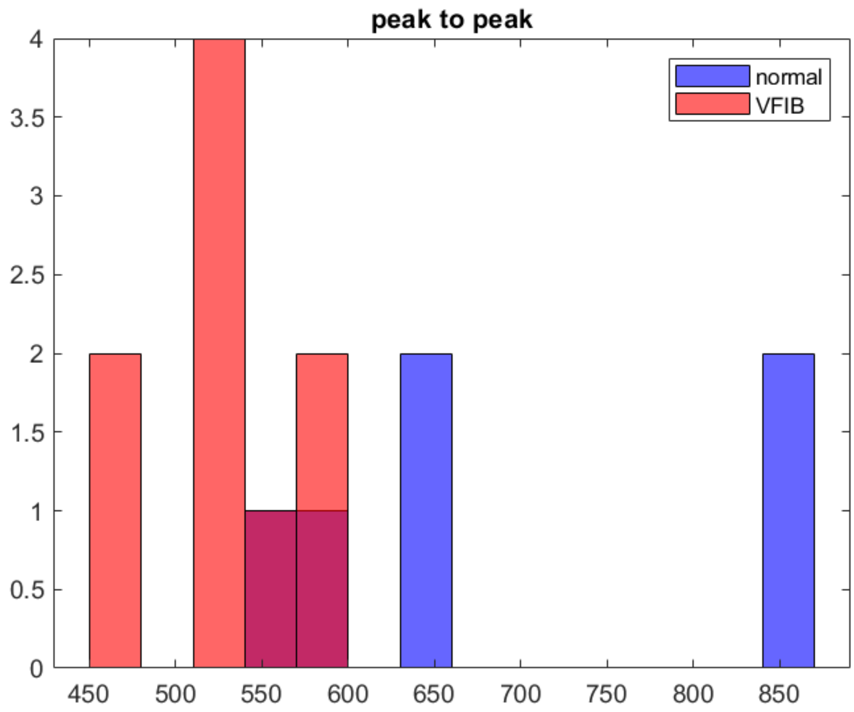

Figure 16: Peak-to-Peak Comparison for Normal and VFIB Segments

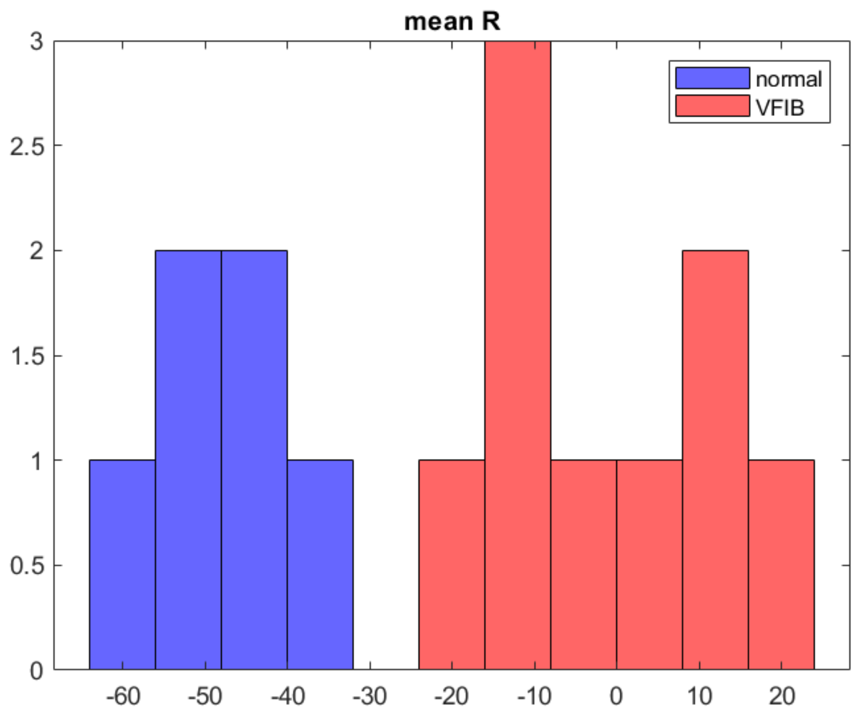

Figure 17: Mean R Comparison for Normal and VFIB Segments

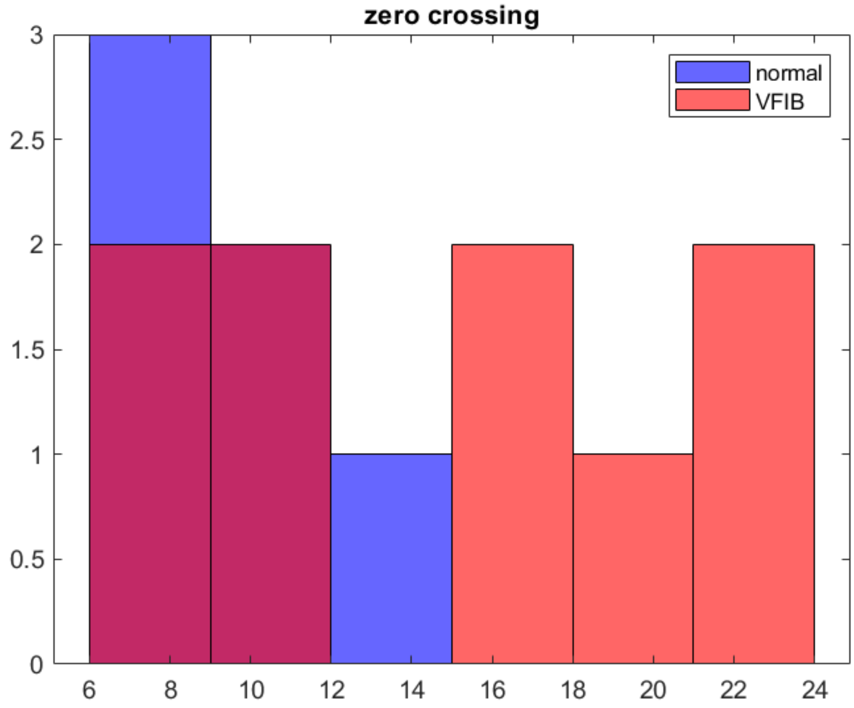

Figure 18: Zero Crossing Comparison for Normal and VFIB Segments

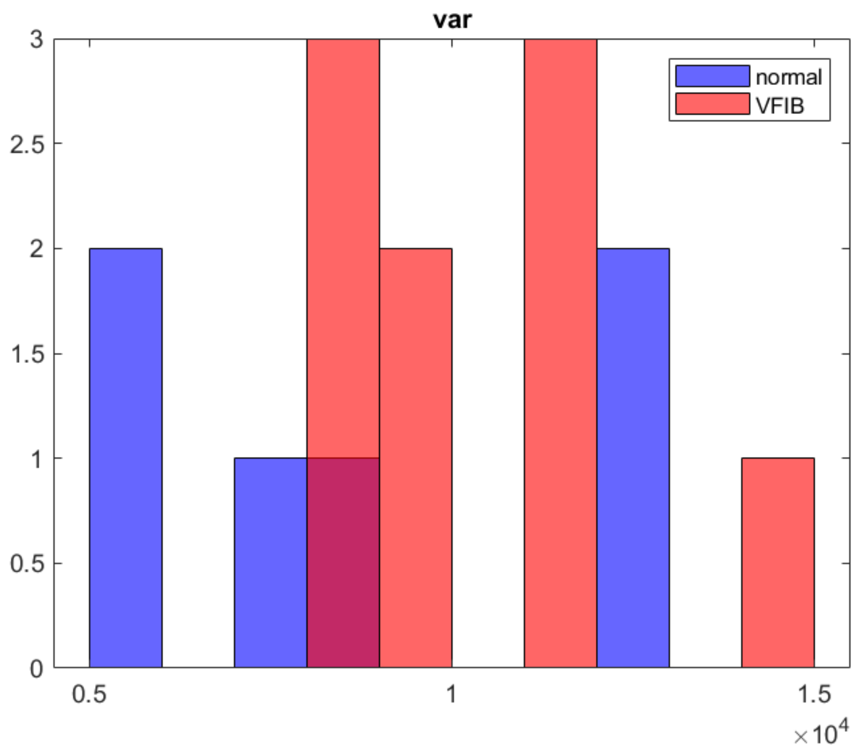

Figure 19: Variance Comparison for Normal and VFIB Segments

Soloution

Observations from the feature comparison:

