Thurs, 3-21-19

8. Filtering (cont.)

1. Types of lowpass filters
2. Lowpass filter design by windowed method
3. Demo08a: interactive filter design by windowed method
4. Sample runs of geosa8

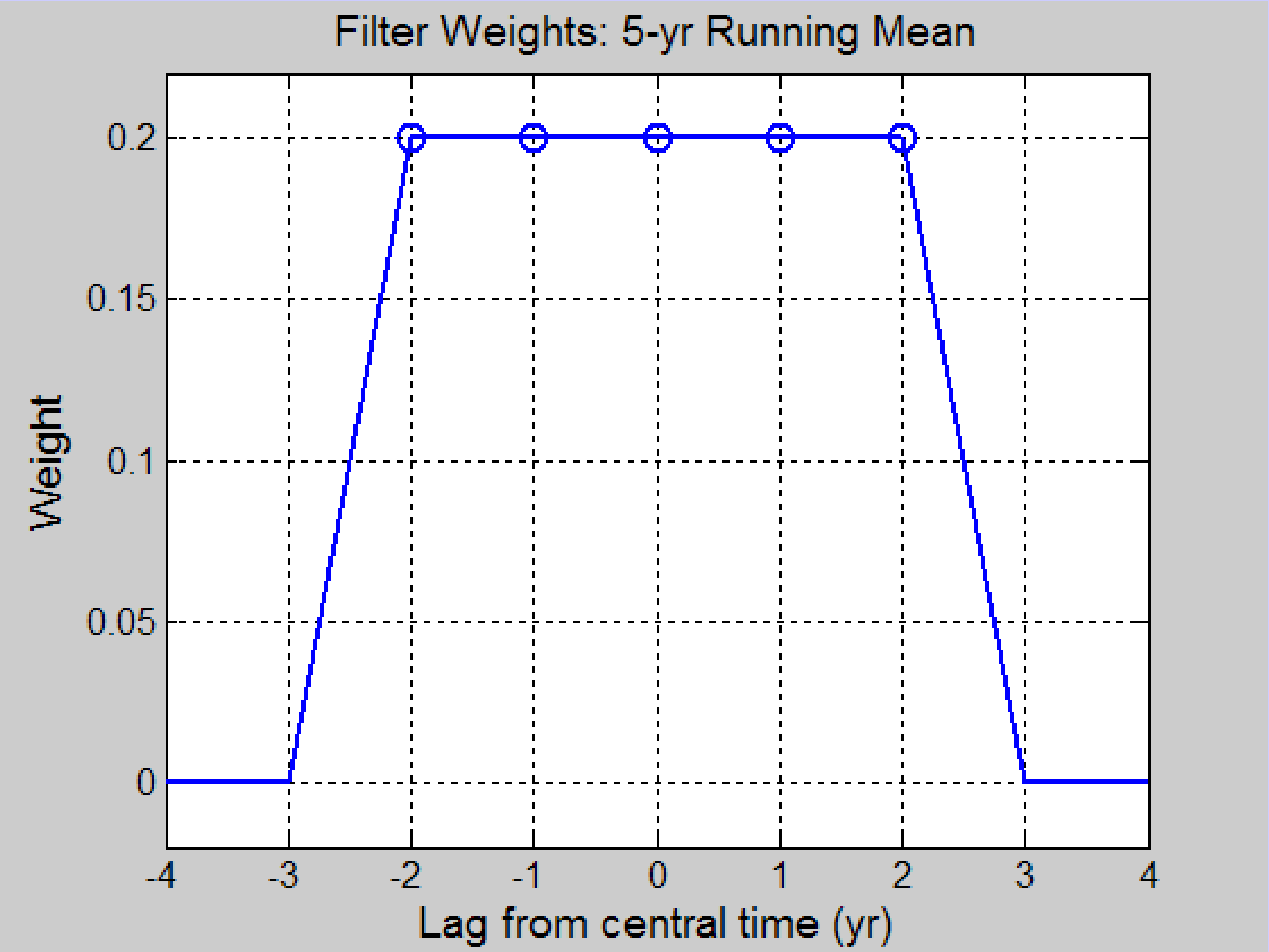
Assignment a8: due Tues, Apr 2 (after Earth Week)

