EE 565: Position, Navigation, and Timing

Power Spectral Density Estimation

Aly El-Osery Kevin Wedeward

Electrical Engineering Department, New Mexico Tech
Socorro, New Mexico, USA

In Collaboration with
Stephen Bruder
Electrical and Computer Engineering Department
Embry-Riddle Aeronautical Univesity
Prescott, Arizona, USA

March 11, 2020

Sensors suffer from noise effects that can not be removed through calibration, consquently, we need to

- understand the nature of the noise
- be able to extract parameters from actual data
- develop models to mimic noise in simulation to provide performance capabilities

Estimate the distribution of power in a signal. Unfortunately, truth and what is practical cause a problem.

Truth

- Infinitely long.

Estimate the distribution of power in a signal. Unfortunately, truth and what is practical cause a problem.

Truth

- Infinitely long.

Practice

- Finite length.

NEW MEXICO TECH
SCIENCE • ENGINEERING • RESEARCH UNIVERSITY

Estimate the distribution of power in a signal. Unfortunately, truth and what is practical cause a problem.

Truth

- Infinitely long.
- Continuous in time and value.

Practice

- Finite length.

Estimate the distribution of power in a signal. Unfortunately, truth and what is practical cause a problem.

Truth

- Infinitely long.
- Continuous in time and value.

Practice

- Finite length.
- Discrete in time and value.

Estimate the distribution of power in a signal. Unfortunately, truth and what is practical cause a problem.

Truth

- Infinitely long.
- Continuous in time and value.
- Provides true distribution of power.

Practice

- Finite length.
- Discrete in time and value.

Estimate the distribution of power in a signal. Unfortunately, truth and what is practical cause a problem.

Truth

- Infinitely long.
- Continuous in time and value.
- Provides true distribution of power.

Practice

- Finite length.
- Discrete in time and value.
- Only approximation of distribution of power.

Purpose

Estimate the distribution of power in a signal. Unfortunately, truth and what is practical cause a problem.

Truth

- Infinitely long.
- Continuous in time and value.
- Provides true distribution of power.

Practice

- Finite length.
- Discrete in time and value.
- Only approximation of distribution of power.

Let's make it more interesting

Estimate the distribution of power in a signal. Unfortunately, truth and what is practical cause a problem.

Truth

- Infinitely long.
- Continuous in time and value.
- Provides true distribution of power.

Practice

- Finite length.
- Discrete in time and value.
- Only approximation of distribution of power.

Let's make it more interesting

The signal is stochastic in nature.

Assume the voltage across a resistor R is $e(t)$ and is producing a current $i(t)$. The instantaneous power per ohm is $p(t) = e(t)i(t)/R = i^2(t)$.

Total Energy

$$E = \lim_{T\to\infty} \int_{-T}^{T} i^2(t)dt \tag{1}$$

Average Power

$$P = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} i^2(t)dt \tag{2}$$

Total Normalized Energy

$$E \triangleq \lim_{T\to\infty} \int_{-T}^{T} |x(t)|^2 dt = \int_{-\infty}^{\infty} |x(t)|^2 dt \tag{3}$$

Normalized Power

$$P \triangleq \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} |x(t)|^2 dt \tag{4}$$

For Energy Signals

$$\phi(\tau) = \int_{-\infty}^{\infty} x(t)x(t+\tau)dt \tag{5}$$

For Power Signals

$$R(\tau) = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} x(t)x(t+\tau)dt \tag{6}$$

For Periodic Signals

$$R(\tau) = \frac{1}{T_0} \int_{T_0} x(t)x(t+\tau)dt \tag{7}$$

Rayleigh's Energy Theorem or Parseval's theorem

$$E = \int_{-\infty}^{\infty} |x(t)|^2 dt = \int_{-\infty}^{\infty} |X(F)|^2 dF \qquad (8)$$

Energy Spectral Density

$$G(F) \triangleq |X(F)|^2 \qquad (9)$$

with units of *volts*^{2}-*sec*^{2} or, if considered on a per-ohm basis, *watts-sec/Hz=joules/Hz*

$$P = \int_{-\infty}^{\infty} S(F) dF = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} |x(t)|^2 dt \tag{10}$$

where we define $S(F)$ as the power spectral density with units of watts/Hz.

