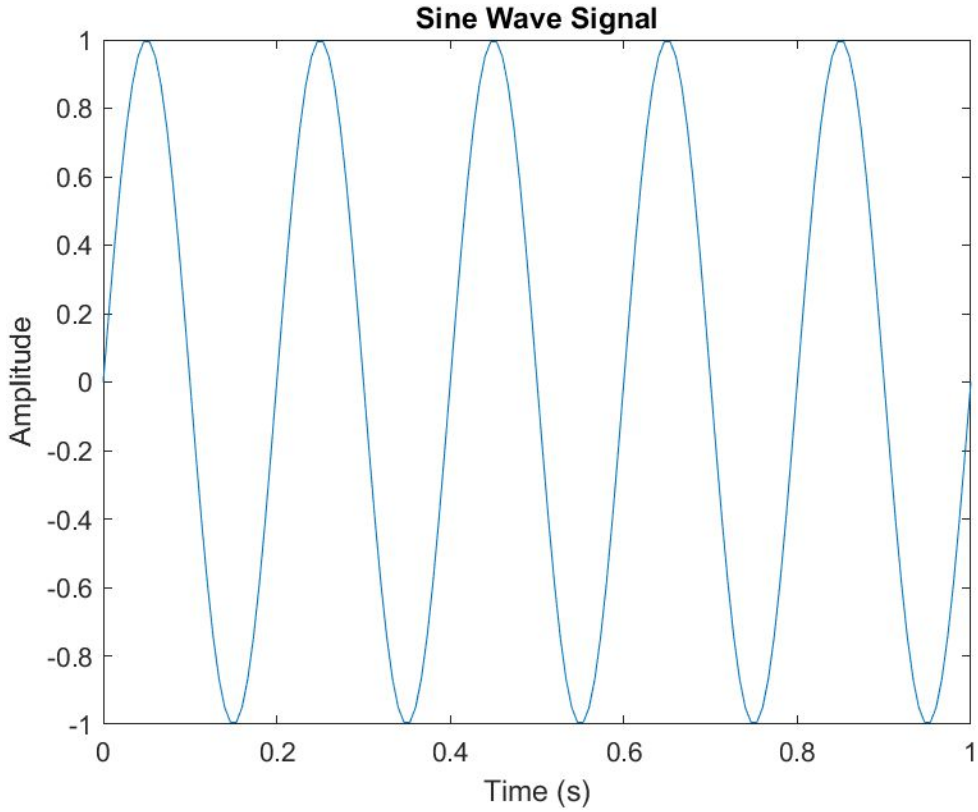
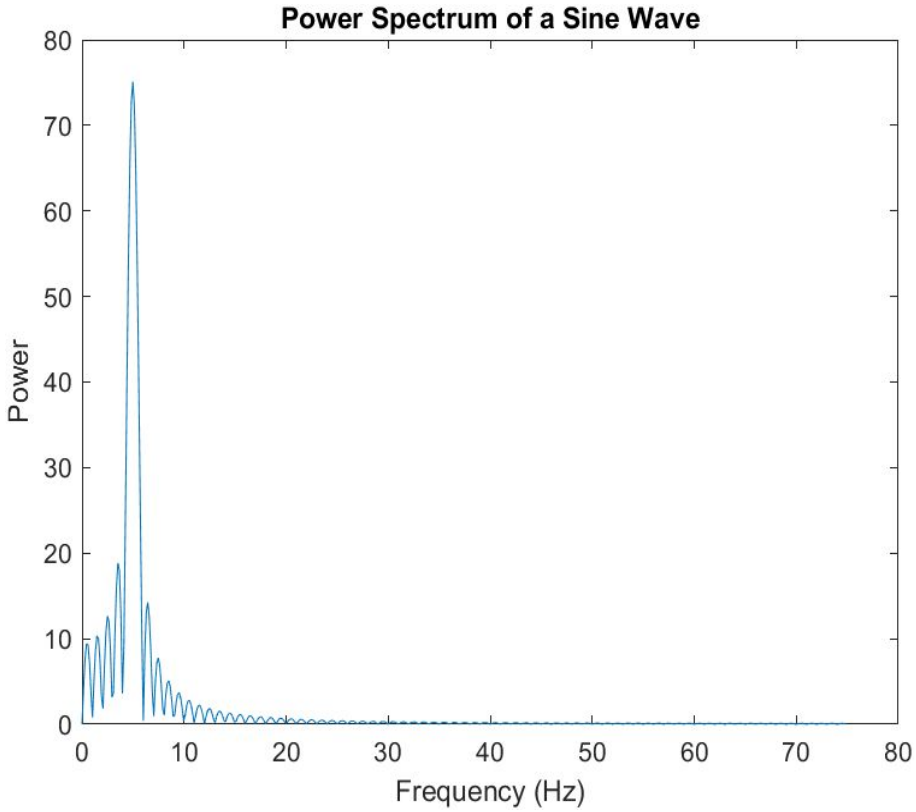
Fast Fourier Transform and MATLAB Implementation