Some types of lowpass filters

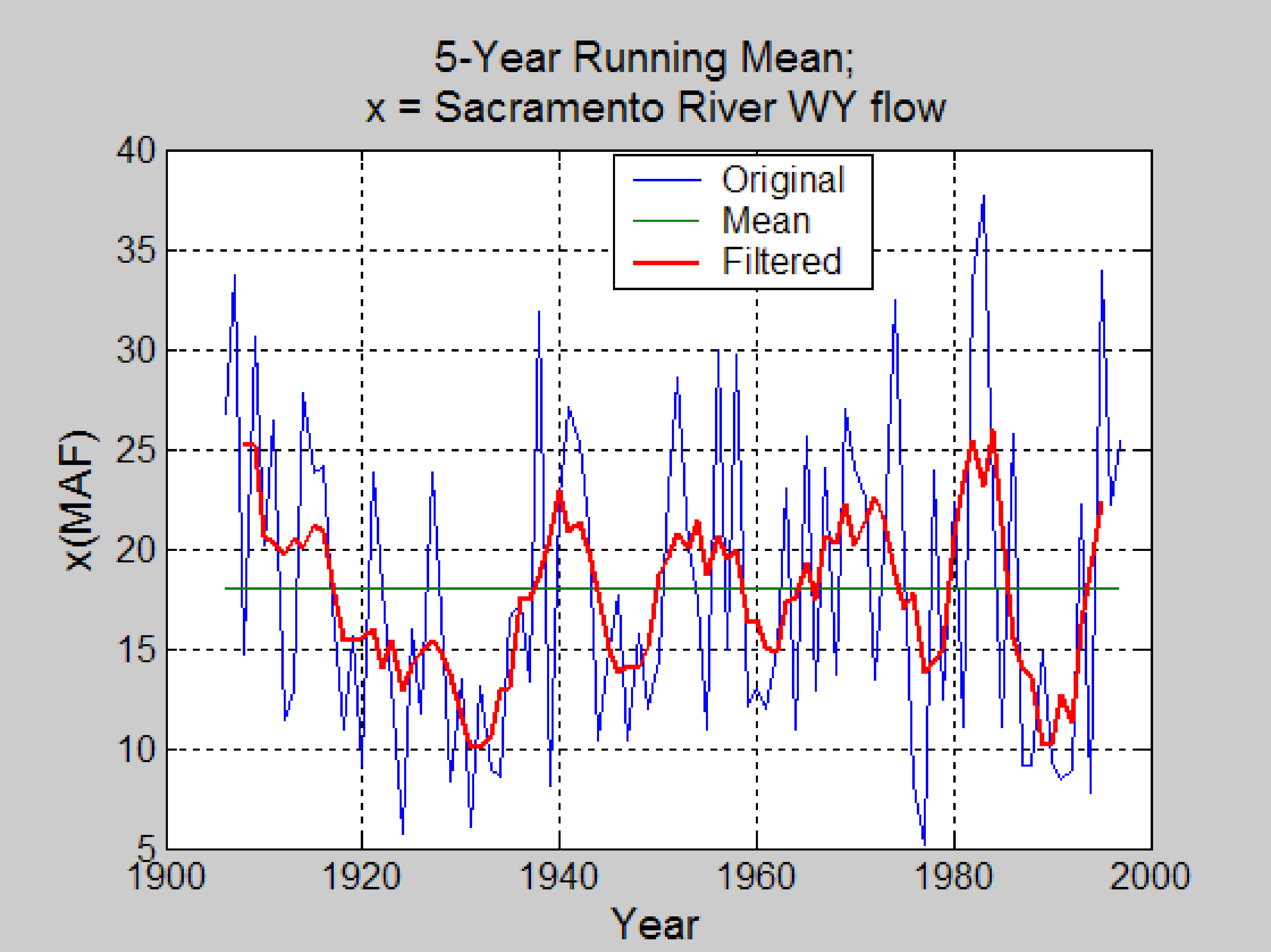
1. (Cubic smoothing spline)
2. Running mean
3. Binomial
4. Gaussian
5. Windowed ideal