- Define an *experiment* with random *outcome*.
- Mapping of the outcome to a variable $\Rightarrow$ random variable.
- Mapping of the outcome to a function $\Rightarrow$ random function.

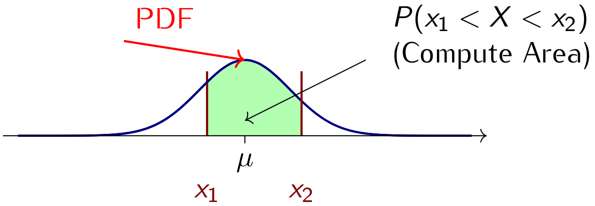
$$F_X(x) = \texttt{probability that } X \leq x = P(X \leq x) \tag{11}$$

Describes the manner random variables take different values.

$$f_X(x) = \frac{dF_X(x)}{dx} \tag{12}$$

and

$$P(x_1 < X \leq x_2) = F_X(x_2) - F_X(x_1) = \int_{x_1}^{x_2} f_X(x)dx \tag{13}$$



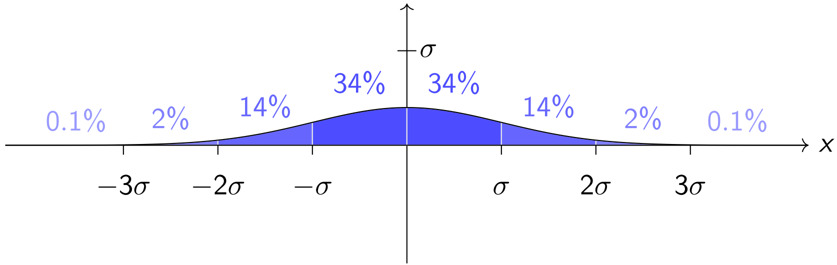
If the random variable X takes a set of discrete values x_i with probability p_i, the pdf of X is expressed in terms of Dirac delta functions, i.e.,

$$f_X(x) = \sum_i p_i \delta(x - x_i) \tag{14}$$

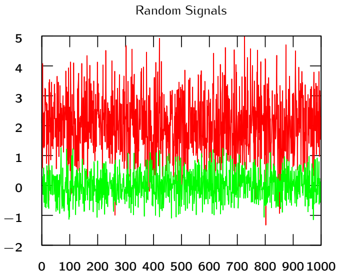
$$f_X(x) = \frac{1}{\sigma_x\sqrt{2\pi}} \exp\left[-\frac{x-\mu_x}{2\sigma_x^2}\right] \tag{15}$$

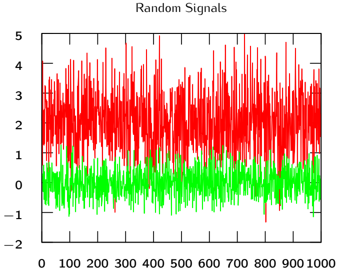
For example if $\sigma_x = \sigma$ and $\mu_x = 0$

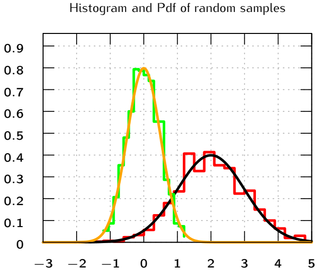
$$\frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-x^2}{2\sigma^2}\right)$$



PDF of White Noise




Random Signals
5
4
3
2
1
0
−1
−2
0
100
200
300
400
500
600
700
800
900
1000


Histogram and Pdf of random samples
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
−3
−2
−1
0
1
2
3
4
5

Mean of a Discrete RV

$$\bar{X} = \mathbb{E}[X] = \sum_{j=1}^{M} x_j P_j \tag{16}$$

Mean of a Continuous RV

$$\bar{X} = \mathbb{E}[X] = \int_{-\infty}^{\infty} x f_X(x) dx \tag{17}$$

Variance of a RV

$$\sigma_X^2 \triangleq \mathbb{E}\left\{[X - \mathbb{E}(X)]^2\right\} = \mathbb{E}[X^2] - \mathbb{E}^2[X] \tag{18}$$

Given a two random variables X and Y.

