# Time Series Analysis

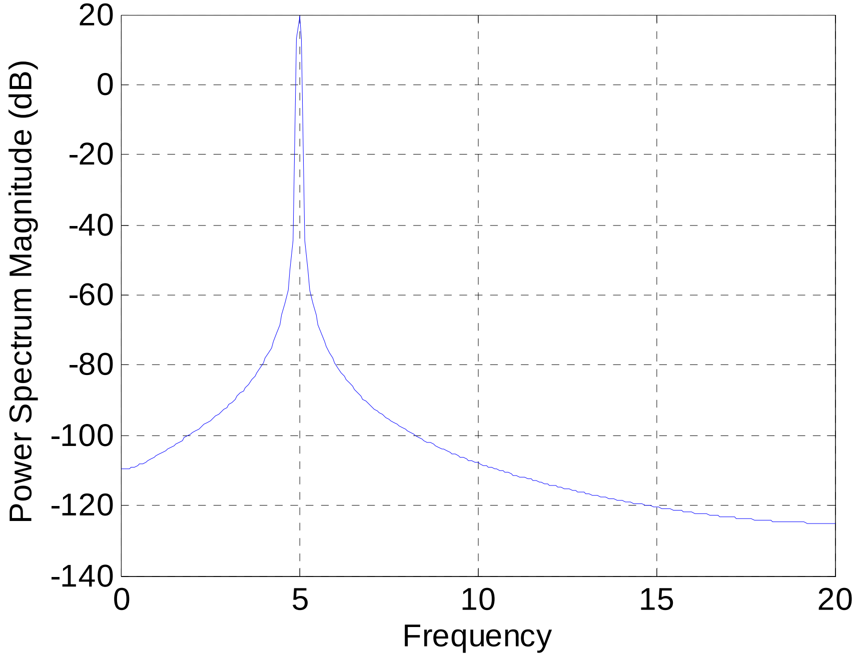
Session II

# Outline

- spectral analysis
- FFT
- complex numbers
- periodogram
- power spectrum
- windowing
- coherence

# Spectral Analysis

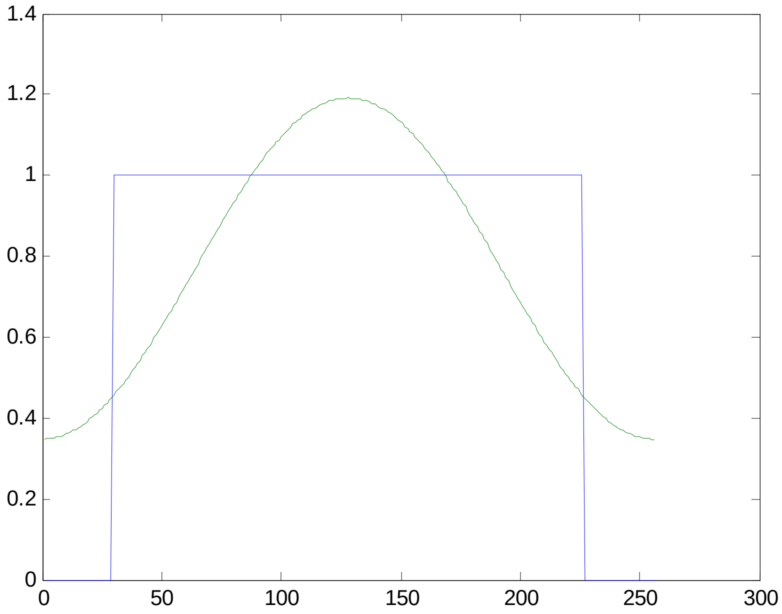
- psd(signal,512,sf,[],256,'mean')
- what are the parameter?



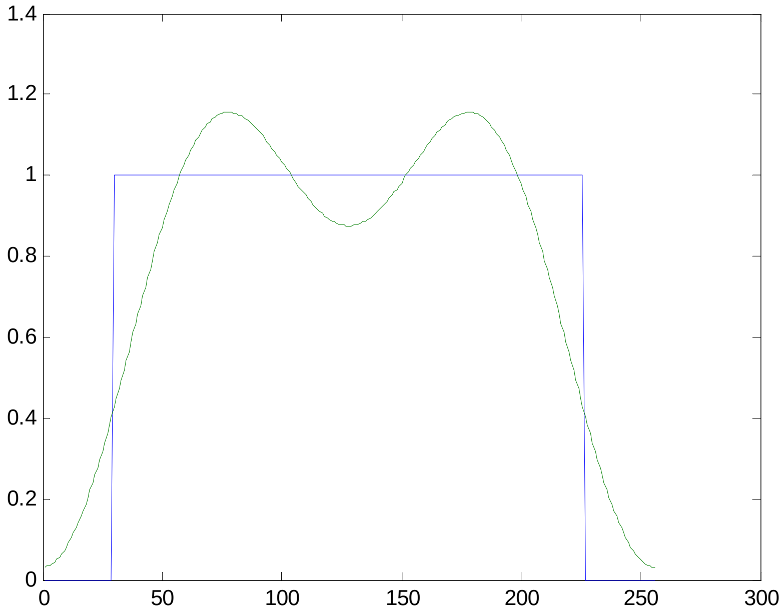
# Fourier transform – the idea

- fitting a function with sinusoids
- transformation of time series to frequency domain

