NAG Fortran Library Routine Document

F01BLF

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

F01BLF calculates the rank and pseudo-inverse of an m by n real matrix, $m \geq n$, using a QR factorization with column interchanges.

2 Specification

```
      SUBROUTINE F01BLF(M, N, T, A, IA, AIJMAX, IRANK, INC, D, U, IU, DU,
     1                  IFAIL)
      INTEGER           M, N, IA, IRANK, INC(N), IU, IFAIL
      real              T, A(IA,N), AIJMAX(N), D(M), U(IU,N), DU(N)
```

3 Description

Householder's factorization with column interchanges is used in the decomposition $F = QU$, where F is A with its columns permuted, Q is the first r columns of an m by m orthogonal matrix and U is an r by n upper-trapezoidal matrix of rank r. The pseudo-inverse of F is given by X where

$$X = U^T(UU^T)^{-1}Q^T.$$

If the matrix is found to be of maximum rank, $r = n$, U is a non-singular n by n upper-triangular matrix and the pseudo-inverse of F simplifies to $X = U^{-1}Q^T$. The transpose of the pseudo-inverse of A is overwritten on A.

4 References

Peters G and Wilkinson J H (1970) The least-squares problem and pseudo-inverses *Comput. J.* **13** 309–316

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

5 Parameters

1: M – INTEGER *Input*

2: N – INTEGER *Input*

On entry: m and n, the number of rows and columns in the matrix A.

Constraint: M $\geq$ N.

3: T – ***real*** *Input*

On entry: the tolerance used to decide when elements can be regarded as zero (see Section 8).

4: A(IA,N) – ***real*** array *Input/Output*

On entry: the m by n rectangular matrix A.

On exit: the transpose of the pseudo-inverse of A.

5: IA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F01BLF is called.

Constraint: IA $\geq$ M.

6: AIJMAX(N) – ***real*** array *Output*

On exit: AIJMAX(i) contains the element of largest modulus in the reduced matrix at the ith stage. If $r < n$, then only the first $r + 1$ elements of AIJMAX have values assigned to them; the remaining elements are unused. The ratio AIJMAX$(1)/$AIJMAX(r) usually gives an indication of the condition number of the original matrix (see Section 8).

7: IRANK – INTEGER *Output*

On exit: r, the rank of A as determined using the tolerance T.

8: INC(N) – INTEGER array *Output*

On exit: the record of the column interchanges in the Householder factorization.

9: D(M) – ***real*** array *Workspace*

10: U(IU,N) – ***real*** array *Workspace*

11: IU – INTEGER *Input*

On entry: the first dimension of the array U as declared in the (sub)program from which F01BLF is called.

Constraint: IU $\geq$ N.

12: DU(N) – ***real*** array *Workspace*

13: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

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

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, for users not familiar with this parameter the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

6 Error Indicators and Warnings

If on entry IFAIL $= 0$ or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL $= 1$

Inverse not found, due to an incorrect determination of IRANK (see Section 8).

IFAIL $= 2$

Invalid tolerance, due to

(i) T is negative, IRANK $= -1$;

(ii) T too large, IRANK $= 0$;

(iii) T too small, IRANK > 0.

IFAIL = 3

On entry, M < N.

7 Accuracy

For most matrices the pseudo-inverse is the best possible having regard to the condition of A and the choice of T. Note that only the singular value decomposition method can be relied upon to give maximum accuracy for the precision of computation used and correct determination of the condition of a matrix (see Wilkinson and Reinsch (1971)).

The computed factors Q and U satisfy the relation $QU = F + E$ where

$$\|E\|_2 < c\epsilon\|A\|_2 + \eta\sqrt{(m-r)(n-r)}$$

in which c is a modest function of m and n, η is the value of T, and ϵ is the ***machine precision***.

8 Further Comments

The time taken by the routine is approximately proportional to mnr.

The most difficult practical problem is the determination of the rank of the matrix (see pages 314–315 of Peters and Wilkinson (1970)); only the singular value decomposition method gives a reliable indication of rank deficiency (see pages 134–151 of Wilkinson and Reinsch (1971) and F02WEF). In F01BLF a tolerance, T, is used to recognise 'zero' elements in the remaining matrix at each step in the factorization. The value of T should be set at n times the bound on possible errors in individual elements of the original matrix. If the elements of A vary widely in their orders of magnitude, of course this presents severe difficulties. Sound decisions can only be made by somebody who appreciates the underlying physical problem.

If the condition number of A is 10^p we expect to get p figures wrong in the pseudo-inverse. An estimate of the condition number is usually given by AIJMAX(1)/AIJMAX(r).

