Application of time series and spectral methods in Solar & Astrophysics

Ding Yuan

Email: Ding.Yuan@wis.kuleuven.be

Centre for mathematical Plasma Astrophysics
Department of Mathematics
Katholieke Universiteit Leuven (KU Leuven)
Celestijnenlaan 200B, bus 2400
B-3001 Leuven, Belgium

Content

- Fundamentals of statistics
- Pre-processing methods
- Fourier transform
- Windowed Fourier transform
- Wavelet
- Periodogram
- Date-compensated Discrete Fourier transform
- Filtering method: time and spectral domain
- Significance tests and noise analysis

Fundamental of Statistics

$y = \{y_1, y_2, ..., y_N\}$

$y_j = x_j + s_j,$ where x is a signal and s is a random noise

$E[y] = E[x] = E[s] = 0;\ Var[x] = \sigma_x^2;\ Var[s] = \sigma_s^2$

$Var[y] = E[y^2] - (E[y])^2$

$= E[x^2] + E[2xs] + E[s^2] - (E[x])^2$

$= Var[x] + E[2xs] + E[s^2]$

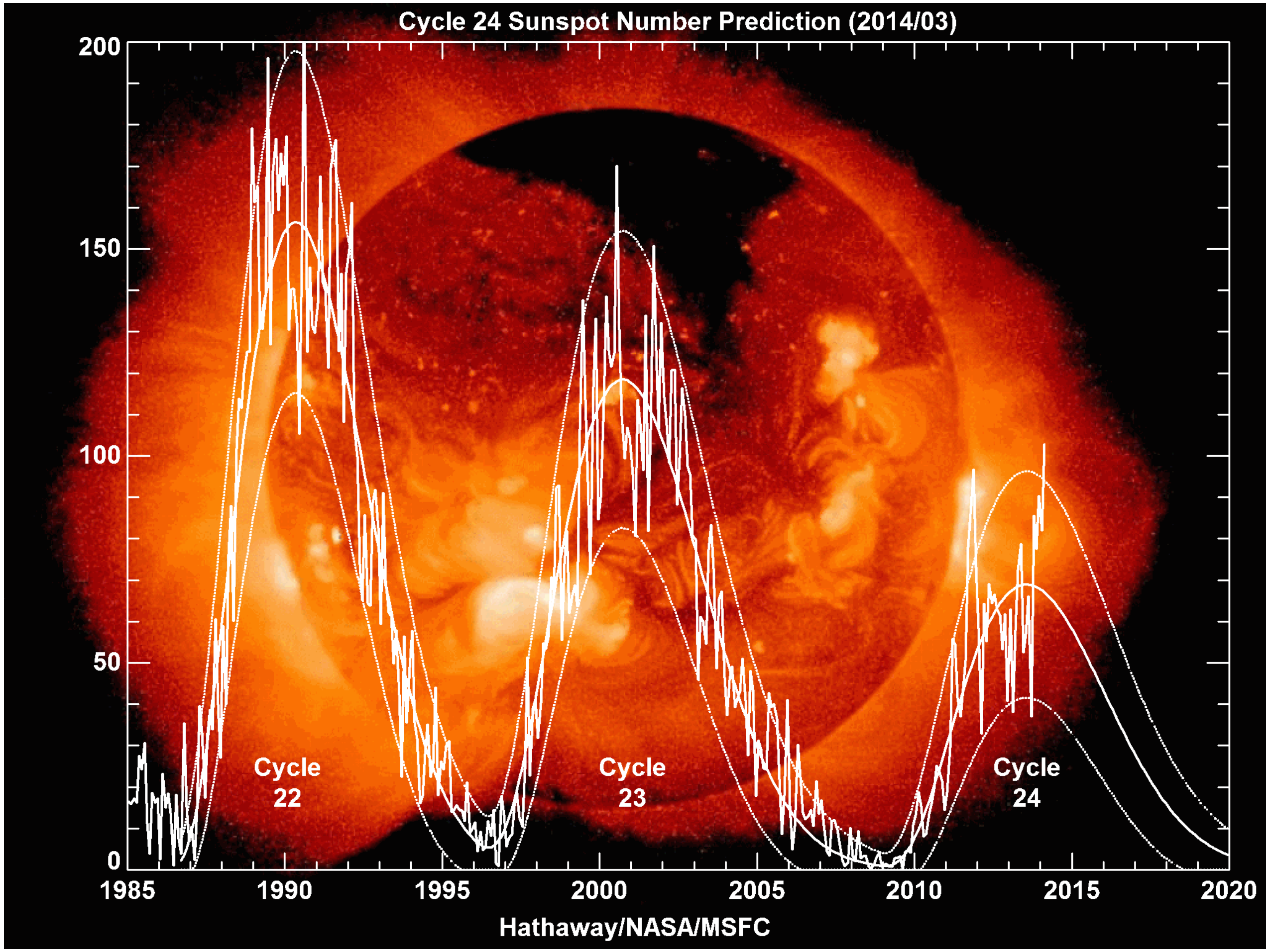
$= \sigma_x^2 + \sigma_s^2$ (error propagation)

$E[(s/\sigma_s)^2] = E[\chi_1^2] = 1,$

χ_1^2 is a χ^2-distribution with 1-degree of freedom

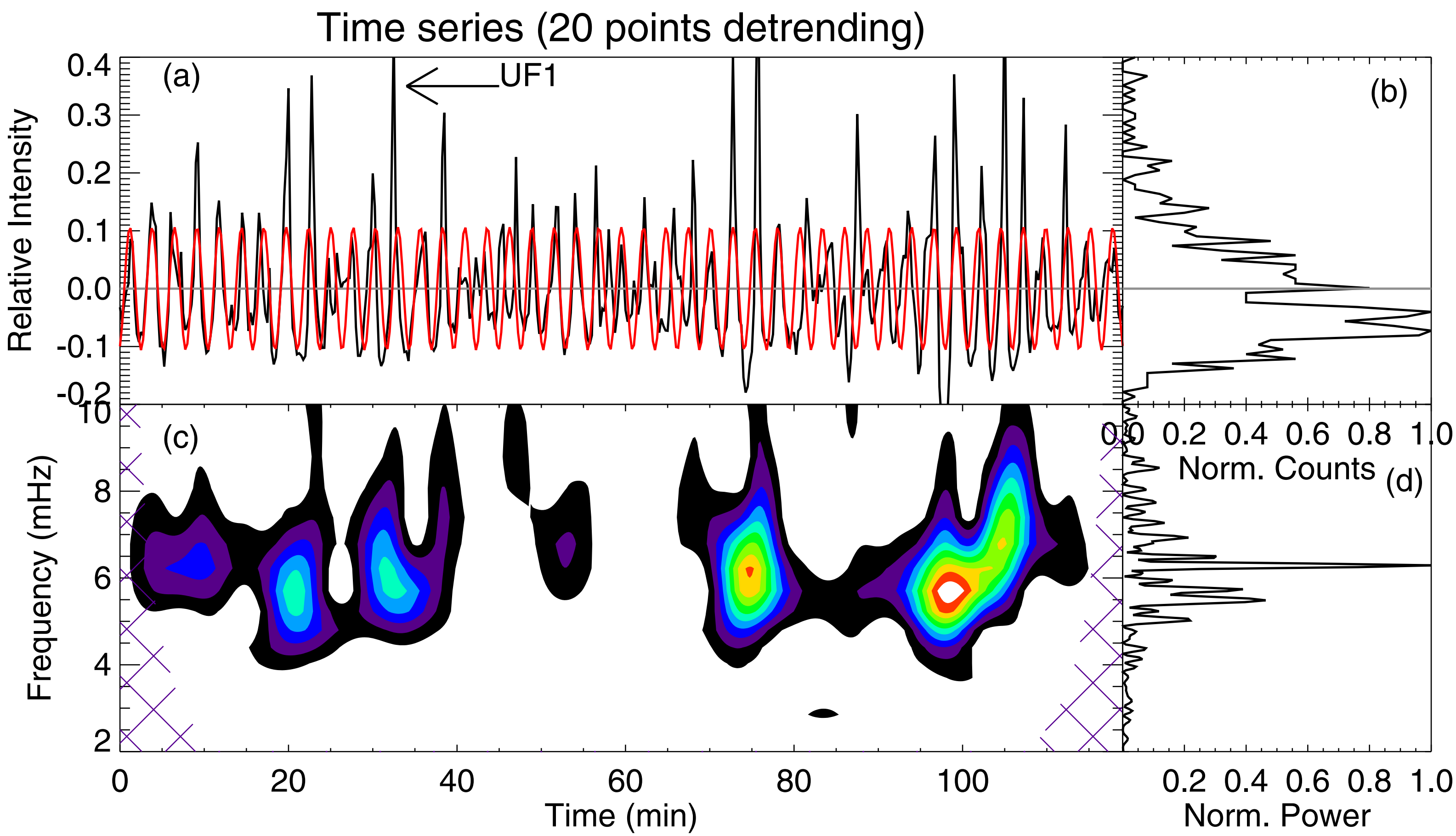
$E[2xs] \approx 0$ by assuming x and s are independent.