1 functions

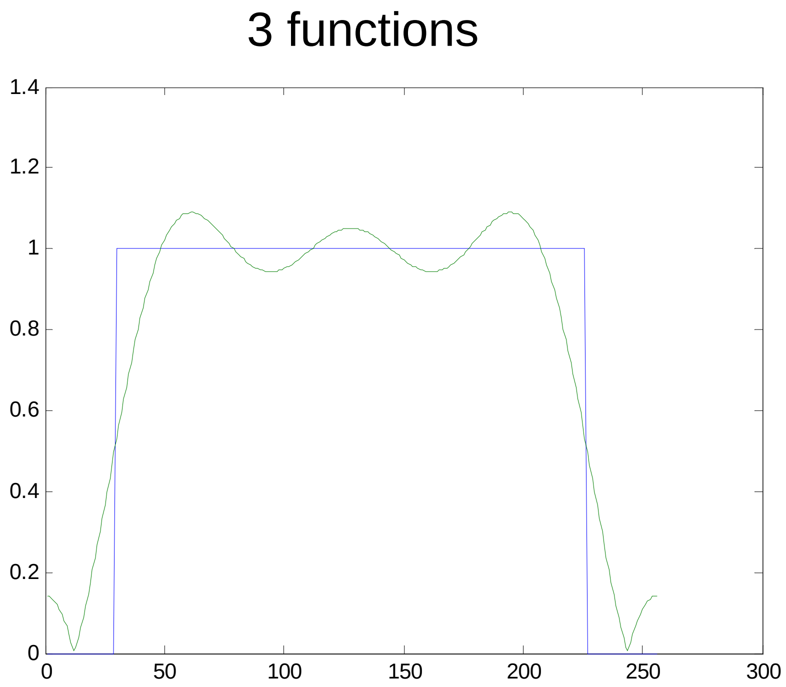


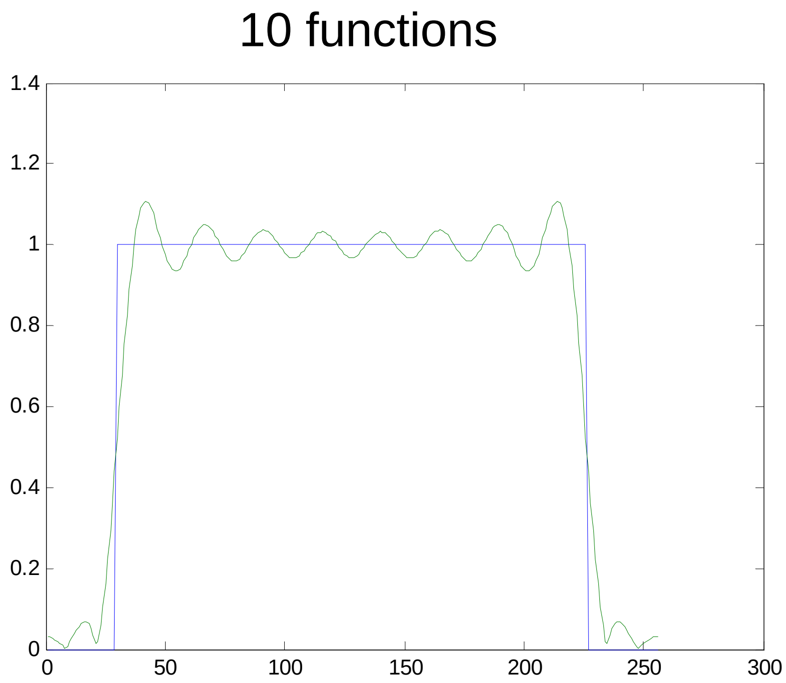

2 functions



# Fourier transform – the idea

- fitting a function with sinusoids
- transformation of time series to frequency domain



