Spectral Analysis of a random signal using Matlab

Lab # 05

Fall 2023

CSE-402L Digital Signal Processing Lab

Submitted by: **Ali Asghar**

Registration No.: **21PWCSE2059**

Class Section: **C**

"On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Submitted to:

Dr. Yasir Saleem Afridi

Date:

27th November 2023

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Lab No: 5.

Title: Spectral Analysis of a random signal using Matlab

Provide .m file with detailed comments

Hint: Find Power Spectral Density, a measurement of the energy at various frequencies,

Procedure:

1. First create some data. Consider data sampled at 1000 samples/sec. Start by forming a time axis for the data, running from t=0 until t=.25 in steps of 1 millisecond. Then form a signal, x, containing sine waves at 50 Hz and 120 Hz.
   (Hint: `x = sin(2*pi*50*t) + sin(2*pi*120*t);`)

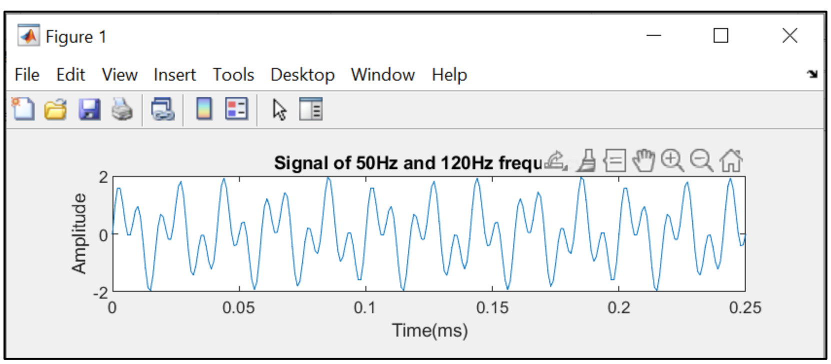
Code:

Lab5.m

```
%%
fs = 1000;
t=0:1/fs:1/4;
f1=50;
f2=120;

figure(1);
x = sin(2*pi*f1*t) + sin(2*pi*f2*t);
subplot(3,1,1);
plot(t,x)
title('Signal of 50Hz and 120Hz frequency');
xlabel('Time(ms)');
ylabel('Amplitude');
```

Output:



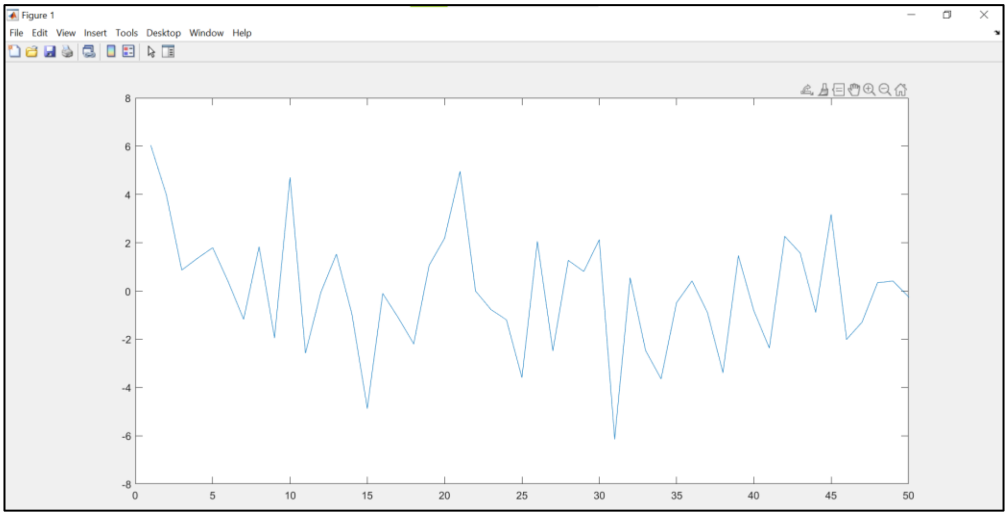
2. Add some random noise with a standard deviation of 2 to produce a noisy signal y. Take a look at this noisy signal y by plotting it. (Hint: `y = x + randn(size(t));`)

Code:

```
%%
y = x + 2*randn(size(t));

%subplot(3,1,2);
plot(y);
xlim([0 50]);
```

Output:

