NAG Toolbox

nag_linsys_real_posdef_band_solve (f04bf)

1 Purpose

nag_linsys_real_posdef_band_solve (f04bf) computes the solution to a real system of linear equations $AX = B$, where A is an n by n symmetric positive definite band matrix of band width $2k + 1$, and X and B are n by r matrices. An estimate of the condition number of A and an error bound for the computed solution are also returned.

2 Syntax

```
[ab, b, rcond, errbnd, ifail] = nag_linsys_real_posdef_band_solve(uplo, kd, ab,
b, 'n', n, 'nrhs_p', nrhs_p)
```

```
[ab, b, rcond, errbnd, ifail] = f04bf(uplo, kd, ab, b, 'n', n, 'nrhs_p', nrhs_p)
```

3 Description

The Cholesky factorization is used to factor A as $A = U^{T}U$, if **uplo** = 'U', or $A = LL^{T}$, if **uplo** = 'L', where U is an upper triangular band matrix with k superdiagonals, and L is a lower triangular band matrix with k subdiagonals. The factored form of A is then used to solve the system of equations $AX = B$.

4 References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A and Sorensen D (1999) *LAPACK Users' Guide* (3rd Edition) SIAM, Philadelphia http://www.netlib.org/lapack/lug

Higham N J (2002) *Accuracy and Stability of Numerical Algorithms* (2nd Edition) SIAM, Philadelphia

5 Parameters

5.1 Compulsory Input Parameters

1: **uplo** – CHARACTER(1)

If **uplo** = 'U', the upper triangle of the matrix A is stored.

If **uplo** = 'L', the lower triangle of the matrix A is stored.

Constraint: **uplo** = 'U' or 'L'.

2: **kd** – INTEGER

The number of superdiagonals k (and the number of subdiagonals) of the band matrix A.

Constraint: **kd** ≥ 0.

3: **ab**$(ldab,:)$ – REAL (KIND=nag_wp) array

The first dimension of the array **ab** must be at least **kd** $+ 1$.

The second dimension of the array **ab** must be at least $\max(1, \mathbf{n})$.

The n by n symmetric band matrix A. The upper or lower triangular part of the symmetric matrix is stored in the first **kd** $+ 1$ rows of the array. The jth column of A is stored in the jth column of the array **ab** as follows:

The matrix is stored in rows 1 to $k+1$, more precisely,

if **uplo** = 'U', the elements of the upper triangle of A within the band must be stored with element A_{ij} in $\mathbf{ab}(k+1+i-j, j)$ for $\max(1, j-k) \leq i \leq j$;

if **uplo** = 'L', the elements of the lower triangle of A within the band must be stored with element A_{ij} in $\mathbf{ab}(1+i-j, j)$ for $j \leq i \leq \min(n, j+k)$.

See Section 9 below for further details.

4: **b**(ldb, :) – REAL (KIND=nag_wp) array

The first dimension of the array **b** must be at least $\max(1, \mathbf{n})$.

The second dimension of the array **b** must be at least $\max(1, \mathbf{nrhs\_p})$.

The n by r matrix of right-hand sides B.

5.2 Optional Input Parameters

1: **n** – INTEGER

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

The number of linear equations n, i.e., the order of the matrix A.

Constraint: $\mathbf{n} \geq 0$.

2: **nrhs_p** – INTEGER

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

The number of right-hand sides r, i.e., the number of columns of the matrix B.

Constraint: $\mathbf{nrhs\_p} \geq 0$.

5.3 Output Parameters

1: **ab**($ldab$, :) – REAL (KIND=nag_wp) array

The first dimension of the array **ab** will be $\mathbf{kd} + 1$.

The second dimension of the array **ab** will be $\max(1, \mathbf{n})$.

If **ifail** $= 0$ or $\mathbf{n} + 1$, the factor U or L from the Cholesky factorization $A = U^{\mathrm{T}}U$ or $A = LL^{\mathrm{T}}$, in the same storage format as A.

2: **b**(ldb, :) – REAL (KIND=nag_wp) array

The first dimension of the array **b** will be $\max(1, \mathbf{n})$.

The second dimension of the array **b** will be $\max(1, \mathbf{nrhs\_p})$.

If **ifail** $= 0$ or $\mathbf{n} + 1$, the n by r solution matrix X.

3: **rcond** – REAL (KIND=nag_wp)

If **ifail** $= 0$ or $\mathbf{n} + 1$, an estimate of the reciprocal of the condition number of the matrix A, computed as $\mathbf{rcond} = 1/\left(\|A\|_1 \|A^{-1}\|_1\right)$.

4: **errbnd** – REAL (KIND=nag_wp)

If **ifail** $= 0$ or $\mathbf{n} + 1$, an estimate of the forward error bound for a computed solution $\hat{x}$, such that $\|\hat{x} - x\|_1 / \|x\|_1 \leq \mathbf{errbnd}$, where $\hat{x}$ is a column of the computed solution returned in the array **b** and x is the corresponding column of the exact solution X. If **rcond** is less than ***machine precision***, then **errbnd** is returned as unity.

