NAG Library Routine Document

F08KAF (DGELSS)

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

F08KAF (DGELSS) computes the minimum norm solution to a real linear least squares problem

$$\min_x \|b - Ax\|_2.$$

2 Specification

```
SUBROUTINE F08KAF (M, N, NRHS, A, LDA, B, LDB, S, RCOND, RANK, WORK,       &
                   LWORK, INFO)
INTEGER            M, N, NRHS, LDA, LDB, RANK, LWORK, INFO
REAL (KIND=nag_wp) A(LDA,*), B(LDB,*), S(*), RCOND, WORK(max(1,LWORK))
```

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

3 Description

F08KAF (DGELSS) uses the singular value decomposition (SVD) of A, where A is an m by n matrix which may be rank-deficient.

Several right-hand side vectors b and solution vectors x can be handled in a single call; they are stored as the columns of the m by r right-hand side matrix B and the n by r solution matrix X.

The effective rank of A is determined by treating as zero those singular values which are less than RCOND times the largest singular value.

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 http://www.netlib.org/lapack/lug

Golub G H and Van Loan C F (1996) *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

1: M – INTEGER *Input*

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

Constraint: $M \geq 0$.

2: N – INTEGER *Input*

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

Constraint: $N \geq 0$.

3: NRHS – INTEGER *Input*

On entry: r, the number of right-hand sides, i.e., the number of columns of the matrices B and X.

Constraint: $NRHS \geq 0$.

4: A(LDA, $*$) – REAL (KIND=nag_wp) array *Input/Output*

Note: the second dimension of the array A must be at least max(1, N).

On entry: the m by n matrix A.

On exit: the first min(m, n) rows of A are overwritten with its right singular vectors, stored row-wise.

5: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08KAF (DGELSS) is called.

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

6: B(LDB, $*$) – REAL (KIND=nag_wp) array *Input/Output*

Note: the second dimension of the array B must be at least max(1, NRHS).

On entry: the m by r right-hand side matrix B.

On exit: B is overwritten by the n by r solution matrix X. If $m \geq n$ and RANK $= n$, the residual sum of squares for the solution in the ith column is given by the sum of squares of elements $n + 1, \dots, m$ in that column.

7: LDB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F08KAF (DGELSS) is called.

Constraint: LDB $\geq$ max(1, M, N).

8: S($*$) – REAL (KIND=nag_wp) array *Output*

Note: the dimension of the array S must be at least max(1, min(M, N)).

On exit: the singular values of A in decreasing order.

9: RCOND – REAL (KIND=nag_wp) *Input*

On entry: used to determine the effective rank of A. Singular values S(i) $\leq$ RCOND $\times$ S(1) are treated as zero. If RCOND < 0, ***machine precision*** is used instead.

10: RANK – INTEGER *Output*

On exit: the effective rank of A, i.e., the number of singular values which are greater than RCOND $\times$ S(1).

11: WORK(max(1, LWORK)) – REAL (KIND=nag_wp) array *Workspace*

On exit: if INFO $= 0$, WORK(1) contains the minimum value of LWORK required for optimal performance.

12: LWORK – INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which F08KAF (DGELSS) is called.

If LWORK $= -1$, a workspace query is assumed; the routine only calculates the optimal size of the WORK array, returns this value as the first entry of the WORK array, and no error message related to LWORK is issued.

Suggested value: for optimal performance, LWORK should generally be larger. Consider increasing LWORK by at least $nb \times \min(\mathrm{M},\mathrm{N})$, where nb is the optimal ***block size***.

Constraint: LWORK ≥ 1, and also
LWORK $\geq 3 \times \min(\mathrm{M},\mathrm{N}) + \max(2 \times \min(\mathrm{M},\mathrm{N}), \max(\mathrm{M},\mathrm{N}), \mathrm{NRHS})$.

13: INFO – INTEGER *Output*

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

6 Error Indicators and Warnings

INFO < 0

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

INFO > 0

The algorithm for computing the SVD failed to converge; if INFO $= i$, i off-diagonal elements of an intermediate bidiagonal form did not converge to zero.

7 Accuracy

See Section 4.5 of Anderson *et al.* (1999) for details.