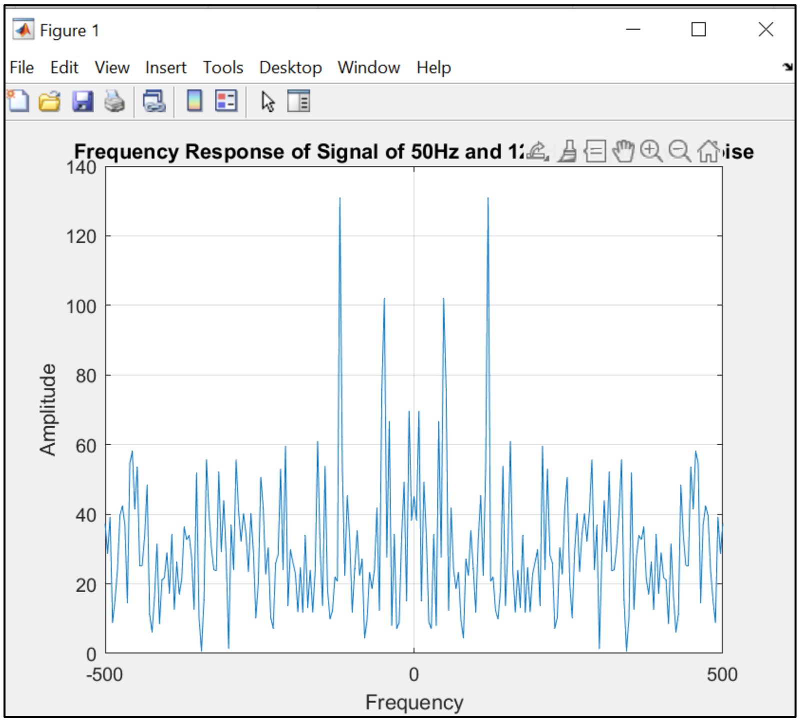


3. Finding the discrete Fourier transform of the noisy signal y (Hint: `Y = fft(y,251);`)

Code:

```
%%
w=-(length(x)-1)/2:(length(x) -1)/2;
y_f = fft(y, length(y));
y_fshift = fftshift(y_f);
%subplot(3,1,3);
plot(4*w,abs(y_fshift))
title('Frequency Response of Signal of 50Hz and 120Hz frequency + Noise');
xlabel('Frequency');
ylabel('Amplitude');
grid on;
```

Output:



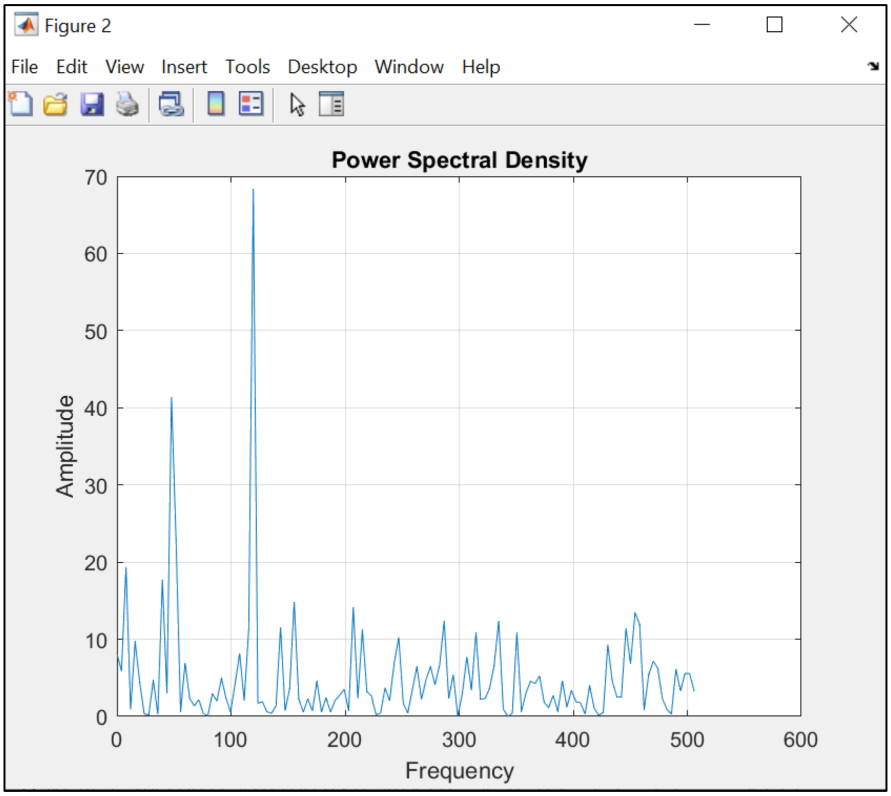
4. Compute the power spectral density, a measurement of the energy at various frequencies, using the complex conjugate (CONJ). Form a frequency axis for the first 127 points and use it to plot the result. (Hint: `Pyy = Y.*conj(Y)/251;`   `f = 1000/251*(0:127);`)

