NAG Fortran Library Routine Document

F07AFF (DGEEQU)

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

F07AFF (DGEEQU) computes diagonal scaling matrices D_R and D_C intended to equilibrate a real m by n matrix A and reduce its condition number.

2 Specification

```
SUBROUTINE F07AFF (M, N, A, LDA, R, C, ROWCND, COLCND, AMAX, INFO)
INTEGER             M, N, LDA, INFO
double precision    A(LDA,*), R(*), C(*), ROWCND, COLCND, AMAX
```

The routine may be called by its LAPACK name ***dgeequ***.

3 Description

F07AFF (DGEEQU) computes the diagonal scaling matrices. The diagonal scaling matrices are chosen to try to make the elements of largest absolute value in each row and column of the matrix B given by

$$B = D_R A D_C$$

have absolute value 1. The diagonal elements of D_R and D_C are restricted to lie in the safe range $(\delta, 1/\delta)$, where δ is the value returned by routine X02AMF. Use of these scaling factors is not guaranteed to reduce the condition number of A but works well in practice.

4 References

None.

5 Parameters

1: M – INTEGER *Input*

On entry: m, the number of rows of the matrix A.

Constraint: $\mathrm{M} \geq 0$.

2: N – INTEGER *Input*

On entry: n, the number of columns of the matrix A.

Constraint: $\mathrm{N} \geq 0$.

3: A(LDA,$*$) – ***double precision*** array *Input*

Note: the second dimension of the array A must be at least $\max(1, \mathrm{N})$.

On entry: the matrix A whose scaling factors are to be computed.

4: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F07AFF (DGEEQU) is called.

Constraint: $\mathrm{LDA} \geq \max(1, \mathrm{M})$.

5: R$(*)$ – ***double precision*** array *Output*

Note: the dimension of the array R must be at least $\max(1, \mathrm{M})$.

On exit: if INFO $= 0$ or INFO $>$ M, R contains the row scale factors, the diagonal elements of D_R. The elements of R will be positive.

6: C$(*)$ – ***double precision*** array *Output*

Note: the dimension of the array C must be at least $\max(1, \mathrm{N})$.

On exit: if INFO $= 0$, C contains the column scale factors, the diagonal elements of D_C. The elements of C will be positive.

7: ROWCND – ***double precision*** *Output*

On exit: if INFO $= 0$ or INFO $>$ M, ROWCND contains the ratio of the smallest value of R(i) to the largest value of R(i). If ROWCND ≥ 0.1 and AMAX is neither too large nor too small, it is not worth scaling by D_R.

8: COLCND – ***double precision*** *Output*

On exit: if INFO $= 0$, COLCND contains the ratio of the smallest value of C(i) to the largest value of C(i).

If COLCND ≥ 0.1, it is not worth scaling by D_C.

9: AMAX – ***double precision*** *Output*

On exit: $\max |a_{ij}|$. If AMAX is very close to overflow or underflow, the matrix A should be scaled.

10: INFO – INTEGER *Output*

On exit: INFO $= 0$ unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the routine:

INFO < 0

If INFO $= -i$, the ith argument had an illegal value. An explanatory message is output, and execution of the program is terminated.

INFO > 0 and INFO $\leq$ M

If INFO $= i$, the ith row of A is exactly zero.

INFO $>$ M

If INFO $= i$, the $(i - \mathrm{M})$th column of A is exactly zero.

7 Accuracy

The computed scale factors will be close to the exact scale factors.

8 Further Comments

The complex analogue of this routine is F07ATF (ZGEEQU).

9 Example

To equilibrate the general matrix A given by

$$A = \begin{pmatrix} 1.80 \times 10^{10} & 2.88 \times 10^{10} & 2.05 & -8.90 \times 10^{9} \\ 5.25 & -2.95 & -9.50 \times 10^{-9} & -3.80 \\ 1.58 & -2.69 & -2.90 \times 10^{-10} & -1.04 \\ -1.11 & -0.66 & -5.90 \times 10^{-11} & 0.80 \end{pmatrix}.$$

Details of the scaling factors, and the scaled matrix are output.

