

Power Spectrum Estimation

Lecture 15

Spring 2002

Wiener-Khintchine Theorem

Let $x(n)$ be a WSS random process with autocorrelation sequence

$$r_{xx}(m) = E[x(n+m)x^*(n)]$$

The power spectral density is *defined* as the Discrete Time Fourier Transform of the autocorrelation sequence

$$\mathcal{P}_{xx}(f) = T \sum_{n=-\infty}^{\infty} r_{xx}(m) e^{-i2\pi f m T}$$

where T is the sampling interval.

The signal is assumed to be bandlimited in frequency to $\pm 1/2T$ and is periodic in frequency with period $1/T$.

Lecture 15

1

Wiener-Khintchine Theorem

The inverse DTFT is

$$r_{xx}(m) = \int_{-1/2T}^{1/2T} \mathcal{P}_{xx}(f) e^{i2\pi f m T} df$$

And, $r_{xx}(0)$ is the average power

$$r_{xx}(0) = \int_{-1/2T}^{1/2T} \mathcal{P}_{xx}(f) df$$

Due to the property $r_{xx}(-m) = r_{xx}^*(m)$, the PSD must be a strictly real, nonnegative function.

Lecture 15

2

White Noise

A white noise process is zero for all lags except $m = 0$. Thus,

$$r_{ww}(m) = \sigma_w^2 \delta(m)$$

The PSD is therefore

$$\mathcal{P}_{ww}(f) = \sigma_w^2 T$$

If $y(n)$ is the sequence that is produced by exciting a discrete linear system with white noise, then

$$P_{yy}(f) = \sigma_w^2 H(z) H(1/z) \Big|_{z=e^{i2\pi f T}}$$

Lecture 15

3

Ergodic Random Processes

If a process is ergodic then ensemble averages can be replaced with "time" averages. We will assume ergodic sequences from here on.

Power spectrum definition using time averages:

$$\mathcal{P}_{xx}(f) = \lim_{m \rightarrow \infty} E \left\{ \frac{1}{(2M+1)T} \left| T \sum_{n=-M}^M x(n) e^{-i2\pi f n T} \right|^2 \right\}$$

This is the basis of the periodogram method for estimating the PSD. The Fourier transform of $x(n)$ is computed, which can be computed via the FFT. To reduce the variance in the estimate, many spectra can be averaged.

Correlogram Estimate

Begin with the definition:

$$\mathcal{P}_{xx}(f) = \lim_{m \rightarrow \infty} E \left\{ \frac{1}{(2M+1)T} \left| T \sum_{n=-M}^M x(n) e^{-i2\pi f n T} \right|^2 \right\}$$

After some algebra this can be converted to a DTFT of the autocorrelation sequence

$$\mathcal{P}_{xx}(f) = T \sum_{n=-\infty}^{\infty} r_{xx}(m) e^{-i2\pi f m T}$$

The Correlogram method of estimating the PSD first estimates the autocorrelation sequence and then transforms it to estimate the PSD.

Autocorrelation Sequence Estimation

Assume that n data samples indexed $n = 0$ to $N - 1$ are available. A possible estimator of the ACS is

$$\hat{r}_{xx}(m) = \frac{1}{N-m} \sum_{n=0}^{N-m-1} x(n+m)x^*(n)$$

The values of m for which the ACS is computed are known as the "lag" indexes.

This estimator is unbiased, since

$$E[\hat{r}_{xx}(m)] = \frac{1}{N-m} \sum_{n=0}^{N-m-1} E[x(n+m)x^*(n)] = r_{xx}(m)$$

ACS Estimation

The variance is approximately

$$\text{var}[\hat{r}_{xx}(m)] \approx \frac{N}{(N-m)^2} \sum_{-\infty}^{\infty} (r_{xx}^2(k) + r_{xx}(k+m)r_{xx}(k-m))$$

for $N \gg m$. Note that the variance increases for lags close to N .

Biased ACS Estimator

An alternative estimate is

$$\tilde{r}_{xx}(m) = \frac{1}{N} \sum_{n=0}^{N-m-1} x(n+m)x^*(n)$$

This estimate is related to $\hat{r}_{xx}(m)$ by

$$\tilde{r}_{xx}(m) = \frac{N-m}{N} \hat{r}_{xx}(m)$$

It is a biased estimator with

$$E[\tilde{r}_{xx}(m)] = \frac{N-m}{N} r_{xx}(m)$$

and variance

$$\text{var}[\tilde{r}_{xx}(m)] = \frac{N-m}{N} \text{var}[\hat{r}_{xx}(m)]$$

Numerical Issues

The biased estimator can never produce an autocorrelation matrix that is not positive semi-definite, whereas the unbiased estimator sometimes does.

Hence, there are numerical advantages to the biased estimator in some cases.

Correlogram Estimate of PSD

Assume that the ACS has been estimated for lags $-L \leq m \leq L$. Then the PSD can be estimated by

$$\hat{\mathcal{P}}_{xx}(f) = T \sum_{m=-L}^L \hat{r}_{xx}(m) e^{-i2\pi f m T}$$

The value of the maximum lag index L should be much less than the number of data samples, N . Typically, $L \approx N/10$.