- **Criterion 1**: Min Amplitude is not suitable for accurate classification.
- **Criterion 2**: Max Amplitude is suitable for classification, with a threshold set at approximately 325.
- **Criterion 3**: Peak-to-Peak Value is not suitable for accurate classification.
- **Criterion 4**: Mean R is suitable for classification, with a threshold set at approximately -28.
- **Criterion 5**: Zero Crossing is not suitable for accurate classification.
- **Criterion 6**: Variance is not suitable for accurate classification.

Question 10

Considering the results of previous question, select two features and, for each one, complete the `va_detect.m` function. Obtain and examine the `alarm` vector (the detection output generated by the selected feature).

```
[alarm_max,t_max] = va_detect(n_422,fs,'max');
[alarm_peak,t_peak] = va_detect(n_422,fs, 'mean-R');
```

Question 11

Considering the results of previous question, select two features and, for each one, complete the `va_detect.m` function. Obtain and examine the `alarm` vector (the detection output generated by the selected feature).

```
targets = labels((labels(:,1) == 1 | labels(:,1) == 2) , 1) == 2;
outputs_max = alarm_max((labels(:,1) == 1 | labels(:,1) == 2)) == 1;
outputs_peak = alarm_peak((labels(:,1) == 1 | labels(:,1) == 2)) == 1;

figure
cm1 = confusionmat(targets, outputs_peak);
confusionchart(cm1)

figure
cm2 = confusionmat(targets, outputs_max);
confusionchart(cm2)

targets_all= labels(: , 1);
outputs_max_all= alarm_max+1;
outputs_peak_all = alarm_peak+1;

cm1_all = confusionmat(targets_all, outputs_max_all);
confusionchart(cm1_all)
title("max amplitude")

figure
cm2_all = confusionmat(targets_all, outputs_peak_all);
confusionchart(cm2_all)
title("peak to peak")
```