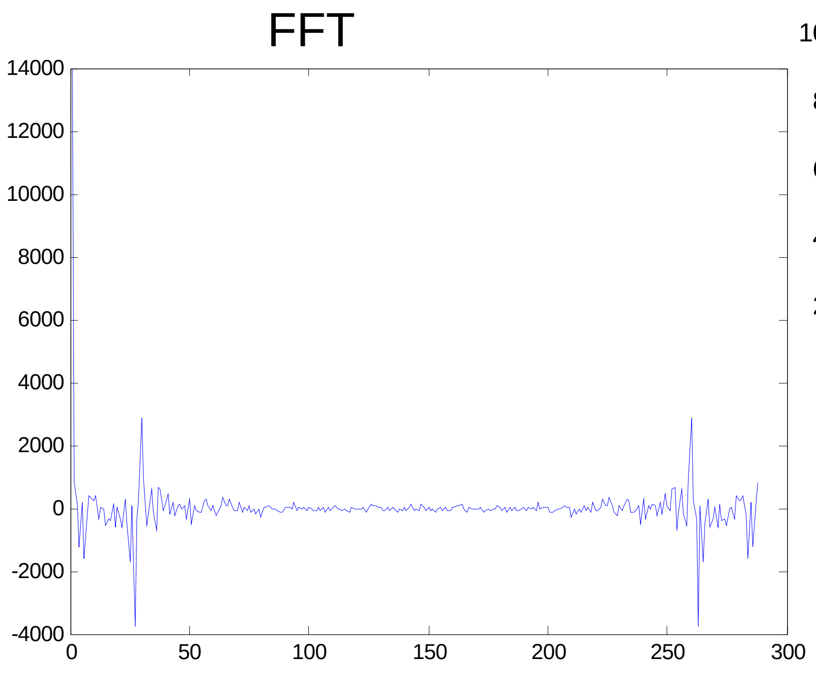



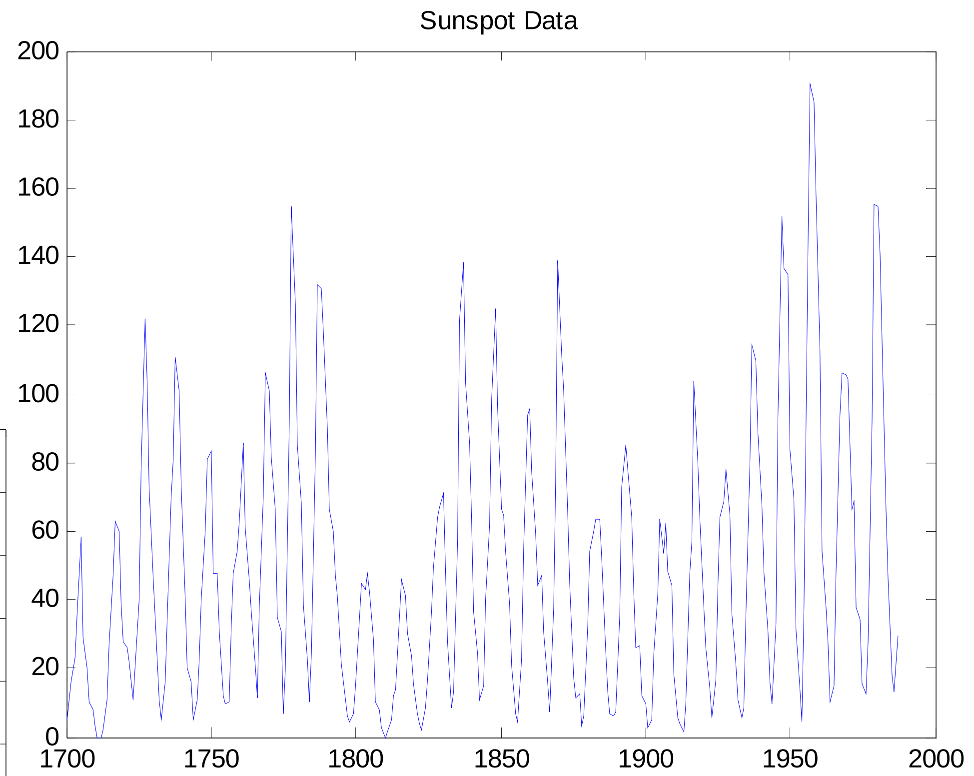
# FFT- general remarks

- Fast Fourier Transform
- optimised algorithm
- 2^N number of samples (128,256,512,1024)