Time series: sunspot cycle



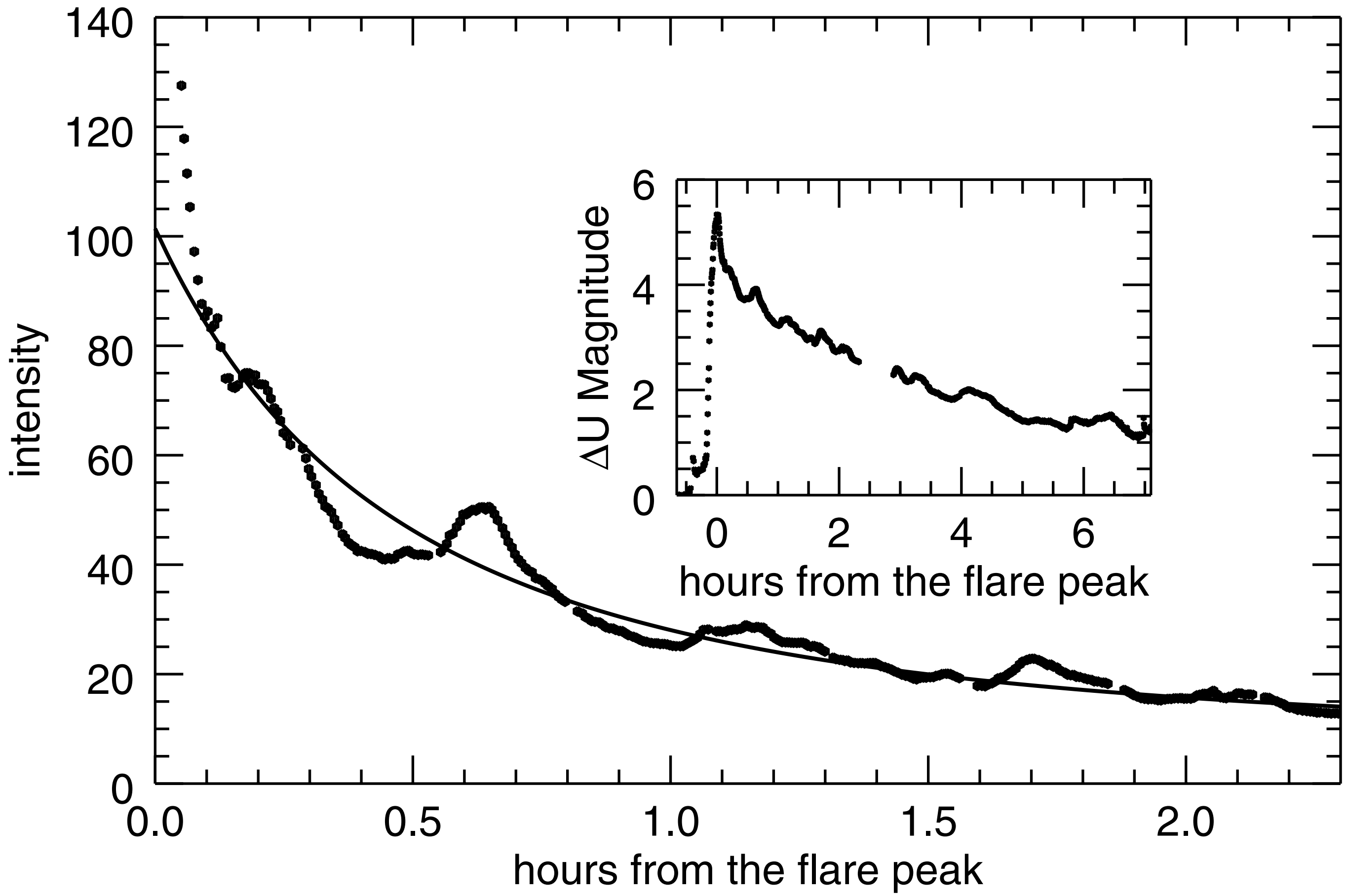
@Courtesy of NASA

Time series: sunspot waves



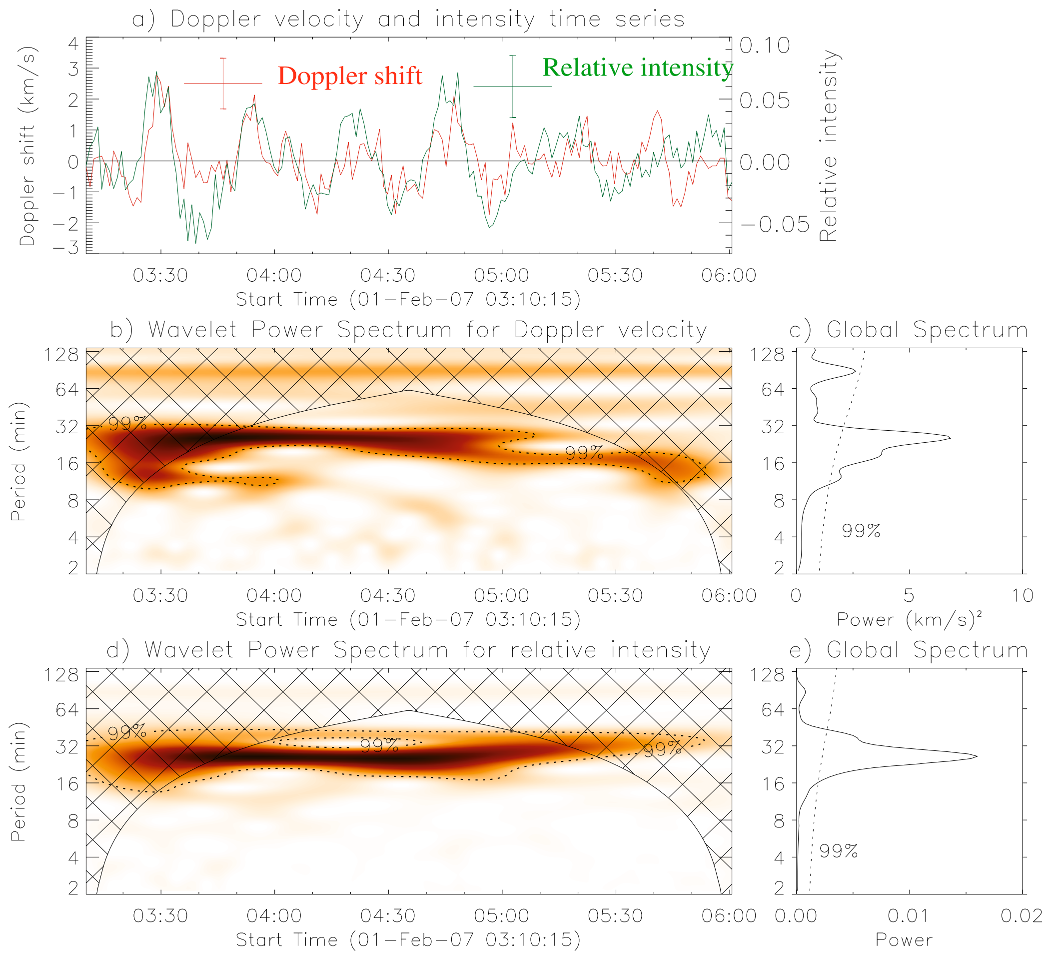
Yuan et al 2014, ApJ 792

Time series: flare pulsations



Anfinogentov et al. 2013 ApJ 773

Time series: MHD waves

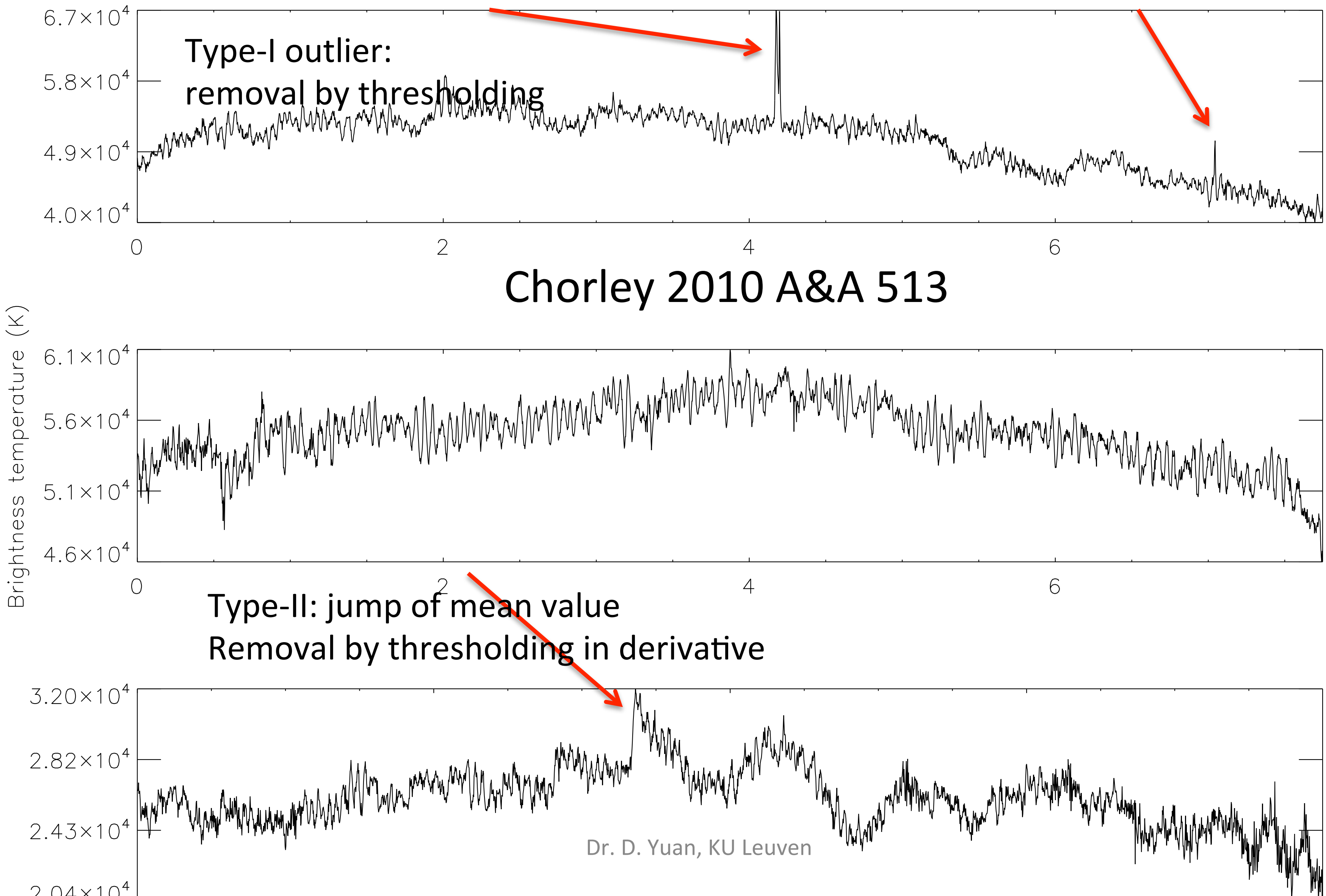


Wang et al. 2009 A&A 503 L25

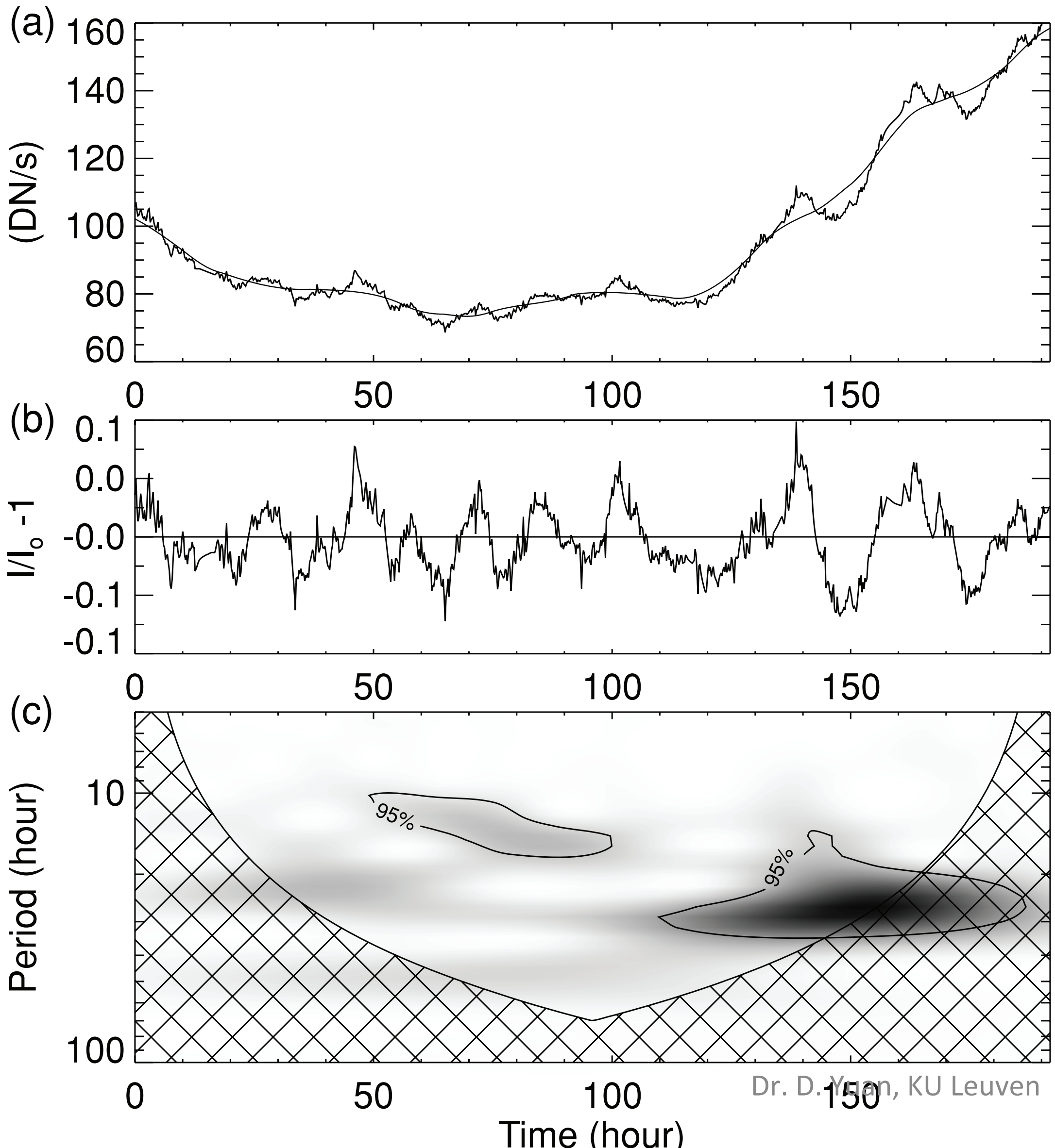
A raw time series

1. Check for faulty data: missing points (zero or interpolation), outliers
2. Data metrics: uniform or uneven time series, mean, variance, histogram.
3. Detrending: moving average, polynomial (exponential, linear) fit, running difference.