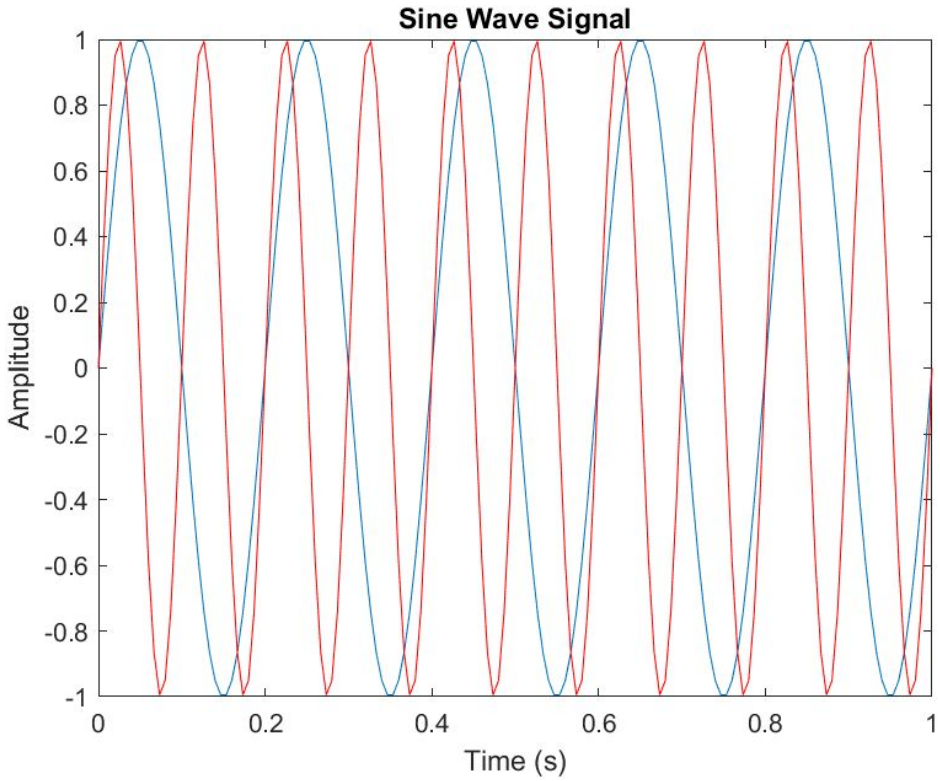
```
clear all

Fs = 150; % Sampling frequency
t = 0:1/Fs:1; % Time vector of 1 second
f_sig = 5; % Create a sine wave of f Hz.
x = sin(2*pi*t*f_sig);

nfft = 1024; % Length of FFT
% Take fft, padding with zeros so that length(X)  is equal to nfft
X = fft(x,nfft);
% FFT is symmetric, throw away second half
X = X(1:nfft/2);
% Take the magnitude of fft of x
mx = abs(X);
% Frequency vector
f = (0:nfft/2-1)*Fs/nfft;

% Generate the plot, title and labels.  figure(1);
plot(t,x);
title('Sine Wave Signal');  xlabel('Time (s)');  ylabel('Amplitude');
figure(2);
plot(f,mx);
title('Power Spectrum of a Sine Wave');  xlabel('Frequency (Hz)');
ylabel('Power');
```

