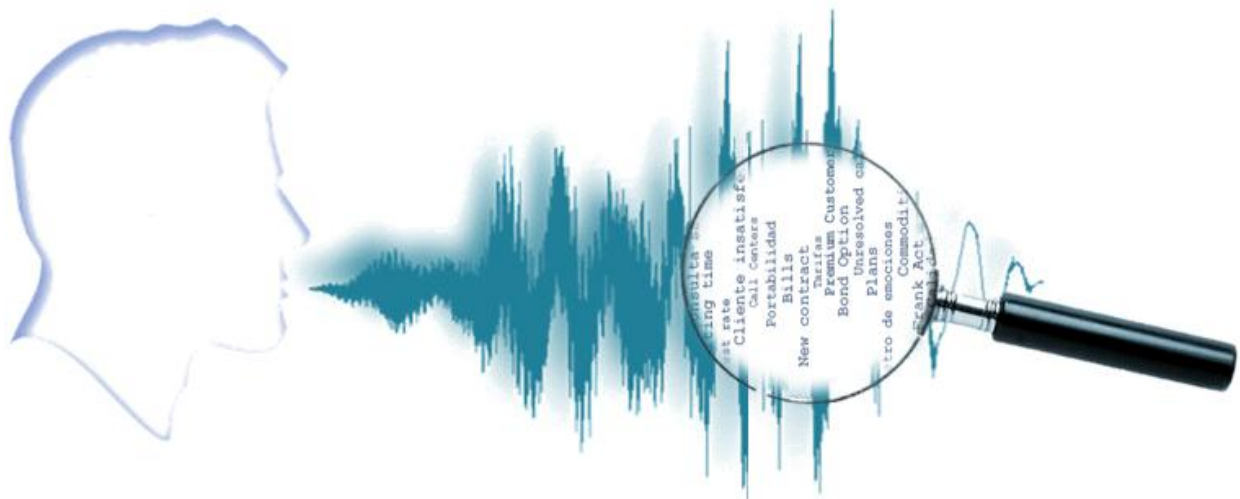


(Tutorial Lessons on Audio & Speech Signal Processing using Matlab)

Lesson 003: Spectral Analysis of Speech/Sound using Matlab



**By: JCBRO Training Division
(JCBRO)**

Introduction

Analysis of any signal can be done mainly in two domains namely as: Time Domain Analysis and Frequency/Spectral Domain analysis. While time domain analysis of a signal tells mainly about the amplitude of the signal, spectral or frequency domain analysis tells about the individual frequency component present in the signal. Frequency domain also tells about the phase information of the system at different frequencies. Fourier Transform also has several application in speech processing. Fourier transform is also helpful in filter designing in order to remove any specific band or frequency component present in the speech signal. Fourier Transform can also act as a feature vector for speaker recognition application. Further applications of FT will be discussed in upcoming lessons.

Fourier Transform

There is Mathematical formula known as Fourier Transform, in order to convert time domain signal into frequency domain signal. Discrete version of the same is known as Discrete Time Fourier Transform also known as DTFT. Mathematical Formula for the DTFT is given as follows:

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}$$

Where $x(n)$ is the discrete sequence and $X(\omega)$ is the continuous function of the frequency. Due to discrete nature of computers this continuous function can not be computed on digital processors and hence discrete version of $X(\omega)$ is required in order to make it computable on digital machines. Discrete version of DTFT or $X(\omega)$ is known as Discrete Fourier Transform (DFT) and it's obtained by uniform sampling of $X(\omega)$. This is given as follows $X(\omega) = X(\omega_k)$. Where $\omega_k = 2\pi k$, $k = 0, 1, \dots, (N - 1)$, where N : no. of samples of $X(\omega)$. In order to avoid aliasing in the reconstructed time domain signal $x(n)$, N should be greater than or equal to the length of $x(n)$. There are several algorithms available for fast computing of DFT and those are known as Fast Fourier Transform (FFT). It should be noted that FFT is complex quantity whose absolute value gives the information about the amplitude of individual frequency component while angle part gives the phase information of the signal.

FFT Computation in Matlab

Matlab provides built in command to find the Fourier transform of the signal. Several other commands are also required in order to plot the FT of the signal. Here, we have provided a short description of those commands. Full and complete description can be found in Matlab Help.

1. **Y = fft(x, n)** : This command is used to determine the FFT of the sequence x and n points. Generally n is the no. greater than length of x. Normally it is chosen in multiple of 2^n .
2. **Y = abs(x)** : This command returns the absolute value of the sequence x.