9 Example

A complete program follows which outputs the maximum of the moduli of the 'remaining' elements at each step in the factorization, the rank, as determined by the given value of T, and the transposed pseudo-inverse. Data and results are given for an example which is a 6 by 5 matrix of deficient rank in which the last column is a linear combination of the other four. Using T = 119ϵ (119 is the norm of the matrix) the rank is correctly determined as 4 and the pseudo-inverse is computed to full implementation accuracy.

9.1 Program Text

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     F01BLF Example Program Text
*     Mark 14 Revised.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          MMAX, NMAX, IA, IU
      PARAMETER        (MMAX=6,NMAX=MMAX,IA=MMAX,IU=NMAX)
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      real             CXIX, T
      INTEGER          I, IFAIL, IRANK, J, M, N
*     .. Local Arrays ..
      real             A(IA,NMAX), AIJMAX(NMAX), D(MMAX), DU(NMAX),
     +                 U(IU,NMAX)
      INTEGER          INC(NMAX)
*     .. External Functions ..
```

```
      real              X02AJF
      EXTERNAL          X02AJF
*     .. External Subroutines ..
      EXTERNAL          F01BLF
*     .. Intrinsic Functions ..
      INTRINSIC         MIN, SQRT
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F01BLF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) M, N
      WRITE (NOUT,*)
      IF (M.GT.0 .AND. M.LE.MMAX .AND. N.GT.0 .AND. N.LE.M) THEN
         READ (NIN,*) ((A(I,J),J=1,N),I=1,M)
*        Set T to N times norm of A.
         CXIX = 0.0e0
         DO 40 I = 1, M
            DO 20 J = 1, N
               CXIX = CXIX + A(I,J)**2
   20       CONTINUE
   40    CONTINUE
         T = SQRT(CXIX)*X02AJF()
         IFAIL = 1
*
         CALL F01BLF(M,N,T,A,IA,AIJMAX,IRANK,INC,D,U,IU,DU,IFAIL)
*
         IF (IFAIL.NE.0) THEN
            WRITE (NOUT,99998) 'Error in F01BLF. IFAIL =', IFAIL
         ELSE
            WRITE (NOUT,*)
     +        'Maximum element in A(K) for I.GE.K and J.GE.K'
            WRITE (NOUT,*)
            WRITE (NOUT,*) '   K    Modulus'
            WRITE (NOUT,99997) (I,AIJMAX(I),I=1,MIN(N,IRANK+1))
            WRITE (NOUT,*)
            WRITE (NOUT,99998) 'Rank = ', IRANK
            WRITE (NOUT,*)
            WRITE (NOUT,99995) 'T = ', T, '  (machine dependent)'
            WRITE (NOUT,*)
            WRITE (NOUT,*) 'Transpose of pseudo-inverse'
            DO 60 I = 1, M
               WRITE (NOUT,99996) (A(I,J),J=1,N)
   60       CONTINUE
         END IF
      ELSE
         WRITE (NOUT,99999) 'M or N out of range: M = ', M, '   N = ', N
      END IF
      STOP
*
99999 FORMAT (1X,A,I5,A,I5)
99998 FORMAT (1X,A,I2)
99997 FORMAT (1X,I4,2X,1P,e12.4)
99996 FORMAT (1X,1P,6e12.4)
99995 FORMAT (1X,A,1P,e11.4,A)
      END
```

9.2 Program Data

```
F01BLF Example Program Data
  6  5
   7.0   -2.0    4.0    9.0    1.8
   3.0    8.0   -4.0    6.0    1.3
   9.0    6.0    1.0    5.0    2.1
  -8.0    7.0    5.0    2.0    0.6
   4.0   -1.0    2.0    8.0    1.3
   1.0    6.0    3.0   -5.0    0.5
```

9.3 Program Results

```
F01BLF Example Program Results

Maximum element in A(K) for I.GE.K and J.GE.K

   K    Modulus
   1    9.0000E+00
   2    9.3101E+00
   3    8.7461E+00
   4    5.6832E+00
   5    3.9064E-16

Rank =  4

T =  2.9977E-15  (machine dependent)

Transpose of pseudo-inverse
  1.7807E-02 -2.1565E-02  5.2029E-02  2.3686E-02  7.1957E-03
 -1.1826E-02  4.3417E-02 -8.1265E-02  3.5717E-02 -1.3957E-03
  4.7157E-02  2.9446E-02  1.3926E-02 -1.3808E-02  7.6720E-03
 -5.6636E-02  2.9132E-02  4.7442E-02  3.0478E-02  5.0415E-03
 -3.6741E-03 -1.3781E-02  1.6647E-02  3.5665E-02  3.4857E-03
  3.8408E-02  3.4256E-02  5.7594E-02 -5.7134E-02  7.3123E-03
```