See notes

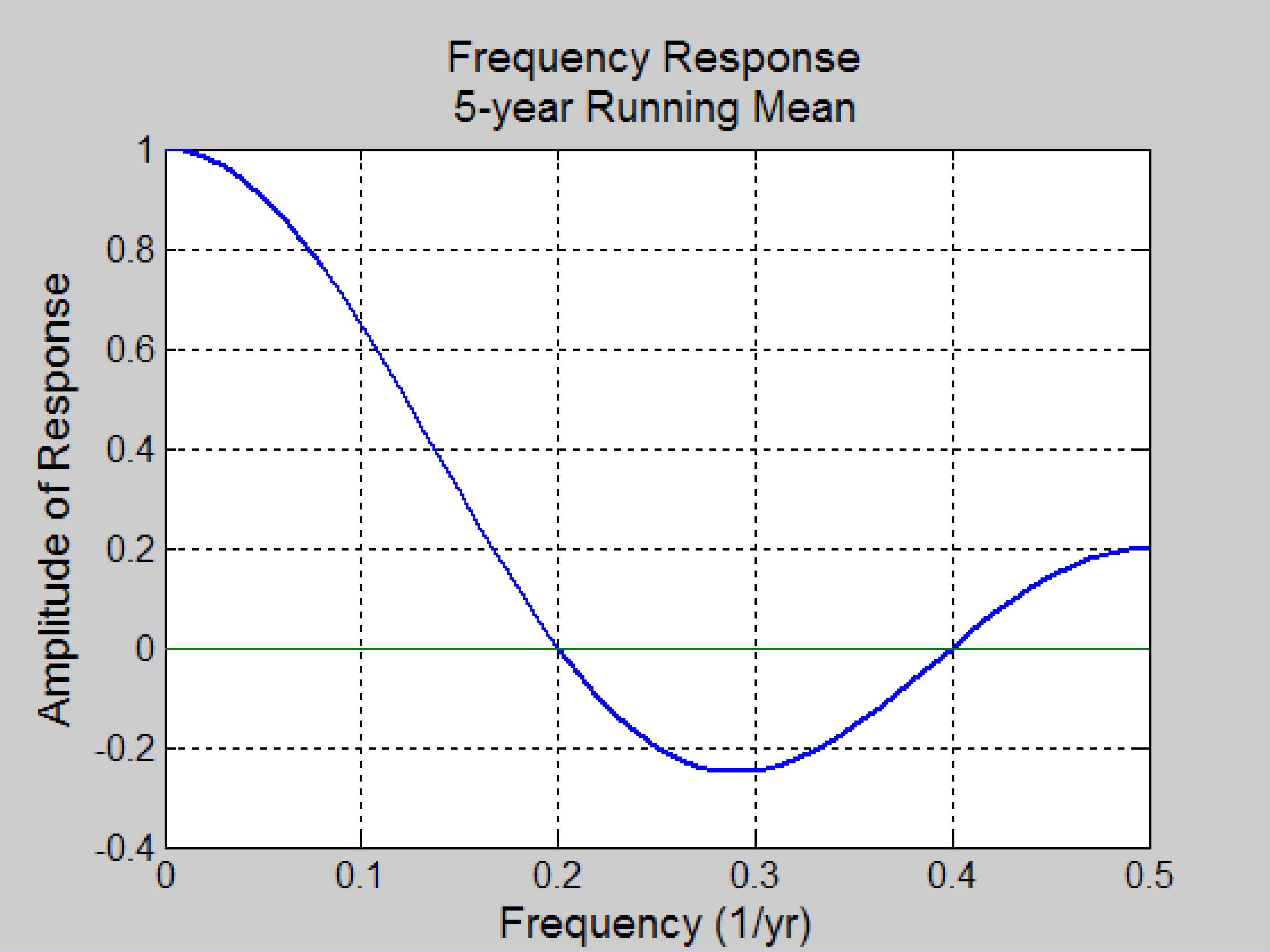
Filter Weights, Running Mean



Running Mean (Moving Average)

