NAG Fortran Library Routine Document

F08AAF (DGELS)

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

F08AAF (DGELS) solves linear least-squares problems of the form

$$\min_x \|b - Ax\|_2 \quad \text{or} \quad \min_x \|b - A^T x\|_2,$$

where A is an m by n matrix of full rank, using a QR or LQ factorization of A.

2 Specification

```
SUBROUTINE F08AAF (TRANS, M, N, NRHS, A, LDA, B, LDB, WORK, LWORK, INFO)
INTEGER            M, N, NRHS, LDA, LDB, LWORK, INFO
double precision   A(LDA,*), B(LDB,*), WORK(*)
CHARACTER*1        TRANS
```

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

3 Description

The following options are provided:

1. If TRANS = 'N' and $m \geq n$: find the least-squares solution of an overdetermined system, i.e., solve the least-squares problem

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

2. If TRANS = 'N' and $m < n$: find the minimum norm solution of an underdetermined system $Ax = b$.
3. If TRANS = 'T' and $m \geq n$: find the minimum norm solution of an undetermined system $A^T x = b$.
4. If TRANS = 'T' and $m < n$: find the least-squares solution of an overdetermined system, i.e., solve the least-squares problem

$$\min_x \|b - A^T x\|_2.$$

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.

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 URL: 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: TRANS – CHARACTER*1 *Input*

On entry: if TRANS = 'N', the linear system involves A; if TRANS = 'T', the linear system involves A^T.

Constraint: TRANS = 'N' or 'T'.

2: M – INTEGER *Input*

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

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

3: N – INTEGER *Input*

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

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

4: 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: $\mathrm{NRHS} \geq 0$.

5: A(LDA,$*$) – ***double precision*** array *Input/Output*

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

On entry: the m by n matrix A.

On exit: if $\mathrm{M} \geq \mathrm{N}$, A is overwritten by details of its QR factorisation as returned by F08AEF (DGEQRF); if $\mathrm{M} < \mathrm{N}$, A is overwritten by details of its LQ factorisation as returned by F08AHF (DGELQF).

6: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08AAF (DGELS) is called.

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

7: B(LDB,$*$) – ***double precision*** array *Input/Output*

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

On entry: the matrix B of right-hand side vectors, stored columnwise; B is m by r if TRANS = 'N', or n by r if TRANS = 'T'.

On exit: is overwritten by the solution vectors, z, stored columnwise:

if TRANS = 'N' and $m \geq n$, rows 1 to n of B contain the least squares solution vectors; the residual sum of squares for the solution in each column is given by the sum of squares of the modulus of elements $(n + 1)$ to m in that column;
if TRANS = 'N' and $m < n$, rows 1 to n of B contain the minimum norm solution vectors;
if TRANS = 'T' and $m \geq n$, rows 1 to m of B contain the minimum norm solution vectors;
if TRANS = 'T' and $m < n$, rows 1 to m of B contain the least-squares solution vectors; the residual sum of squares for the solution in each column is given by the sum of squares of elements $(m + 1)$ to n in that column.

8: LDB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F08AAF (DGELS) is called.

Constraint: $\mathrm{LDB} \geq \mathrm{MAX}(1, \mathrm{M}, \mathrm{N})$.

9: WORK($*$) – ***double precision*** array *Workspace*

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

On exit: if INFO $= 0$, WORK(1) returns the optimal LWORK.

10: LWORK – INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which F08AAF (DGELS) is called.

For optimal performance, $\mathrm{LWORK} \geq \min(\mathrm{M}, \mathrm{N}) + \max(1, \mathrm{M}, \mathrm{N}, \mathrm{NRHS}) \times nb$, where nb is the optimal block size.

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.

Constraint: $\mathrm{LWORK} \geq \min(\mathrm{M}, \mathrm{N}) + \max(1, \mathrm{M}, \mathrm{N}, \mathrm{NRHS})$.

11: 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.

7 Accuracy

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

8 Further Comments

The total number of floating point operations required to factorize A is approximately $\frac{2}{3}n^2(3m - n)$ if $m \geq n$ and $\frac{2}{3}m^2(3n - m)$ otherwise. Following the factorization the solution for a single vector x requires $O\left(\min\left(m^2, n^2\right)\right)$ operations.

The complex analogue of this routine is F08ANF (ZGELS).

9 Example

To solve the linear least squares problem

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

where

$$A = \begin{pmatrix} -0.57 & -1.28 & -0.39 & 0.25 \\ -1.93 & 1.08 & -0.31 & -2.14 \\ 2.30 & 0.24 & -0.40 & -0.35 \\ -1.93 & 0.64 & -0.66 & 0.08 \\ 0.15 & 0.30 & 0.15 & -2.13 \\ -0.02 & 1.03 & -1.43 & 0.50 \end{pmatrix}$$

and

$$b = \begin{pmatrix} -2.67 \\ -0.55 \\ 3.34 \\ -0.77 \\ 0.48 \\ 4.10 \end{pmatrix}.$$

The square root of the residual sum of squares is also output.

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.

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.

```
*     F08AAF Example Program Text
*     Mark 21 Release. NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          MMAX, NB, NMAX
      PARAMETER        (MMAX=16,NB=64,NMAX=8)
      INTEGER          LDA, LWORK
      PARAMETER        (LDA=MMAX,LWORK=NMAX+NB*MMAX)
*     .. Local Scalars ..
      DOUBLE PRECISION RNORM
      INTEGER          I, INFO, J, M, N
*     .. Local Arrays ..
      DOUBLE PRECISION A(LDA,NMAX), B(MMAX), WORK(LWORK)
*     .. External Functions ..
      DOUBLE PRECISION DNRM2
      EXTERNAL         DNRM2
*     .. External Subroutines ..
      EXTERNAL         DGELS
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F08AAF Example Program Results'
      WRITE (NOUT,*)
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) M, N
      IF (M.LE.MMAX .AND. N.LE.NMAX .AND. M.GE.N) THEN
*
*        Read A and B from data file
*
         READ (NIN,*) ((A(I,J),J=1,N),I=1,M)
         READ (NIN,*) (B(I),I=1,M)
*
*        Solve the least squares problem min( norm2(b - Ax) ) for x
*
         CALL DGELS('No transpose',M,N,1,A,LDA,B,M,WORK,LWORK,INFO)
*
*        Print solution
*
         WRITE (NOUT,*) 'Least squares solution'
         WRITE (NOUT,99999) (B(I),I=1,N)
*
*        Compute and print estimate of the square root of the residual
*        sum of squares
*
         RNORM = DNRM2(M-N,B(N+1),1)
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'Square root of the residual sum of squares'
         WRITE (NOUT,99998) RNORM
      ELSE
         WRITE (NOUT,*) 'MMAX and/or NMAX too small, and/or M.LT.N'
```

```
      END IF
      STOP
*
99999 FORMAT (1X,7F11.4)
99998 FORMAT (3X,1P,E11.2)
      END
```

9.2 Program Data

```
F08AAF Example Program Data

  6      4                  :Values of M and N

 -0.57  -1.28  -0.39   0.25
 -1.93   1.08  -0.31  -2.14
  2.30   0.24   0.40  -0.35
 -1.93   0.64  -0.66   0.08
  0.15   0.30   0.15  -2.13
 -0.02   1.03  -1.43   0.50 :End of matrix A

 -2.67
 -0.55
  3.34
 -0.77
  0.48
  4.10                      :End of vector b
```

9.3 Program Results

```
 F08AAF Example Program Results

 Least squares solution
      1.5339     1.8707    -1.5241     0.0392

 Square root of the residual sum of squares
      2.22E-02
```