NAG Toolbox

nag_lapack_dgelsd (f08kc)

1 Purpose

nag_lapack_dgelsd (f08kc) computes the minimum norm solution to a real linear least squares problem

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

2 Syntax

```
[a, b, s, rank, info] = nag_lapack_dgelsd(a, b, rcond, 'm', m, 'n', n, 'nrhs_p',
nrhs_p)
```

```
[a, b, s, rank, info] = f08kc(a, b, rcond, 'm', m, 'n', n, 'nrhs_p', nrhs_p)
```

3 Description

nag_lapack_dgelsd (f08kc) uses the singular value decomposition (SVD) of A, where A is a real 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 problem is solved in three steps:

1. reduce the coefficient matrix A to bidiagonal form with Householder transformations, reducing the original problem into a 'bidiagonal least squares problem' (BLS);
2. solve the BLS using a divide-and-conquer approach;
3. apply back all the Householder transformations to solve the original least squares problem.

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

5.1 Compulsory Input Parameters

1: **a**(lda, :) – REAL (KIND=nag_wp) array

The first dimension of the array **a** must be at least max(1, **m**).

The second dimension of the array **a** must be at least max(1, **n**).

The m by n coefficient matrix A.

2: **b**(ldb, :) – REAL (KIND=nag_wp) array

The first dimension of the array **b** must be at least max(1, **m**, **n**).

The second dimension of the array **b** must be at least $\max(1, \mathbf{nrhs\_p})$.

The m by r right-hand side matrix B.

3: **rcond** – REAL (KIND=nag_wp)

Used to determine the effective rank of A. Singular values $\mathbf{s}(i) \leq \mathbf{rcond} \times \mathbf{s}(1)$ are treated as zero. If $\mathbf{rcond} < 0$, ***machine precision*** is used instead.

5.2 Optional Input Parameters

1: **m** – INTEGER

Default: the first dimension of the array **a**.

m, the number of rows of the matrix A.

Constraint: $\mathbf{m} \geq 0$.

2: **n** – INTEGER

Default: the second dimension of the array **a**.

n, the number of columns of the matrix A.

Constraint: $\mathbf{n} \geq 0$.

3: **nrhs_p** – INTEGER

Default: the second dimension of the array **b**.

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

Constraint: $\mathbf{nrhs\_p} \geq 0$.

5.3 Output Parameters

1: **a**(lda, :) – REAL (KIND=nag_wp) array

The first dimension of the array **a** will be $\max(1, \mathbf{m})$.

The second dimension of the array **a** will be $\max(1, \mathbf{n})$.

The contents of **a** are destroyed.

2: **b**(ldb, :) – REAL (KIND=nag_wp) array

The first dimension of the array **b** will be $\max(1, \mathbf{m}, \mathbf{n})$.

The second dimension of the array **b** will be $\max(1, \mathbf{nrhs\_p})$.

b stores the n by r solution matrix X. If $m \geq n$ and $\mathbf{rank} = n$, the residual sum of squares for the solution in the ith column is given by the sum of squares of elements $n + 1, \ldots, m$ in that column.

3: **s**(:) – REAL (KIND=nag_wp) array

The dimension of the array **s** will be $\max(1, \min(\mathbf{m}, \mathbf{n}))$

The singular values of A in decreasing order.

4: **rank** – INTEGER

The effective rank of A, i.e., the number of singular values which are greater than $\mathbf{rcond} \times \mathbf{s}(1)$.

5: **info** – INTEGER

$\mathbf{info} = 0$ unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

info $= -i$

If **info** $= -i$, parameter i had an illegal value on entry. The parameters are numbered as follows:

1: **m**, 2: **n**, 3: **nrhs_p**, 4: **a**, 5: **lda**, 6: **b**, 7: **ldb**, 8: **s**, 9: **rcond**, 10: **rank**, 11: **work**, 12: **lwork**, 13: **iwork**, 14: **info**.

It is possible that **info** refers to a parameter that is omitted from the MATLAB interface. This usually indicates that an error in one of the other input parameters has caused an incorrect value to be inferred.

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 Further Comments

The complex analogue of this function is nag_lapack_zgelsd (f08kq).

9 Example

This example solves the linear least squares problem

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

for the solution, x, of minimum norm, where

$$A = \begin{pmatrix} -0.09 & -1.56 & -1.48 & -1.09 & 0.08 & -1.59 \\ 0.14 & 0.20 & -0.43 & 0.84 & 0.55 & -0.72 \\ -0.46 & 0.29 & 0.89 & 0.77 & -1.13 & 1.06 \\ 0.68 & 1.09 & -0.71 & 2.11 & 0.14 & 1.24 \\ 1.29 & 0.51 & -0.96 & -1.27 & 1.74 & 0.34 \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} 7.4 \\ 4.3 \\ -8.1 \\ 1.8 \\ 8.7 \end{pmatrix}.$$

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.

9.1 Program Text

```
function f08kc_example

fprintf('f08kc example results\n\n');

% Least squares problem min ||b - Ax|| where A and b are:
a = [-0.09, -1.56, -1.48, -1.09,  0.08, -1.59;
      0.14,  0.20, -0.43,  0.84,  0.55, -0.72;
     -0.46,  0.29,  0.89,  0.77, -1.13,  1.06;
      0.68,  1.09, -0.71,  2.11,  0.14,  1.24;
      1.29,  0.51, -0.96, -1.27,  1.74,  0.34];
[m,n] = size(a);
b = [ 7.4;
      4.3;
     -8.1;
      1.8;
      8.7;
      0.0];
```

```
% Treat singular values less than 0.01 as zero
rcond = 0.01;
[vr, x, s, rank, info] = f08kc( ...
                                a, b, rcond);

disp('Least squares solution');
disp(x(1:n)');
disp('Tolerance used to estimate the rank of A');
fprintf('%12.2e\n',rcond);
disp('Estimated rank of A');
fprintf('%5d\n\n',rank);
disp('Singular values of A');
disp(s');
```

9.2 Program Results

```
     f08kc example results

Least squares solution
    1.5938   -0.1180   -3.1501    0.1554    2.5529   -1.6730

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
```