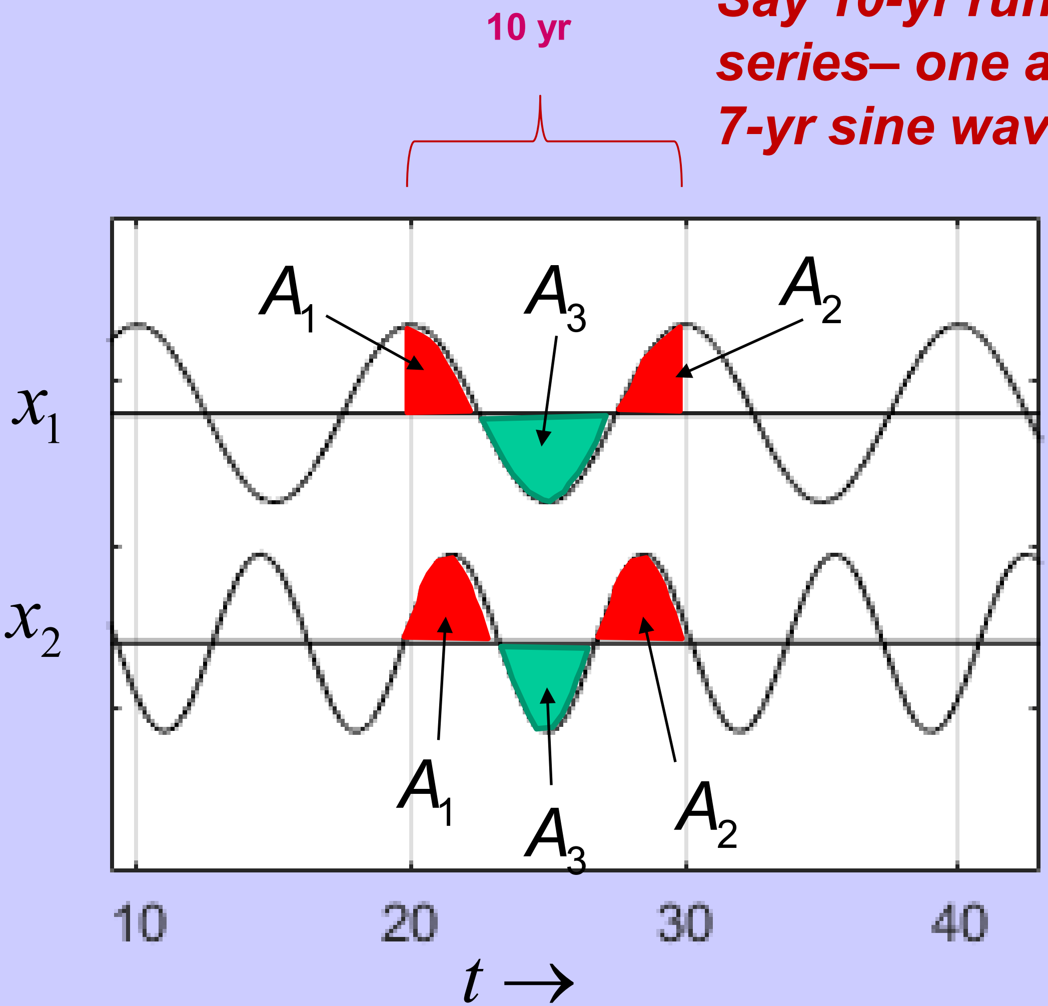


Frequency Response, Running Mean



A drawback of running mean: distortion of peaks and troughs

Say 10-yr running mean of two time series– one a 10-yr sine wave and one a 7-yr sine wave