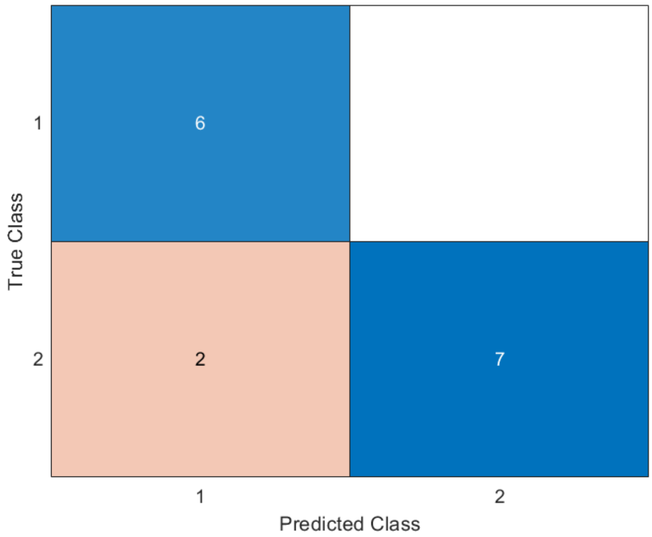

Figure 20: Two class classification based on Max Amplitude confusion matrix

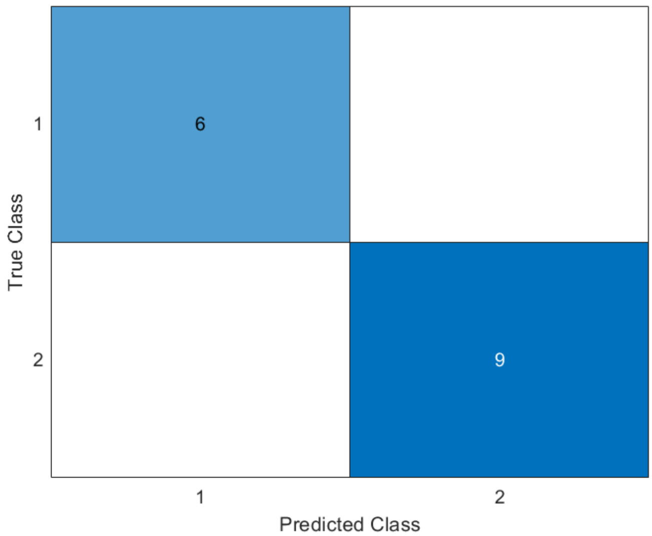

Figure 21: Two class classification based on Mean R confusion matrix

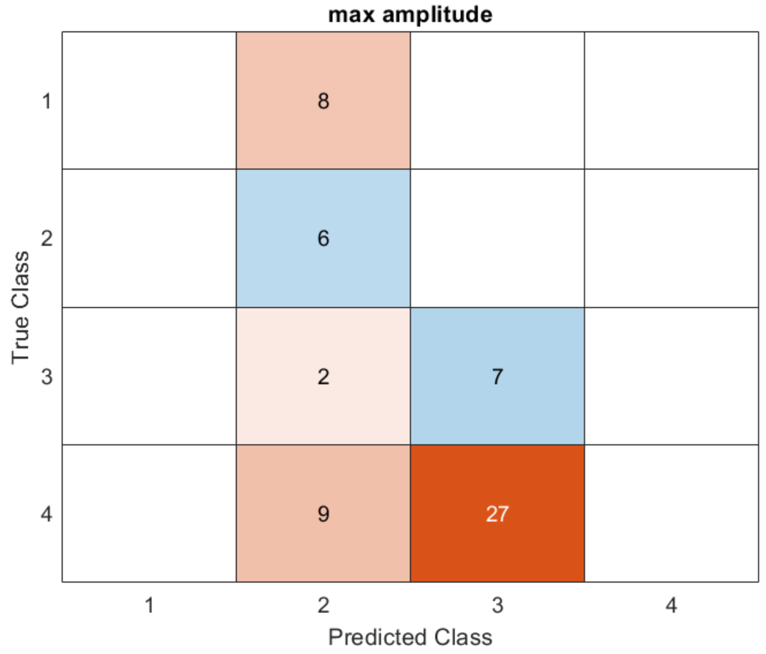

Figure 22: 4 class classification based on Max Amplitude confusion matrix

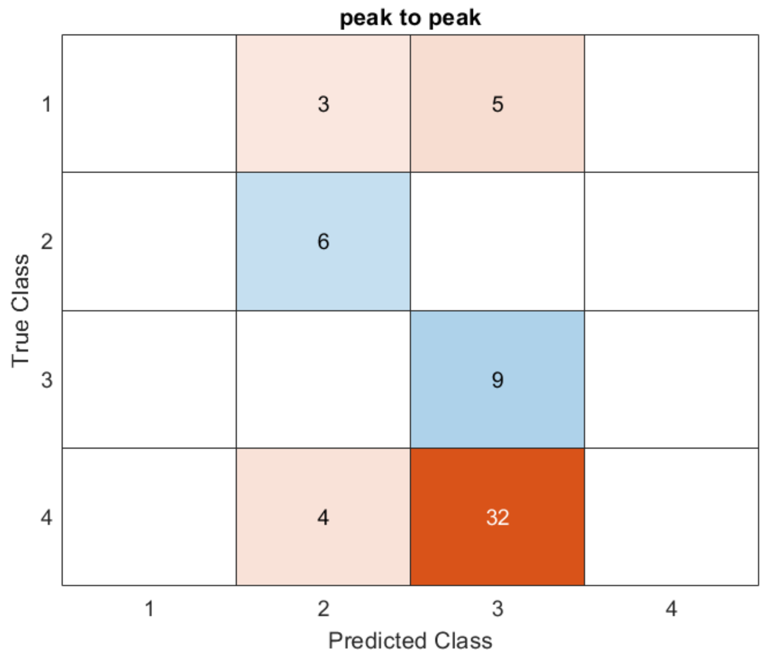

Figure 23: 4 class classification based on Mean R confusion matrix

Soloution

Two-Class Classification Based on Max Amplitude

- **Accuracy** $= \frac{6+7}{6+7+2+0} = 0.86$
- **Sensitivity** $= \frac{6}{6+2} = 0.75$
- **Specificity** $= \frac{7}{7+0} = 1.0$

Two-Class Classification Based on Mean R

- **Accuracy** $= \frac{19+19}{19+19+0+0} = 1.0$
- **Sensitivity** $= \frac{19}{19+0} = 1.0$
- **Specificity** $= \frac{19}{19+0} = 1.0$

four-Class Classification Based on Max Amplitude:

The data from the "none" group has been classified into the Normal and VFIB classes with a ratio of 8 to 0.
The data from the "VT" group has been classified into the Normal and VFIB classes with a ratio of 9 to 27.

four-Class Classification Based on Mean R:

The data from the "none" group has been classified into the Normal and VFIB classes with a ratio of 3 to 5.
The data from the "VT" group has been classified into the Normal and VFIB classes with a ratio of 4 to 32.

Question 12

Do all previous parts for `n_424.mat`.