Covariance

$$\mu_{XY} = \mathbb{E}\left\{[X - \bar{x}][Y - \bar{Y}]\right\} = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y] \tag{19}$$

Correlation Coefficient

$$\rho_{XY} = \frac{\mu_{XY}}{\sigma_X \sigma_Y} \tag{20}$$

Autocorrelation

$$\Gamma_X(\tau) = \mathbb{E}[X(t)X(t+\tau)] \tag{21}$$

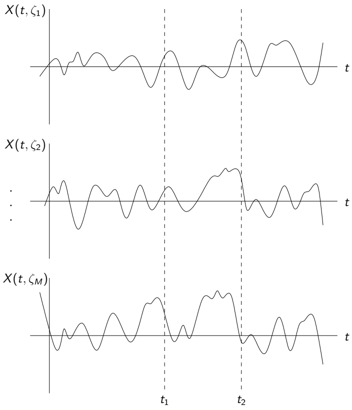


Figure: Sample functions of a random process

- $X(t,\zeta_i)$: sample function.
- The governing experiment: random or stochastic process.
- All sample functions: ensemble.
- $X(t_j,\zeta)$: random variable.

If the joint pdfs depend only on the time difference regardless of the time origin, then the random process is known as *stationary*.

For stationary process means and variances are independent of time and the covariance depends only on the time difference.

If the joint pdfs depends on the time difference but the mean and variances are time-independent, then the random process is known as *wide-sense-stationary*.

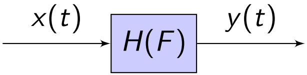
If the time statistics equals ensemble statistics, then the random process is known as *ergodic*.

Any statistic calculated by averaging of all members of an ergodic ensemble at a fixed time can also be calculated by using a single representative waveform and averging over all time.

Given a sample function $X(t, \zeta_i)$ of a random process, we obtain the power spectral density by

$$S(F) \overset{\mathcal{F}}{\longleftrightarrow} \Gamma(\tau) \tag{22}$$

i.e., for a wide sense stationary signal, the power spectral density and autocorrelation are Fourier transform pairs.

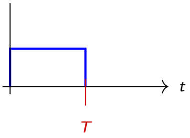


$$S_Y(F) = |H(F)|^2 S_X(F) \tag{23}$$

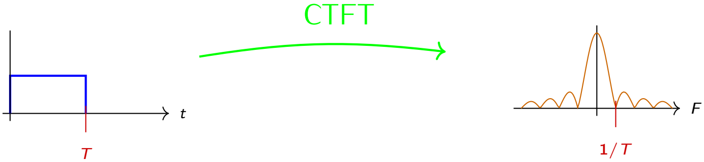
Noise Shaping

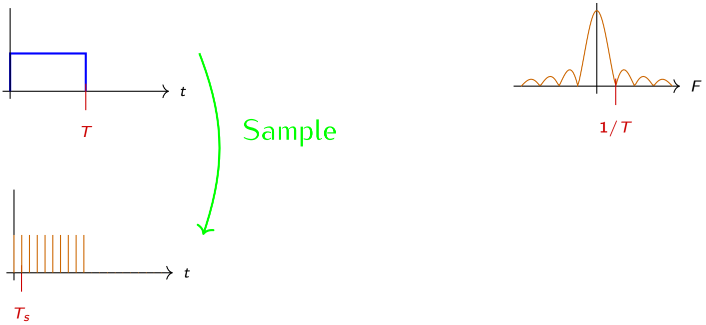
If $x(t)$ is white noise, we can design the filter $h(t)$ to "shape" the noise.