8 Parallelism and Performance

F08KAF (DGELSS) is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

F08KAF (DGELSS) makes calls to BLAS and/or LAPACK routines, which may be threaded within the vendor library used by this implementation. Consult the documentation for the vendor library for further information.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

9 Further Comments

The complex analogue of this routine is F08KNF (ZGELSS).

10 Example

This example solves the linear least squares problem

$$\min_x \|b - Ax\|_2$$

for the solution, x, of minimum norm, where

$$A = \begin{pmatrix} -0.09 & 0.14 & -0.46 & 0.68 & 1.29 \\ -1.56 & 0.20 & 0.29 & 1.09 & 0.51 \\ -1.48 & -0.43 & 0.89 & -0.71 & -0.96 \\ -1.09 & 0.84 & 0.77 & 2.11 & -1.27 \\ 0.08 & 0.55 & -1.13 & 0.14 & 1.74 \\ -1.59 & -0.72 & 1.06 & 1.24 & 0.34 \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} 7.4 \\ 4.2 \\ -8.3 \\ 1.8 \\ 8.6 \\ 2.1 \end{pmatrix}.$$

A tolerance of 0.01 is used to determine the effective rank of A.

Note that the block size (NB) of 64 assumed in this example is not realistic for such a small problem, but should be suitable for large problems.

10.1 Program Text

```
    Program f08kafe

!     F08KAF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: dgelss, dnrm2, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter              :: nb = 64, nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)              :: rcond, rnorm
      Integer                         :: i, info, lda, lwork, m, n, rank
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: a(:,:), b(:), s(:), work(:)
!     .. Executable Statements ..
      Write (nout,*) 'F08KAF Example Program Results'
      Write (nout,*)
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) m, n
      lda = m
      lwork = 3*n + nb*(m+n)
      Allocate (a(lda,n),b(m),s(n),work(lwork))

!     Read A and B from data file

      Read (nin,*)(a(i,1:n),i=1,m)
      Read (nin,*) b(1:m)

!     Choose RCOND to reflect the relative accuracy of the input data

      rcond = 0.01_nag_wp

!     Solve the least squares problem min( norm2(b - Ax) ) for the x
!     of minimum norm.

!     The NAG name equivalent of dgelss is f08kaf
      Call dgelss(m,n,1,a,lda,b,m,s,rcond,rank,work,lwork,info)

      If (info==0) Then

!       Print solution

        Write (nout,*) 'Least squares solution'
        Write (nout,99999) b(1:n)

!       Print the effective rank of A

        Write (nout,*)
        Write (nout,*) 'Tolerance used to estimate the rank of A'
        Write (nout,99998) rcond
        Write (nout,*) 'Estimated rank of A'
        Write (nout,99997) rank

!       Print singular values of A

        Write (nout,*)
        Write (nout,*) 'Singular values of A'
        Write (nout,99999) s(1:n)

!       Compute and print estimate of the square root of the
!       residual sum of squares

        If (rank==n) Then
!         The NAG name equivalent of dnrm2 is f06ejf
          rnorm = dnrm2(m-n,b(n+1),1)
```

```
          Write (nout,*)
          Write (nout,*) 'Square root of the residual sum of squares'
          Write (nout,99998) rnorm
        End If
      Else
        Write (nout,*) 'The SVD algorithm failed to converge'
      End If

99999 Format (1X,7F11.4)
99998 Format (3X,1P,E11.2)
99997 Format (1X,I6)
    End Program f08kafe
```

10.2 Program Data

```
F08KAF Example Program Data

  6      5                        :Values of M and N

 -0.09   0.14  -0.46   0.68   1.29
 -1.56   0.20   0.29   1.09   0.51
 -1.48  -0.43   0.89  -0.71  -0.96
 -1.09   0.84   0.77   2.11  -1.27
  0.08   0.55  -1.13   0.14   1.74
 -1.59  -0.72   1.06   1.24   0.34 :End of matrix A

  7.4
  4.2
 -8.3
  1.8
  8.6
  2.1                             :End of vector b
```

10.3 Program Results

```
 F08KAF Example Program Results

 Least squares solution
      0.6344     0.9699    -1.4403     3.3678     3.3992

 Tolerance used to estimate the rank of A
      1.00E-02
 Estimated rank of A
      4

 Singular values of A
      3.9997     2.9962     2.0001     0.9988     0.0025
```