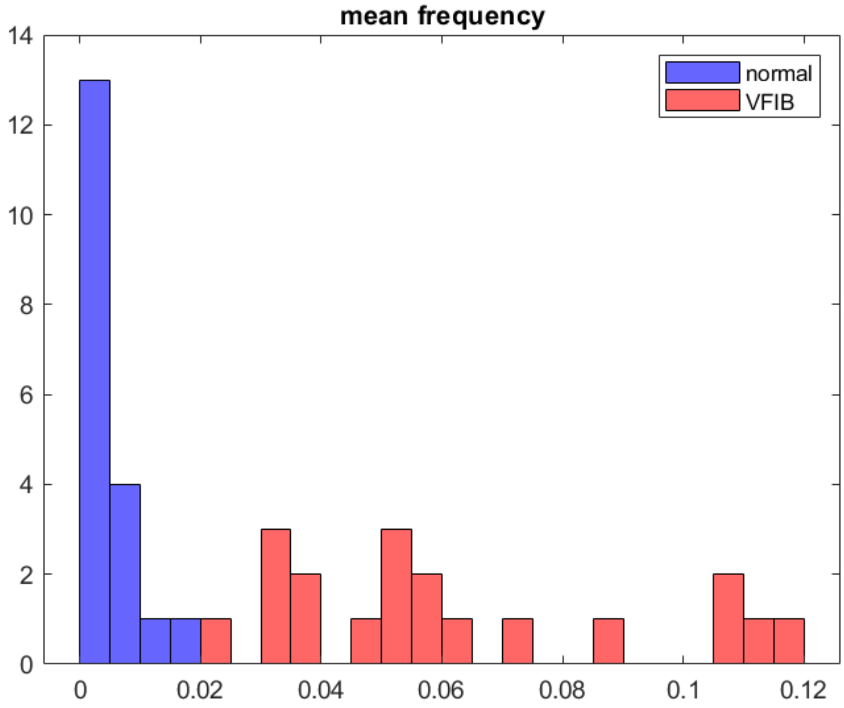

Figure 24: Mean Frequency Comparison for Normal and VFIB Segments

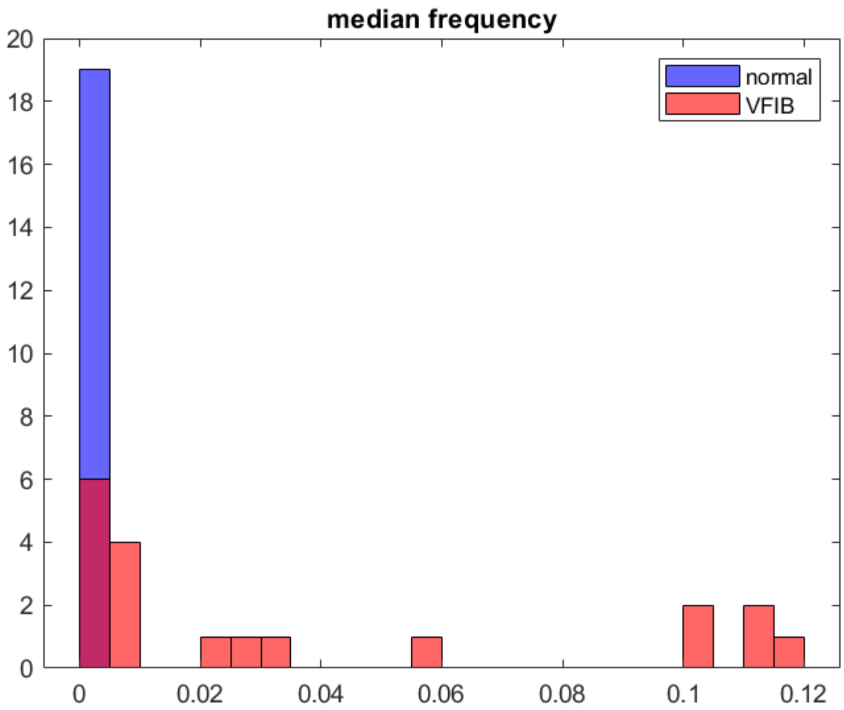

Figure 25: Median Frequency Comparison for Normal and VFIB Segments

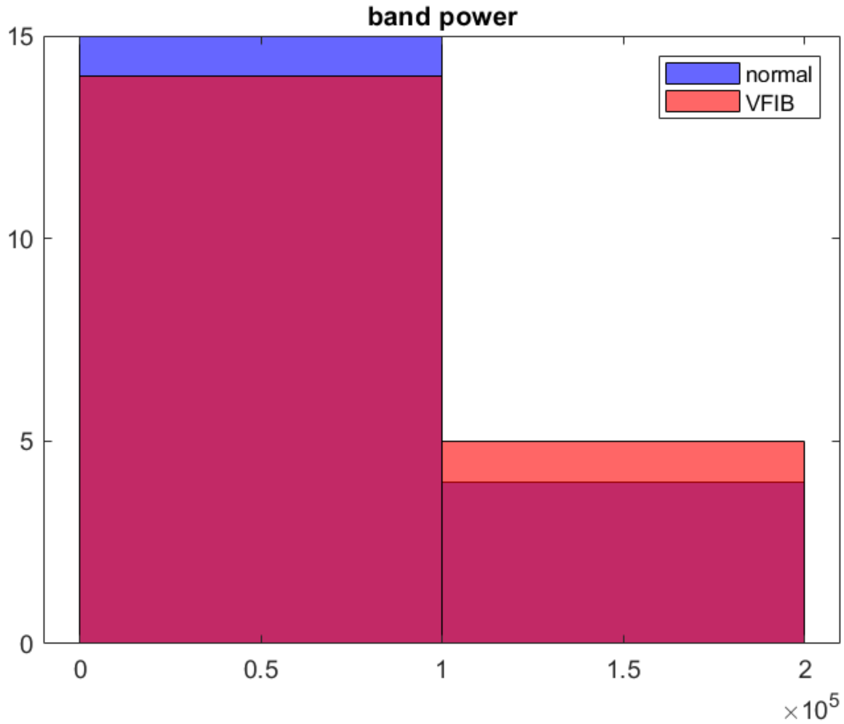

Figure 26: Delta Power in 10-30 Hz Band Comparison for Normal and VFIB Segments

Soloution

Observations from the feature comparison:

- **Criterion 1**: Mean frequency is suitable for classification, with a threshold set at approximately 0.02.
- **Criterion 2**: Median frequency is not suitable for accurate classification.
- **Criterion 3**: Power in the 10 to 30 Hz band is suitable for classification, with a threshold set at approximately 2×10^6.

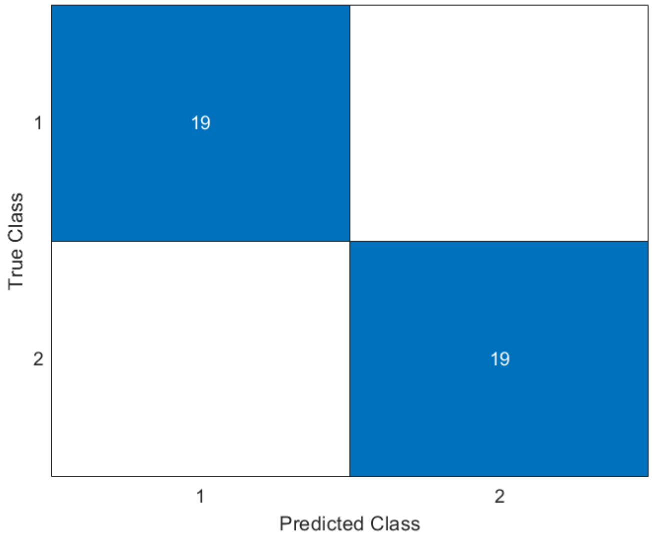

Figure 27: Two class classification based on Mean confusion matrix

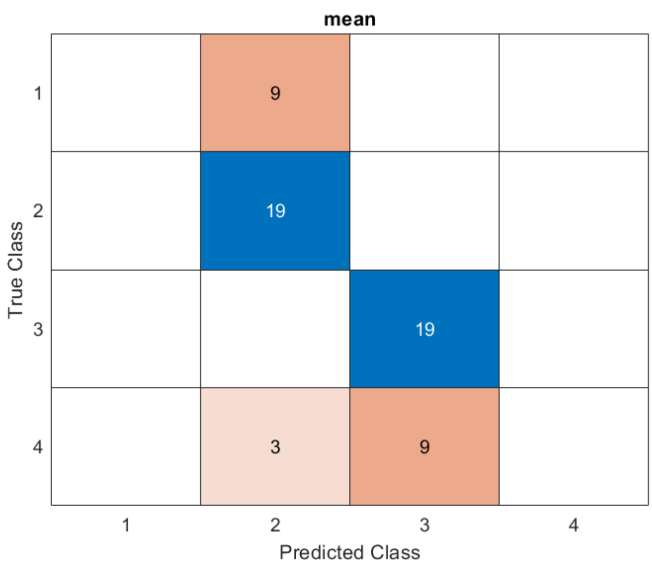

Figure 28: 4 class classification based on Mean confusion matrix

Soloution

Two-Class Classification Based on Mean

- **Accuracy** $= \frac{19+19}{19+19+0+0} = 1.0$
- **Sensitivity** $= \frac{19}{19+0} = 1.0$
- **Specificity** $= \frac{19}{19+0} = 1.0$

four-Class Classification Based on Mean:

The data from the "none" group has been classified into the Normal and VFIB classes with a ratio of 9 to 0.
The data from the "VT" group has been classified into the Normal and VFIB classes with a ratio of 3 to 9.