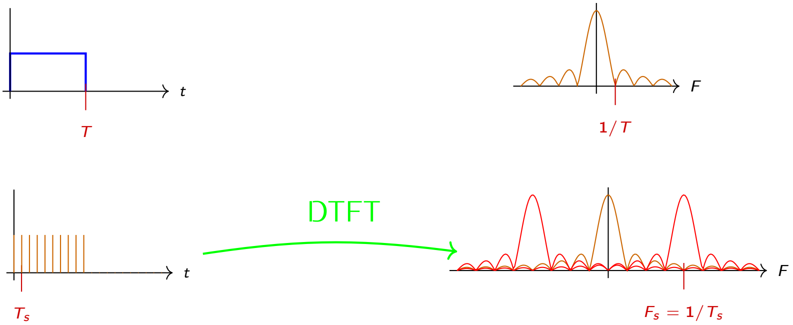
Big Picture

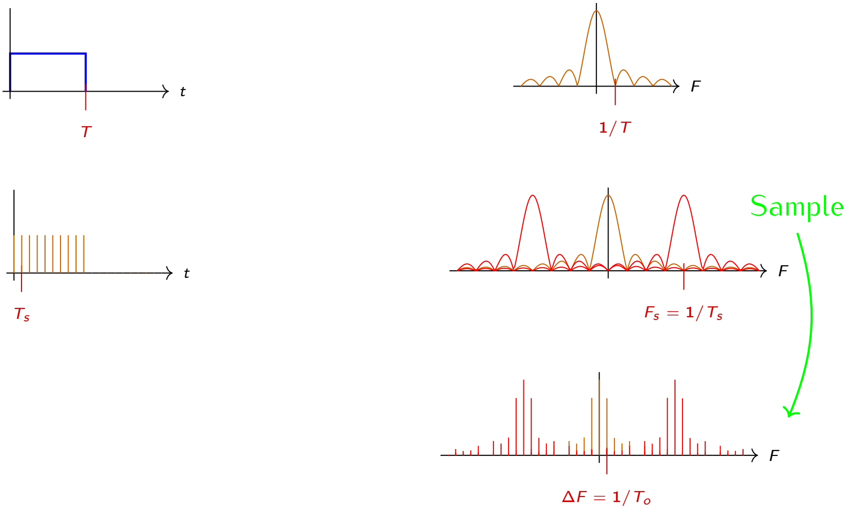


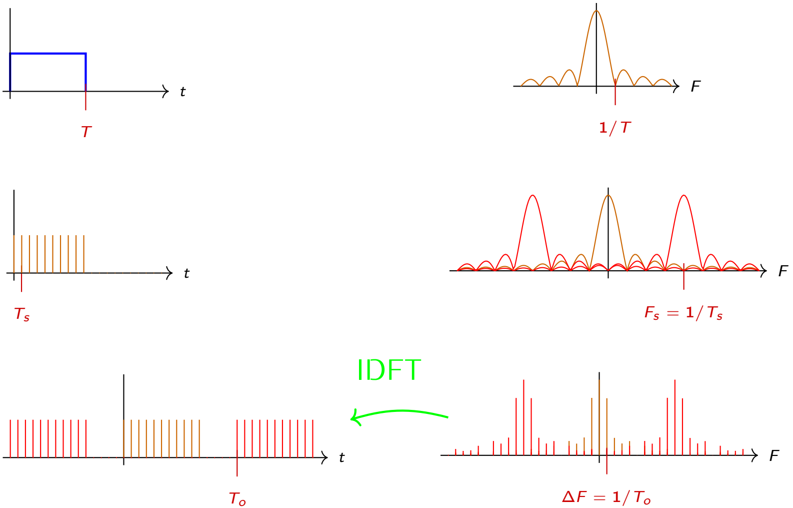
Big Picture




t
T
Sample
t
T_s
F
1/T


t
T
F
1/T
DTFT
t
T_s
F
$F_s = 1/T_s$


t
T
F
1/T
t
T_s
Sample
F
$F_s = 1/T_s$
F
$\Delta F = 1/T_o$



t
T
F
1/T
t
T_s
F
$F_s = 1/T_s$
IDFT
t
T_o
F
$\Delta F = 1/T_o$

- Must sample more than twice bandwidth to avoid aliasing.
- FFT represents a periodic version of the time domain signal $\rightarrow$ could have time domain aliasing.
- Number of points in FFT is the same as number of points in time domain signal.

What we want is

$$\Gamma_X(\tau) = \mathbb{E}[X(t)X(t+\tau)] \xrightarrow{\mathcal{CTFT}} S_X(F)$$

For infinitely long signals.