9.1 Program Text

```
*     F07AFF Example Program Text
*     Mark 21 Release. NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NMAX
      PARAMETER        (NMAX=8)
      INTEGER          LDA
      PARAMETER        (LDA=NMAX)
*     .. Local Scalars ..
      DOUBLE PRECISION AMAX, BIG, CJ, COLCND, ROWCND, SMALL
      INTEGER          I, IFAIL, INFO, J, N
*     .. Local Arrays ..
      DOUBLE PRECISION A(LDA,NMAX), C(NMAX), R(NMAX)
*     .. External Functions ..
      DOUBLE PRECISION X02AJF, X02AMF
      INTEGER          X02BHF
      EXTERNAL         X02AJF, X02AMF, X02BHF
*     .. External Subroutines ..
      EXTERNAL         DGEEQU, X04CAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F07AFF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      IF (N.LE.NMAX) THEN
*
*        Read the N by N matrix A from data file
*
         READ (NIN,*) ((A(I,J),J=1,N),I=1,N)
*
*        Print the matrix A
*
         IFAIL = 0
         CALL X04CAF('General',' ',N,N,A,LDA,'Matrix A',IFAIL)
         WRITE (NOUT,*)
*
*        Compute row and column scaling factors
*
         CALL DGEEQU(N,N,A,LDA,R,C,ROWCND,COLCND,AMAX,INFO)
*
         IF (INFO.GT.0) THEN
            IF (INFO.LE.N) THEN
               WRITE (NOUT,99999) 'Row ', INFO, ' of A is exactly zero'
            ELSE
               WRITE (NOUT,99999) 'Column ', INFO - N,
     +           ' of A is exactly zero'
            END IF
         ELSE
*
*           Print ROWCND, COLCND, AMAX and the scale factors
*
            WRITE (NOUT,99998) 'ROWCND = ', ROWCND, ', COLCND = ',
     +        COLCND, ', AMAX = ', AMAX
            WRITE (NOUT,*)
```

```
            WRITE (NOUT,*) 'Row scale factors'
            WRITE (NOUT,99997) (R(I),I=1,N)
            WRITE (NOUT,*)
            WRITE (NOUT,*) 'Column scale factors'
            WRITE (NOUT,99997) (C(I),I=1,N)
            WRITE (NOUT,*)
*
*           Compute values close to underflow and overflow
*
            SMALL = X02AMF()/(X02AJF()*X02BHF())
            BIG = 1.0D0/SMALL
            IF ((ROWCND.GE.0.1D0) .AND. (AMAX.GE.SMALL)
     +          .AND. (AMAX.LE.BIG)) THEN
               IF (COLCND.LT.0.1D0) THEN
*
*                 Just column scale A
*
                  DO 40 J = 1, N
                     CJ = C(J)
                     DO 20 I = 1, N
                        A(I,J) = A(I,J)*CJ
   20                CONTINUE
   40             CONTINUE
*
*                 Print the column scaled matrix
*
                  IFAIL = 0
                  CALL X04CAF('General',' ',N,N,A,LDA,'Scaled matrix',
     +                        IFAIL)
*
               END IF
            ELSE IF (COLCND.GE.0.1D0) THEN
*
*              Just row scale A
*
               DO 80 J = 1, N
                  DO 60 I = 1, N
                     A(I,J) = R(I)*A(I,J)
   60             CONTINUE
   80          CONTINUE
*
*              Print the row scaled matrix
*
               IFAIL = 0
               CALL X04CAF('General',' ',N,N,A,LDA,'Scaled matrix',
     +                     IFAIL)
*
            ELSE
*
*              Row and column scale A
*
               DO 120 J = 1, N
                  CJ = C(J)
                  DO 100 I = 1, N
                     A(I,J) = R(I)*A(I,J)*CJ
  100             CONTINUE
  120          CONTINUE
*
*              Print the row and column scaled matrix
*
               IFAIL = 0
               CALL X04CAF('General',' ',N,N,A,LDA,'Scaled matrix',
     +                     IFAIL)
*
            END IF
         END IF
      ELSE
         WRITE (NOUT,*) 'NMAX too small'
      END IF
      STOP
*
```

```
99999 FORMAT (1X,A,I4,A)
99998 FORMAT (1X,3(A,1P,E7.1))
99997 FORMAT ((1X,1P,7E11.1))
      END
```

9.2 Program Data

```
F07AFF Example Program Data

  4                                          :Value of N

  1.80D+10   2.88D+10   2.05D+00  -8.90D+09
  5.25D+00  -2.95D+00  -9.50D-09  -3.80D+00
  1.58D+00  -2.69D+00  -2.90D-10  -1.04D+00
 -1.11D+00  -6.60D-01  -5.90D-11   8.00D-01 :End of matrix A
```

9.3 Program Results

```
 F07AFF Example Program Results

 Matrix A
                1            2            3            4
 1    1.8000E+10   2.8800E+10   2.0500E+00  -8.9000E+09
 2    5.2500E+00  -2.9500E+00  -9.5000E-09  -3.8000E+00
 3    1.5800E+00  -2.6900E+00  -2.9000E-10  -1.0400E+00
 4   -1.1100E+00  -6.6000E-01  -5.9000E-11   8.0000E-01

 ROWCND = 3.9E-11, COLCND = 1.8E-09, AMAX = 2.9E+10

 Row scale factors
     3.5E-11    1.9E-01    3.7E-01    9.0E-01

 Column scale factors
     1.0E+00    1.0E+00    5.5E+08    1.4E+00

 Scaled matrix
          1       2       3       4
 1   0.6250  1.0000  0.0393 -0.4269
 2   1.0000 -0.5619 -1.0000 -1.0000
 3   0.5874 -1.0000 -0.0596 -0.5341
 4  -1.0000 -0.5946 -0.0294  0.9957
```