The biased estimator can be substituted for the unbiased estimator.

Correlogram Program

```
FUNCTION CORRELOGRAMPSD,X,Y,LAG,T,NF,R
;+
; PSD=CORRELOGRAMPSD(X,Y,LAG,T,NF)
; computes the power spectrum density by
; the Blackman-Tukey correlogram method.
;
; INPUTS
; X - A Sequence of length NX
; Y - A Sequence of length NY
; LAG - The maximum lag to be used.
; T - The interval between samples
; NF - The number of frequency points
;
; OUTPUT
; PSD = Power Spectral Density estimate at NF points
;
; Notes:
; The length of the correlation sequence is
; N=NX<NY. The minimum is used.
;
; H. Rhody
; May, 1998
;
; REFERENCE
; S. L. Marple,
; Digital Spectral Analysis with Applications,
; Prentice Hall, 1987
; Appendix 5.B
;-
```

Correlogram program, continued

```
lags= FINDGEN(LAG)+1
lags=[lags-LAG-1,0,lags]
w=0.538+0.462*cos(!pi*lags/LAG)

NX=N_ELEMENTS(X)
NY=N_ELEMENTS(Y)
N=NX<NY
R=C_CORRELATE(X[0:N-1],Y[0:N-1],lags,/COVARIANCE)
R=R*N/(N-ABS(lags))
R=R*w

S=[R[LAG:2*LAG],FLTARR(NF-2*LAG-1),R[0:LAG-1]]
PSD=ABS(FFT(S))
RETURN,SHIFT(PSD,NF/2)
END
```

Periodogram PSD Estimator

The periodogram is based on the DTFT of the sample sequence. From the definition,

$$\mathcal{P}_{xx}(f) = \lim_{m \rightarrow \infty} E \left\{ \frac{1}{(2M+1)T} \left| T \sum_{n=-M}^M x(n) e^{-i2\pi f n T} \right|^2 \right\}$$

we can form an estimator like

$$\tilde{\mathcal{P}}(f) = \frac{T}{N} \left| \sum_{n=0}^{N-1} x(n) e^{-i2\pi f n T} \right|^2$$

Improved Periodogram PSD Estimator

This estimator can be improved by

- Smoothing the sample sequence before computing the transform
- Segmenting the sample sequence and computing more estimators that can be averaged
- Using overlapping sample sequences

These techniques can be combined to construct the Welch Periodogram estimator.

Periodogram Estimator Program

```
FUNCTION PERIODOGRAM,X,Y,D,S,WINDOW=window,WINFUN=w
;+
; PXY=PERIODOGRAM(X,Y,D,S)
; computes the cross power spectral density between
; sequences X and Y. If X and Y are the same sequence
; then a direct PSD is computed.
;
;
; INPUT PARAMETERS
; X   An array of data samples
; Y   An array of data samples
; D   The size of the analysis window
; S   The shift of the analysis window
;
; The window function is determined by the keyword WINDOW:
; WINDOW=0 (Default) All values equally weighted (no window)
; WINDOW=1 Hamming Window
; WINDOW=2 Bartlett Window
; WINDOW=3 Hann Window
; WINDOW=4 Truncated Gaussian
;
```

Periodogram Estimator Program (continued)

```
; The window function can be accessed via WINFUN keyword
;
; If X and Y are not the same length then the shorter
; one sets the length for the analysis.
;
; Example: Find the PSD of a sequence X at 500 points
; using a window of size 80 and shift of size 40.
;
; PSD=PERIODOGRAM(X,X,500,80,Window=1)
; PLOT,PSD
; Note that the PSD plot contains 500 points.
;
; H. Rhody May, 1998
; Revised April 2000 to include window functions and
; remove extraneous parameter NF. Added access to
; window function via WINFUN.
;
; Reference:
; S. L. Marple,
; Digital Spectral Analysis with Applications,
; Prentice Hall, 1987
; Appendix 5.C
;-
```

Periodogram Estimator Program (continued)

```
PSD=FLTARR(D)
XS=FLTARR(D)
YS=FLTARR(D)

IF NOT KEYWORD_SET(window) THEN window=0
k=(findgen(D)-(D-1)/2.)/(D-1)
CASE window OF
0: w=FLTARR(D)+1 ; No window. Default
1: BEGIN ; Hamming Window
    w=0.538+0.462*COS(2*!pi*k)
    END
2: BEGIN ; Bartlett Window
    w=1-2*ABS(k)
    END
3: BEGIN ; Squared Cosine (Hann) Window
    w=0.5+0.5*cos(2*!pi*k)
    END
4: BEGIN
    w=exp(-12.5*k^2)
    END
ELSE: MESSAGE, 'Unrecognized WINDOW selection'
ENDCASE
```

Periodogram program, continued

```
U=TOTAL(w)/D
w=w/U
NX=N_Elements(X)
NY=N_Elements(Y)
N=NX<NY
NP=(N-D)/S+1

FOR p=0, NP-1 DO BEGIN
    XS[0:D-1]=X(p*S:p*S+D-1)*w
    YS[0:D-1]=Y(p*S:p*S+D-1)*w
    XSF=FFT(XS)
    YSF=FFT(YS)
    PSD=PSD+XSF*CONJ(YSF)
ENDFOR
RETURN, ABS(PSD)/NP
END
```

Example

In this example we will produce a random sequence by passing white noise through a pair of narrowband filters.