Code:

```
%%
Pyy = (y_f.* conj(y_f))/length(y_f);
%plot(w,10*log10(Pyy))
f = 1000/251*(0:127);

figure(2);
%plot(10*log10(Pyy))
plot(f(1:128), Pyy(1:128));
title('Power Spectral Density');
xlabel('Frequency');
ylabel('Amplitude');
grid on;
```

Output:



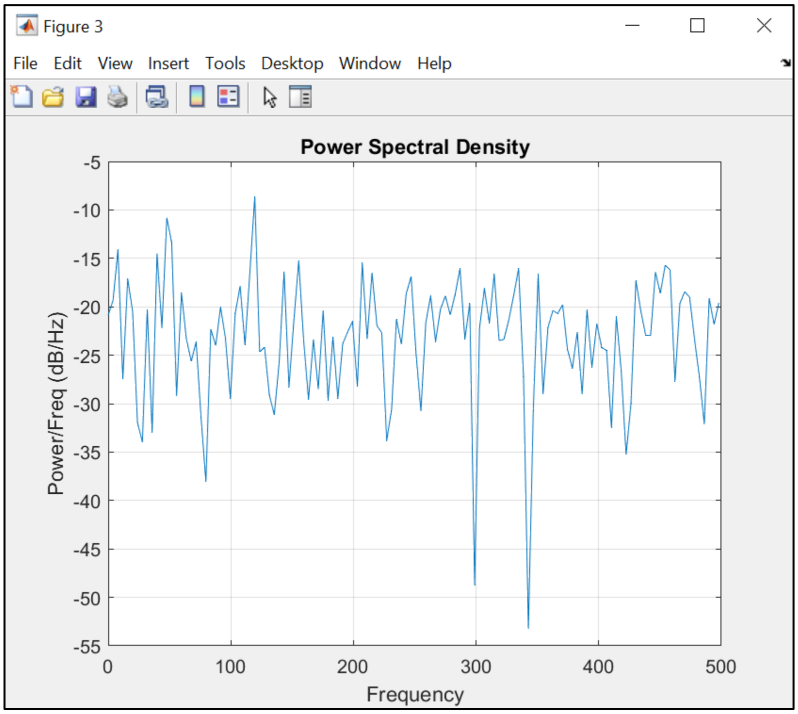
5. Compute and plot the periodogram using periodogram. Show that the two results are identical.

```
[Pyy2,w] = periodogram(y,rectwin(length(y)),length(y),1000)
figure;
plot(w,10*log10(Pyy2))
```

Code:

```
%%
[Pyy2,w2] = periodogram(y,rectwin(length(y)),length(y),1000)
figure(3);
plot(w2,10*log10(Pyy2))
title('Power Spectral Density');
xlabel('Frequency');
ylabel('Power/Freq (dB/Hz)');
grid on;
```

Output:

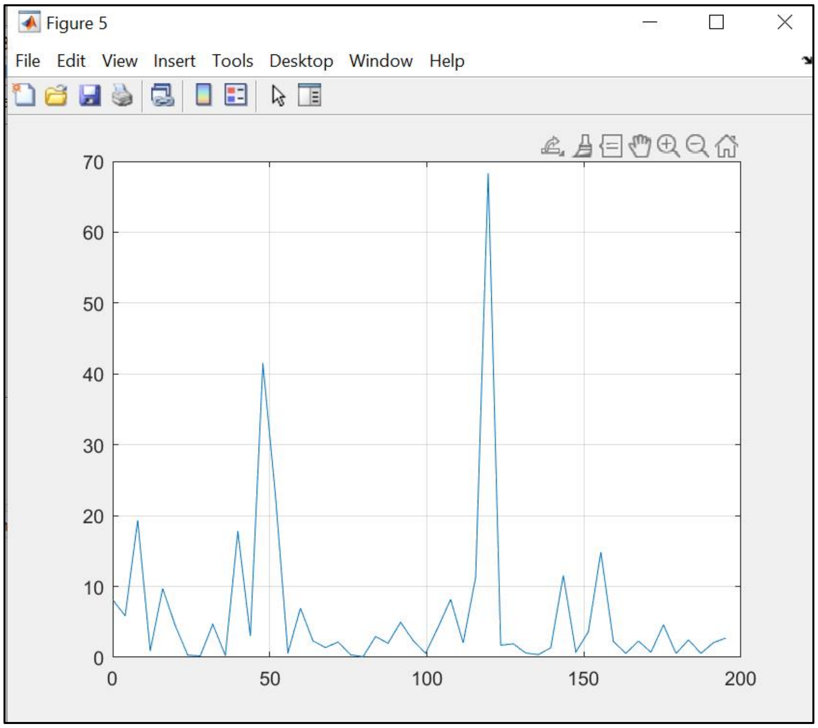


6. Zoom in and plot only up to 200 Hz. Notice the peaks at 50 Hz and 120 Hz. These are the frequencies of the original signal. (Hint: `plot(f(1:50),Pyy(1:50))`

Code:

```
%%
figure(5);
plot(f(1:50),Pyy(1:50))
grid on;
```

Output:



7. Final Remarks/Conclusion.

Full Code:

Editor - C:\Users\PMLS\Desktop\MATLAB\Lab 05\Lab5.m

Lab5.m

```
%%
fs = 1000; % Sampling frequency
t=0:1/fs:1/4; % Time vector
f1=50; % Frequency of first sine wave
f2=120; % Frequency of second sine wave

figure(1);
x = sin(2*pi*f1*t) + sin(2*pi*f2*t); % Sum of two sine waves
subplot(3,1,1);
plot(t,x) % Plot of the signal
title('Signal of 50Hz and 120Hz frequency');
xlabel('Time(ms)');
ylabel('Amplitude');
%%
y = x + 2*randn(size(t)); % Adding noise to the signal

%subplot(3,1,2);
plot(y); % Plot of the noisy signal
xlim([0 50]);
%%
w=-(length(x)-1)/2:(length(x) -1)/2;
y_f = fft(y, length(y)); % Fourier transform of the noisy signal
y_fshift = fftshift(y_f); % Shift zero frequency component to center of spectrum
%subplot(3,1,3);
```

Editor - C:\Users\PMLS\Desktop\MATLAB\Lab 05\Lab5.m

Lab5.m

```
%subplot(3,1,3);
plot(4*w,abs(y_fshift)) % Plot of the frequency response
title('Frequency Response of Signal of 50Hz and 120Hz frequency + Noise');
xlabel('Frequency');
ylabel('Amplitude');
grid on;

%%
Pyy = (y_f.* conj(y_f))/length(y_f); % Power spectral density
%plot(w,10*log10(Pyy))
f = 1000/251*(0:127);

figure(2);
%plot(10*log10(Pyy))
plot(f(1:128), Pyy(1:128)); % Plot of the power spectral density
title('Power Spectral Density');
xlabel('Frequency');
ylabel('Amplitude');
grid on;

%%
[Pyy2,w2] = periodogram(y,rectwin(length(y)),length(y),1000) % Periodogram estimate of the power spectral density
figure(3);
plot(w2,10*log10(Pyy2)) % Plot of the periodogram
```

```
plot(w2,10*log10(Pyy2)) % Plot of the periodogram
title('Power Spectral Density');
xlabel('Frequency');
ylabel('Power/Freq (dB/Hz)');
grid on;


%%
figure(5);
plot(f(1:50),Pyy(1:50)) % Plot of the first 50 points of the power spectral density
grid on;

```

Remarks/Conclusion:

Initially, I created a random signal characterized by frequencies of 50Hz and 120Hz. Subsequently, random noise was introduced to the signal. Following the addition of noise, the signal was transformed into the frequency domain. The power density spectrum was then computed using the relevant formula. The next step involved plotting the periodogram of the power density signal, providing a visual representation of the signal's frequency components. To focus on specific frequencies of interest, a zoom-in was performed, honing in on the 50Hz and 120Hz components for closer examination.