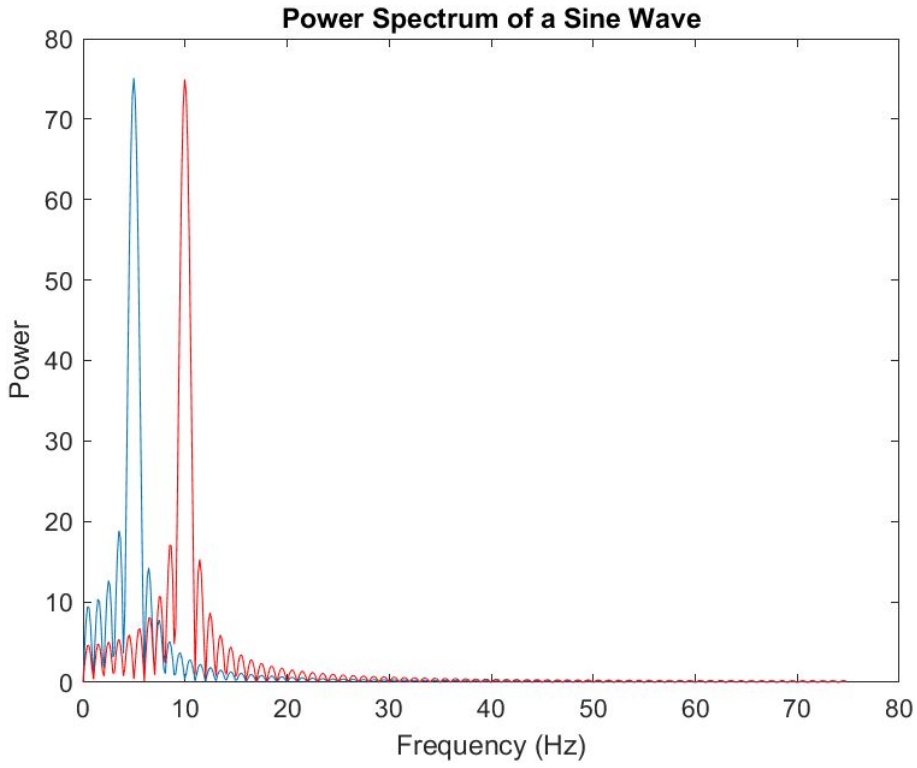
```
x = sin(2*pi*t*5);
```


Sine Wave Signal
Amplitude
Time (s)
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
0.2
0.4
0.6
0.8
1


Power Spectrum of a Sine Wave
Power
Frequency (Hz)
80
70
60
50
40
30
20
10
0
0
10
20
30
40
50
60
70
80

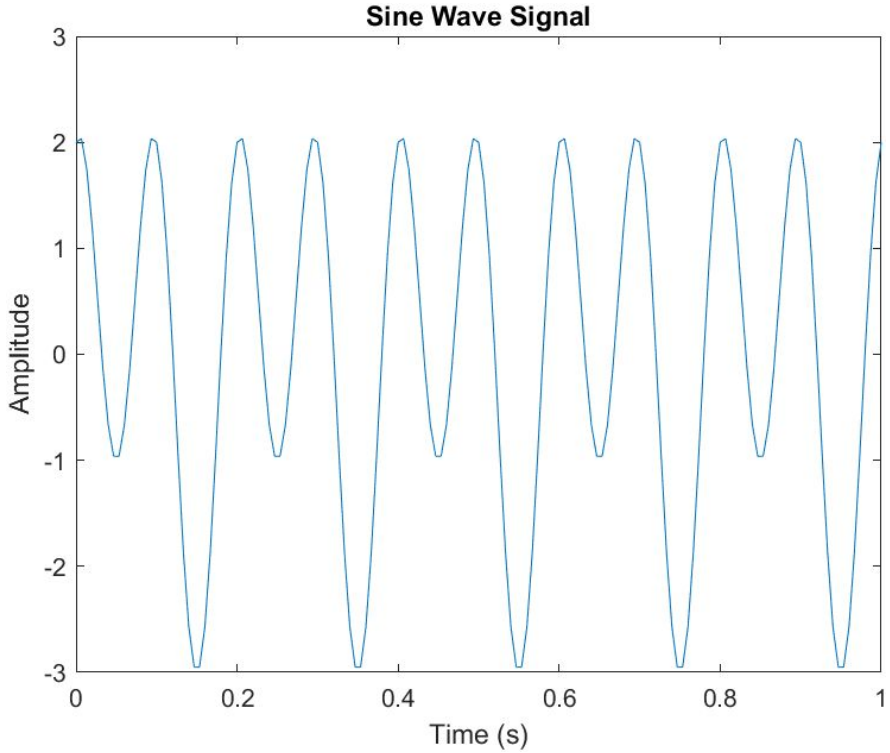
```
x1 = sin(2*pi*t*5);
x2 = sin(2*pi*t*10);
```