What we want is

$$\Gamma_X(\tau) = \mathbb{E}[X(t)X(t+\tau)] \xrightarrow{\mathcal{CTFT}} S_X(F)$$

For infinitely long signals.

What we can compute is

$$\gamma_X(m) = \mathbb{E}[X(n)X(n+m)] \xrightarrow{\mathcal{DFT}} P_X(f)$$

For finite length signals.

As $N \to \infty$ and in the mean squared sense

Unbiased

Asymptotically the mean of the estimate approaches the true power.

Variance

Variance of the estimate approaches zero.

Resulting in a consistent estimate of the power spectrum.

Possible PSD Options

Periodogram

computed using $1/N$ times the magnitude squared of the FFT

$$\lim_{N\to\infty} \mathbb{E}[P_X(f)] = S_X(f)$$

$$\lim_{N\to\infty} var[P_X(f)] = S_X^2(f)$$

Periodogram

computed using $1/N$ times the magnitude squared of the FFT

$$\lim_{N\to\infty} \mathbb{E}[P_X(f)] = S_X(f)$$

$$\lim_{N\to\infty} var[P_X(f)] = S_X^2(f)$$

Welch Method

computed by segmenting the data (allowing overlaps), windowing the data in each segment then computing the average of the resultant priodogram

$$\mathbb{E}[P_X(f)] = \frac{1}{2\pi MU} S_X(f) \circledast W(f)$$

$$var[P_X(f)] \approx \frac{9}{8L} S_X^2(f)$$

Assuming data length N, segment length M, Bartlett window, and 50% overlap

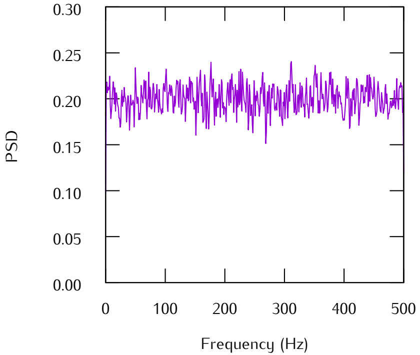
- FFT length $= M = 1.28/\Delta f = 1.28F_s/\Delta F$
- Resulting number of segments $= L = \frac{2N}{M}$
- Length of data collected in sec. $= \frac{1.28L}{2\Delta F}$

```
[Pxx,f] = pwelch(x,window,noverlap,...
                 nfft,fs,'range')
```

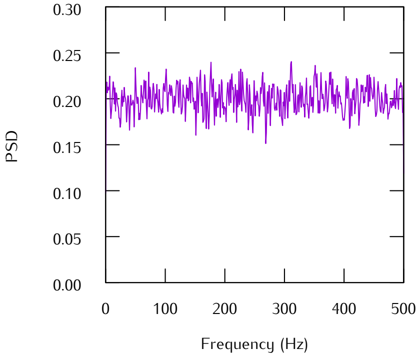
You can use [] in fields that you want the default to be used.

```
Fs = 1000;
x = sqrt(0.1*Fs)*randn(1,100000);
[Pxx,f] = pwelch(x,1024,[],[],Fs,'onesided');
```

```
Fs = 1000;
x = sqrt(0.1*Fs)*randn(1,100000);
[Pxx,f] = pwelch(x,1024,[],[],Fs,'onesided');
```



```
Fs = 1000;
x = sqrt(0.1*Fs)*randn(1,100000);
[Pxx,f] = pwelch(x,1024,[],[],Fs,'onesided');
```

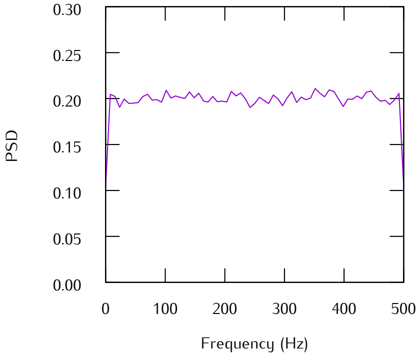


- Variance to high.