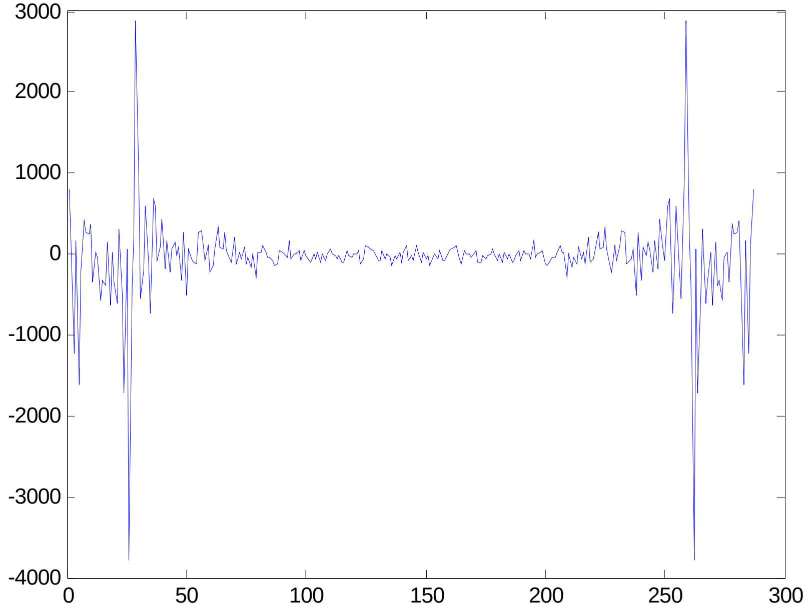
# FFT in matlab

- Example: sunspot data
- Y = fft(sunspot);
- Y(1)=[]; %just the sum

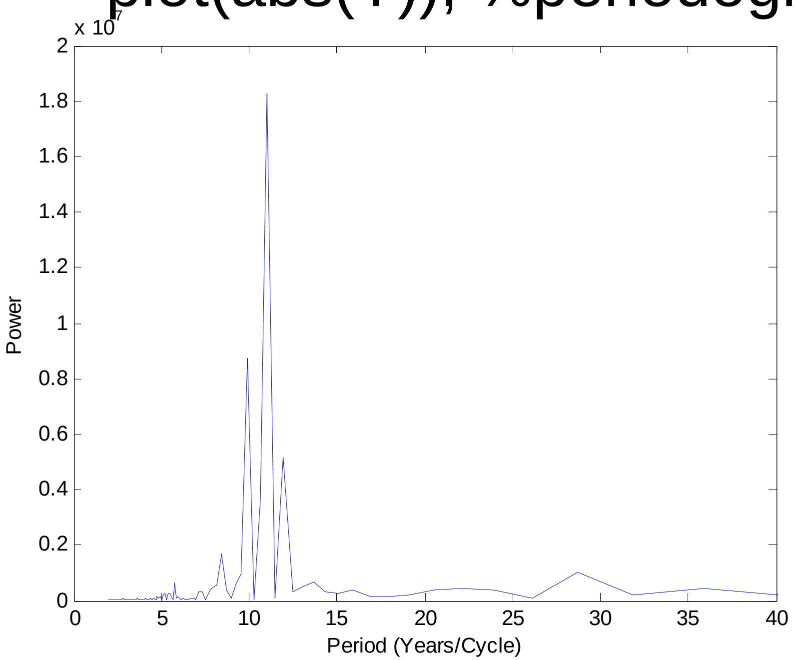


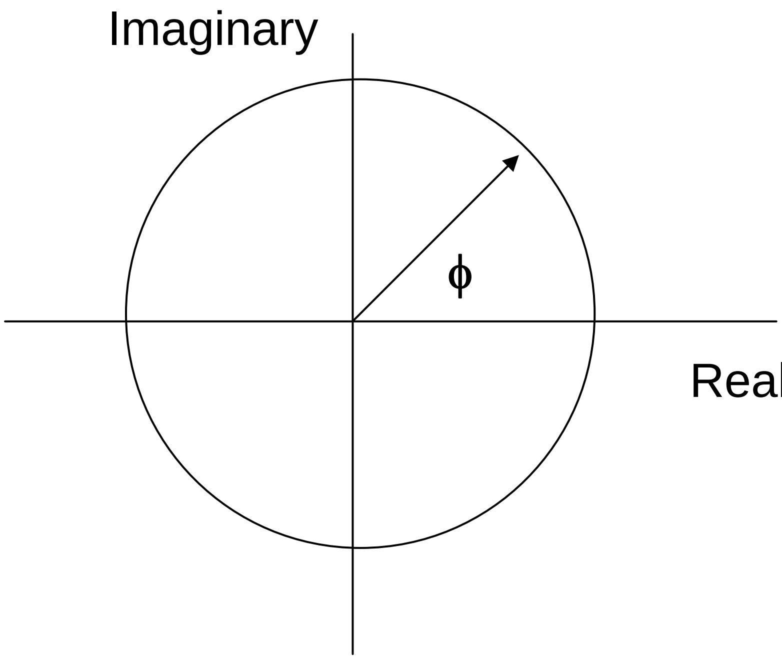
# FFT in matlab

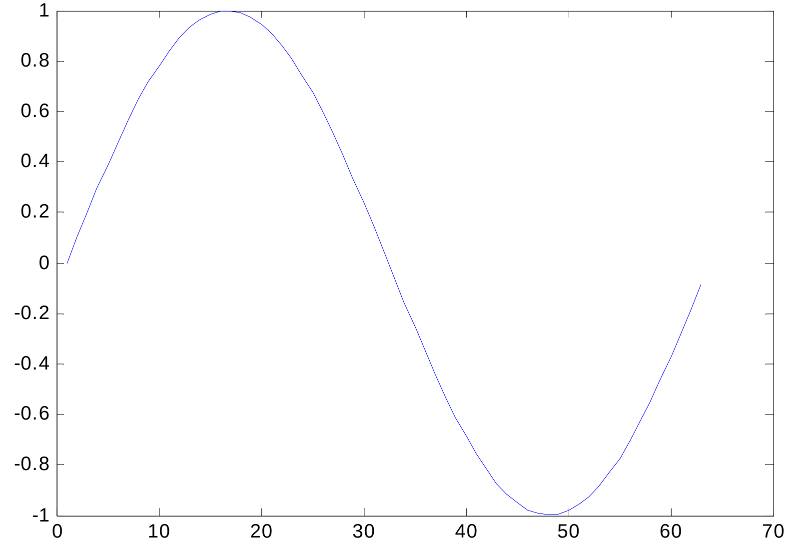



- symmetric
- remove negative frequencies
- n=length(Y);
- Y=Y(1:floor(n/2));
- plot(abs(Y)); %periodogram