Matlab Code Example for FFT Plot of Speech Signal

Code for FFT computation and plotting the single sided spectrum of the input sequence is given below. Figure 1 shows the speech signal and it's Fourier transform.

```
%% Read signal  
[data,fs] = wavread('10a07La_boredom');  
  
%% take fourier transform and plot single sided spectrum  
l = length(data);  
NFFT = 2nextpow2(l);  
f = fs/2*linspace(0,1,NFFT/2+1);  
xf = abs(fft(data, NFFT));  
  
plot(f, xf(1:NFFT/2+1));
```

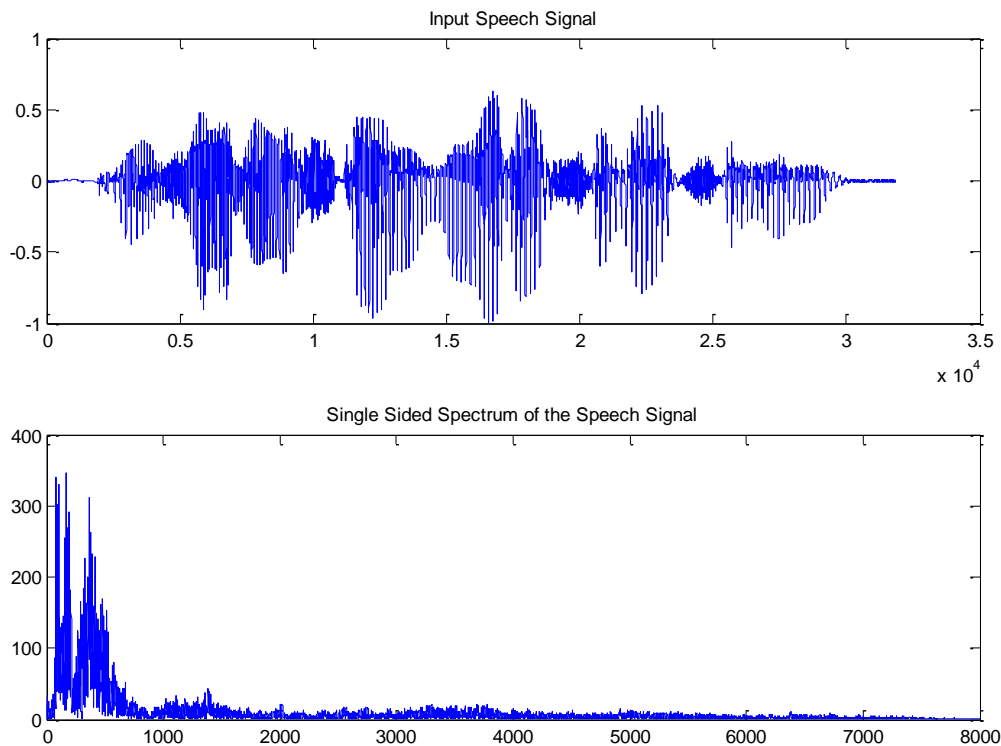


Fig. 1: Input Speech Signal and It's Amplitude Spectrum

Power Spectral Density

Power spectral density (PSD) describes that variation of power present in the signal as a function of frequency. In other words it tells which frequency components contain higher power and which frequency contains lower power. It shows the variation in the form of a graph with frequency of x-axis and power (db/Hz or watt/Hz) on y-axis. There are several ways to compute PSD in theory as well as in Matlab.

1. **PSD estimate using Periodogram Command:** PSD calculation can be done by simply by periodogram command in Matlab. It takes data, window and sampling frequency as input and returns the PSD and frequency of the data. Figure 2 shows the PSD estimate of the input speech signal based on periodogram command in matlab. Matlab code for PSD calculation using periodogram is given as follows:

```
[psdestx, Fxx] = periodogram(data, rectwin(length(data)), length(data), fs);
plot(Fxx, 10*log10(psdestx)); grid on;
xlabel('Hz'); ylabel('Power/Frequency (dB/Hz)');
```

title('Periodogram Power Spectral Density Estimate');

