

F01ABF – NAG Fortran Library Routine Document

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

F01ABF calculates the accurate inverse of a real symmetric positive-definite matrix, using a Cholesky factorization and iterative refinement.

2 Specification

```
SUBROUTINE F01ABF(A, IA, N, B, IB, Z, IFAIL)
  INTEGER          IA, N, IB, IFAIL
  real           A(IA,N), B(IB,N), Z(N)
```

3 Description

To compute the inverse X of a real symmetric positive-definite matrix A , this routine first computes a Cholesky factorization of A as $A = LL^T$, where L is lower triangular. An approximation to X is found by computing L^{-1} and then the product $(L^{-1})^T L^{-1}$. The residual matrix $R = I - AX$ is calculated using *additional precision*, and a correction D to X is found by solving $LL^T D = R$. X is replaced by $X + D$, and this iterative refinement of the inverse is repeated until full machine accuracy has been obtained.

4 References

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

5 Parameters

- 1:** A(IA,N) — *real* array *Input/Output*
On entry: the upper triangle of the n by n positive-definite symmetric matrix A . The elements of the array below the diagonal need not be set.
On exit: the lower triangle of the inverse matrix X is stored in the elements of the array below the diagonal, in rows 2 to $n + 1$; x_{ij} is stored in $A(i + 1, j)$ for $i \geq j$. The upper triangle of the original matrix is unchanged.
- 2:** IA — INTEGER *Input*
On entry: the first dimension of the array A as declared in the (sub)program from which F01ABF is called.
Constraint: $IA \geq N + 1$.
- 3:** N — INTEGER *Input*
On entry: n , the order of the matrix A .
Constraint: $N \geq 1$.
- 4:** B(IB,N) — *real* array *Output*
On exit: the lower triangle of the inverse matrix X , with x_{ij} stored in $B(i, j)$, for $i \geq j$.

- 5:** IB — INTEGER *Input*
On entry: the first dimension of the array B as declared in the (sub)program from which F01ABF is called.
Constraint: $IB \geq N$.
- 6:** Z(N) — *real* array *Workspace*
7: IFAIL — INTEGER *Input/Output*
On entry: IFAIL must be set to 0, -1 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.
On exit: IFAIL = 0 unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

Errors detected by the routine:

IFAIL =1

The matrix A is not positive-definite, possibly due to rounding errors.

IFAIL =2

The refinement process fails to converge, i.e., the matrix A is ill-conditioned.

7 Accuracy

The computed inverse should be correct to full machine accuracy. For a detailed error analysis see Wilkinson and Reinsch [1] page 40.

8 Further Comments

The time taken by the routine is approximately proportional to n^3 .

9 Example

To find the inverse of the 4 by 4 matrix:

$$\begin{pmatrix} 5 & 7 & 6 & 5 \\ 7 & 10 & 8 & 7 \\ 6 & 8 & 10 & 9 \\ 5 & 7 & 9 & 10 \end{pmatrix}.$$

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.

```
*   F01ABF Example Program Text
*   Mark 14 Revised.  NAG Copyright 1989.
*   .. Parameters ..
      INTEGER          NMAX, IA, IB
      PARAMETER        (NMAX=8, IA=NMAX+1, IB=NMAX)
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5, NOUT=6)
*   .. Local Scalars ..
      INTEGER          I, IFAIL, J, N
*   .. Local Arrays ..
```

```

      real          A(IA,NMAX), B(IB,NMAX), Z(NMAX)
*    .. External Subroutines ..
      EXTERNAL      F01ABF
*    .. Executable Statements ..
      WRITE (NOUT,*) 'F01ABF Example Program Results'
*    Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      WRITE (NOUT,*)
      IF (N.GT.0 .AND. N.LE.NMAX) THEN
        READ (NIN,*) ((A(I,J),J=1,N),I=1,N)
        IFAIL = 1
*
        CALL F01ABF(A,IA,N,B,IB,Z,IFAIL)
*
        IF (IFAIL.NE.0) THEN
          WRITE (NOUT,99999) 'Error in F01ABF. IFAIL =', IFAIL
        ELSE
          WRITE (NOUT,*) 'Lower triangle of inverse'
          DO 20 I = 1, N
            WRITE (NOUT,99998) (B(I,J),J=1,I)
20          CONTINUE
          END IF
        ELSE
          WRITE (NOUT,99999) 'N is out of range: N = ', N
        END IF
      STOP
*
99999 FORMAT (1X,A,I5)
99998 FORMAT (1X,8F9.4)
      END

```

9.2 Program Data

F01ABF Example Program Data

```

4
5.  7.  6.  5.
7. 10.  8.  7.
6.  8. 10.  9.
5.  7.  9. 10.

```

9.3 Program Results

F01ABF Example Program Results

Lower triangle of inverse

```

68.0000
-41.0000 25.0000
-17.0000 10.0000  5.0000
10.0000 -6.0000 -3.0000  2.0000

```