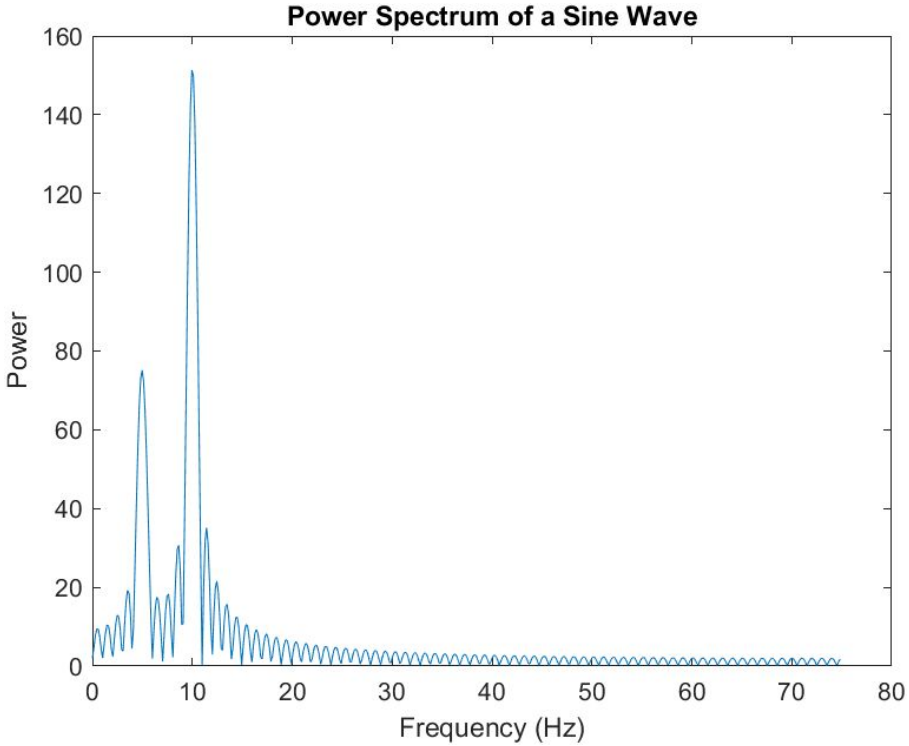




```
clear all

Fs = 150; % Sampling frequency
t = 0:1/Fs:1; % Time vector of 1 second
f_sig = 5; % Create a sine wave of f Hz.
x = sin(2*pi*t*5)+2*cos(2*pi*t*10);

nfft = 1024; % Length of FFT
% Take fft, padding with zeros so that length(X)  is equal to nfft
X = fft(x,nfft);
% FFT is symmetric, throw away second half
X = X(1:nfft/2);
% Take the magnitude of fft of x
mx = abs(X);
% Frequency vector
f = (0:nfft/2-1)*Fs/nfft;

% Generate the plot, title and labels.  figure(1);
plot(t,x);
title('Sine Wave Signal');  xlabel('Time (s)');  ylabel('Amplitude');
figure(2);
plot(f,mx);
title('Power Spectrum of a Sine Wave');  xlabel('Frequency (Hz)');
ylabel('Power');
```