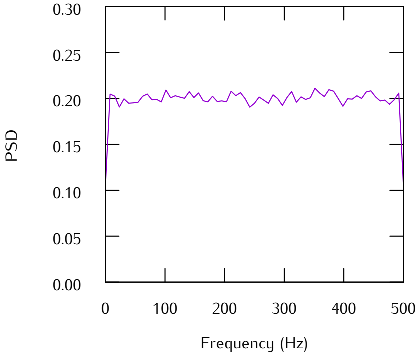
```
[Pxx,f] = pwelch(x,128,[],[],Fs,'onesided')
```

pwelch Function - WGN signal

```
[Pxx,f] = pwelch(x,128,[],[],Fs,'onesided')
```



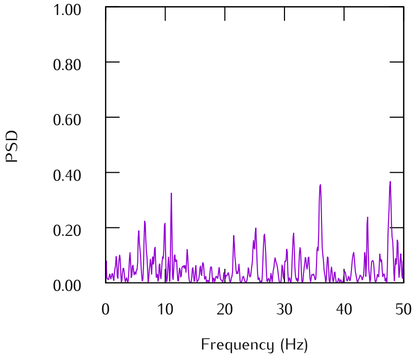
```
[Pxx,f] = pwelch(x,128,[],[],Fs,'onesided')
```



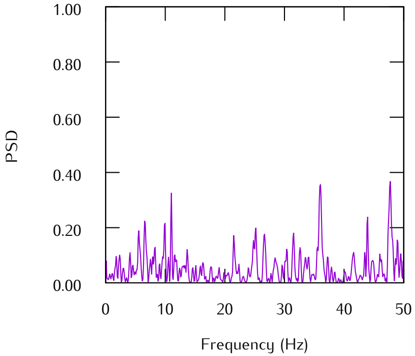
- Reduced window size.
- Variance is now smaller.

```
Fs = 100;    t = 0:1/Fs:5;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,1024,[],[],Fs,'onesided');
```

```
Fs = 100;   t = 0:1/Fs:5;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,1024,[],[],Fs,'onesided');
```



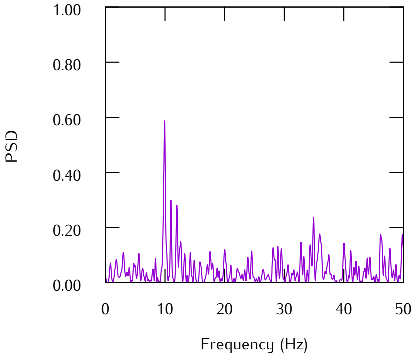
```
Fs = 100;   t = 0:1/Fs:5;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,1024,[],[],Fs,'onesided');
```



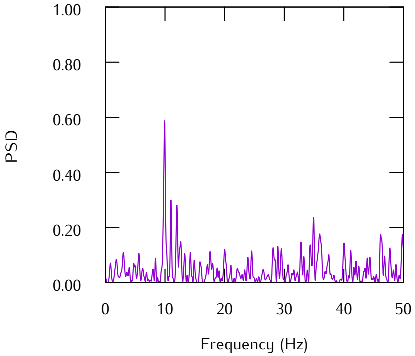
- Window larger than length of data.
- Frequency components can't be resolved.
- Variance high.

```
Fs = 100;   t = 0:1/Fs:5;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,1024,[],4096,Fs,'onesided');
```

```
Fs = 100;   t = 0:1/Fs:5;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
          sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,1024,[],4096,Fs,'onesided');
```



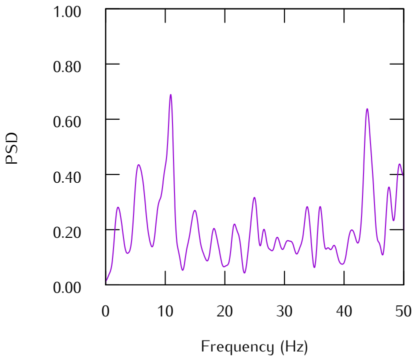
```
Fs = 100;   t = 0:1/Fs:5;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,1024,[],4096,Fs,'onesided');
```



