Estimation of spectral density

Let $x_t, t = 0, 1, 2, \ldots n-1$ be a sequence of data. Compute the Fourier transform,

$$\mathcal{X}(f) = \sum_{t=0}^{n-1} x_t e^{-i2\pi ft}.$$

The *periodogram* is defined as

$$\widehat{R}_x(f) = \frac{1}{n}|\mathcal{X}(f)|^2,$$

and is an estimate of the spectral density.

The periodogram was invented by Sir Arthur Schuster already in 1898, see the famous paper - "On the investigation of hidden periodicities with application to a supposed 26 day period of meteorological phenomena," Terrestrial Magnetism, 3, 13-41, 1898. This was long before the invention of the computers, so the calculations were made by hand......

The DFT and FFT

With the Discrete Fourier Transform (DFT),

$$\mathcal{X}(\frac{k}{N}) = \sum_{t=0}^{n-1} x_t e^{-i2\pi \frac{k}{N} t},$$

where $k = 0, 1, \ldots, N-1$, the periodogram can be computed as,

$$\widehat{R}_x(\frac{k}{N}) = \frac{1}{n} |\mathcal{X}(\frac{k}{N})|^2.$$

The calculations are often made using the Fast Fourier Transform (FFT) algorithm, with $N = 2^i$ for any integer i where $N \geq n$.

Example

The DFT of a realization of the random harmonic function $X_t = A\cos(2\pi\frac{k_0}{N}t + \phi)$, $t = 0, 1, 2, \ldots N-1$, where $k_0 < N/2$ is

$$\mathcal{X}(\frac{k}{N}) = \frac{A \cdot N}{2} e^{i\phi}, \quad k = k_0 \text{ and } k = N - k_0,$$

and zero for other values $0 \leq k \leq N-1$. The component at $k = N - k_0$ is actually located at $k = -k_0$ but using the DFT and FFT, frequencies $-0.5 < f < 0$ are shifted to be located $0.5 < f < 1$.The estimated spectral density is

$$\widehat{R}_x(\frac{k}{N}) = \frac{1}{N}|\mathcal{X}(\frac{k}{N})|^2 = \frac{A^2 \cdot N}{4}, \quad k = k_0 \text{ and } k = N - k_0,$$

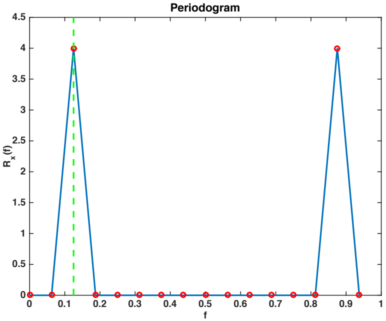
and zero for other values $0 \leq k \leq N-1$. For proof see section 4.3.2 p.95.

Example

The periodogram of a discrete time sequence $x_t = \cos(2\pi f_0 t)$, $t = 0, 1, \ldots, N-1$, with frequency $f_0 = 0.125$ and $N = 16$ is calculated. The result is shown below where all values are zero except for $\widehat{R}_x(\frac{2}{16}) = \widehat{R}_x(\frac{14}{16}) = A^2 \cdot 16/4 = 4$ as $A = 1$. As already discussed, the component at $f = \frac{14}{16}$ actually comes from $f = -\frac{2}{16}$, i.e. the negative complex exponential component of the cosine-function.

Matlab code:

```
x=cos(2*pi*0.125*[0:N-1]');
X=fft(x);
stem([0:N-1]/N,1/N*abs(X).^ 2);
```



Matlab-built in periodogram

You might prefer to use built-in Matlab functions. Then it is good to know what they actually calculate. This is not always obvious from the description...And sometimes, the default option is not what you expect. We will study a simple example of this.

Syntax

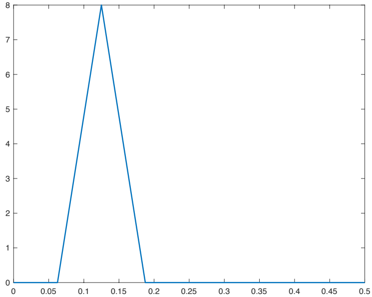
```
pxx = periodogram(x)
pxx = periodogram(x,window)
pxx = periodogram(x,window,nfft)

[pxx,w] = periodogram( __ )
[pxx,f] = periodogram( __ ,fs)

[pxx,w] = periodogram(x,window,w)
[pxx,f] = periodogram(x,window,f,fs)

[ __ ] = periodogram(x,window, __ ,freqrange)
[ __ ] = periodogram(x,window, __ ,spectrumtype)

[ __ ,pxxc] = periodogram( __ ,'ConfidenceLevel',probability)

[rpxx,f] = periodogram( __ ,'reassigned')
[rpxx,f,pxx,fc] = periodogram( __ ,'reassigned')
```

Example: Matlab periodogram

The 'one-sided' periodogram is the default option if you have a real-valued signal input. Note that all power is then moved to the positive side giving a double height of the peak in the above cosine example!

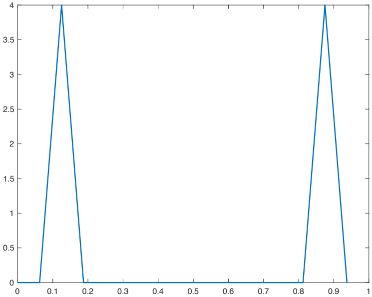
Matlab code:

```
[P,f]=periodogram(x,[],16,1);
plot(f,P)
```



Example

If we instead specify the 'two-sided' periodogram, the power is equally spread at the positive and negative (translated) frequencies, and the result is exactly the same as computing the actual periodogram according to theory.



Matlab code:

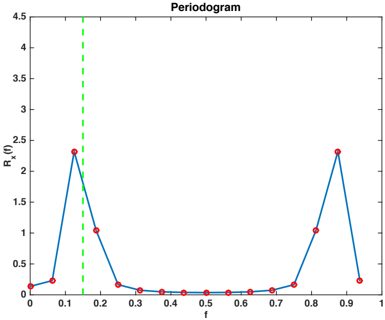
```
[P,f]=periodogram(x,[],'twosided',16,1);
plot(f,P)
```

Another example: A new cosine

If the frequency of the random harmonic function is slightly changed, e.g., to $f_0 = 0.15$, the view changes considerably. Obviously, we can no longer estimate the power from the peak value, as this is no longer at the level of 4, which is the correct power, similar as in the above example.

Matlab code:

```
x=cos(2*pi*0.15*[0:N-1]');
X=fft(x);
stem([0:N-1]/N,1/N*abs(X).^ 2);
```



Zero-padding

We use zero-padding (noll-utfyllnad). What this means is that we extend the signal length with zeros (which does not actually change the signal) but the number of frequency values for which the DFT is calculated increases. E.g., we extend the signal length so the total length is $N = 64$, and thereby we will also use $N = 64$ frequency values, located at frequencies $f = k/N$, $k = 0 \dots 63$ for the periodogram. The resulting estimate is close to the actual spectral density of the cosine function. Why this no longer is just a single peak at the exact cosine frequency, we will come back to in connection to spectral estimation and window techniques.

Matlab code:

```
X=fft(x,64)
plot([0:63]/64,1/16*abs(X).^2)
```