Outliers (spikes)



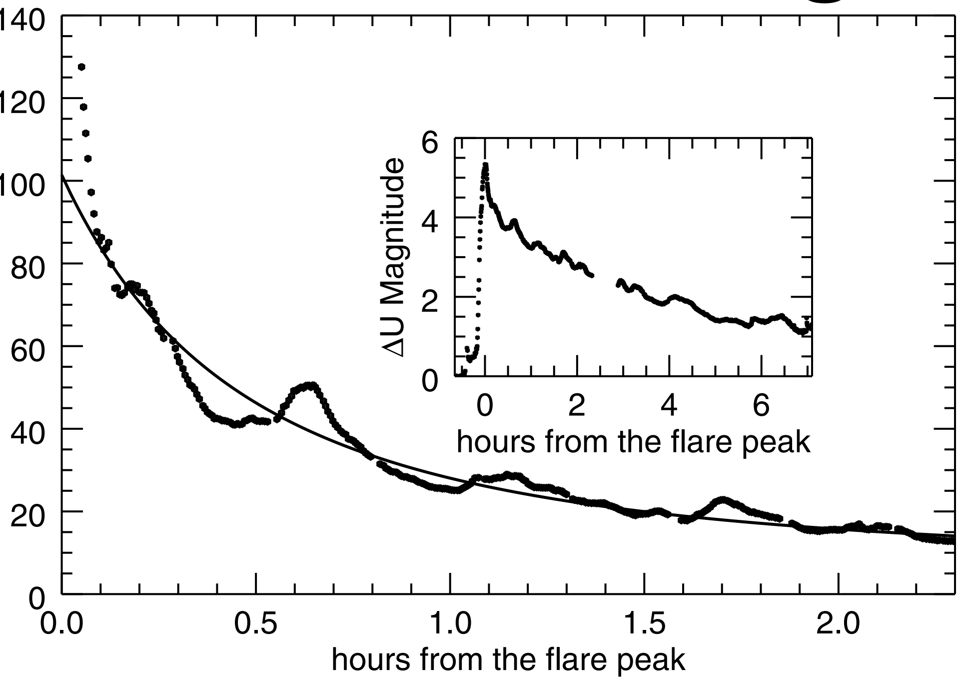
Detrending: polynomial fit

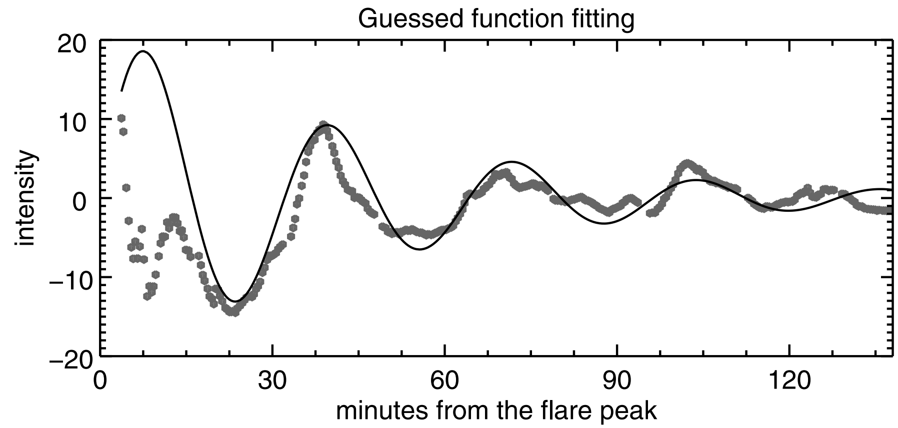


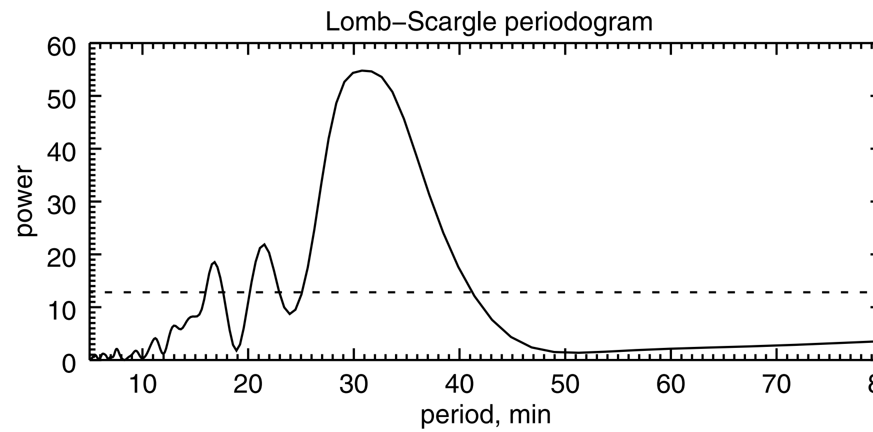
Foullon et al. 2009 ApJ 700

Dr. D. Yuan, KU Leuven

Detrending: exponential fit







Anfinogentov et al. 2013 ApJ 773

Uniform or non-uniform data

- Uniform data (or uniformly interpolated data): FFT, Windowed FFT, wavelet, etc.
- Non-uniform data: Periodogram, DCDFT, etc.
- Short time series (a few oscillation cycles): Nonlinear fit (mpfit.pro, https://www.physics.wisc.edu/~craigm/idl/fitting.html), Optimization methods (IDL routine:powell.pro), Bayesian inference (Marsh ApJ 2008, 681; Irregui, ApJ, 2011 740).

Discrete Fourier Transform (DFT)

$$Y_k = \sum_{j=0}^{N-1} y_j e^{-i2\pi k \frac{j}{N}}, k = 0, \cdots, N-1$$

- DFT requires $O(N^2)$ operations;
- FFT is an **algorithm** that compute accurate DFT with *O*(*NlogN)* operations.
- FFTPACK (Fortran), FFT (IDL), numpy.fft(Python)
- IDL and Fortran FFT comparison (www.ssec.wisc.edu/~paulv/fft/fft_comparison.html)

#1: How to use FFT (IDL)

- tt: *the vector of time;*
- xx: *the vector of variables*
- n=n_elements(xx)*; number of samples*
- tt=tt-tt[0] *; time invariance*
- xx=xx-mean(xx) *; remove mean value*
- dt=tt[1:n-1]-tt[0:n-2] *; calculate cadence*
- dt_min=min(dt,max=dt_max)
- ti=0.5*(dt_min+dt_max)*; ensure uniform data*

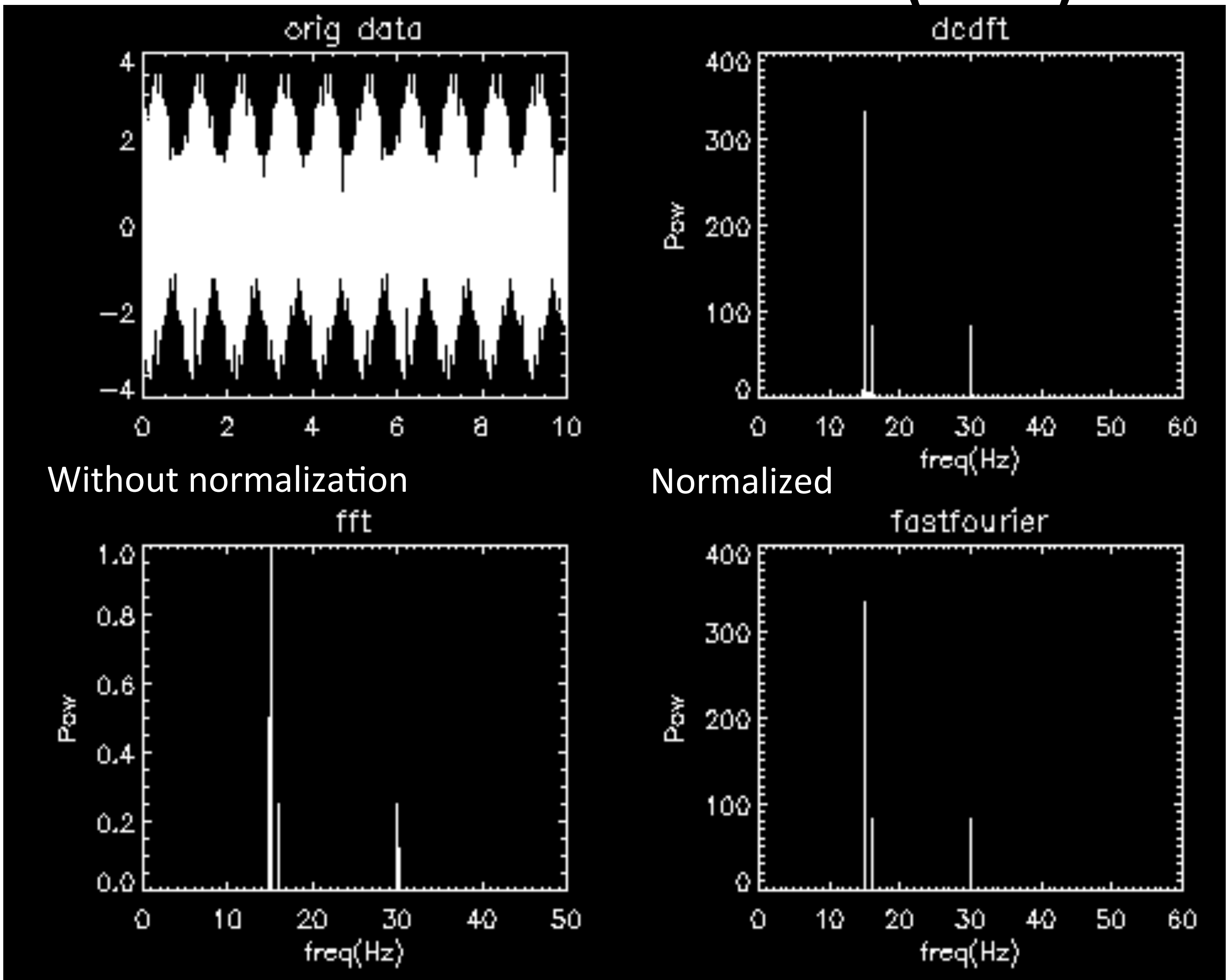
#2: How to use FFT (IDL)

- temp = fft(xx) ; calculate FFT
- if n mod 2 eq 0 then begin ; even number
- freq = findgen(n/2+1)/(n*ti)
- pow_fft = abs(temp[0:n/2])^2
- phase_fft=atan(temp[0:n/2],/phase)
- endif else begin ; odd number
- freq=findgen((n+1)/2)/(n*ti)
- pow_fft=abs(temp[0:(n-1)/2])^2
- phase_fft=atan(temp[0:(n-1)/2],/phase)
- endelse

#3: How to use FFT (IDL)

- norm=variance(xx)/float(N)
- pow_fft=pow_fft/norm ; Normalization
- power=pow_fft
- fs=0.01 & Num=1000.
- tt=findgen(Num)*fs
- xx=2*sin(2*!pi*15*tt)-cos(2*!pi*16*tt) +sin(2*!pi*30*tt) ; A time series of freq=[15, 16, 30]

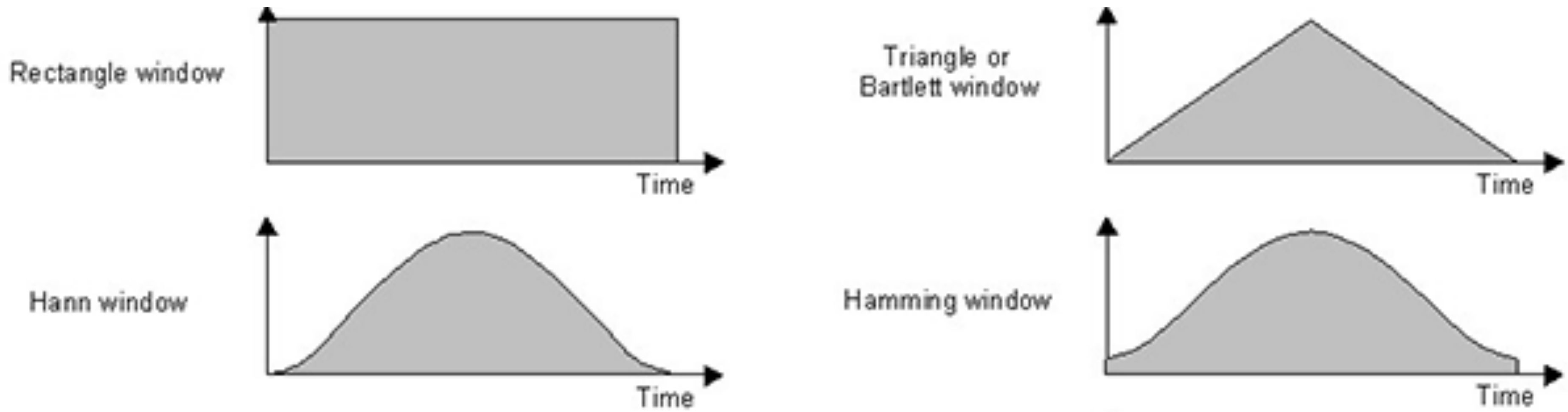
#4: How to use FFT (IDL)

