NAG Toolbox

nag_matop_real_symm_posdef_fac (f01bu)

1 Purpose

nag_matop_real_symm_posdef_fac (f01bu) performs a $ULDL^{\mathrm{T}}U^{\mathrm{T}}$ decomposition of a real symmetric positive definite band matrix.

2 Syntax

```
[a, ifail] = nag_matop_real_symm_posdef_fac(k, a, 'n', n, 'm1', m1)
[a, ifail] = f01bu(k, a, 'n', n, 'm1', m1)
```

Note: the interface to this routine has changed since earlier releases of the toolbox:

At Mark 22: **m1** was made optional.

3 Description

The symmetric positive definite matrix A, of order n and bandwidth $2m+1$, is divided into the leading principal sub-matrix of order k and its complement, where $m \le k \le n$. A UDU^{T} decomposition of the latter and an LDL^{T} decomposition of the former are obtained by means of a sequence of elementary transformations, where U is unit upper triangular, L is unit lower triangular and D is diagonal. Thus if $k=n$, an LDL^{T} decomposition of A is obtained.

This function is specifically designed to precede nag_matop_real_symm_posdef_geneig (f01bv) for the transformation of the symmetric-definite eigenproblem $Ax = \lambda Bx$ by the method of Crawford where A and B are of band form. In this context, k is chosen to be close to $n/2$ and the decomposition is applied to the matrix B.

4 References

Wilkinson J H (1965) *The Algebraic Eigenvalue Problem* Oxford University Press, Oxford

Wilkinson J H and Reinsch C (1971) *Handbook for Automatic Computation II, Linear Algebra* Springer–Verlag

5 Parameters

5.1 Compulsory Input Parameters

1: **k** – INTEGER

k, the change-over point in the decomposition.

Constraint: $\mathbf{m1} - 1 \le \mathbf{k} \le \mathbf{n}$.

2: $\mathbf{a}(lda,\mathbf{n})$ – REAL (KIND=nag_wp) array

lda, the first dimension of the array, must satisfy the constraint $lda \ge \mathbf{m1}$.

The upper triangle of the n by n symmetric band matrix A, with the diagonal of the matrix stored in the $(m+1)$th row of the array, and the m superdiagonals within the band stored in the first m rows of the array. Each column of the matrix is stored in the corresponding column of the array. For example, if $n=6$ and $m=2$, the storage scheme is

$$\begin{array}{cccccc} * & * & a_{13} & a_{24} & a_{35} & a_{46} \\ * & a_{12} & a_{23} & a_{34} & a_{45} & a_{56} \\ a_{11} & a_{22} & a_{33} & a_{44} & a_{55} & a_{66} \end{array}$$

Elements in the top left corner of the array are not used. The following code assigns the matrix elements within the band to the correct elements of the array:

```
for j=1:n
   for i=max(1,j-m1+1):j
     a(i-j+m1,j) = matrix (i,j);
   end
 end
```

5.2 Optional Input Parameters

1: **n** – INTEGER

Default: the second dimension of the array **a**.

n, the order of the matrix A.

2: **m1** – INTEGER

Default: the first dimension of the array **a**.

$m+1$, where m is the number of nonzero superdiagonals in A. Normally **m1** $\ll$ **n**.

5.3 Output Parameters

1: **a**(lda, **n**) – REAL (KIND=nag_wp) array

A stores the corresponding elements of L, D and U.

2: **ifail** – INTEGER

ifail $= 0$ unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

On entry, **k** $<$ **m1** $- 1$ or **k** $>$ **n**.

ifail $= 2$
ifail $= 3$

The matrix A is not positive definite, perhaps as a result of rounding errors, giving an element of D which is zero or negative. **ifail** $= 3$ when the failure occurs in the leading principal sub-matrix of order **k** and **ifail** $= 2$ when it occurs in the complement.

ifail $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

ifail $= -399$

Your licence key may have expired or may not have been installed correctly.

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

The Cholesky decomposition of a positive definite matrix is known for its remarkable numerical stability (see Wilkinson (1965)). The computed U, L and D satisfy the relation $ULDL^{\mathrm{T}}U^{\mathrm{T}} = A + E$ where the 2-norms of A and E are related by $\|E\| \leq c(m+1)^2\epsilon\|A\|$ where c is a constant of order unity and ϵ is the ***machine precision***. In practice, the error is usually appreciably smaller than this.

8 Further Comments

The time taken by nag_matop_real_symm_posdef_fac (f01bu) is approximately proportional to $nm^2 + 3nm$.

This function is specifically designed for use as the first stage in the solution of the generalized symmetric eigenproblem $Ax = \lambda Bx$ by Crawford's method which preserves band form in the transformation to a similar standard problem. In this context, for maximum efficiency, k should be chosen as the multiple of m nearest to $n/2$.

The matrix U is such that $U^{-1}AU^{-\mathrm{T}}$ is diagonal in its last $n-k$ rows and columns, L is such that $L^{-1}U^{-1}AU^{-\mathrm{T}}L^{-\mathrm{T}} = D$ and D is diagonal. To find U, L and D where $A = ULDL^{\mathrm{T}}U^{\mathrm{T}}$ requires $nm(m+3)/2 - m(m+1)(m+2)/3$ multiplications and divisions which, is independent of k.

9 Example

This example finds a $ULDL^{\mathrm{T}}U^{\mathrm{T}}$ decomposition of the real symmetric positive definite matrix

$$\begin{pmatrix} 3 & -9 & 6 & & & & \\ -9 & 31 & -2 & -4 & & & \\ 6 & -2 & 123 & -66 & 15 & & \\ & -4 & -66 & 145 & -24 & 4 & \\ & & 15 & -24 & 61 & -74 & -18 \\ & & & 4 & -74 & 98 & 24 \\ & & & & -18 & 24 & 6 \end{pmatrix}.$$

9.1 Program Text

```
     function f01bu_example

fprintf('f01bu example results\n\n');

% A Banded matrix A in banded storage format
m1 = nag_int(3); n = nag_int(7);
a = [0,  0,   6,  -4,  15,   4, -18;
     0, -9,  -2, -66, -24, -74,  24;
     3, 31, 123, 145,  61,  98,   6];

k = nag_int(4);
[a, ifail] = f01bu(k, a);

ptitle = 'Factorized form of the matrix';

[ifail] = x04ce( ...
                 n, n, nag_int(0), m1-1, a, ptitle);
```

9.2 Program Results

```
     f01bu example results

 Factorized form of the matrix
             1          2          3          4          5          6          7
 1      3.0000    -3.0000     2.0000
 2                 4.0000     4.0000    -1.0000
```

```
 3                         2.0000    5.0000    3.0000
 4                                   3.0000   -4.0000    2.0000
 5                                             5.0000   -1.0000   -3.0000
 6                                                       2.0000    4.0000
 7                                                                 6.0000
```