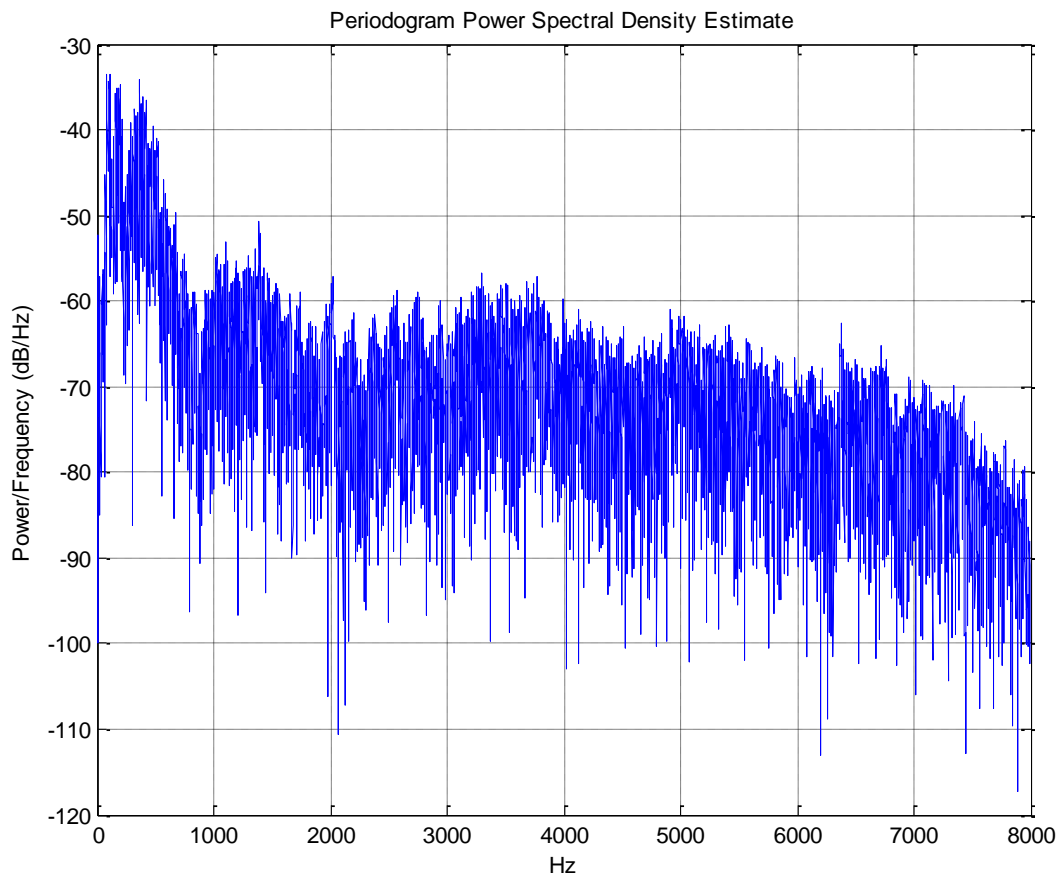


Fig. 2: PSD Estimate of the Input Speech Signal

- 2. Welch's Method for PSD Estimate:** Welch's method is an improved way of estimating the PSD of input Data. It is better than the periodogram method as it reduces the amount of noise in estimated PSD at the cost of frequency resolution. Figure 3 shows the PSD estimate of the input speech signal using Welch's Method. Further, following the Matlab code for doing the same:

```
h = spectrum.welch; % create welch spectrum object  
d = psd(h, data, 'Fs', fs);  
figure;  
plot(d);
```

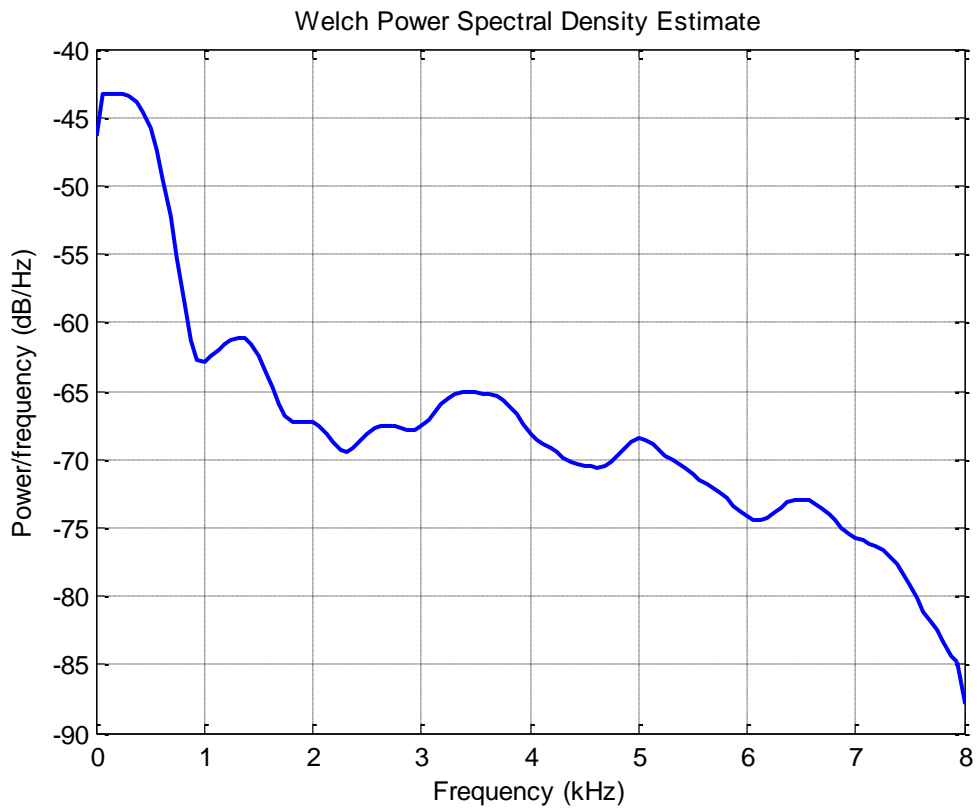


Fig. 3: PSD Estimate of the Input Speech Signal using Welch's Method