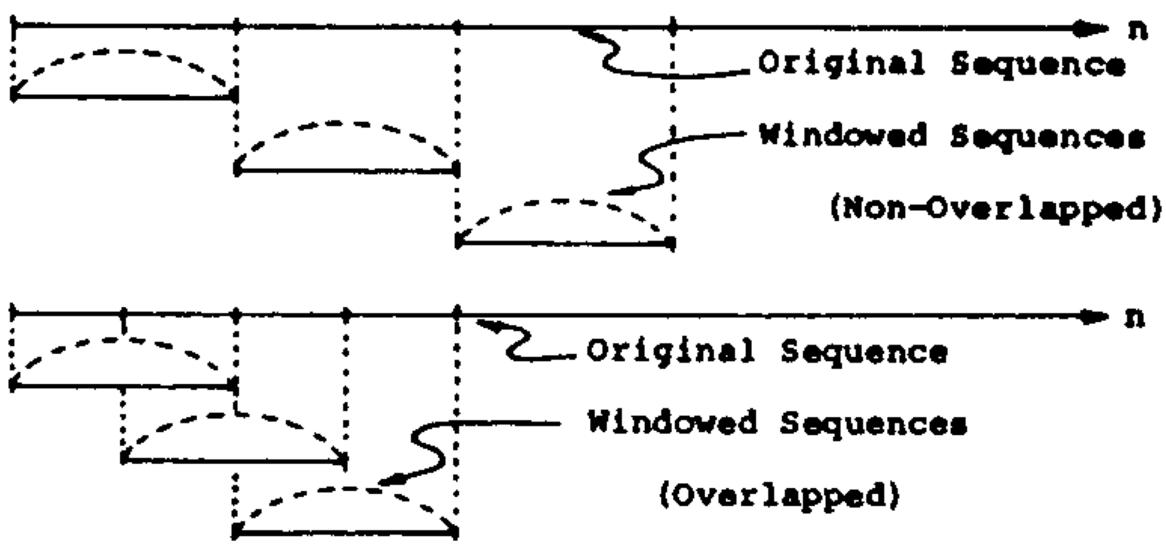


Windowing

- Problems in FFT: aliasing (discretization), spectral leakage (finite time span)
- Windowing -> a) Select a desired range; b) Apply weights to the data; c) Reduce the noise by repetitive measurements.
- Harris, 1978 "*On the use of windows for harmonic analysis with the Discrete Fourier Transform*"

Windowing





Harris, 1978
The selected sequence has to contain sufficient information.

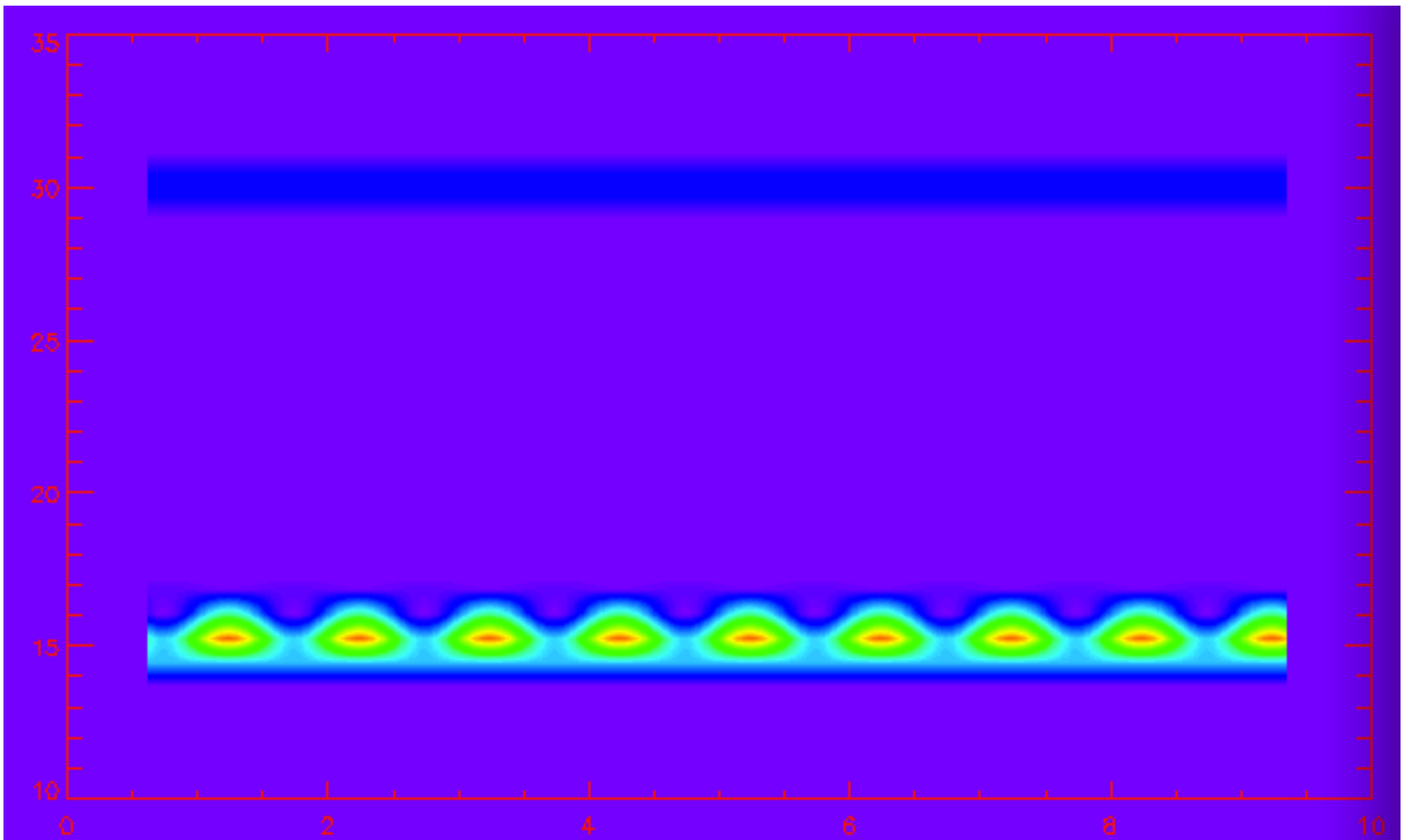
Windowed FFT (Short-time FFT)
Sliding DFT

$$
\begin{aligned}
Y_{k,m} &= \sum_{j=0}^{N-1} y_j w_{j-m} e^{-i2\pi k \frac{j}{N}}, k = 0, \cdots, N-1 \\
&= \{Y_k \star W_k\}(m),
\end{aligned}
$$

1. WFFT reveals a dynamic (time-dependent) spectrum.
2. It is arbitrary to choose a proper window width.
3. The spectral resolution depends on the window width.
4. Different windows produce slightly different spectra.

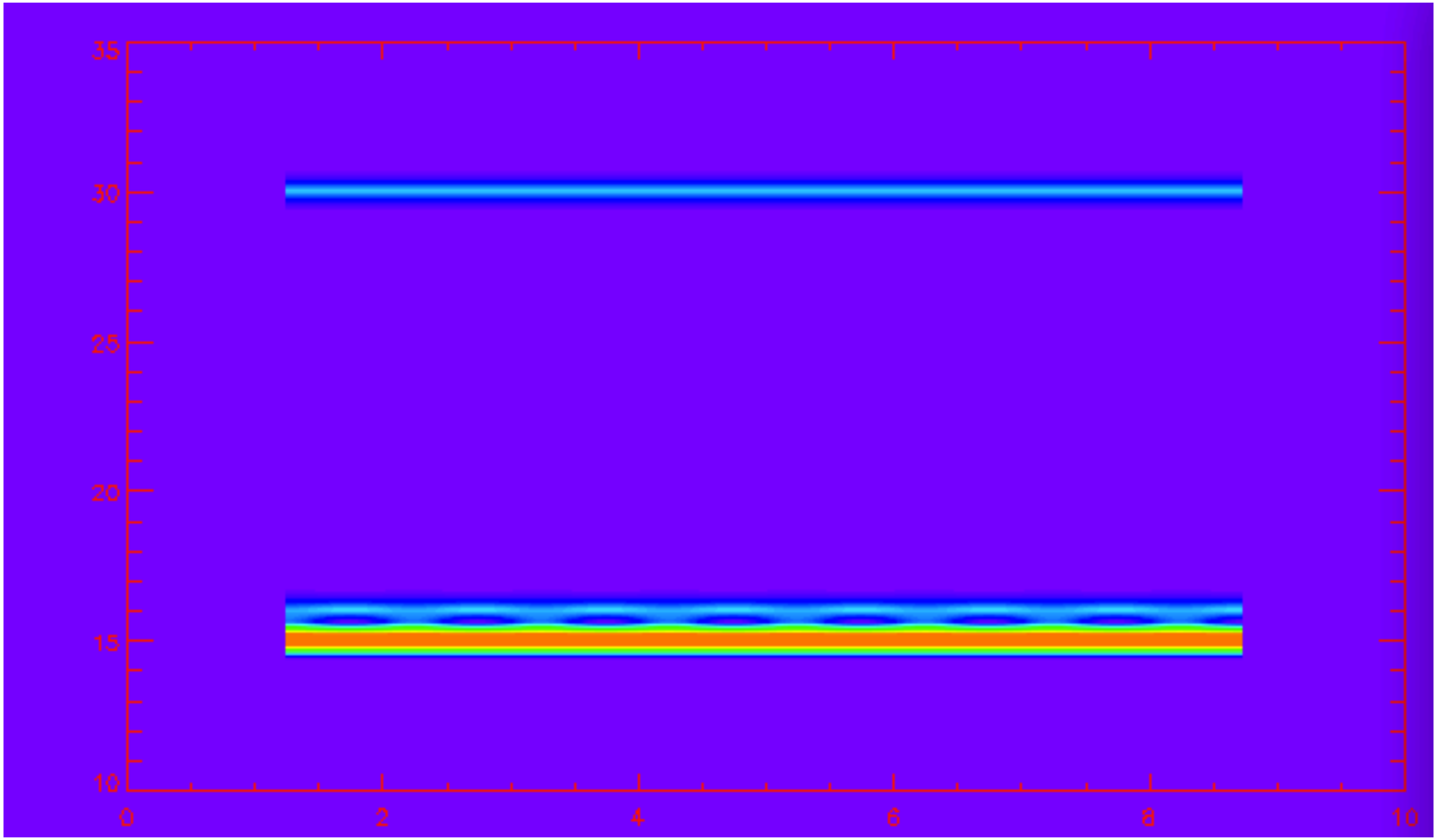
Windowed FFT
window width=1/8 time span

➢xx=2*sin(2*!pi*15*tt)-cos(2*!pi*16*tt)+sin(2*!pi*30*tt) ; A time series of freq=[15, 16, 30]