# complex numbers




amplitude: length of the arrow
phase: $\phi$

# Spectral analysis with FFT

- Exercise:
- type “edit fftdemo” and do the steps in the script

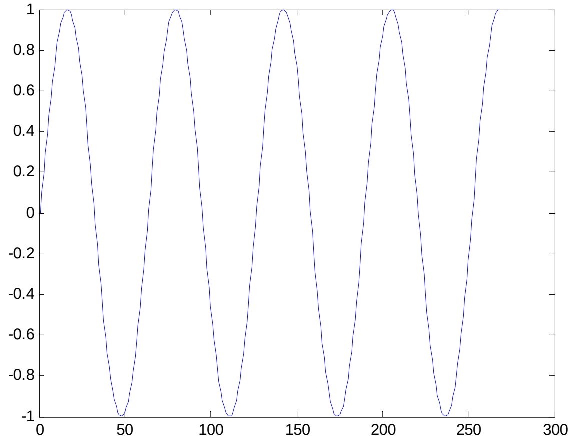
# Problems with periodogram

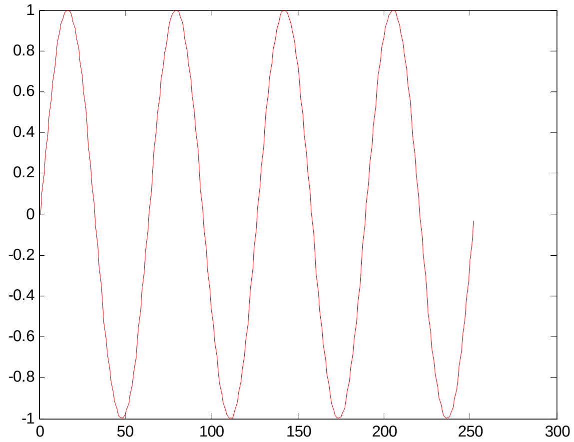
- Leakage
- Accuracy

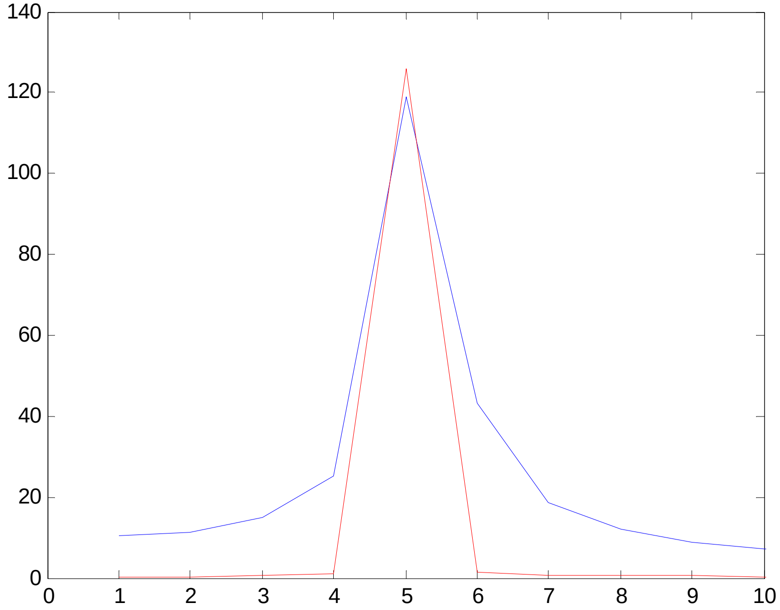
# Leakage

TS

FFT



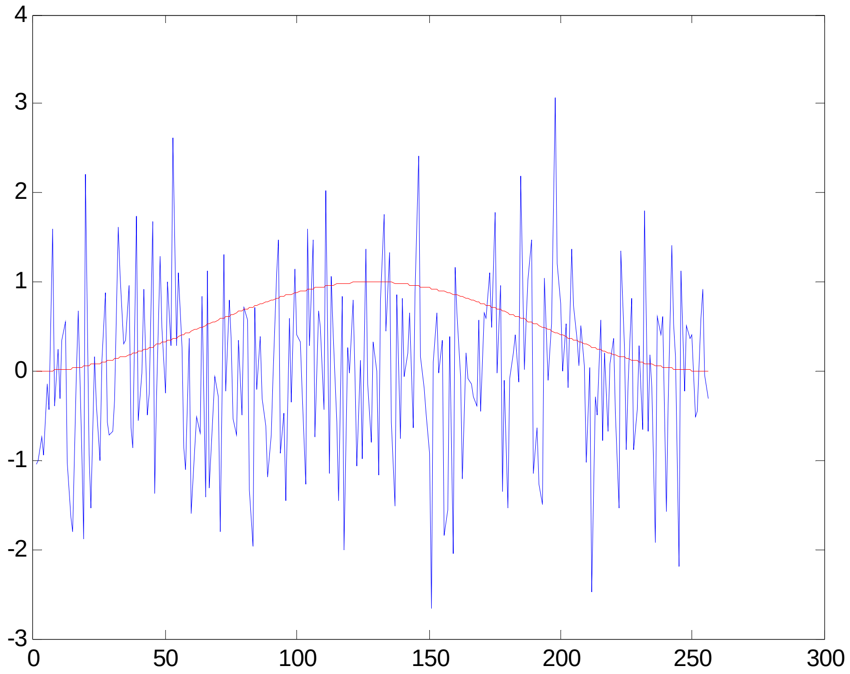



# Leakage

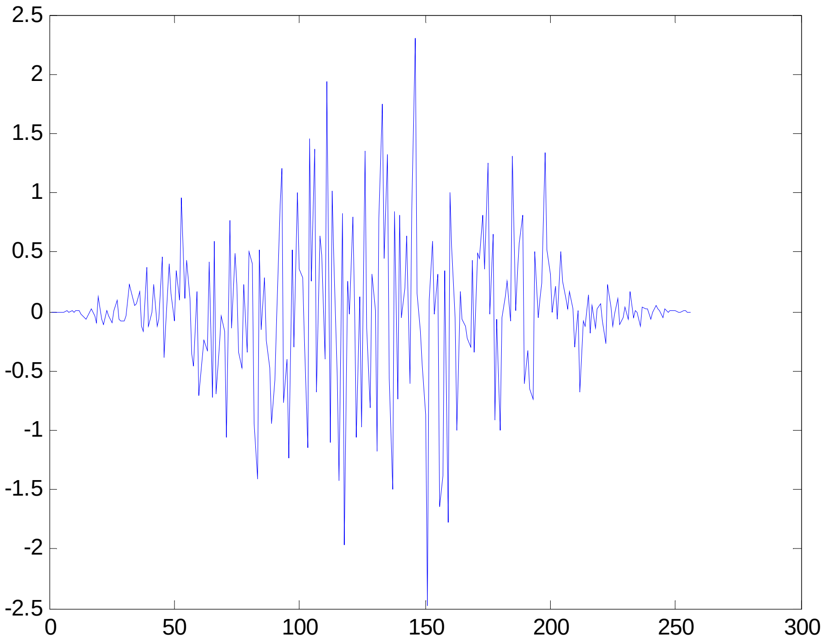
- occurs for non-periodic signals
- non-periodic signals violate FFT assumptions
- problem for real-world signals
- solution: windowing