$A_1 + A_2 = A_3$

Pos. and neg. cancel

Frequency response 0 at $\lambda = 10yr$

$(A_1 + A_2) > A_3$

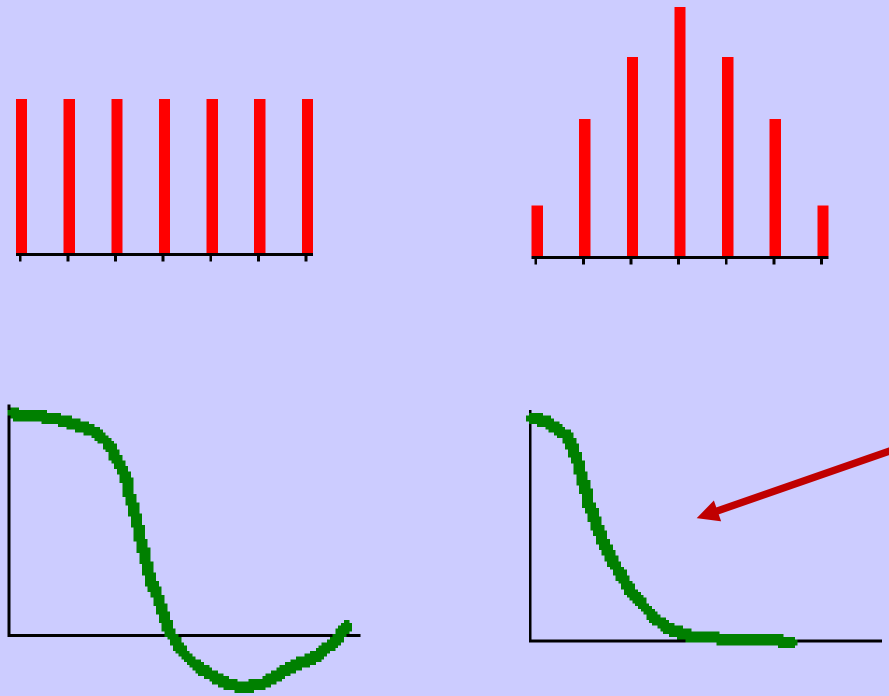
Original series has trough at t=25

Smoothed series has a peak at t=25

Frequency response 0 at $\lambda = 10yr$

m-yr running mean can reverse phase of fluctuations with $\lambda<m$

Running mean vs bell-shaped filter



Filter weights

Preferred form for lowpass filter

Frequency response

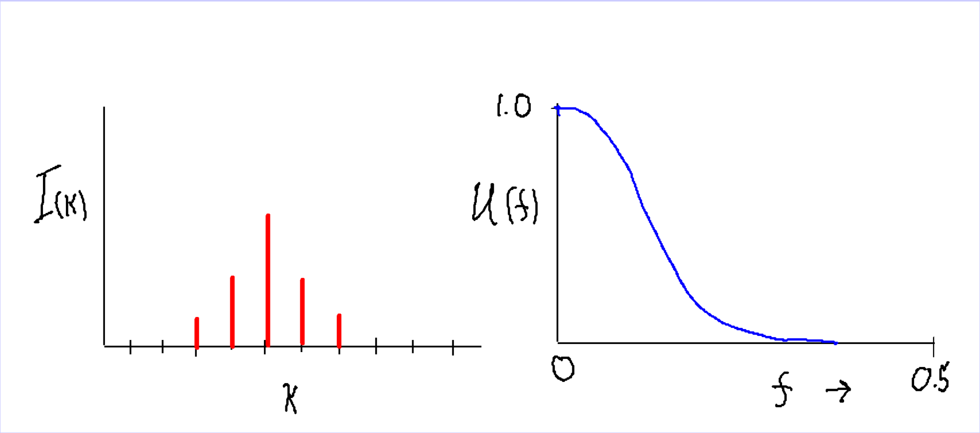
Gaussian
Binomial
Windowed
Other
(Matlab)