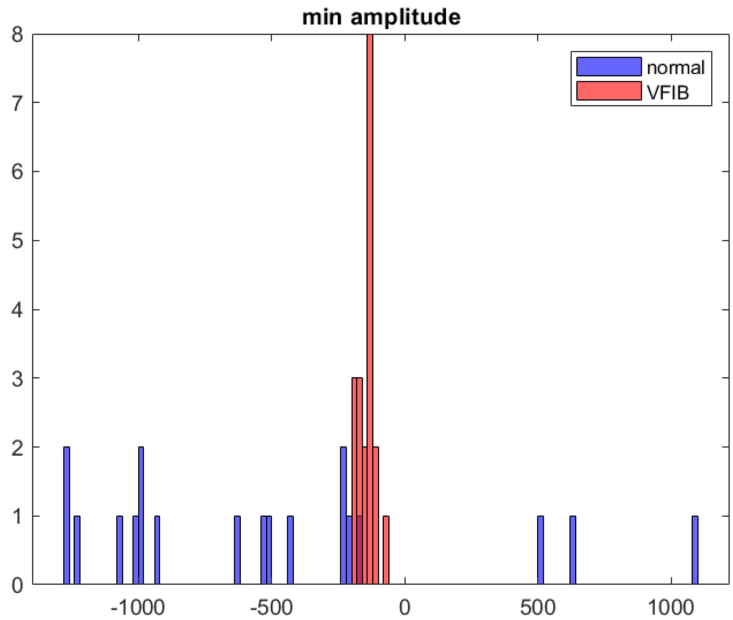

Figure 29: Min Amplitude Comparison for Normal and VFIB Segments

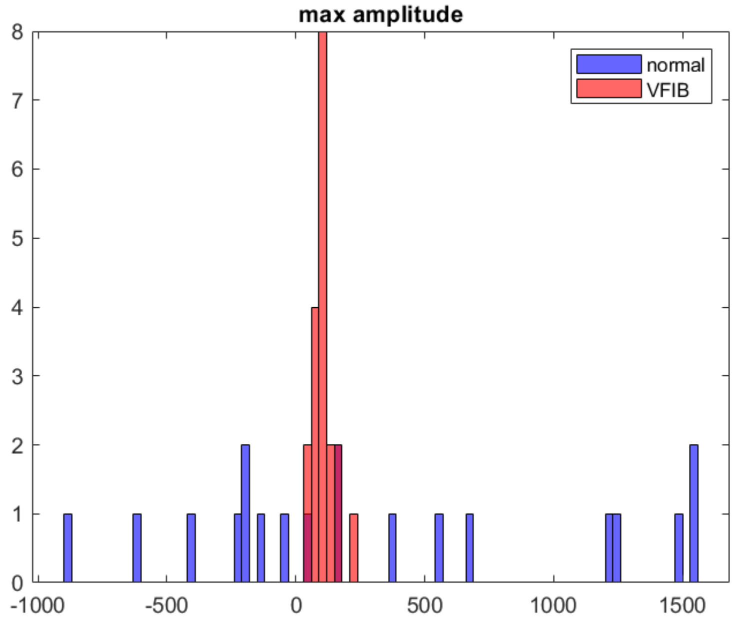

Figure 30: Max Amplitude Comparison for Normal and VFIB Segments

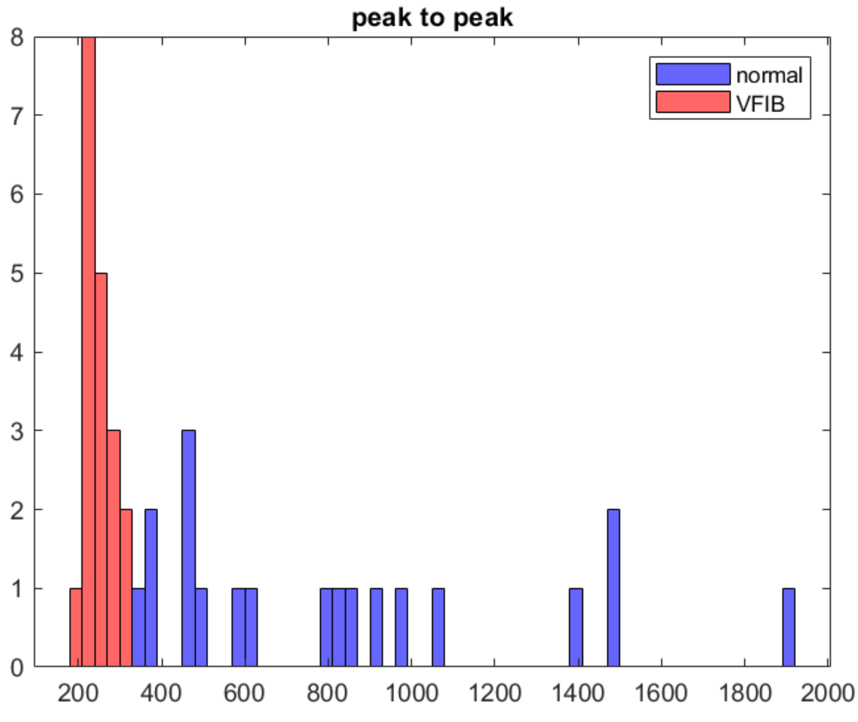

Figure 31: Peak-to-Peak Comparison for Normal and VFIB Segments

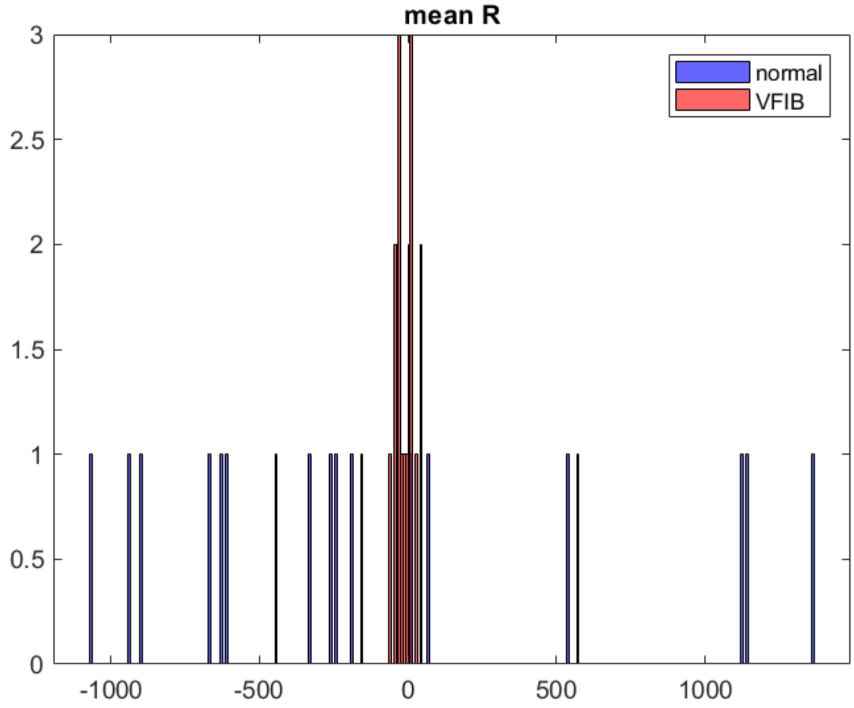

Figure 32: Mean R Comparison for Normal and VFIB Segments

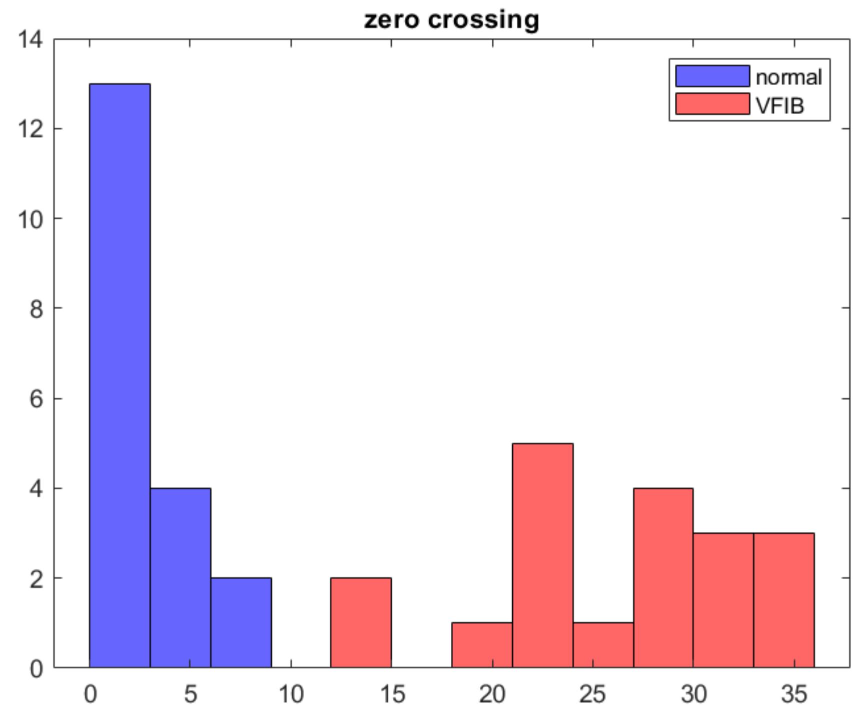

Figure 33: Zero Crossing Comparison for Normal and VFIB Segments

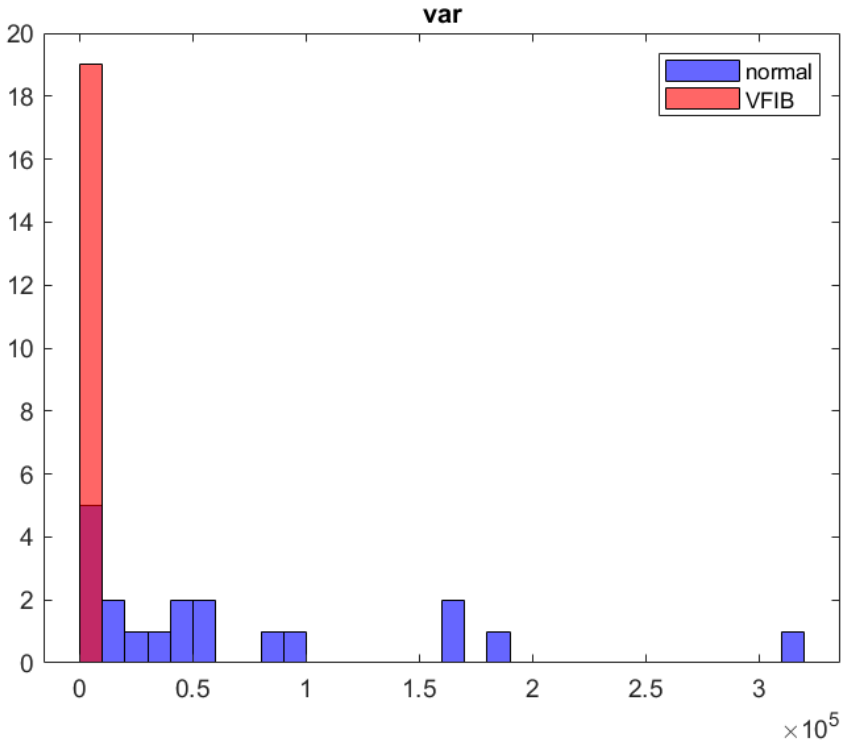

Figure 34: Variance Comparison for Normal and VFIB Segments

Soloution

Observations from the feature comparison:

- **Criterion 1**: Min Amplitude is not suitable for accurate classification.
- **Criterion 2**: Max Amplitude is not suitable for accurate classification.
- **Criterion 3**: Peak-to-Peak is suitable for classification, with a threshold set at approximately 315.
- **Criterion 4**: Mean R is not suitable for accurate classification.
- **Criterion 5**: Zero Crossing is suitable for classification, with a threshold set at approximately 10.
- **Criterion 6**: Variance is not suitable for accurate classification.