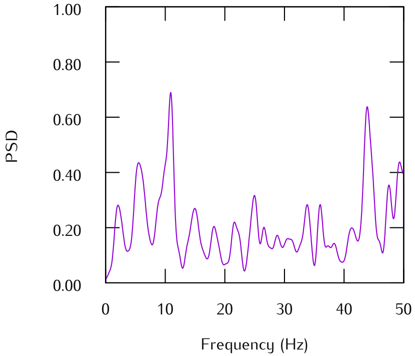
- As expected increasing `nFFT` does not help.

```
Fs = 100;   t = 0:1/Fs:5;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,128,[],4096,Fs,'onesided');
```

```
Fs = 100;    t = 0:1/Fs:5;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,128,[],4096,Fs,'onesided');
```



```
Fs = 100;   t = 0:1/Fs:5;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,128,[],4096,Fs,'onesided');
```

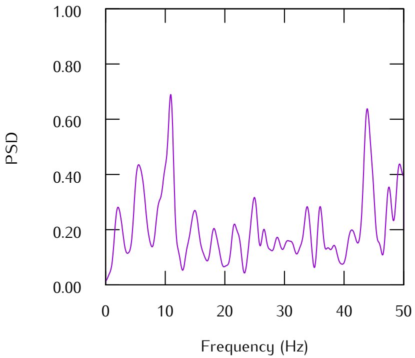


- Decreasing the window size decreases the variance.
- Still can't resolve the two frequencies.

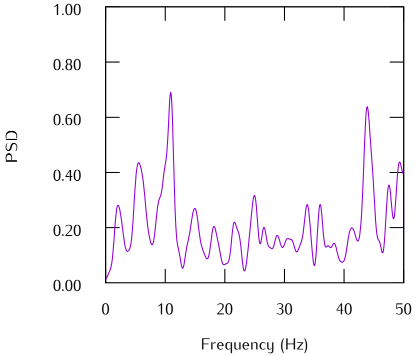
pwelch Function - cos + WGN signal

```
Fs = 100;    t = 0:1/Fs:50;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,128,[],4096,Fs,'onesided');
```

```
Fs = 100;   t = 0:1/Fs:50;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
          sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,128,[],4096,Fs,'onesided');
```



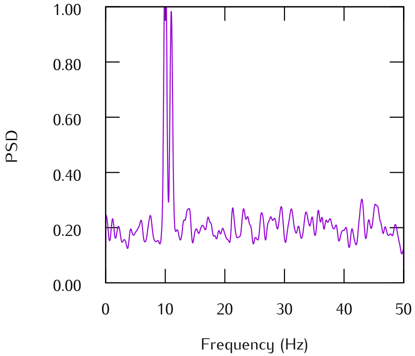
```
Fs = 100;   t = 0:1/Fs:50;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,128,[],4096,Fs,'onesided');
```



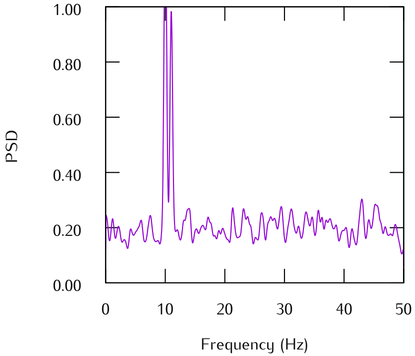
- Length of data sequence must be increased.
- Still can't resolve the two frequencies as the window size is too small.

```
Fs = 100;   t = 0:1/Fs:50;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,256,[],4096,Fs,'onesided');
```

```
Fs = 100;   t = 0:1/Fs:50;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
          sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,256,[],4096,Fs,'onesided');
```



```
Fs = 100;   t = 0:1/Fs:50;
x = cos(2*pi*10*t)+cos(2*pi*11*t)+...
        sqrt(0.1*Fs)*randn(1,length(t));
[Pxx,f] = pwelch(x,256,[],4096,Fs,'onesided');
```



- Now we can resolve the two frequencies.

- The length of the data sequence determines the maximum resolution that can be observed.
- Increasing the window length of each segment in the data increases the resolution.
- Decreasing the window length of each segment in the data decreases the variance of the estimate.
- `nFFT` only affects the amount of details shown and not the resolution.