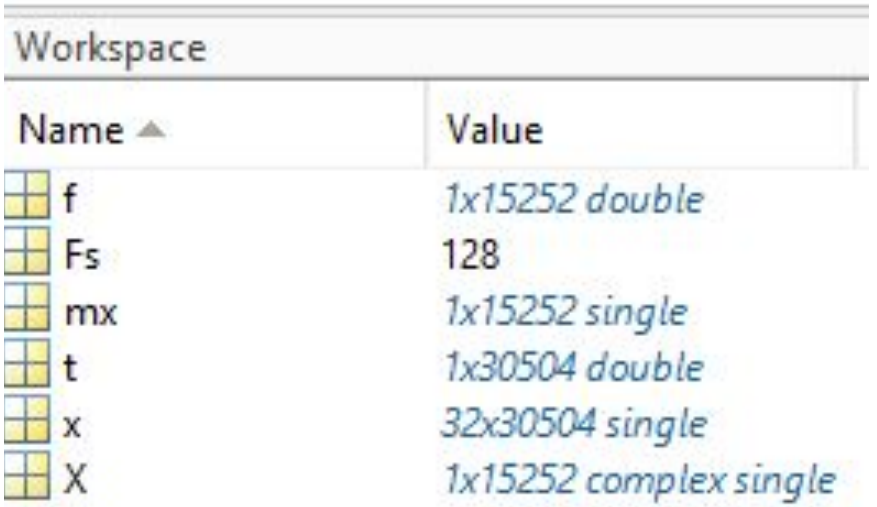
```
x = sin(2*pi*t*5)+2*cos(2*pi*t*10);
```