# Windowing

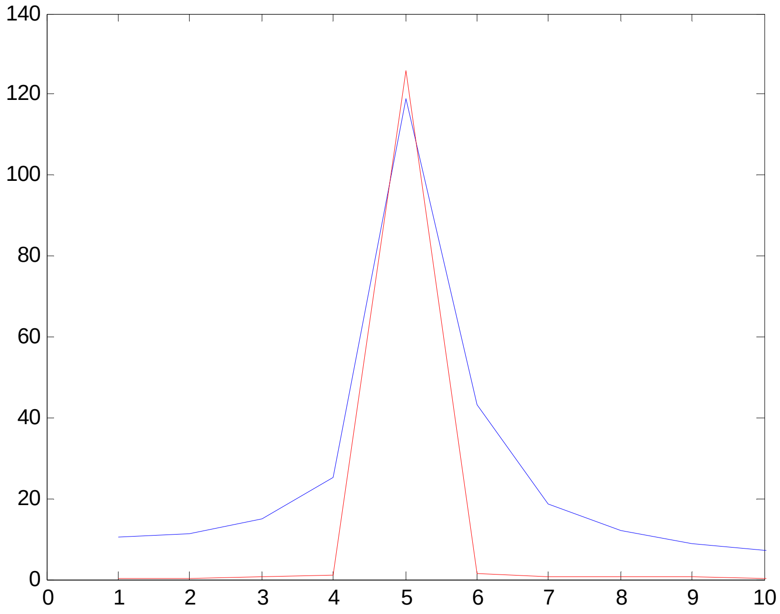
- purpose: make signal periodic (zero at beginning and end)

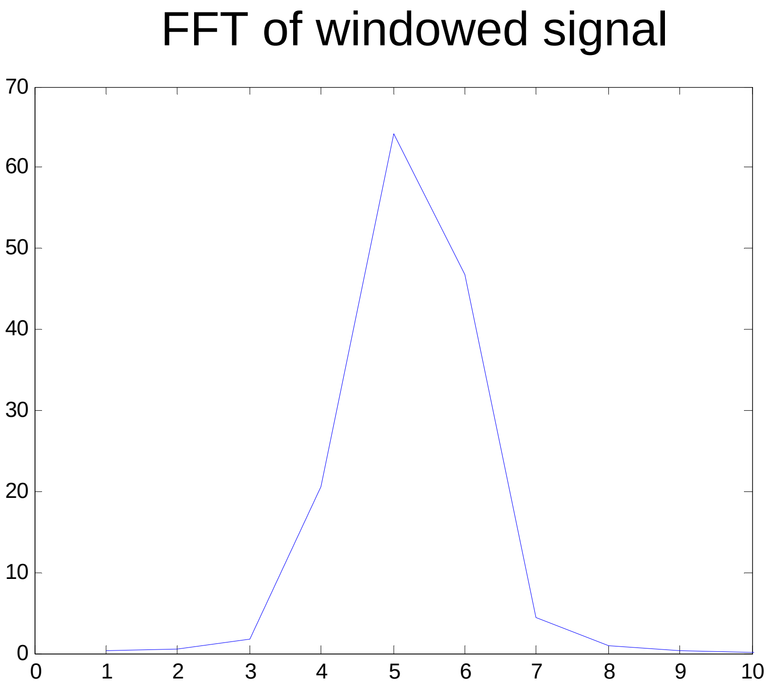



windowed signal



# Windowed sinusoid



# Windowing

- windowing does not eliminate leakage but reduces it
- actually: it changes the shape of leakage
- wintool: to display and analyze windows
- standard window is Hanning window
- hw=hanning(256);

# Accuracy of Periodogram

- Question: how does variance of power estimate decrease with increasing number of samples?
- Answer: No at all!
- additional information is used to compute power at finer frequency resolution

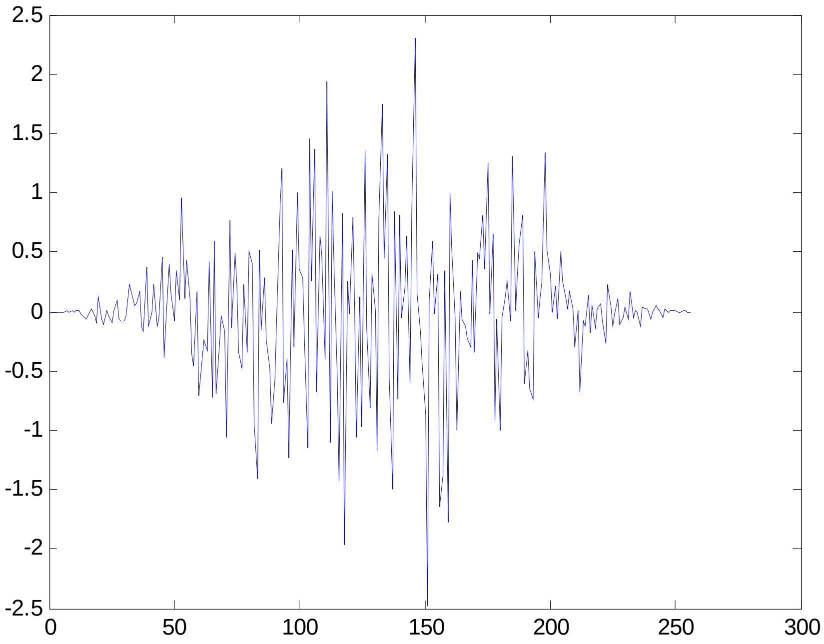
# Welchs Method

- Welchs Method:
  - divide time series in segments of equal length (typically 2^N samples)
  - apply window
  - compute fft
  - average power spectra

# Overlap

- we loose information!
- solution: overlap
- typically: half window length

windowed signal



# psd function

- psd(signal,512,sf,[],256,’mean’)

segment

hanning window

overlap

- [p,f]=psd(signal,512,sf,[],256,’mean’);
- p: power spectrum
- f: frequency vector