Available bell-shaped filters

Windowed method of filter design

Basis: duality of IRF and frequency response function

$$I(k) \leftrightarrow u(f)$$



$S_x(f)$ and $S_y(f)$ are spectra of original and smoothed series

$I(k)$ is equivalent to the filter weights

$u(f)$ is Fourier transform of $I(k)$

$I(k)$ is inverse Fourier transform of $u(f)$

$$S_y(f) = u^2(f) S_x(f)$$

Windowed method of filter design
convolution vs multiplication

$x_t, \mathrm{t} = 1, \mathrm{N}$ original time series

y_t filtered time series

$I(k)$ filter weights

$u(f)$ frequency response of filter

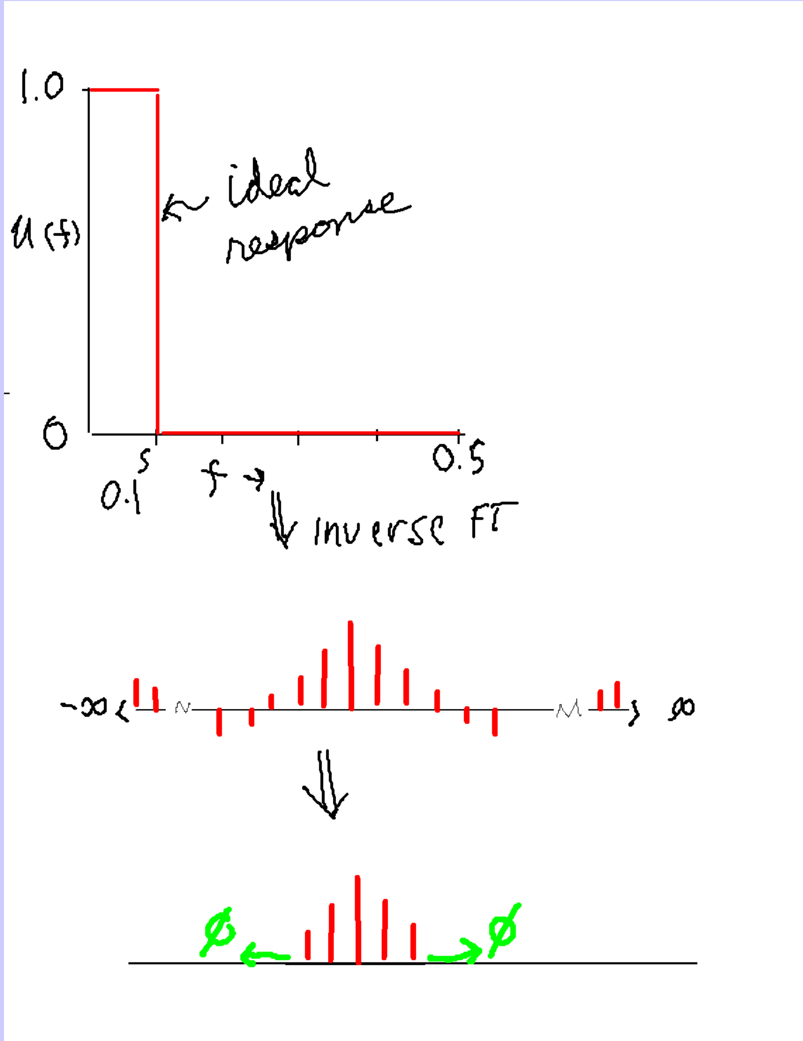
y_t is the convolution of x_t with $I(k)$

The spectrum of y_t is the product of the spectrum of x_t and $u^2(f)$

Convolution in the time domain is equivalent to multiplication in the frequency domain

Windowed method of filter design

steps



1. Specify desired cutoff frequency of ideal filter
2. Specify desired filter length
3. Take inverse Fourier transform of the ideal, or brick-wall filter, and use the duality relationship:
   - Frequency response is Fourier transform of Impulse response
   - Impulse response is inverse Fourier transform of frequency response
4. Truncate the filter to the desired number span
5. Frequency response of resulting filter will not be “ideal”, but will approximate the ideal given the constraint of the filter span

Windowed method of filter design

Matlab function fir1

$$b = fir1(m, f^*_{cutoff}, 'low')$$

Filter Weights

Filter-length -1
(7 weight filter: m=6)

"normalized" frequency at which desired response is 0.5

$$f: \quad 0 \rightarrow 0.5 \quad \text{cycles/time step}$$
$$f^*: \; 0 \rightarrow 1.0$$
$$\omega: \quad 0 \rightarrow \pi$$