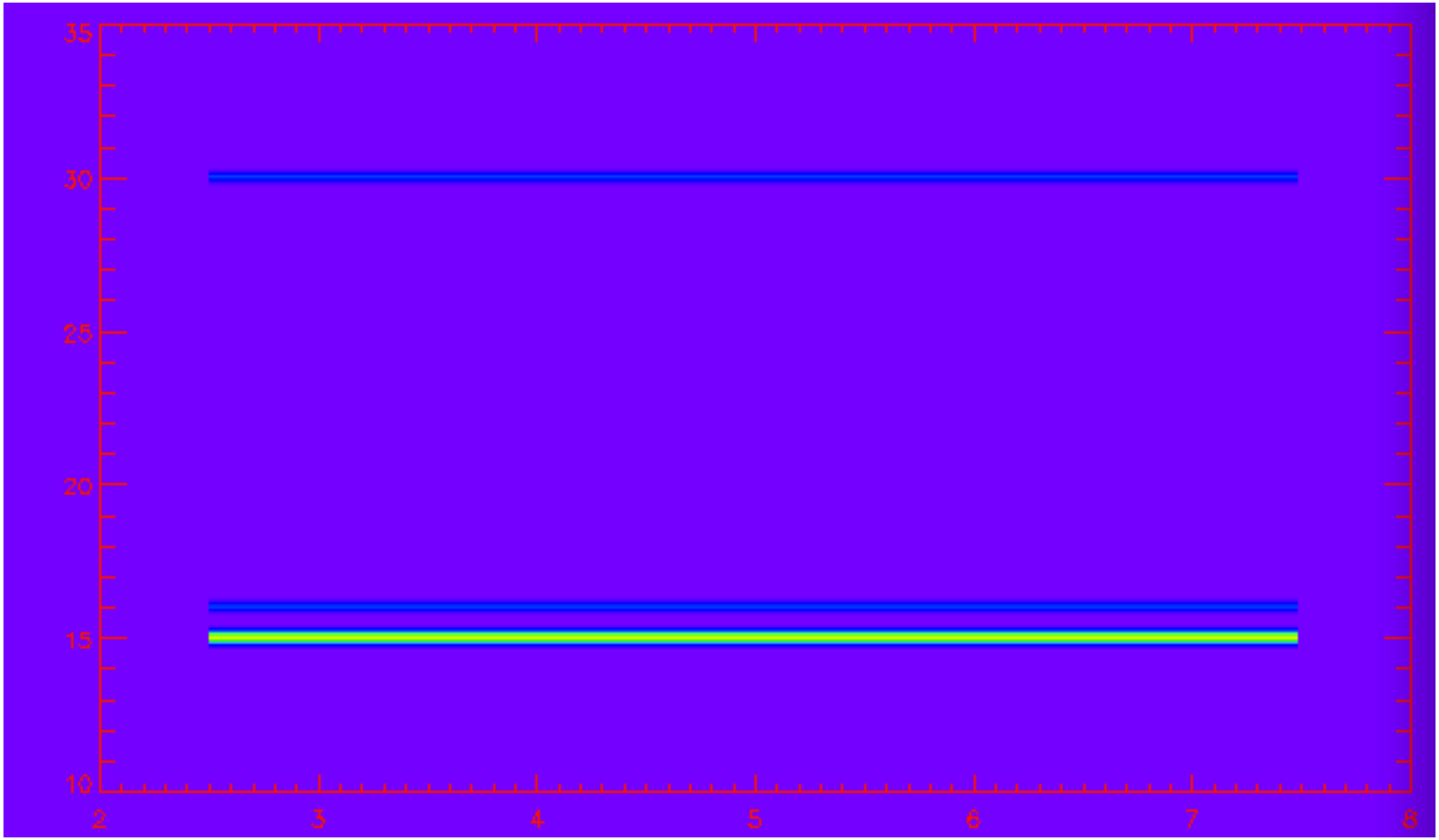
Windowed FFT
window width=1/4 time span

- xx=2*sin(2*!pi*15*tt)-cos(2*!pi*16*tt)+sin(2*!pi*30*tt) ; A time series of freq=[15, 16, 30]



Windowed FFT
window width=1/2 time span

- xx=2*sin(2*!pi*15*tt)-cos(2*!pi*16*tt)+sin(2*!pi*30*tt) ; A time series of freq=[15, 16, 30]



Wavelet Transform

$$W_h(s) = \sum_{j=0}^{N-1} y_j \Psi^*[\frac{(j-h)\delta t}{s}]$$

$$= \sum_{k=0}^{N-1} Y_k \hat{\Psi}^*(s\omega_k) e^{i\omega_k h \delta t}$$

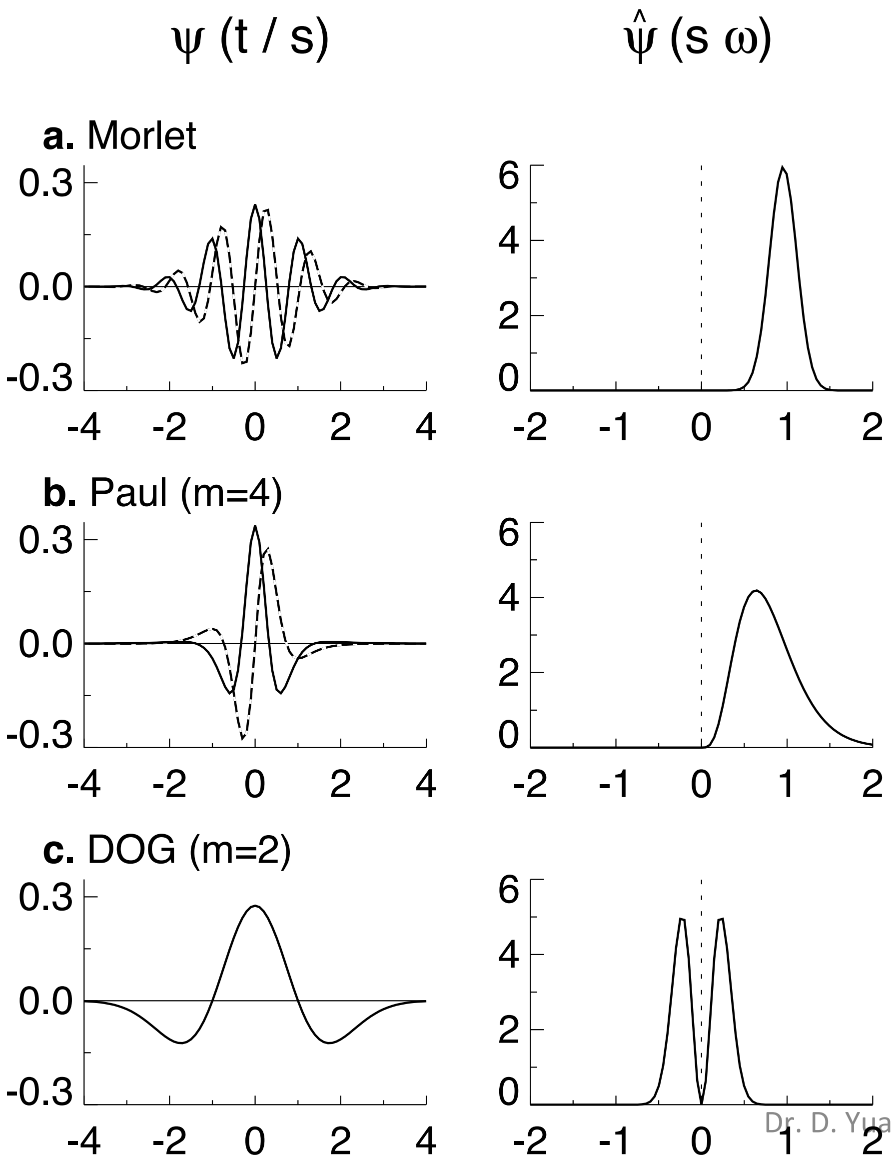
Wavelet is defined as the convolution of a time series with a scaled mother function Ψ.

Torrence & Compo, A practical guide to wavelet, BAMS 1998.

$$\Psi_0(\eta) = \pi^{-1/4} e^{i\omega_0 \eta} e^{-\eta^2/2},$$

Ψ_0 is the original mother function

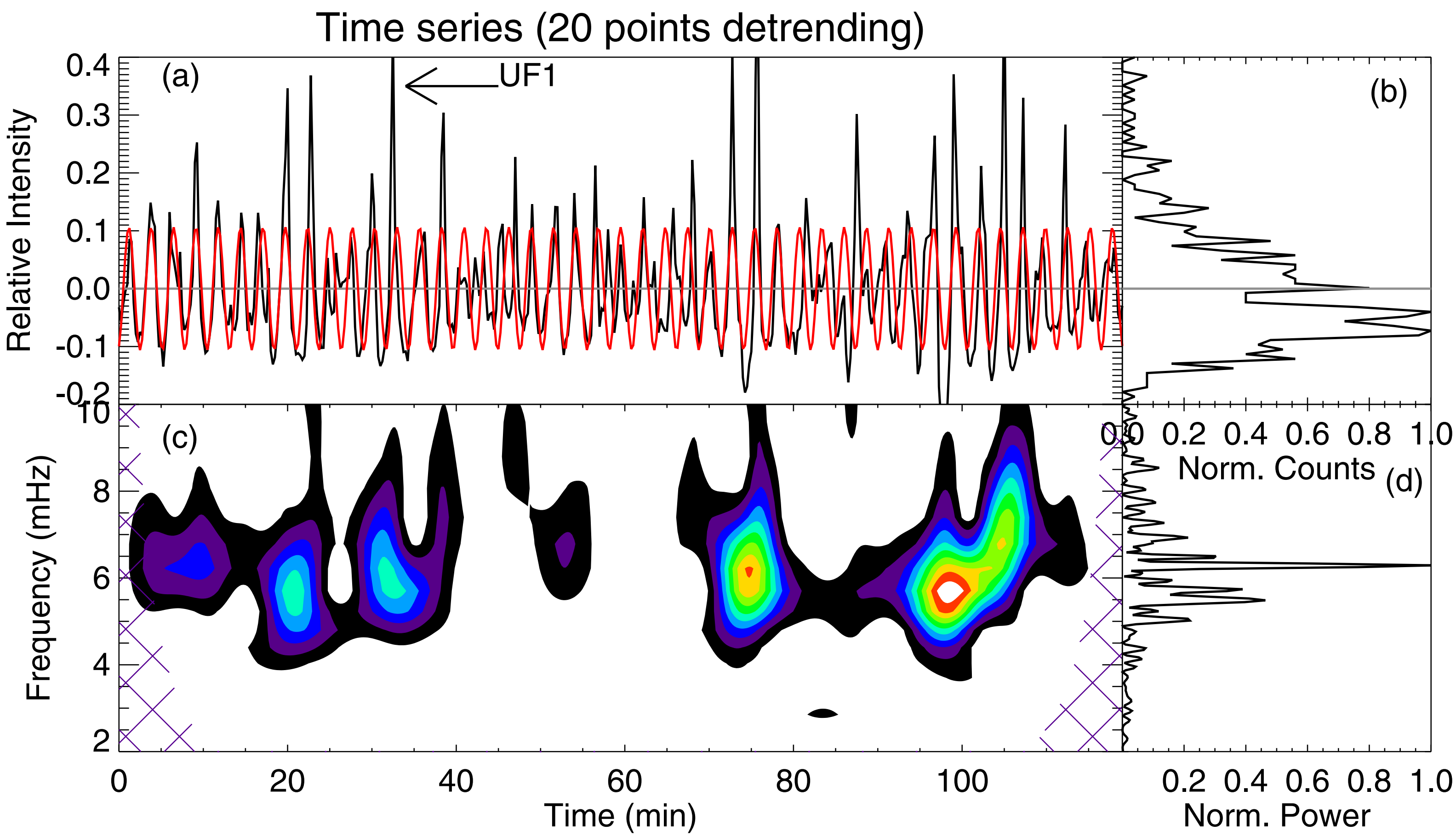
Mother function



Morlet mother function is frequently used in analyzing oscillatory signals.