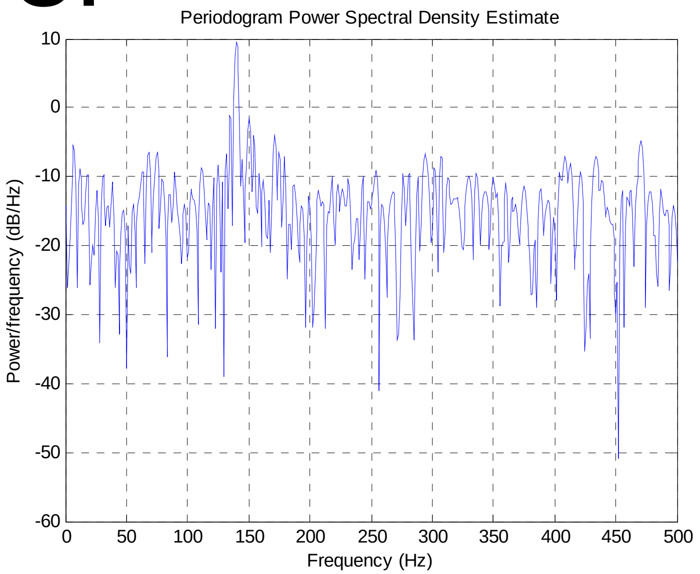
# frequency resolution

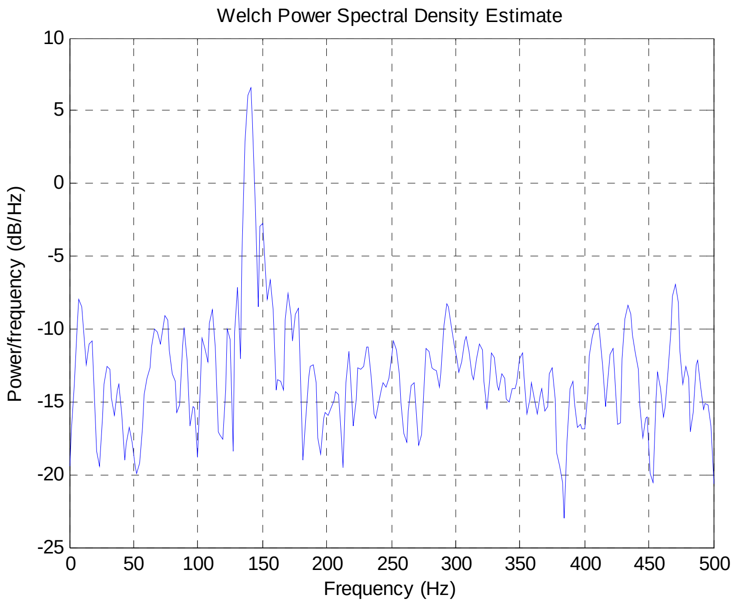
- depends on sampling frequency and length of FFT segment
- sf=256; segment: 256 => 1 Hz resolution
- sf=256; segment: 512 => 0.5 Hz resolution

