NAG Toolbox for MATLAB

f04bj

1 Purpose

f04bj computes the solution to a real system of linear equations $AX = B$, where A is an n by n symmetric matrix, stored in packed format 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

```
[ap, ipiv, b, rcond, errbnd, ifail] = f04bj(uplo, ap, b, 'n', n,
'nrhs_p', nrhs_p)
```

3 Description

The diagonal pivoting method is used to factor A as $A = UDU^{\mathrm{T}}$, if **uplo** = 'U', or $A = LDL^{\mathrm{T}}$, if **uplo** = 'L', where U (or L) is a product of permutation and unit upper (lower) triangular matrices, and D is symmetric and block diagonal with 1 by 1 and 2 by 2 diagonal blocks. 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 URL: 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 – string**

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: **ap**($*$) **– double array**

Note: the dimension of the array **ap** must be at least $\max(1, \mathbf{n} \times (\mathbf{n} + 1)/2)$.

The n by n symmetric matrix A, packed column-wise in a linear array. The jth column of the matrix A is stored in the array **ap** as follows:

if **uplo** = 'U', $\mathbf{ap}(i + (j-1)j/2) = a_{ij}$ for $1 \leq i \leq j$
if **uplo** = 'L', $\mathbf{ap}(i + (j-1)(2n-j)/2) = a_{ij}$ for $j \leq i \leq n$

See Section 8 below for further details.

3: **b**(**ldb**,$*$) **– double array**

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

The second dimension of the array must be at least $\max(1, \mathbf{nrhs\_p})$. To solve the equations $Ax = b$, where b is a single right-hand side, **b** may be supplied as a one-dimensional array with length $\mathbf{ldb} = \max(1, \mathbf{n})$

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

5.2 Optional Input Parameters

1: **n – int32 scalar**

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

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

2: **nrhs_p – int32 scalar**

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 Input Parameters Omitted from the MATLAB Interface

ldb

5.4 Output Parameters

1: **ap**$(*)$ **– double array**

Note: the dimension of the array **ap** must be at least $\max(1, \mathbf{n} \times (\mathbf{n}+1)/2)$.

If **ifail** ≥ 0, the block diagonal matrix D and the multipliers used to obtain the factor U or L from the factorization $A = UDU^{\mathrm{T}}$ or $A = LDL^{\mathrm{T}}$ as computed by f07pd, stored as a packed triangular matrix in the same storage format as A.

2: **ipiv**$(*)$ **– int32 array**

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

If **ifail** ≥ 0, details of the interchanges and the block structure of D, as determined by f07pd.

If $\mathbf{ipiv}(k) > 0$, then rows and columns k and $\mathbf{ipiv}(k)$ were interchanged, and d_{kk} is a 1 by 1 diagonal block;
if **uplo** = 'U' and $\mathbf{ipiv}(k) = \mathbf{ipiv}(k-1) < 0$, then rows and columns $k-1$ and $-\mathbf{ipiv}(k)$ were interchanged and $d_{k-1:k,k-1:k}$ is a 2 by 2 diagonal block;
if **uplo** = 'L' and $\mathbf{ipiv}(k) = \mathbf{ipiv}(k+1) < 0$, then rows and columns $k+1$ and $-\mathbf{ipiv}(k)$ were interchanged and $d_{k:k+1,k:k+1}$ is a 2 by 2 diagonal block.

3: **b**$(\mathbf{ldb},*)$ **– double array**

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

The second dimension of the array must be at least $\max(1, \mathbf{nrhs\_p})$. To solve the equations $Ax = b$, where b is a single right-hand side, **b** may be supplied as a one-dimensional array with length $\mathbf{ldb} = \max(1, \mathbf{n})$

If **ifail** $= 0$ or $Np1$, the n by r solution matrix X.

4: **rcond – double scalar**

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

5: **errbnd – double scalar**

If **ifail** $= 0$ or $Np1$, 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.

6: **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 < 0 and **ifail** $\neq -999$

If **ifail** $= -i$, the ith argument had an illegal value.

ifail $= -999$

Allocation of memory failed. The integer allocatable memory required is **n**, and the double allocatable memory required is $2 \times$ **n**. Allocation failed before the solution could be computed.

ifail > 0 and **ifail** $\leq N$

If **ifail** $= i$, d_{ii} is exactly zero. The factorization has been completed, but the block diagonal matrix D is exactly singular, so the solution could not be computed.

ifail $= N + 1$

rcond is less than ***machine precision***, so that the matrix A is numerically singular. A solution to the equations $AX = B$ has nevertheless been 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. f04bj 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 packed storage scheme is illustrated by the following example when $n = 4$ and **uplo** = 'U'. Two-dimensional storage of the symmetric matrix A:

$$\begin{array}{cccc} a_{11} & a_{12} & a_{13} & a_{14} \\ & a_{22} & a_{23} & a_{24} \\ & & a_{33} & a_{34} \\ & & & a_{44} \end{array} \qquad (a_{ij} = a_{ji})$$

Packed storage of the upper triangle of A:

$$\mathbf{ap} = [a_{11}, \quad a_{12}, \quad a_{22}, \quad a_{13}, \quad a_{23}, \quad a_{33}, \quad a_{14}, \quad a_{24}, \quad a_{34}, \quad a_{44}]$$

The total number of floating-point operations required to solve the equations $AX = B$ is proportional to $\left(\frac{1}{3}n^3 + 2n^2r\right)$. 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 analogues of f04bj are f04cj for complex Hermitian matrices, and f04dj for complex symmetric matrices.

9 Example

```
uplo = 'U';
ap = [-1.81;
     2.06;
     1.15;
     0.63;
     1.87;
     -0.21;
     -1.15;
     4.2;
     3.87;
     2.07];
b = [0.96, 3.93;
     6.07, 19.25;
     8.380000000000001, 9.9;
     9.5, 27.85];
[apOut, ipiv, bOut, rcond, errbnd, ifail] = f04bj(uplo, ap, b)
```

```
apOut =
    0.4074
    0.3031
   -2.5907
   -0.5960
    0.8115
    1.1500
    0.6537
    0.2230
    4.2000
    2.0700
ipiv =
           1
           2
          -2
          -2
bOut =
   -5.0000    2.0000
   -2.0000    3.0000
    1.0000    4.0000
    4.0000    1.0000
rcond =
    0.0132
errbnd =
   8.4111e-15
ifail =
           0
```