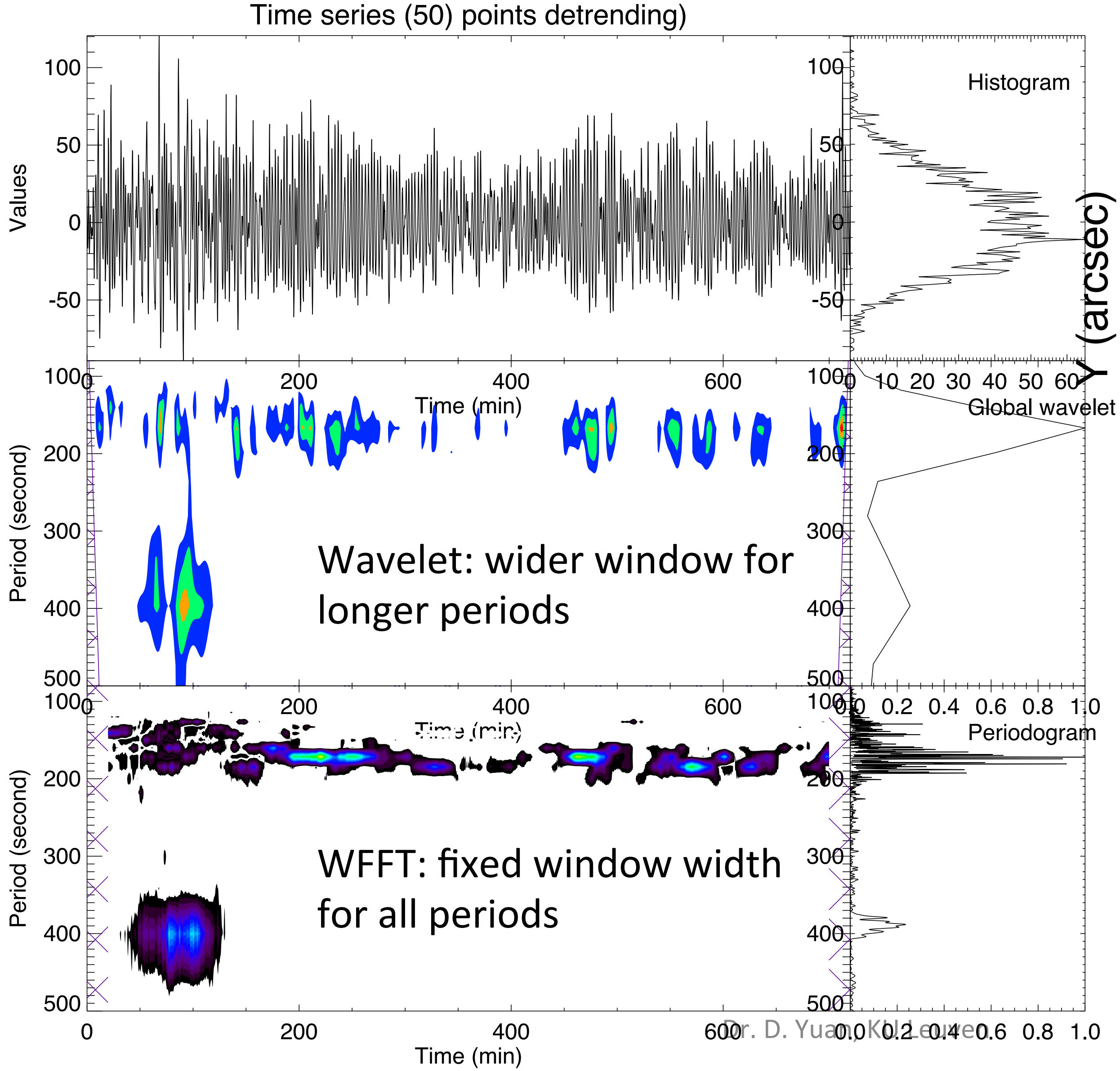
Torrence & Compo 1998, BAMS
IDL routines:
http://paos.colorado.edu/research/wavelets/
Farge, 1992, Annu. Rev. Fluid Mech.
De Moortel 2002 A&A 381, 311
Sych 2008 Sol. Phys. 248, 395

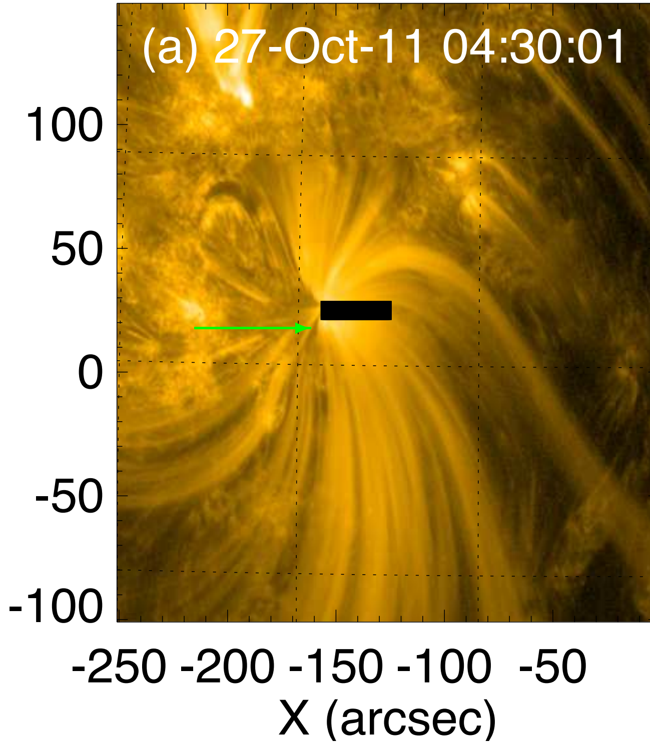
Examples: umbral waves



Yuan et al 2014, ApJ 792

3-min slow wave & instrumental effect





Yuan, 2013, PhD thesis, University of Warwick

A trade-off to make betwenn spectral and temporal resolution

Periodogram

$$P_y(\omega) = \frac{1}{2}\left\{\frac{\left[\sum_j y_j \cos\omega(t_j-\tau)\right]^2}{\sum_j \cos^2\omega(t_j-\tau)} + \frac{\left[\sum_j y_j \sin\omega(t_j-\tau)\right]^2}{\sum_j \sin^2\omega(t_j-\tau)}\right\}.$$

$$\tan(2\omega\tau) = \left(\sum_j \sin 2\omega t_j\right) \bigg/ \left(\sum_j \cos 2\omega t_j\right)$$

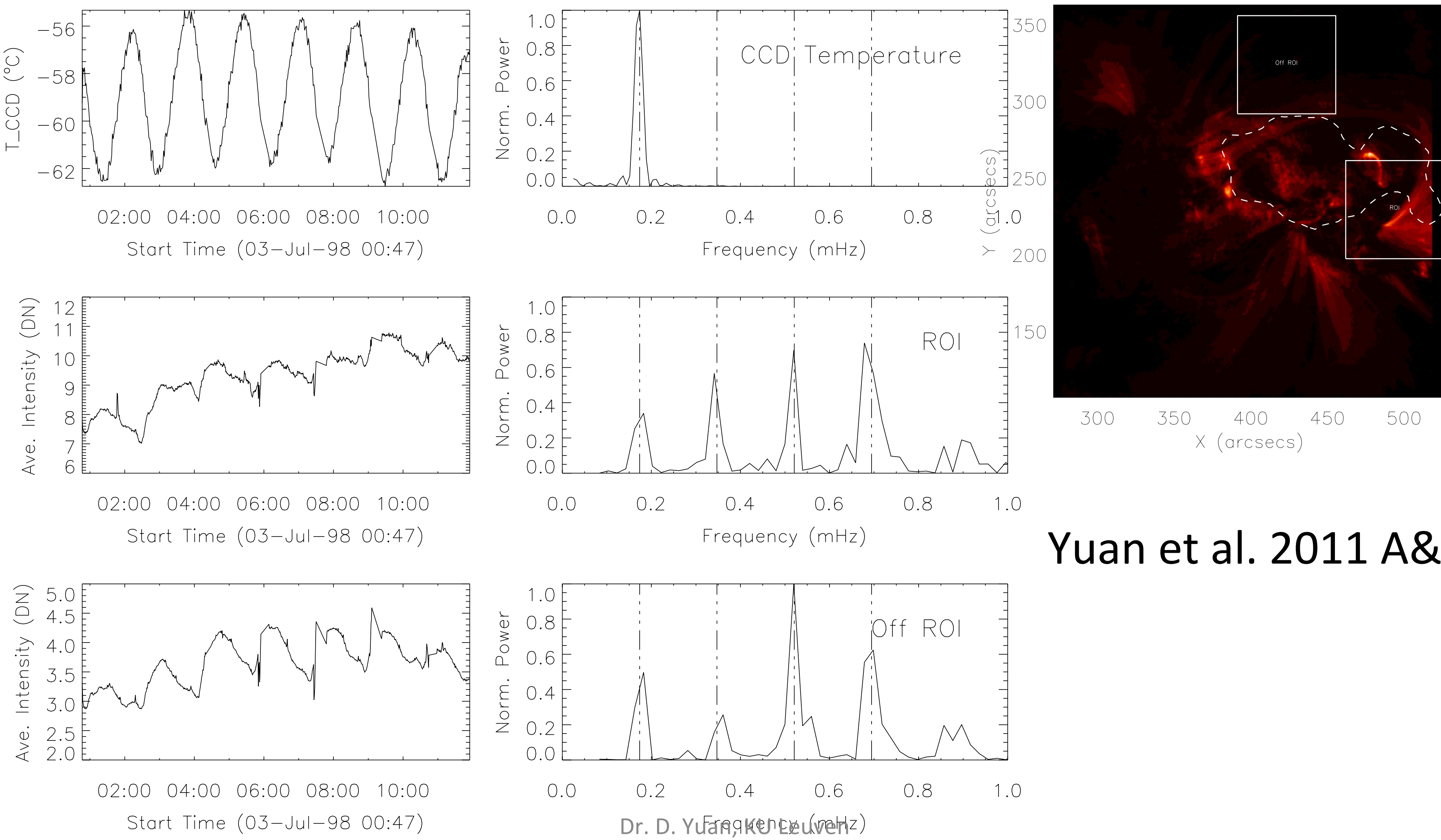
Periodogram is equivalent to least-square fit method that is effective in extracting periodic component in unevenly spaced data.

Scargle, 1982 ApJ, Horne & Baliunas 1986, ApJ

IDL routine:

http://www.arm.ac.uk/~csj/idl/PRIMITIVE/scargle.pro

Example: CCD temperature-induced EUV image intensity variation



Yuan et al. 2011 A&A

Date-compensated DFT

- Orthogonal Basis:

$$H_0(t_j) = 1$$

$$H_1(t_j) = \cos \omega t_j$$

$$H_2(t_j) = \sin \omega t_j$$

Orthonormal basis:

$$h_0 = a_0 H_0$$

$$h_1 = a_1 H_1 - a_1 h_0 < h_0, H_1 >$$

$$h_2 = a_2 H_2 - a_2 h_0 < h_0, H_2 >$$

$$- a_2 h_1 < h_1, H_2 >$$

$$< y_1, y_2 >= \sum_1^N y_1(t_j) y_2(t_j)$$

$$P(\omega) = F(\omega) F^*(\omega)$$

$$F(\omega) = < y, h_1 + i h_2 > / a_0 \sqrt{2},$$

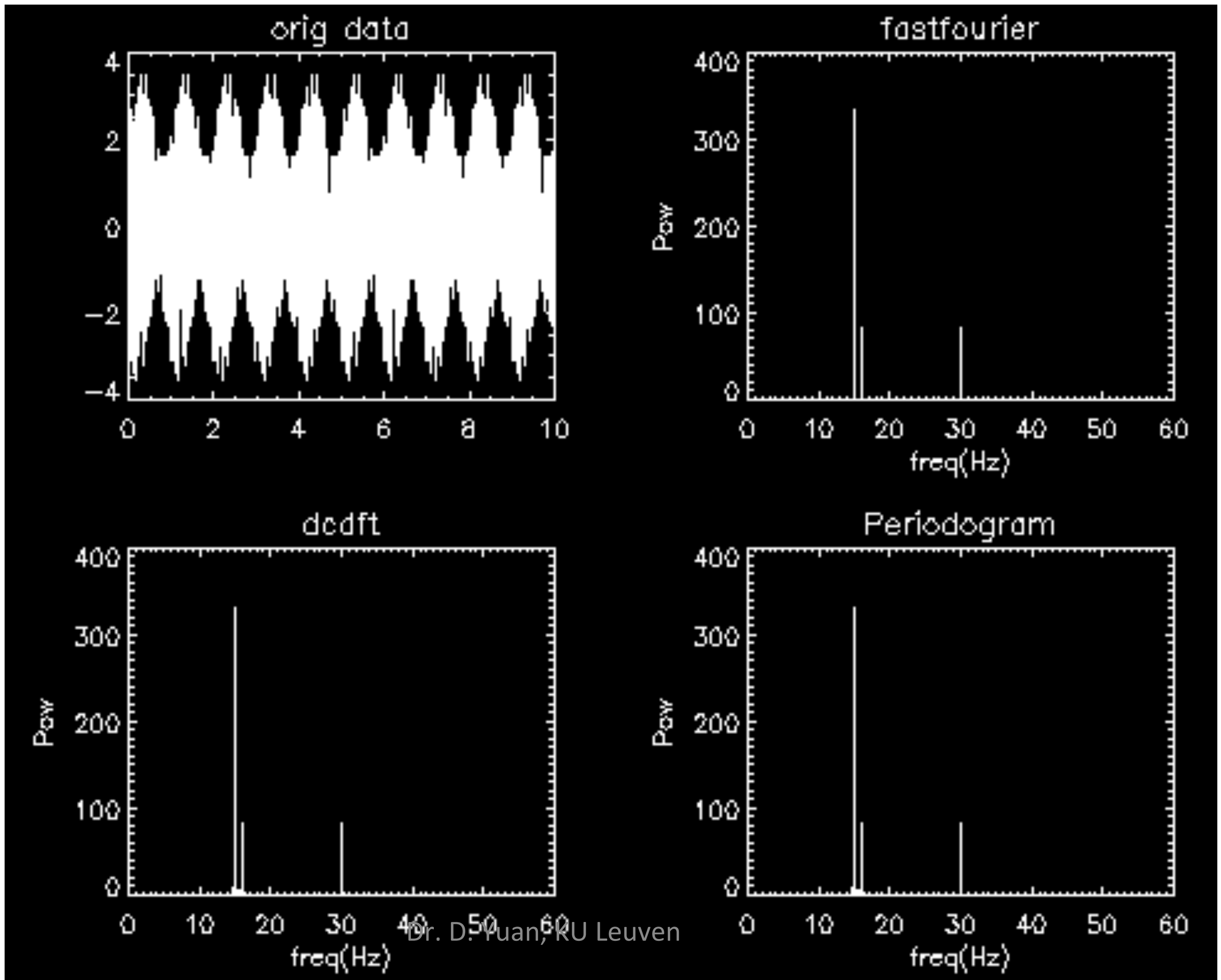
DCDFT: Every frequency component shares a fraction of the mean value.

