NAG Toolbox for MATLAB

f04me

1 Purpose

f04me updates the solution to the Yule–Walker equations for a real symmetric positive-definite Toeplitz system.

2 Syntax

```
[x, v, ifail] = f04me(t, x, v, 'n', n)
```

3 Description

f04me solves the equations

$$T_n x_n = -t_n,$$

where T_n is the n by n symmetric positive-definite Toeplitz matrix

$$T_n = \begin{pmatrix} \tau_0 & \tau_1 & \tau_2 & \dots & \tau_{n-1} \\ \tau_1 & \tau_0 & \tau_1 & \dots & \tau_{n-2} \\ \tau_2 & \tau_1 & \tau_0 & \dots & \tau_{n-3} \\ . & . & . & & . \\ \tau_{n-1} & \tau_{n-2} & \tau_{n-3} & \dots & \tau_0 \end{pmatrix}$$

and t_n is the vector

$$t_n^{\mathrm{T}} = (\tau_1 \tau_2 \dots \tau_n),$$

given the solution of the equations

$$T_{n-1} x_{n-1} = -t_{n-1}.$$

The function will normally be used to successively solve the equations

$$T_k x_k = -t_k, \qquad k = 1, 2, \dots, n.$$

If it is desired to solve the equations for a single value of n, then function f04fe may be called. This function uses the method of Durbin (see Durbin 1960 and Golub and Van Loan 1996).

4 References

Bunch J R 1985 Stability of methods for solving Toeplitz systems of equations *SIAM J. Sci. Statist. Comput.* **6** 349–364

Bunch J R 1987 The weak and strong stability of algorithms in numerical linear algebra *Linear Algebra Appl.* **88/89** 49–66

Cybenko G 1980 The numerical stability of the Levinson–Durbin algorithm for Toeplitz systems of equations *SIAM J. Sci. Statist. Comput.* **1** 303–319

Durbin J 1960 The fitting of time series models *Rev. Inst. Internat. Stat.* **28** 233

Golub G H and Van Loan C F 1996 *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

5.1 Compulsory Input Parameters

1: **t(0 : n) – double array**

t(0) must contain the value τ_0 of the diagonal elements of T, and the remaining **n** elements of **t** must contain the elements of the vector t_n.

Constraint: **t**$(0) > 0.0$. Note that if this is not true, then the Toeplitz matrix cannot be positive-definite.

2: **x(*) – double array**

Note: the dimension of the array **x** must be at least $\max(1, \mathbf{n})$.

With $\mathbf{n} > 1$ the $(n-1)$ elements of the solution vector x_{n-1} as returned by a previous call to f04me. The element **x**(**n**) need not be specified.

Constraint: $|\mathbf{x}(\mathbf{n}-1)| < 1.0$. Note that this is the partial (auto)correlation coefficient, or reflection coefficient, for the $(n-1)$th step. If the constraint does not hold, then T_n cannot be positive-definite.

3: **v – double scalar**

With $\mathbf{n} > 1$ the mean square prediction error for the $(n-1)$th step, as returned by a previous call to f04me.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The dimension of the array **x**.

The order of the Toeplitz matrix T.

Constraint: $\mathbf{n} \geq 0$. When $\mathbf{n} = 0$, then an immediate return is effected.

5.3 Input Parameters Omitted from the MATLAB Interface

work

5.4 Output Parameters

1: **x(*) – double array**

Note: the dimension of the array **x** must be at least $\max(1, \mathbf{n})$.

The solution vector x_n. The element **x**(**n**) returns the partial (auto)correlation coefficient, or reflection coefficient, for the nth step. If $|\mathbf{x}(\mathbf{n})| \geq 1.0$, then the matrix T_{n+1} will not be positive-definite to working accuracy.

2: **v – double scalar**

The mean square prediction error, or predictor error variance ratio, ν_n, for the nth step. (See Section 8 and the Introduction to Chapter G13.)

3: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= -1$

On entry, $\mathbf{n} < 0$,
or $\mathbf{t}(0) \leq 0.0$,
or $\mathbf{n} > 1$ and $|\mathbf{x}(\mathbf{n} - 1)| \geq 1.0$.

ifail $= 1$

The Toeplitz matrix T_{n+1} is not positive-definite to working accuracy. If, on exit, $\mathbf{x}(\mathbf{n})$ is close to unity, then the principal minor was probably close to being singular, and the sequence $\tau_0, \tau_1, \ldots, \tau_{\mathbf{n}}$ may be a valid sequence nevertheless. **x** returns the solution of the equations

$$T_n x_n = -t_n,$$

and **v** returns v_n, but it may not be positive.

7 Accuracy

The computed solution of the equations certainly satisfies

$$r = T_n x_n + t_n,$$

where $\|r\|_1$ is approximately bounded by

$$\|r\|_1 \leq c\epsilon\left(\prod_{i=1}^{n}(1 + |p_i|) - 1\right),$$

c being a modest function of n, ϵ being the ***machine precision*** and p_k being the kth element of x_k. This bound is almost certainly pessimistic, but it has not yet been established whether or not the method of Durbin is backward stable. For further information on stability issues see Bunch 1985, Bunch 1987, Cybenko 1980 and Golub and Van Loan 1996. The following bounds on $\left\|T_n^{-1}\right\|_1$ hold:

$$\max\left(\frac{1}{v_{n-1}}, \frac{1}{\prod_{i=1}^{n-1}(1 - p_i)}\right) \leq \left\|T_n^{-1}\right\|_1 \leq \prod_{i=1}^{n-1}\left(\frac{1 + |p_i|}{1 - |p_i|}\right),$$

where v_n is the mean square prediction error for the nth step. (See Cybenko 1980.) Note that $v_n < v_{n-1}$. The norm of T_n^{-1} may also be estimated using function f04yc.

8 Further Comments

The number of floating-point operations used by this function is approximately $4n$.

The mean square prediction errors, v_i, is defined as

$$v_i = \left(\tau_0 + t_{i-1}^{\mathrm{T}} x_{i-1}\right)/\tau_0.$$

Note that $v_i = \left(1 - p_i^2\right)v_{i-1}$.

9 Example

```
t = [4; 3; 2; 1; 0];
v = 0;
x=[0];
fprintf('\n');
```

```
for k=1:4
  [x, v, ifail] = f04me(t(1:k+1), x, v);
  fprintf('Solution for system of order %d\n', k);
  disp(transpose(x));
  fprintf('Mean square prediction error\n    %6.4g\n\n', v);
  if k < 4
    x = [x; 0]; % Extend x by one element
  end
end
```

```
Solution for system of order 1
   -0.7500
Mean square prediction error
    0.4375

Solution for system of order 2
   -0.8571    0.1429
Mean square prediction error
    0.4286

Solution for system of order 3
   -0.8333    0.0000    0.1667
Mean square prediction error
    0.4167

Solution for system of order 4
   -0.8000    0.0000   -0.0000    0.2000
Mean square prediction error
       0.4
```