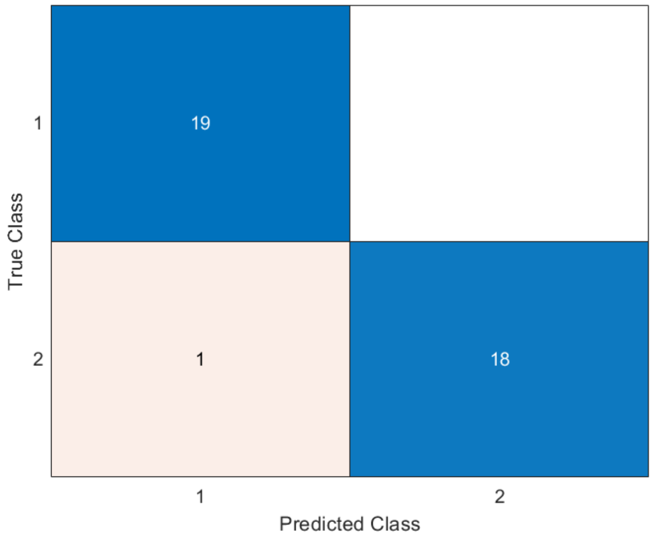

Figure 35: Two class classification based on Peak-to-Peak confusion matrix

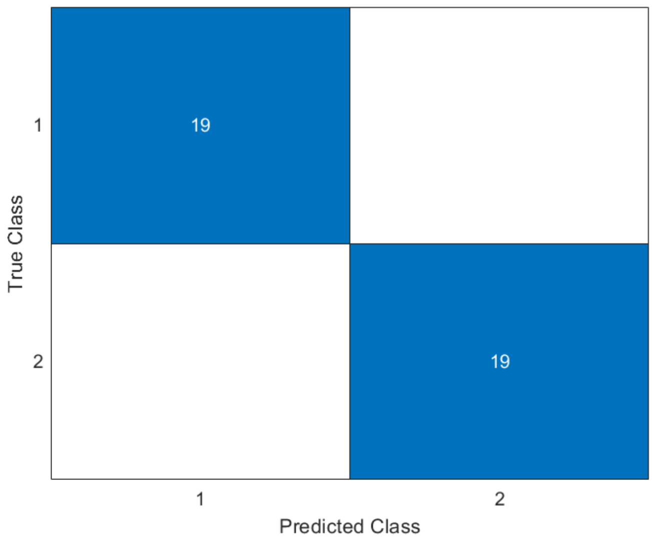

Figure 36: Two class classification based on Zero Crossing confusion matrix

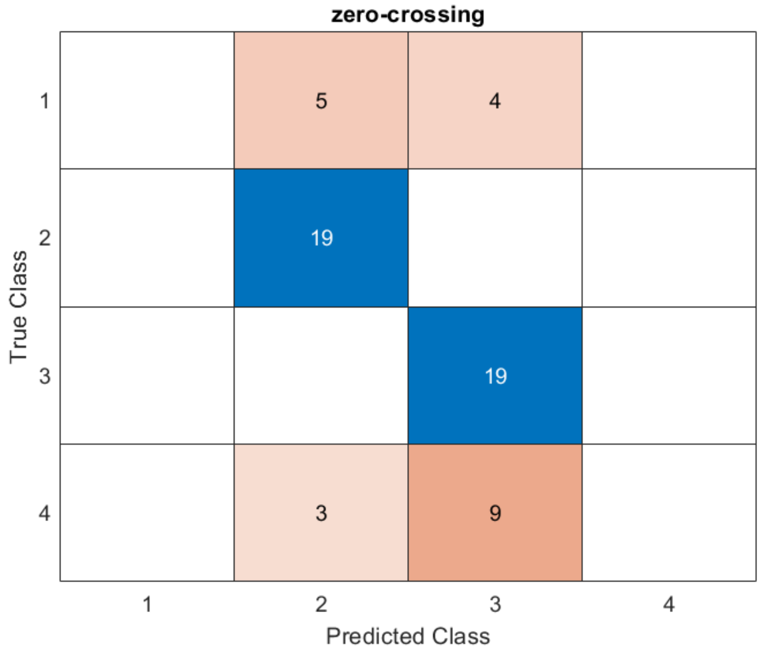

Figure 37: 4 class classification based on Peak-to-Peak confusion matrix

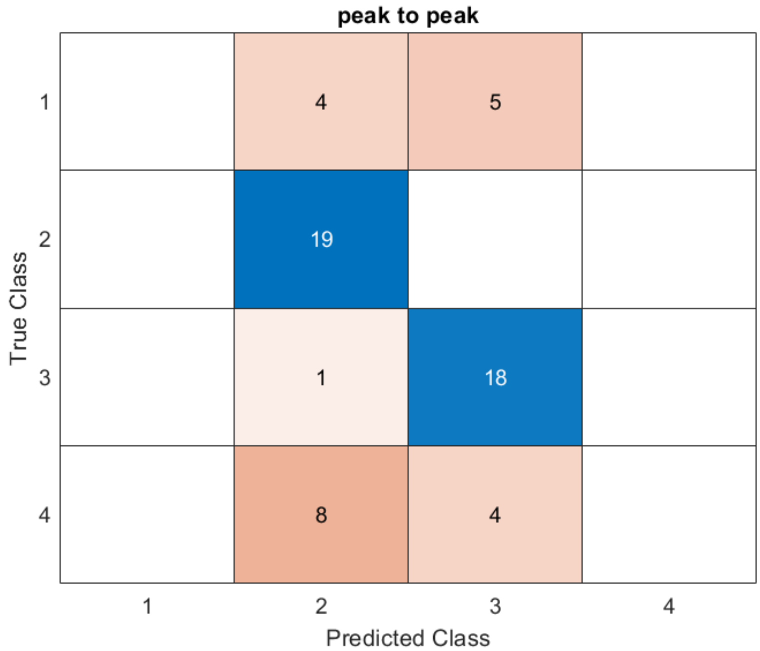

Figure 38: 4 class classification based on Zero Crossing confusion matrix

Soloution

Two-Class Classification Based on Peak-to-Peak

- **Accuracy** $= \frac{19+18}{19+18+1+0} = 0.97$
- **Sensitivity** $= \frac{19}{19+1} = 0.95$
- **Specificity** $= \frac{18}{18+0} = 1.0$

Two-Class Classification Based on Zero Crossing

- **Accuracy** $= \frac{19+19}{19+19+0+0} = 1.0$
- **Sensitivity** $= \frac{19}{19+0} = 1.0$
- **Specificity** $= \frac{19}{19+0} = 1.0$

four-Class Classification Based on Peak-to-Peak:

The data from the "none" group has been classified into the Normal and VFIB classes with a ratio of 5 to 4.
The data from the "VT" group has been classified into the Normal and VFIB classes with a ratio of 3 to 9.

four-Class Classification Based on Zero Crossing:

The data from the "none" group has been classified into the Normal and VFIB classes with a ratio of 4 to 5.
The data from the "VT" group has been classified into the Normal and VFIB classes with a ratio of 8 to 4.

Question 13

Compare the best detector obtained for the first dataset with the best detector obtained for the second dataset.

Soloution

The best detector for first dataset is Max Amplitude and the best for the second dataset is Zero-Crossing. The thresholds are different. Each one will have their best performance on differemt datsets.

Question 14

Apply the best detector obtained for the first dataset to the second dataset and review the result. Similarly, apply the best detector obtained for the second dataset to the first dataset.

Soloution

We use Zero-crossing for the first data and Max Amplitude for the second one.

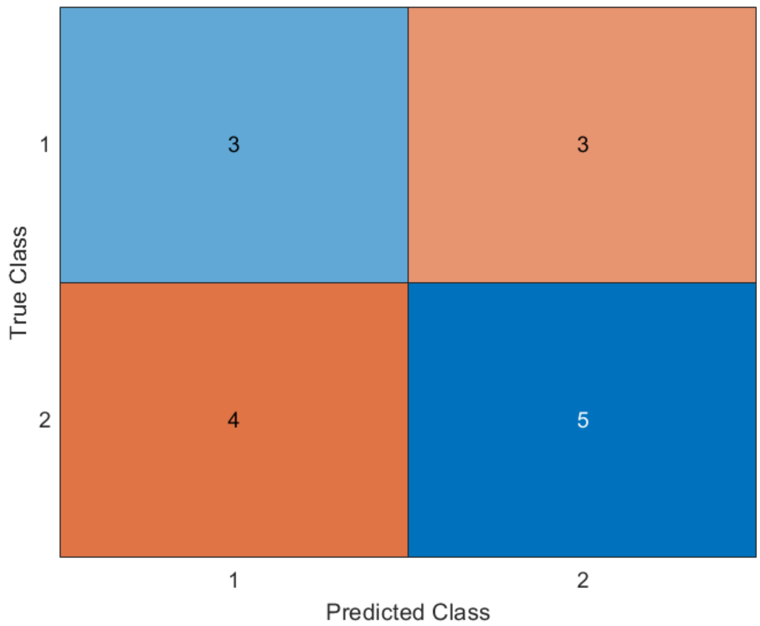

Figure 39: Two class classification based on Zero Crossing confusion matrix for n_422

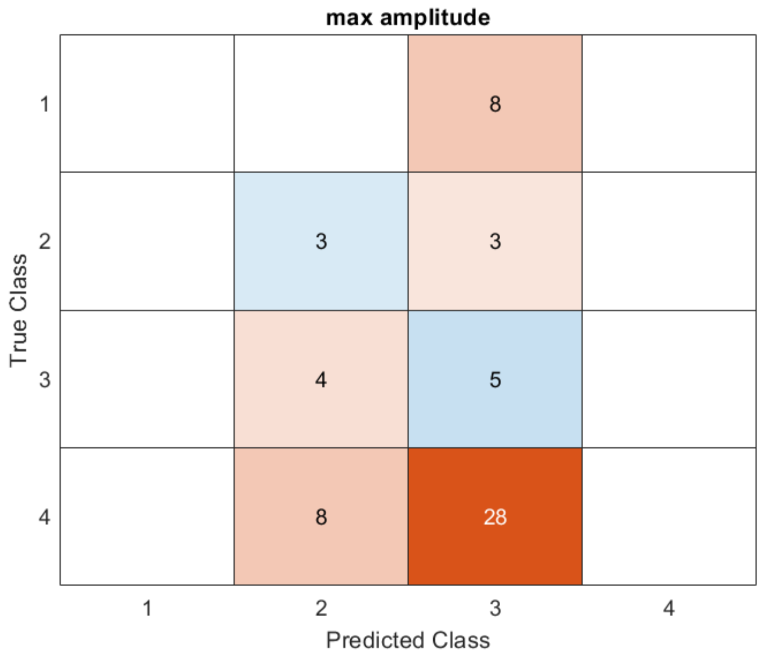

Figure 40: Two class classification based on Zero Crossing confusion matrix for n_422

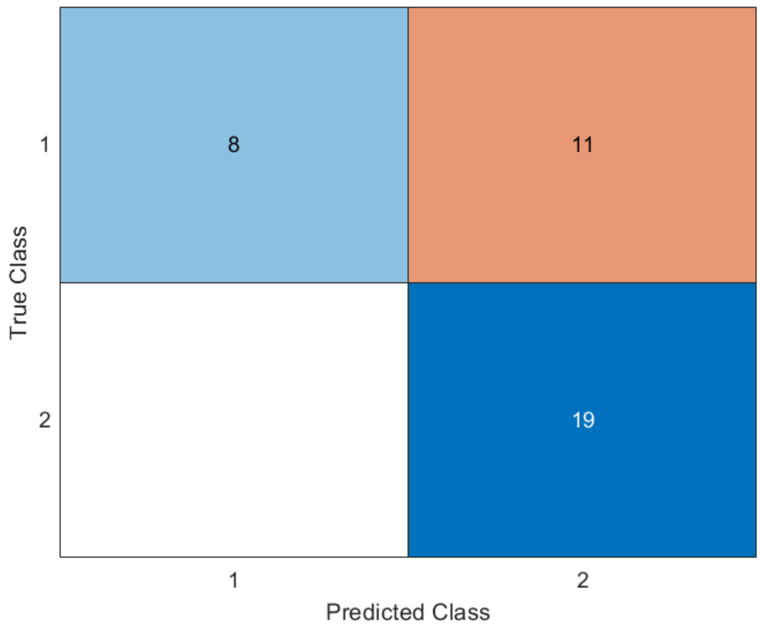

Figure 41: 4 class classification based on Max Amplitude confusion matrix for n_424