DFT: Only the zero frequency component contains the mean value.

Ferraz-Mello 1981 AJ

Examples: DFDFT vs FFT

- xx=2*sin(2*!pi*15*tt)-cos(2*!pi*16*tt)+sin(2*!pi*30*tt) ; A time series of freq=[15, 16, 30]



Advantages of DCDFT and Lamb-Scargle periodogram

- Applicable to unevenly spaced data;
- Compute the power of any frequency or a selected range instantly ;
- Periodogram is associated with a significance test;
- DCDFT estimate the amplitude (power) better than other methods.
- DCDFT could estimate the phase, amplitude and residue, therefore could extract any frequency component without resort to spectral domain (Harmonic filter).

Frequency filter

- Apply a window function in frequency domain
- Remove unwanted signal or noise

$$y_1(t) = y(t) * w(t)$$

$$Y_1(\omega) = Y(\omega) \cdot W(\omega)$$

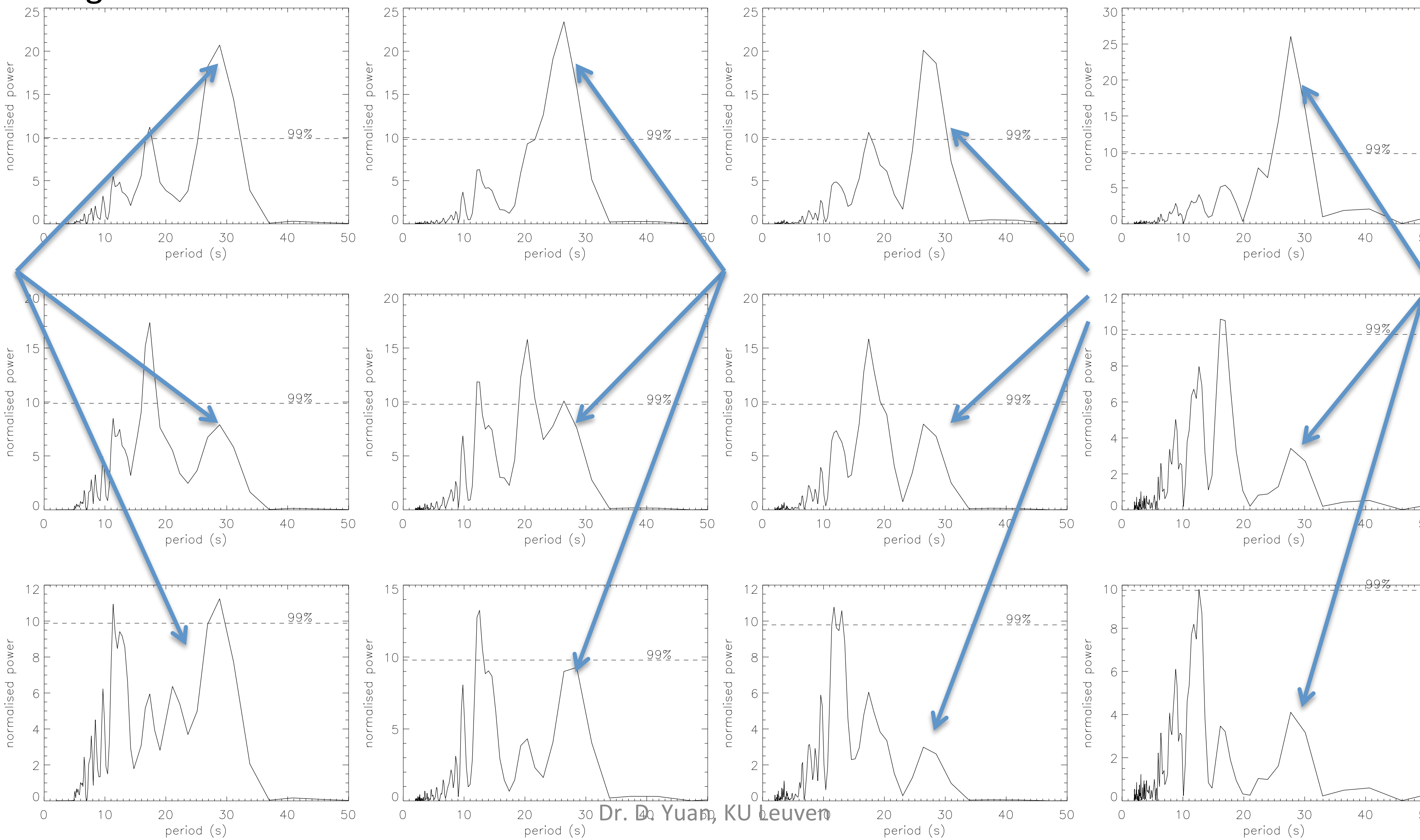
$$Y(\omega) = \mathrm{FFT}[y(t)]$$

$$W(\omega) = \mathrm{FFT}[w(t)]$$

IDL usage: y1=FFT(W*FFT(y),/inverse);

Multi-mode QPP detected with NoRH

Inglis & Nakariakov A&A 2009



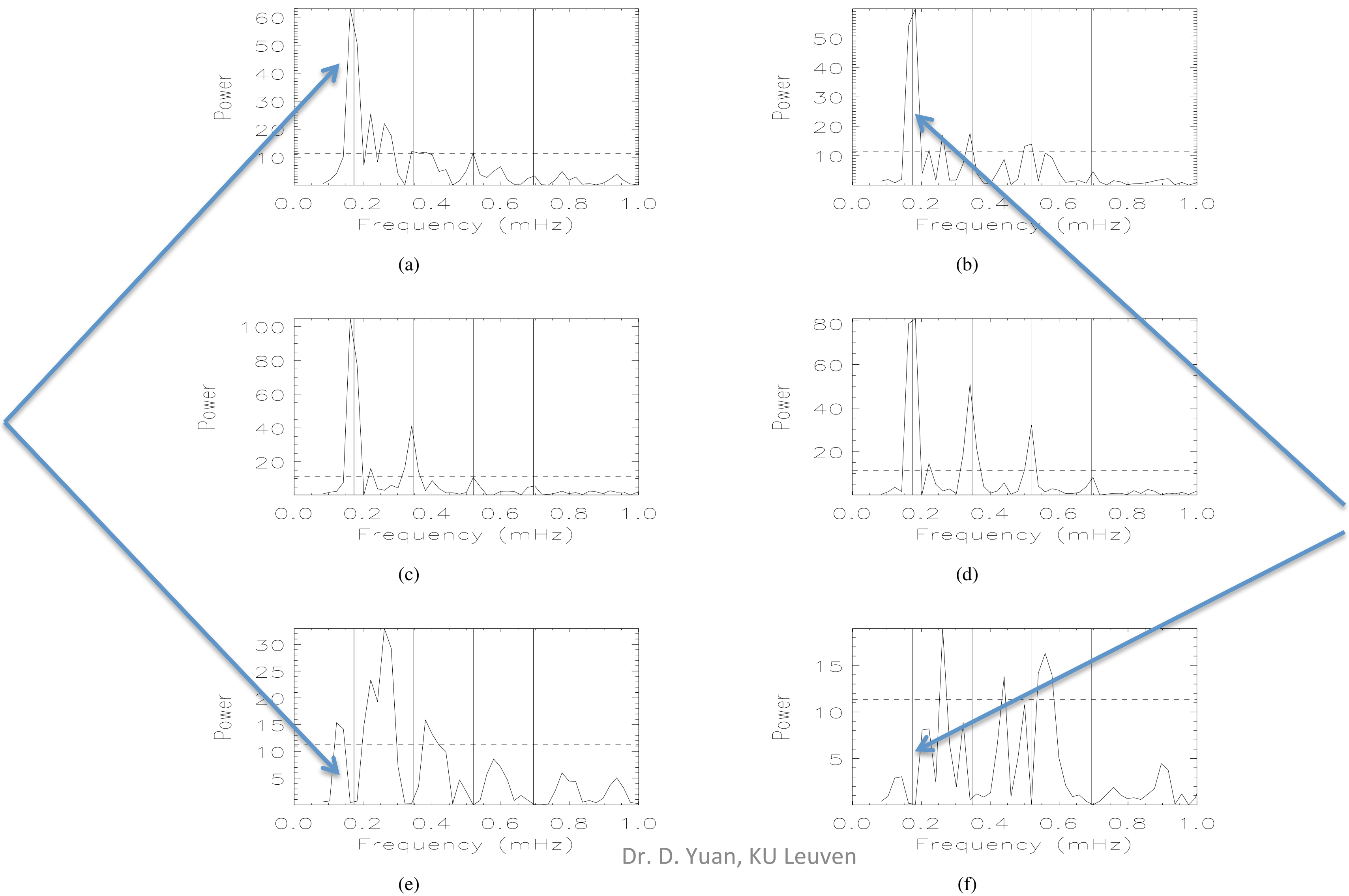
Time-domain Filter: Harmonic filter

$$y_1(t) = y(t) - a - b\cos\omega t - c\sin\omega t$$

a,b, and c are calculated with DCDFT

Ferraz-Mello AJ 1981
Yuan et al A&A 2011

Removing TRACE orbital periods



Frequency *vs* time domain filter

- Frequency filter: No clean removal of a single spectral component due to aliasing,
- Easy to implement and capable of wide band filtering.

- Time domain filter: clean removal
- Good at removing one spectral component.

