
Subject: dpss / multi-taper / ssa toolkit?

Posted by [stoatstoat](#) on Fri, 11 Jun 1999 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi...

anyone out there have IDL code for producing DPSS sequences, or a more complete code to do multi-taper spectral analysis, or have some rough equivalent of the ssa toolkit in IDL?

Matlab, apparently, can do dpss...

- William Connolley "at home"
- "Work as though you were in the early days of a better nation"

Sent via Deja.com <http://www.deja.com/>
Share what you know. Learn what you don't.

Subject: Re: dpss / multi-taper / ssa toolkit?

Posted by [roy.hansen](#) on Tue, 15 Jun 1999 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

William Connolley wrote:

>
> anyone out there have IDL code for producing DPSS sequences, or
> a more complete code to do multi-taper spectral analysis, or have some
> rough equivalent of the ssa toolkit in IDL?
>
> Matlab, apparently, can do dpss...
>

Here is a function that produces the DPSS based on Drosopoulus and Haykin.
An other approach which is less computationally intensive, is to use sinusoidal tapers instead of the DPSS. The spectral estimate based on this method is below (mbmtse.pro).

Hope this helps.

BTW, I agree that the lack of standard functions in signal processing such as window functions, classical/modern spectral estimation and time-frequency methods in IDL is a pain. Especially when all this is implemented in Matlab.

It is not easy to be an IDL-fan in signal processing :-(

--Roy E. Hansen

```

;-----
; slepian.pro
;-----
function slepian, N, K, lambda, NVEC=nvec
;+
; NAME:
;   slepian
;
; PURPOSE:
;   Calculate the Slepian functions (also called the Prolate Spheroidal
;   Wavefunctions).
;   Uses a method based on the matrix form of the DPSS defining equation.
;   See Drosopoulus A. and Haykin S., Adaptive Radar Parameter Estimation
;   with Thomson's Multiple-window Method, In Haykin S. and Steinhardt A.,
;   "Adaptive Radar Detction and Estimation", pp390-395.
;   The method is O(N^3)!
;
; CATEGORY:
;   Math/DSP
;
; CALLING SEQUENCE:
;   ev = slepian(N, K, lambda, NVEC=nvec)
;
; INPUTS:
;   N:   Number of samples in the function
;   K:   Time-bandwidth product. 4 is a good choice.
;
; KEYWORD INPUTS:
;   NVEC: Number of eigenvectors to return. Default is 2K + 1.
;
; OUTPUTS:
;   ev:   dblarr(N, nvec) of eigenvectors (samples of the Slepian
;         functions).
;   lambda: Vector of all eigenvalues.
;
; MODIFICATION HISTORY:
;   Version 1.0
;   040796 Trygve Sparr @ SUSAR.
;-

message, 'Version 1.0', /CONTINUE

if not keyword_set(NVEC) then nvec = 2*K + 1
NW = K
W = NW/N           ; Local bandwidth

;; Make the matrices S and T, S is tridiagonal and T is Toeplitz

```

```

message, 'Initializing matrices...', /CONTINUE
start0 = systime(1)
T = dblarr(N, N)

for i=0, N - 1 do begin
  for j=0, N - 1 do begin

    if j eq i then begin
      T(i, j) = 2d*W
    endif else begin
      T(i, j) = sin(2d*!dpi*W*(i - j))/(!dpi*(i - j))
    endelse

  endfor
endfor

mt = systime(1)
message, 'Done in' + string(mt - start0) + ' sec', /CONTINUE

;; Solve the Toeplitz equation
message, 'Reducing to symmetric tridiagonal form...', /CONTINUE
tt = t
tired, tt, d, e, /double
stt = systime(1)
message, 'Done in' + string(stt - mt) + ' sec, total' + string(stt - start0) + ' sec', /CONTINUE

message, 'Finding all eigenvectors and eigenvalues...', /CONTINUE
lambda = d
ee = e
triql, lambda, ee, tt, /double
evt = systime(1)
message, 'Done in' + string(evt - stt) + ' sec, total' + string(evt - start0) + ' sec', /CONTINUE

idx = reverse(sort(lambda))
lambda = lambda(idx)

return, tt(*, idx(0:nvec - 1))

end ;; slepian.pro

;-----
; mbmtse.pro
;-----
function mbmtse, x, k
;+
; NAME:
;   mbmtse

```

```

;
;
; PURPOSE:
;   Calculate a spectral estimate as
;   described in:
;       Riedel K.S. and Sidorenko A.,
;       Minimum Bias Multiple Spectral Estimation,
;       IEEE Trans. Signal Processing,
;       vol 43, pp188-195, 1995.
;   The routine gives optimum quadratic spectral estimates
;   of stochastic processes in the sense of minimum bias.
;
; CATEGORY:
;   Signal Processing
;
; CALLING SEQUENCE:
;   sp = mbmtse(x, k)
;
; INPUTS:
;   x:   Discrete time series.
;   K:   Number of tapers.
;
; KEYWORD INPUTS:
;
; OUTPUTS:
;   sp:   Spectral estimate.
;
; MODIFICATION HISTORY:
;   Version 1.0
;   160796 Trygve Sparr @ SUSAR.
;-

N = n_elements(x)

s = fltarr(N)

y = fft(x, -1)
mu = (1. - (findgen(k))^2/K^2)
C = 2d*total(mu)

for j=1, K do begin
    s = s + mu(j - 1)*abs(shift(y, -j) - shift(y, j))^2
endfor

return, s/C

end ;; mbmtse.pro

```