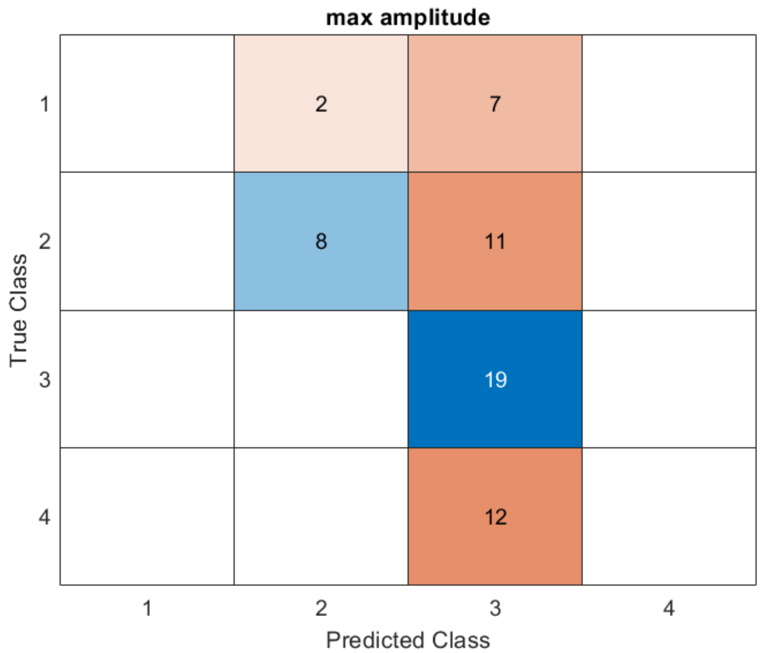

Figure 42: 4 class classification based on Max Amplitude confusion matrix for n_422

Soloution

Two-Class Classification Based on Zero Crossing for n_422

- **Accuracy** $= \frac{3+5}{3+5+3+3} = 0.57$
- **Sensitivity** $= \frac{3}{3+4} = 0.42$
- **Specificity** $= \frac{5}{5+4} = 0.55$

Two-Class Classification Based on Max Amplitude for n_424

- **Accuracy** $= \frac{8+19}{8+19+11+0} = 0.71$
- **Sensitivity** $= \frac{8}{8+0} = 1.0$
- **Specificity** $= \frac{19}{19+30} = 0.63$

four-Class Classification Based on Zero Crossing for n_422:

The data from the ”none” group has been classified into the Normal and VFIB classes with a ratio of 0 to 8.
The data from the ”VT” group has been classified into the Normal and VFIB classes with a ratio of 8 to 28.

four-Class Classification Based on Max Amplitude for n_424:

The data from the ”none” group has been classified into the Normal and VFIB classes with a ratio of 2 to 7.
The data from the ”VT” group has been classified into the Normal and VFIB classes with a ratio of 0 to 12.

Question 15

Among the designed detectors, select one as the best detector and apply it to at least one of the other signals. Review and analyze the results. In particular, these files give you an opportunity to observe whether your system produces false alarms in the presence of high noise or remains stable without unnecessary alerts. Does your detector produce false alarms, miss detections (missed detections), or produce both? Under what conditions is your detector more prone to errors?

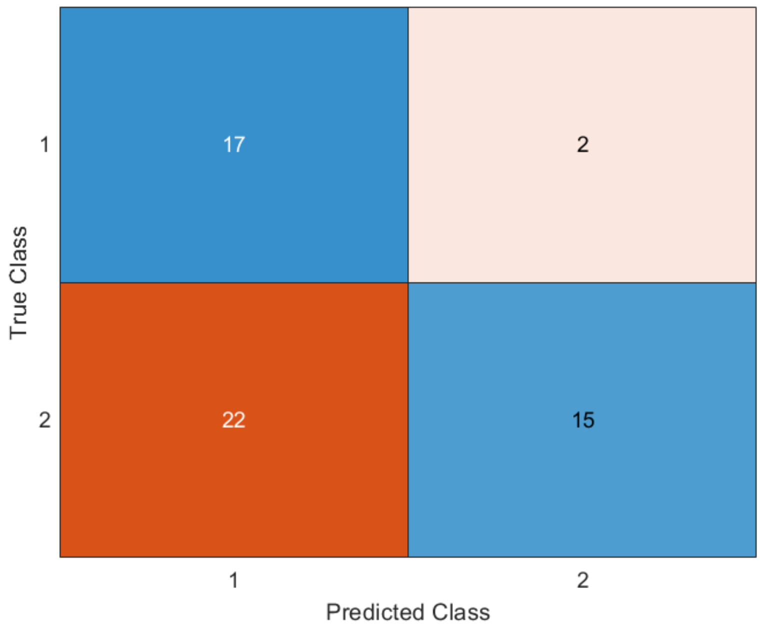

Figure 43: Two class classification based on Zero Crossing confusion matrix for n_426

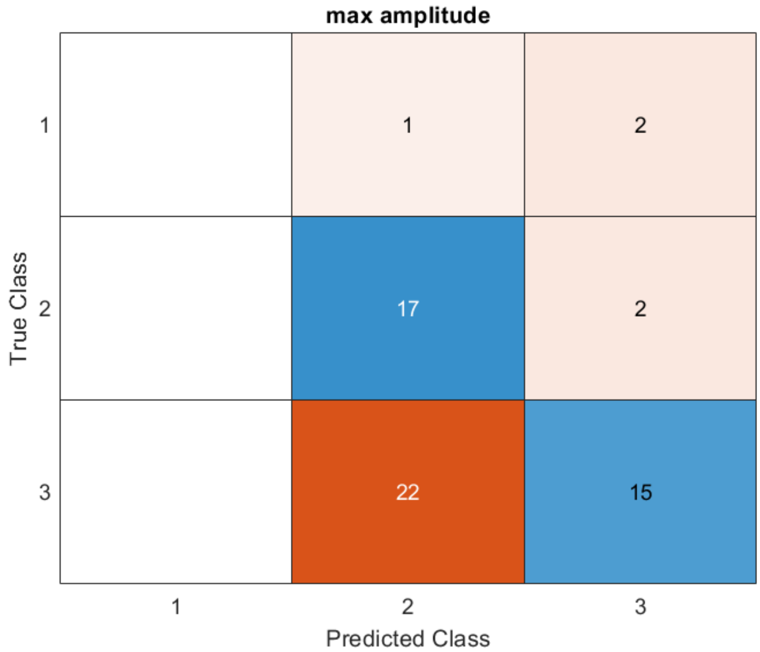

Figure 44: Three class classification based on Zero Crossing confusion matrix for n_426

Soloution

I choose Zero-Crossing as the best detector.

Two-Class Classification Based on Zero Crossing for n_426

- **Accuracy** $= \frac{17+15}{17+15+2+22} = 0.6$
- **Sensitivity** $= \frac{17}{17+22} = 0.43$
- **Specificity** $= \frac{15}{15+2} = 0.88$

Based on the confusion matrix, we can see our detector is not robust to noise. It also can not detect the VT and Noise.