We will then show the spectra by doing a correlogram estimate and a periodogram estimate.

The program is called `sigdemo8.pro`

Sigdemo8 program

```

PRO SIGDEMO8
; This procedure demonstrates the generation of a random waveform
; by filtering a random process. The power spectrum of the waveform
; is then estimated by averaging power spectra over several members
; of the ensemble.

n=1000; Length of the sample function
f0=0.05
f1=0.13
w0=2*pi*f0; Frequency of one filter resonance.
w1=2*pi*f1; Frequency of second filter resonance.

r0=0.95; Damping ratio for first resonance point
r1=0.95; Damping ratio for second resonance point

; Establish the parameters of the filter pair
a0=[1,-2*r0*cos(w0),r0^2] & b0=[0,r0*sin(w0)]
a1=[1,-2*r1*cos(w1),r1^2] & b1=[0,r1*sin(w1)]

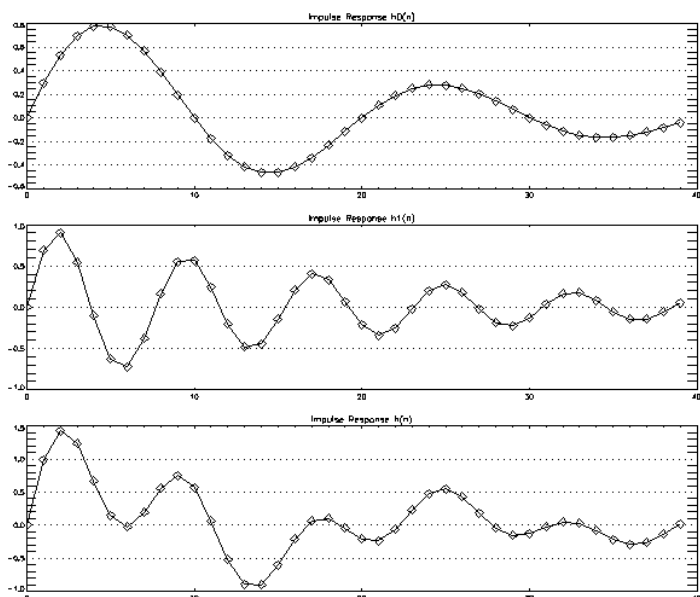
```

Sigdemo8 program

```

WINDOW,1,XSIZE=600,YSIZE=600
!P.MULTI=[0,1,3]
k=ALOG(0.1)/ALOG(r0)>ALOG(0.1)/ALOG(r1)
k=10*(FIX(k)/10)
; Calculate the impulse response
xi=FLTARR(k) & xi[0]=1
y0=filter(b0,a0,xi)
y1=filter(b1,a1,xi)
yi=y0+y1 ;impulse response
PLOT,yi0,yticklen=1,ygridstyle=1,psym=-4,TITLE='Impulse Response h0(n)'
PLOT,yi1,yticklen=1,ygridstyle=1,psym=-4,TITLE='Impulse Response h1(n)'
PLOT,yi,yticklen=1,ygridstyle=1,psym=-4,TITLE='Impulse Response h(n)'

```



Sigdemo8 program

```

WINDOW,2,XSIZE=600,YSIZE=600
!P.MULTI=[0,1,4]
x=randomn(seed,n)
y0=filter(b0,a0,x)
y1=filter(b1,a1,x)
y=y0+y1;
PSD=CORRELOGRAMPSD(y,y,100,400,CORRELATION=R)
n=N_ELEMENTS(PSD)
f=FINDGEN(n/2+1)/n
nr=N_ELEMENTS(R)
tau=FINDGEN(nr)-nr/2.
plot,y[0:40],psym=-4,TITLE='Filter Output with White Noise Input'
PLOT,tau,R,Xtickv=[-nr/2.,-nr/4.,0,nr/4.,nr/2.],$
xticks=6,xticklen=1,yticklen=1,xgridstyle=1,ygridstyle=1,$
TITLE='Autocorrelation Sequence with White Noise Input'
PLOT,f,PSD(0:n/2),TITLE='Correlogram Estimate',XTITLE='Normalized Frequency'
oplot,[f0,f0],[0,1]
oplot,[f1,f1],[0,1]
PSD=PERIODOGRAM(Y,Y,200,120,WINDOW=1)
n=N_ELEMENTS(PSD)
f=FINDGEN(n/2+1)/n
PLOT,f,PSD(0:n/2),TITLE='Periodogram Estimate',XTITLE='Normalized Frequency'
oplot,[f0,f0],[0,1]
oplot,[f1,f1],[0,1]

!P.MULTI=0
END

```