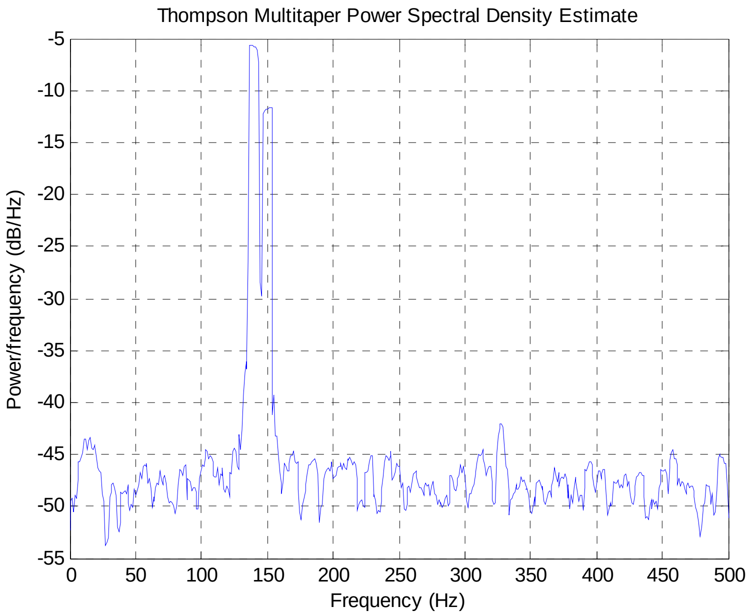
# Multitaper

- reducing the variance further

- sinusoid (140, 150 Hz) +noise



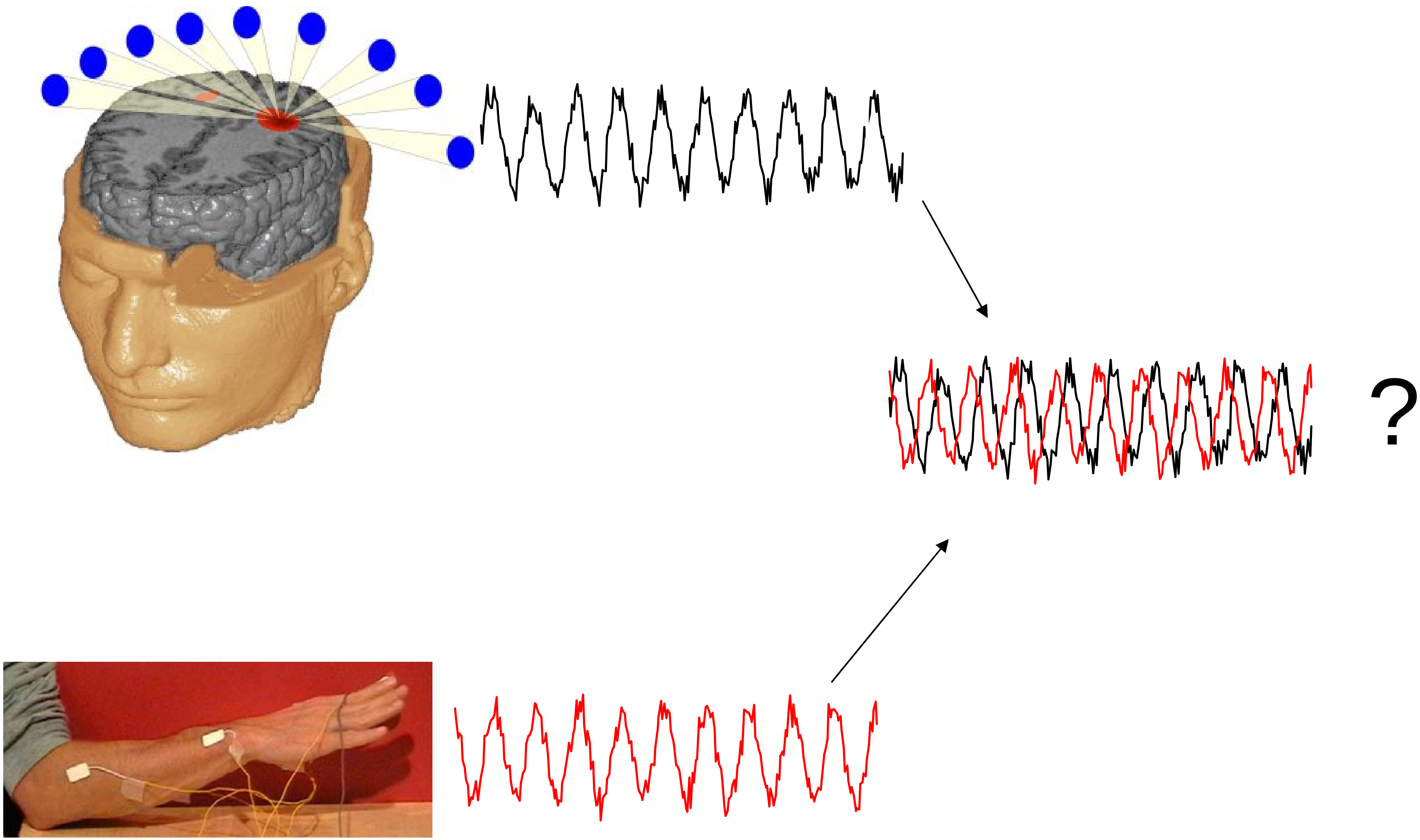
# Exercise

- compute power spectrum of EMG with fft only
- compute power spectrum with psd function, determine tremor frequency with given accuracy
- change length of FFT segment and type of window and observe the effect on power spectrum

# Coherence – general

- Correlation in the frequency domain
- normalized between 0-1 (1: complete dependence)
- preferred phase difference

# Coherence



# Coherence

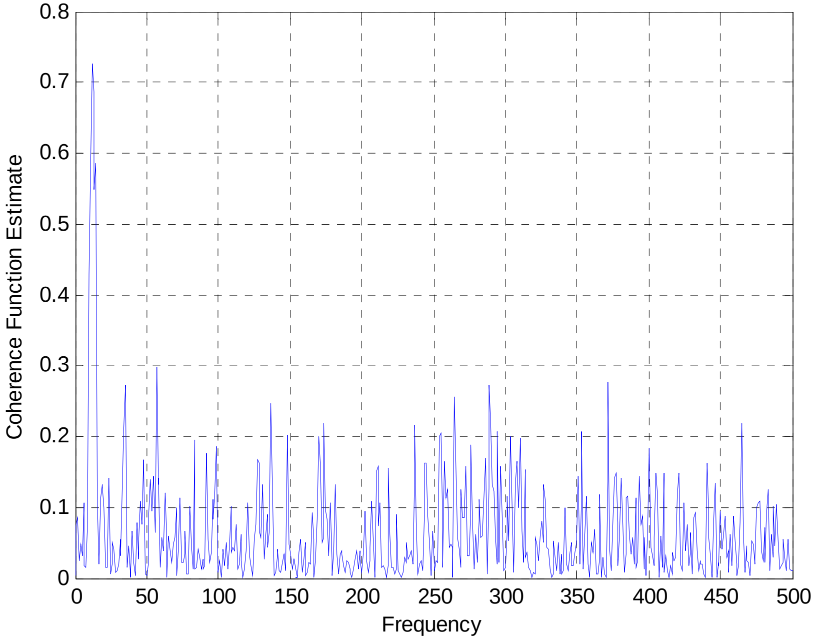
```
cohere(signal1,signal2,512,sf,[],256,’mean’)
```

# Confidence interval

- analytic (Halliday et al, 1995)
- numeric (use random permutation of time series and compute coherence a large number of times)

# Simulation

- x=rand(1,10000);
- [b,a]=butter(4,2*[10 15]/1000);
- xf=filtfilt(b,a,x);
- y=xf+0.5*rand(1,10000);
- z=xf+0.5*rand(1,10000);
- cohere(y,z,1024,1000,[],…
  512,'mean')



# Exercise

- compute power spectra of EMG and MEG using psd
  look for common frequencies
- compute coherence between EMG and MEG with different frequency resolution