NAG Toolbox for MATLAB

f07bg

1 Purpose

f07bg estimates the condition number of a real band matrix A, where A has been factorized by f07bd.

2 Syntax

```
[rcond, info] = f07bg(norm_p, kl, ku, ab, ipiv, anorm, 'n', n)
```

3 Description

f07bg estimates the condition number of a real band matrix A, in either the 1-norm or the ∞-norm:

$$\kappa_1(A) = \|A\|_1 \left\|A^{-1}\right\|_1 \qquad \text{or} \qquad \kappa_\infty(A) = \|A\|_\infty \left\|A^{-1}\right\|_\infty.$$

Note that $\kappa_\infty(A) = \kappa_1\left(A^{\mathrm{T}}\right)$.

Because the condition number is infinite if A is singular, the function actually returns an estimate of the **reciprocal** of the condition number.

4 References

Higham N J 1988 FORTRAN codes for estimating the one-norm of a real or complex matrix, with applications to condition estimation *ACM Trans. Math. Software* **14** 381–396

5 Parameters

5.1 Compulsory Input Parameters

1: **norm_p – string**

Indicates whether $\kappa_1(A)$ or $\kappa_\infty(A)$ is estimated.

norm_p = '1' or 'O'

$\kappa_1(A)$ is estimated.

norm_p = 'I'

$\kappa_\infty(A)$ is estimated.

Constraint: **norm_p** = '1', 'O' or 'I'.

2: **kl – int32 scalar**

k_l, the number of subdiagonals within the band of the matrix A.

Constraint: **kl** ≥ 0.

3: **ku – int32 scalar**

k_u, the number of superdiagonals within the band of the matrix A.

Constraint: **ku** ≥ 0.

4: **ab**(**ldab**,∗) – **double array**

The first dimension of the array **ab** must be at least $2 \times \mathbf{kl} + \mathbf{ku} + 1$

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

The LU factorization of A, as returned by f07bd.

5: **ipiv**(∗) – **int32 array**

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

The pivot indices, as returned by f07bd.

6: **anorm** – **double scalar**

If **norm_p** = '1' or 'O', the 1-norm of the **original** matrix A.

If **norm_p** = 'I', the ∞-norm of the **original** matrix A.

anorm must be computed either **before** calling f07bd or else from a **copy** of the original matrix A.

Constraint: **anorm** ≥ 0.0.

5.2 Optional Input Parameters

1: **n** – **int32 scalar**

Default: The second dimension of the array **ab**.

n, the order of the matrix A.

Constraint: **n** ≥ 0.

5.3 Input Parameters Omitted from the MATLAB Interface

ldab, work, iwork

5.4 Output Parameters

1: **rcond** – **double scalar**

An estimate of the reciprocal of the condition number of A. **rcond** is set to zero if exact singularity is detected or the estimate underflows. If **rcond** is less than ***machine precision***, A is singular to working precision.

2: **info** – **int32 scalar**

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

6 Error Indicators and Warnings

Errors or warnings detected by the function:

info $= -i$

If **info** $= -i$, parameter i had an illegal value on entry. The parameters are numbered as follows:

1: **norm_p**, 2: **n**, 3: **kl**, 4: **ku**, 5: **ab**, 6: **ldab**, 7: **ipiv**, 8: **anorm**, 9: **rcond**, 10: **work**, 11: **iwork**, 12: **info**.

It is possible that **info** refers to a parameter that is omitted from the MATLAB interface. This usually indicates that an error in one of the other input parameters has caused an incorrect value to be inferred.

7 Accuracy

The computed estimate **rcond** is never less than the true value ρ, and in practice is nearly always less than 10ρ, although examples can be constructed where **rcond** is much larger.

8 Further Comments

A call to f07bg involves solving a number of systems of linear equations of the form $Ax = b$ or $A^{\mathrm{T}}x = b$; the number is usually 4 or 5 and never more than 11. Each solution involves approximately $2n(2k_l + k_u)$ floating-point operations (assuming $n \gg k_l$ and $n \gg k_u$) but takes considerably longer than a call to f07be with one right-hand side, because extra care is taken to avoid overflow when A is approximately singular.

The complex analogue of this function is f07bu.

9 Example

```
norm_p = '1';
m = int32(4);
kl = int32(1);
ku = int32(2);
ab = [0, 0, 0, 0;
     0, 0, -3.66, -2.13;
     0, 2.54, -2.73, 4.07;
     -0.23, 2.46, 2.46, -3.82;
     -6.98, 2.56, -4.78, 0];
anorm = 13.63;
[ab, ipiv, info] = f07bd(m, kl, ku, ab);
[rcond, info] = f07bg(norm_p, kl, ku, ab, ipiv, anorm)
```

```
rcond =
    0.0177
info =
           0
```