Specifies "lowpass", and by default this uses a "Hamming" window

Windowed method of filter design

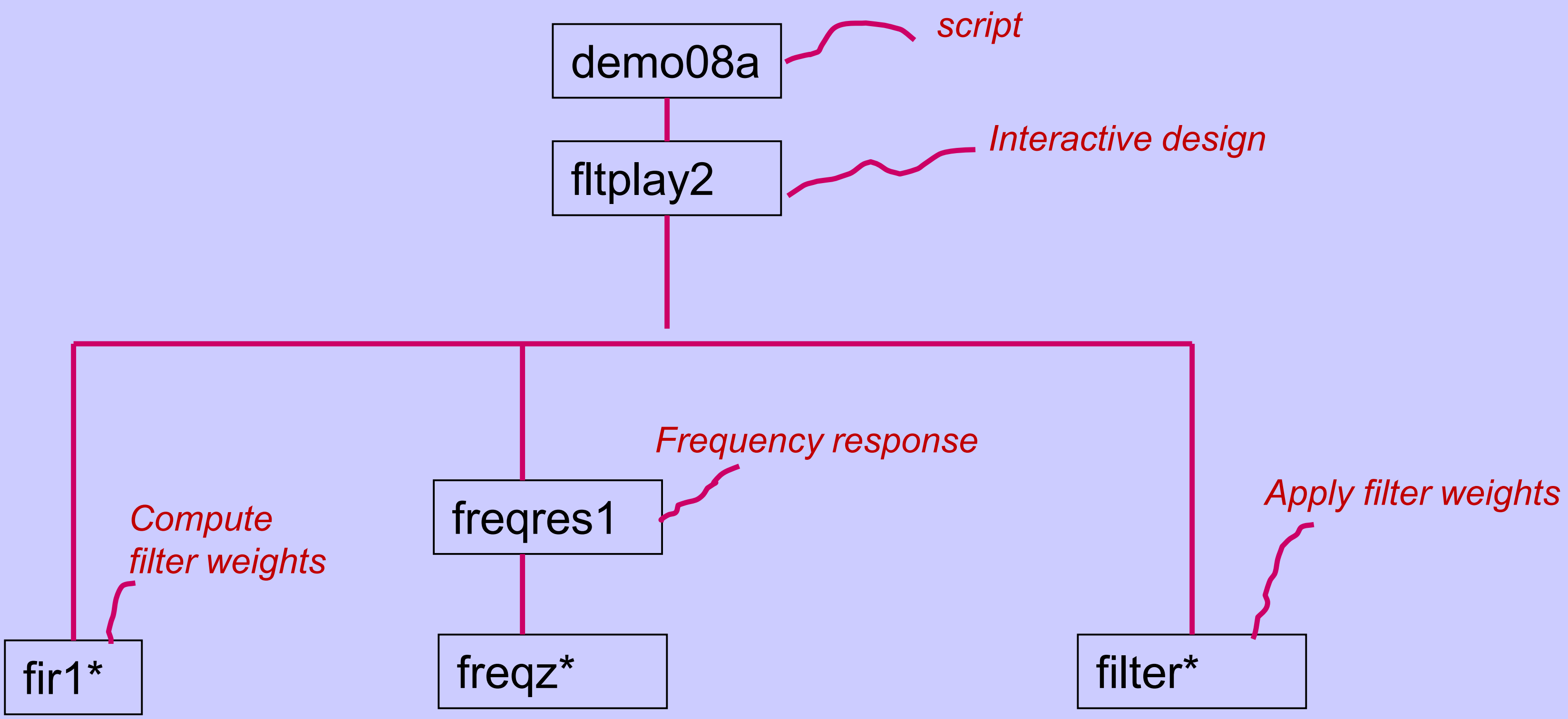
Example: function fir1 – calling from command window

- Assume
  - Time step = 1 yr
  - Desired cutoff frequency: f=0.10 cycles/year, or wavelength 10 yr
  - Desired filter length: 9-weight filter

$$\mathrm{b} = \mathrm{fir1}(8, 0.20, '\mathit{low}')$$

$$f^* = 0.20 \text{ corresponds to}$$
$$f = 0.10$$

Demo8a – filter design by windowed method



*Matlab built-in function

Trial runs of geosa8…