5: **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 > 0 and **ifail** $\leq$ **n**

The principal minor of order $\langle value \rangle$ of the matrix A is not positive definite. The factorization has not been completed and the solution could not be computed.

ifail $=$ **n** $+ 1$ (*warning*)

A solution has been computed, but **rcond** is less than ***machine precision*** so that the matrix A is numerically singular.

ifail $= -1$

On entry, **uplo** not one of `U' or `u' or `L' or `l'.

ifail $= -2$

Constraint: **n** ≥ 0.

ifail $= -3$

Constraint: **kd** ≥ 0.

ifail $= -4$

Constraint: **nrhs_p** ≥ 0.

ifail $= -6$

Constraint: $ldab \geq$ **kd** $+ 1$.

ifail $= -8$

Constraint: $ldb \geq \max(1,$ **n**$)$.

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.

The integer allocatable memory required is ***n****, and the double allocatable memory required is* $3 \times$ **n**. *Allocation failed before the solution could be computed.*

7 Accuracy

The computed solution for a single right-hand side, $\hat{x}$, satisfies an equation of the form

$$(A + E)\hat{x} = b,$$

where

$$\|E\|_1 = O(\epsilon)\|A\|_1$$

and ϵ is the ***machine precision***. An approximate error bound for the computed solution is given by

$$\frac{\|\hat{x}-x\|_1}{\|x\|_1} \leq \kappa(A)\frac{\|E\|_1}{\|A\|_1},$$

where $\kappa(A) = \|A^{-1}\|_1 \|A\|_1$, the condition number of A with respect to the solution of the linear equations. nag_linsys_real_posdef_band_solve (f04bf) uses the approximation $\|E\|_1 = \epsilon\|A\|_1$ to estimate **errbnd**. See Section 4.4 of Anderson *et al.* (1999) for further details.

8 Further Comments

The band storage scheme for the array **ab** is illustrated by the following example, when $n = 6$, $k = 2$, and **uplo** = 'U':

On entry:

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

On exit:

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

Similarly, if **uplo** = 'L' the format of **ab** is as follows:

On entry:

$$\begin{array}{cccccc} a_{11} & a_{22} & a_{33} & a_{44} & a_{55} & a_{66} \\ a_{21} & a_{32} & a_{43} & a_{54} & a_{65} & * \\ a_{31} & a_{42} & a_{53} & a_{64} & * & * \end{array}$$

On exit:

$$\begin{array}{cccccc} l_{11} & l_{22} & l_{33} & l_{44} & l_{55} & l_{66} \\ l_{21} & l_{32} & l_{43} & l_{54} & l_{65} & * \\ l_{31} & l_{42} & l_{53} & l_{64} & * & * \end{array}$$

Array elements marked $*$ need not be set and are not referenced by the function.

Assuming that $n \gg k$, the total number of floating-point operations required to solve the equations $AX = B$ is approximately $n(k+1)^2$ for the factorization and $4nkr$ for the solution following the factorization. The condition number estimation typically requires between four and five solves and never more than eleven solves, following the factorization.

In practice the condition number estimator is very reliable, but it can underestimate the true condition number; see Section 15.3 of Higham (2002) for further details.

The complex analogue of nag_linsys_real_posdef_band_solve (f04bf) is nag_linsys_complex_posdef_band_solve (f04cf).

9 Example

This example solves the equations

$$AX = B,$$

where A is the symmetric positive definite band matrix

$$A = \begin{pmatrix} 5.49 & 2.68 & 0 & 0 \\ 2.68 & 5.63 & -2.39 & 0 \\ 0 & -2.39 & 2.60 & -2.22 \\ 0 & 0 & -2.22 & 5.17 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 22.09 & 5.10 \\ 9.31 & 30.81 \\ -5.24 & -25.82 \\ 11.83 & 22.90 \end{pmatrix}.$$

An estimate of the condition number of A and an approximate error bound for the computed solutions are also printed.

9.1 Program Text

```
function f04bf_example

fprintf('f04bf example results\n\n');

% Solve Ax = b for symmetric banded A with error bound and condition number
uplo = 'U';
kd = nag_int(1);
ab = [ 0.00,  2.68, -2.39, -2.22;
       5.49,  5.63,  2.60,  5.17];
b = [ 22.09,  5.10;
       9.31, 30.81;
      -5.24,-25.82;
      11.83, 22.90];

[ab, x, rcond, errbnd, ifail] = ...
  f04bf(uplo, kd, ab, b);

disp('Solution');
disp(x);
disp('Estimate of condition number');
fprintf('%10.1f\n\n',1/rcond);
disp('Estimate of error bound for computed solutions');
fprintf('%10.1e\n\n',errbnd);
```

9.2 Program Results

```
f04bf example results

Solution
    5.0000   -2.0000
   -2.0000    6.0000
   -3.0000   -1.0000
    1.0000    4.0000

Estimate of condition number
      74.2

Estimate of error bound for computed solutions
   8.2e-15
```