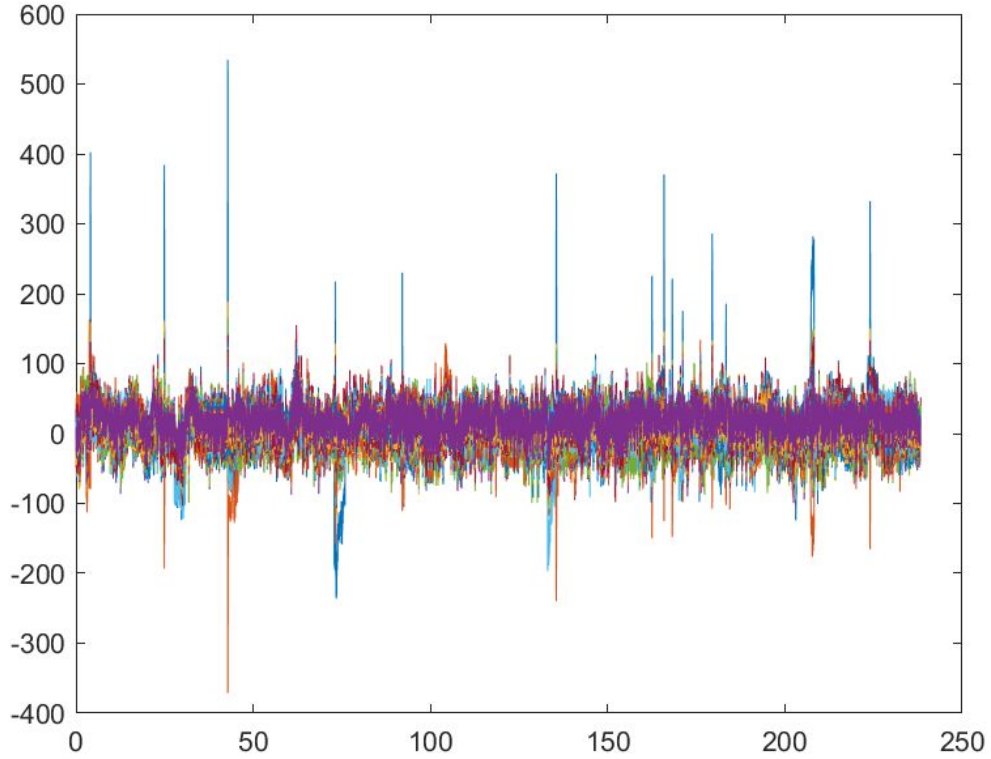


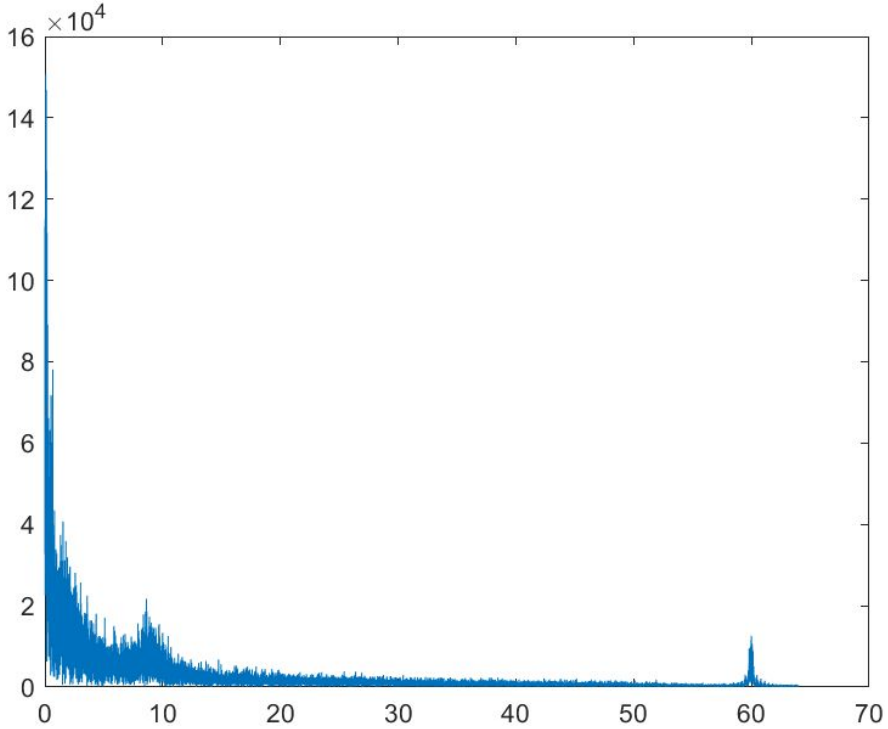
FFT on EEG data

```
clear all

load x
Fs=128;
t = 0:1/Fs:(238.3125-1/Fs);
plot(t,x)

X = fft(x(1,:),30504);
% FFT is symmetric, throw away second half
X = X(1:30504/2);
% Take the magnitude of fft of x
mx = abs(X);
% Frequency vector
f = (0:30504/2-1)*Fs/30504;
figure
plot(f,mx)
```



FFT on EEG data



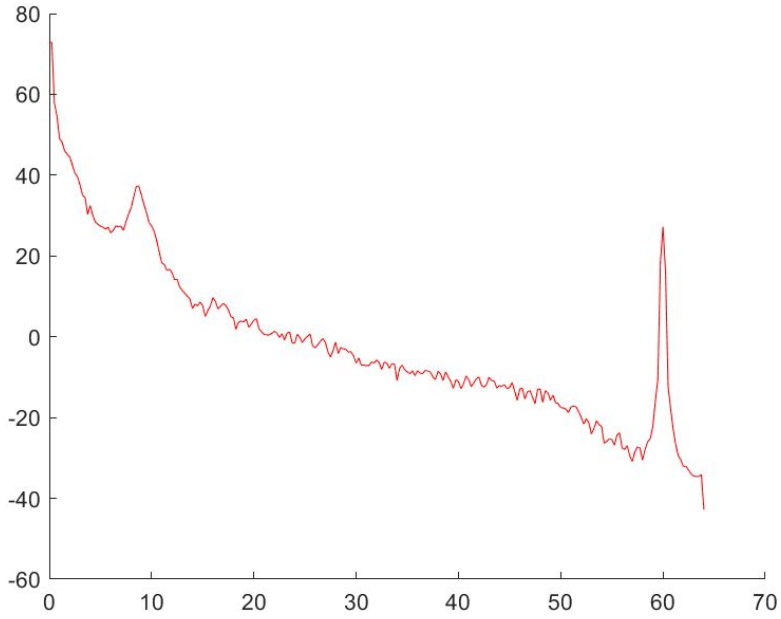


FFT with Hanning Window Function is on EEG data

```
clear all

load x
Fs=128;
[X2,fw]=pwelch(x(1,:),hanning(512),128,512,128);
hold on
plot(fw,10*log(X2),'r')
```

Workspace

Name	Value
Fs	128
fw	257x1 single
x	32x30504 single
X2	257x1 single



Comparing with and without Window Function