Noise estimate in FFT

$P_k^N = NY_k^2 / 2\sigma^2$ is the normalized Fourier power

Y_k is the FFT of y_j

σ^2 is the total variance of y_j

α is the lag-1 auto-correlation coefficient of y_j

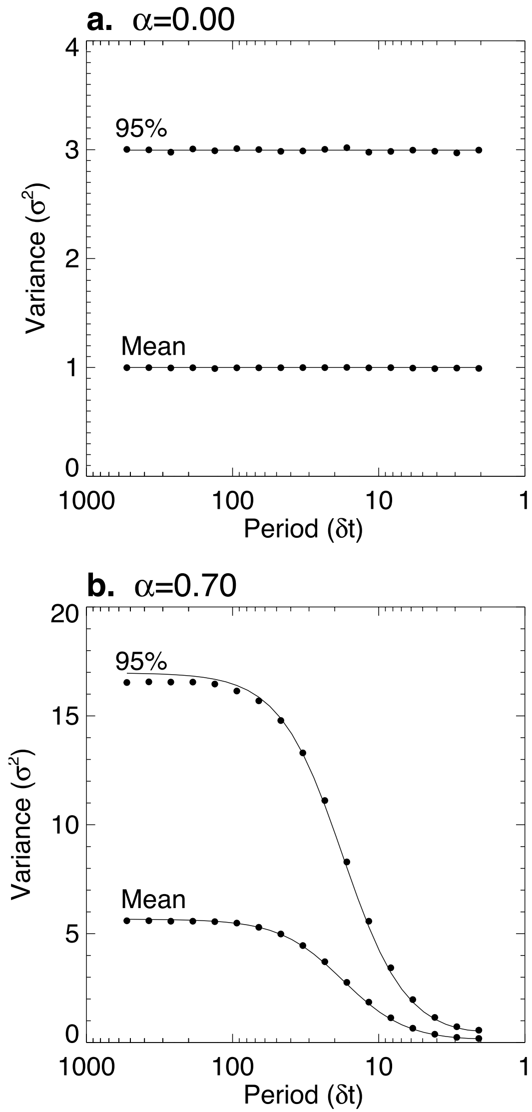
The red noise spectrum is

$$P_k = \frac{1-\alpha^2}{1+\alpha^2 - 2\alpha\,\mathrm{co}(2\pi k / N)}$$

$P_k = 1$ (normalized mean variance) for white noise $\alpha = 0$.

Torrence & Compo 1998

FFT and Wavelet noise level



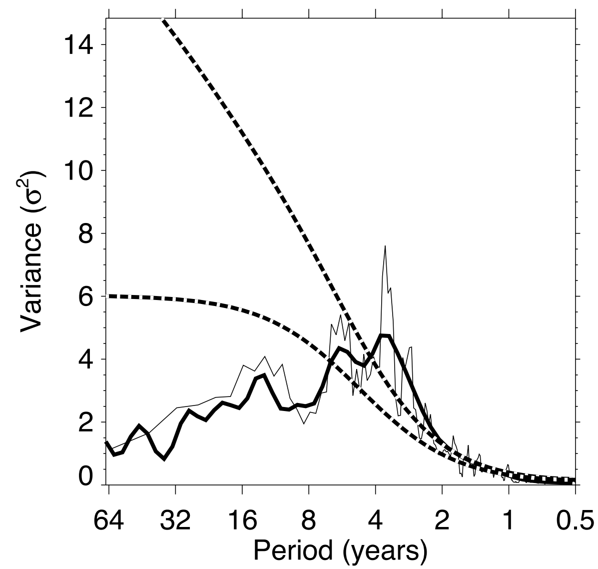
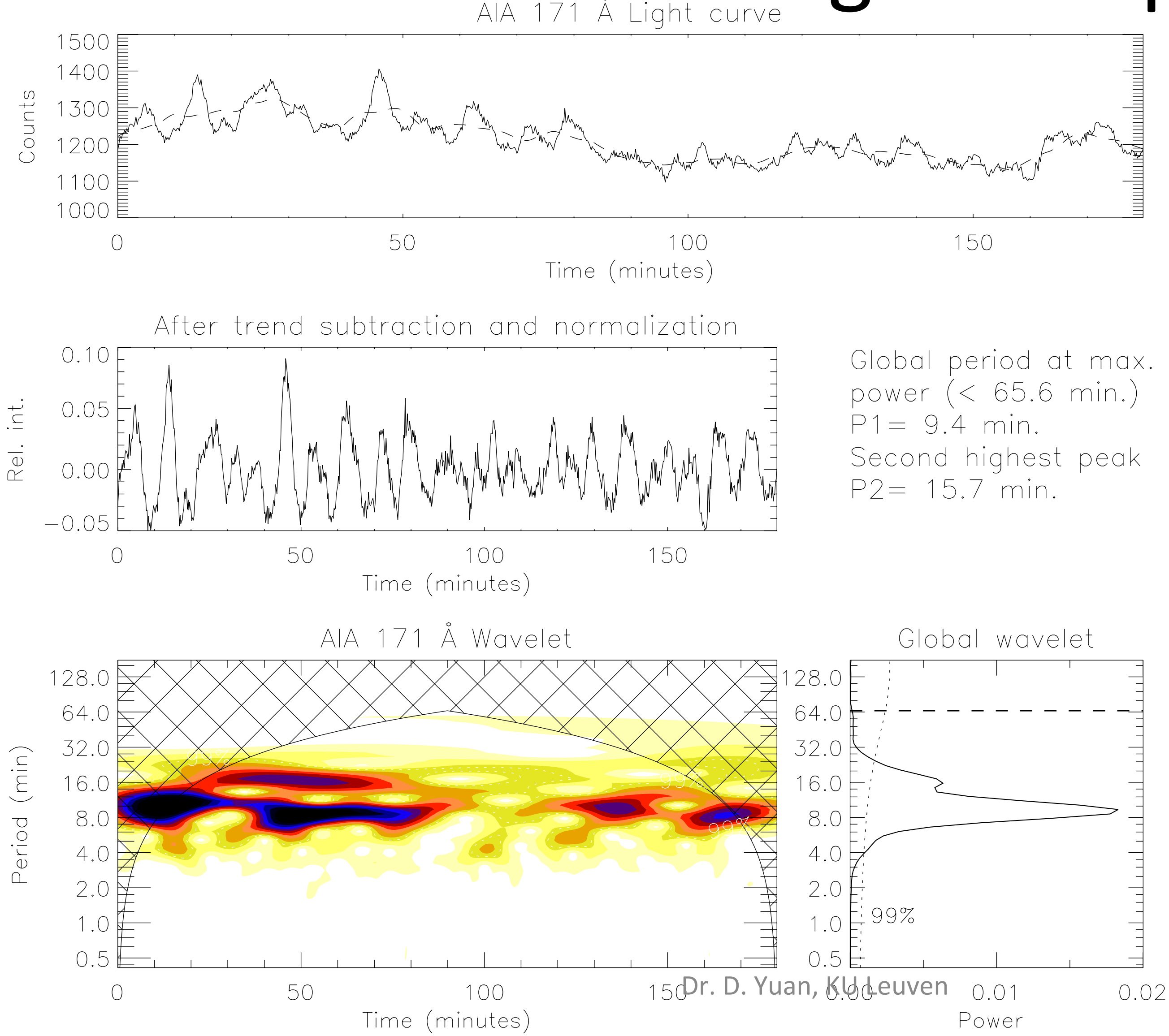


FIG. 6. Fourier power spectrum from Fig. 3, smoothed with a five-point running average (thin solid line). The thick solid line is the global wavelet spectrum for the Niño3 SST. The lower dashed line is the mean red-noise spectrum, while the upper dashed line is the 95% confidence level for the global wavelet spectrum, assuming $\alpha = 0.72$.

Torrence & Compo 1998

Long period oscillating in active region loops



Krishna Prasad et al
A&A 2012

False alarm probability in periodogram

$P_N(\omega_k) = P_y(\omega_k)/\sigma_y^2$ follows exponential distribution.

let $Z = \max\{P_N(\omega_k)\}$, the probability that Z is above a certain power level

$\Pr\{Z > z\} = 1 - [1 - e^{-z}]^M$, M is the number of independent frequencies

At a small probality p_0, a random noise generate a power at level z_0,

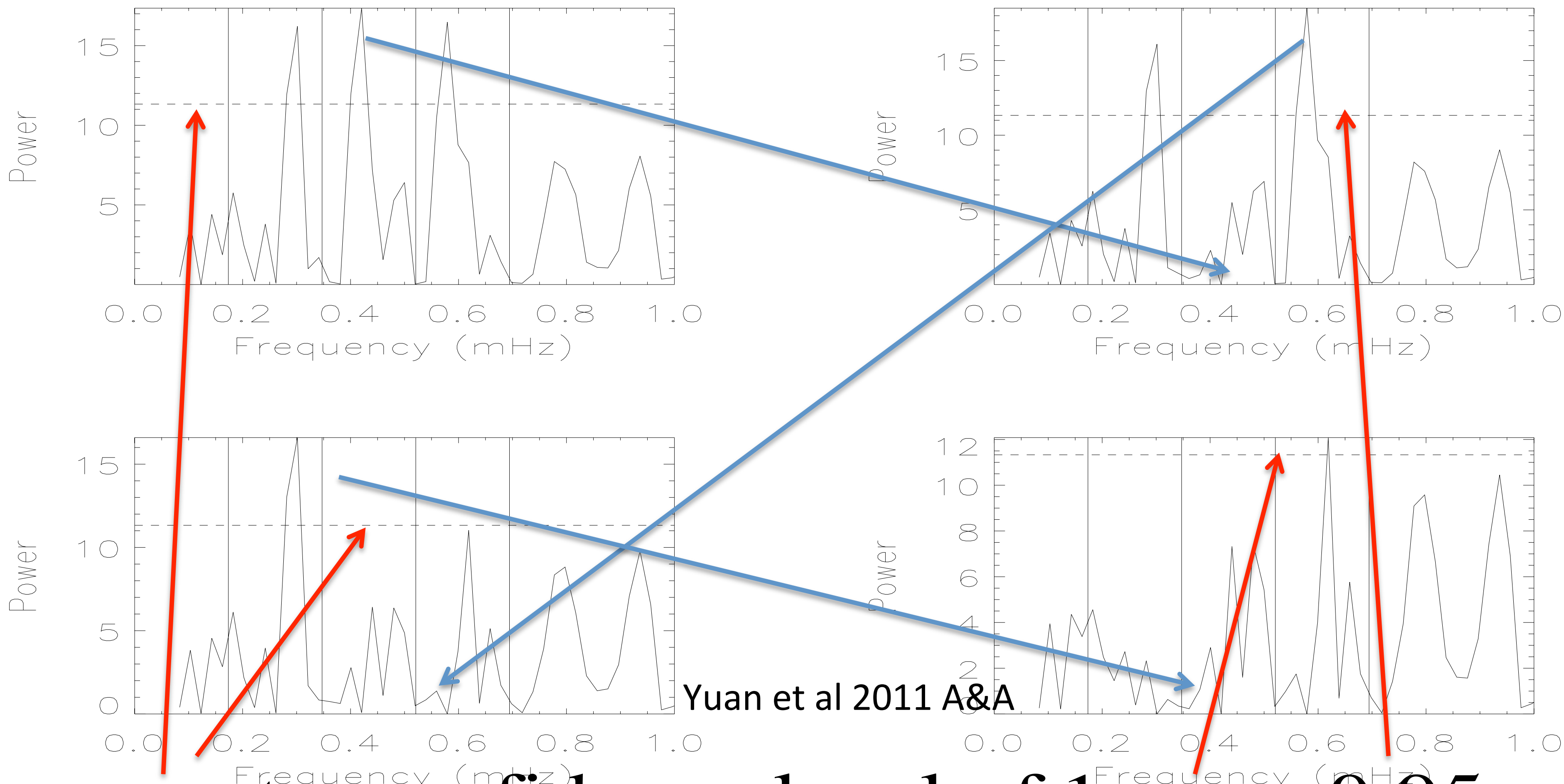
$z_0 = -\ln[1 - (1 - p_0)^{1/M}]$

Above level z_0, the power is signficant at a confidence level of 1- p_0

p_0=0.01,0.03,0.05, e.g.

Horne & Baliunas 1986, ApJ

Interactive significance test for multi peaks



Yuan et al 2011 A&A

z_0 at a confidence level of $1 - p_0 = 0.95$

Fisher's randomization test

Randomly permute two data,

$$y_j = \{y_0, y_1, y_2, \ldots y_{N-1}\} \Rightarrow P_m^{y_j}$$

$$y_{rj} = \{y_{r0}, y_{r1}, y_{r2}, \ldots y_{rN-1}\} \Rightarrow P_m^{y_{rj}}$$

if the time series is better organized

in favor of a dominant peak at frequencey m then,

$$P_m^{y_{rj}} > P_m^{y_j}$$

Repeat M times, and in R cases such scenario occur,

then peak at m is false at a probalbity of $p_0 = R / M$

Linnell Nemec & Nemec 1985 AJ 90,

Yuan et al 2011 A&A

Starlink-PERIOD package: http://starlink.eao.hawaii.edu/starlink

Summary

- Fundamentals of statistics
- Pre-processing methods
- Fourier transform
- Windowed Fourier transform
- Wavelet
- Periodogram
- Date-compensate Discrete Fourier transform
- Filtering method: time and spectral domain
- Significance tests and noise analysis