NAG Toolbox

nag_lapack_dgglse (f08za)

1 Purpose

nag_lapack_dgglse (f08za) solves a real linear equality-constrained least squares problem.

2 Syntax

```
[a, b, c, d, x, info] = nag_lapack_dgglse(a, b, c, d, 'm', m, 'n', n, 'p', p)
[a, b, c, d, x, info] = f08za(a, b, c, d, 'm', m, 'n', n, 'p', p)
```

3 Description

nag_lapack_dgglse (f08za) solves the real linear equality-constrained least squares (LSE) problem

$$\underset{x}{\text{minimize}} \|c - Ax\|_2 \quad \text{subject to} \quad Bx = d$$

where A is an m by n matrix, B is a p by n matrix, c is an m element vector and d is a p element vector. It is assumed that $p \leq n \leq m + p$, $\text{rank}(B) = p$ and $\text{rank}(E) = n$, where $E = \begin{pmatrix} A \\ B \end{pmatrix}$. These conditions ensure that the LSE problem has a unique solution, which is obtained using a generalized RQ factorization of the matrices B and A.

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

Anderson E, Bai Z and Dongarra J (1992) Generalized *QR* factorization and its applications *Linear Algebra Appl. (Volume 162–164)* 243–271

EldÉn L (1980) Perturbation theory for the least squares problem with linear equality constraints *SIAM J. Numer. Anal.* **17** 338–350

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 matrix A.

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

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

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

The p by n matrix B.

3: **c**(**m**) – REAL (KIND=nag_wp) array

The right-hand side vector c for the least squares part of the LSE problem.

4: **d**(**p**) – REAL (KIND=nag_wp) array

The right-hand side vector d for the equality constraints.

5.2 Optional Input Parameters

1: **m** – INTEGER

Default: the dimension of the array **c** and the first dimension of the array **a**. (An error is raised if these dimensions are not equal.)

m, the number of rows of the matrix A.

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

2: **n** – INTEGER

Default: the second dimension of the arrays **a**, **b**.

n, the number of columns of the matrices A and B.

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

3: **p** – INTEGER

Default: the dimension of the array **d** and the first dimension of the array **b**. (An error is raised if these dimensions are not equal.)

p, the number of rows of the matrix B.

Constraint: $0 \leq \mathbf{p} \leq \mathbf{n} \leq \mathbf{m} + \mathbf{p}$.

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 $\max(1, \mathbf{n})$.

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

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

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

3: **c**(**m**) – REAL (KIND=nag_wp) array

The residual sum of squares for the solution vector x is given by the sum of squares of elements $\mathbf{c}(\mathbf{n} - \mathbf{p} + 1), \mathbf{c}(\mathbf{n} - \mathbf{p} + 2), \ldots, \mathbf{c}(\mathbf{m})$; the remaining elements are overwritten.

4: **d**(**p**) – REAL (KIND=nag_wp) array

5: **x**(**n**) – REAL (KIND=nag_wp) array

The solution vector x of the LSE problem.

6: **info** – INTEGER

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

6 Error Indicators and Warnings

$\mathbf{info} = -i$

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

1: **m**, 2: **n**, 3: **p**, 4: **a**, 5: **lda**, 6: **b**, 7: **ldb**, 8: **c**, 9: **d**, 10: **x**, 11: **work**, 12: **lwork**, 13: **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 $= 1$

The upper triangular factor R associated with B in the generalized RQ factorization of the pair (B, A) is singular, so that $\text{rank}(B) < p$; the least squares solution could not be computed.

info $= 2$

The $(N - P)$ by $(N - P)$ part of the upper trapezoidal factor T associated with A in the generalized RQ factorization of the pair (B, A) is singular, so that the rank of the matrix (E) comprising the rows of A and B is less than n; the least squares solutions could not be computed.

7 Accuracy

For an error analysis, see Anderson *et al.* (1992) and EldÉn (1980). See also Section 4.6 of Anderson *et al.* (1999).

8 Further Comments

When $m \geq n = p$, the total number of floating-point operations is approximately $\frac{2}{3}n^2(6m + n)$; if $p \ll n$, the number reduces to approximately $\frac{2}{3}n^2(3m - n)$.

nag_opt_lsq_lincon_solve (e04nc) may also be used to solve LSE problems. It differs from nag_lapack_dgglse (f08za) in that it uses an iterative (rather than direct) method, and that it allows general upper and lower bounds to be specified for the variables x and the linear constraints Bx.

9 Example

This example solves the least squares problem

$$\underset{x}{\text{minimize}} \|c - Ax\|_2 \quad \text{subject to} \quad Bx = d$$

where

$$c = \begin{pmatrix} -1.50 \\ -2.14 \\ 1.23 \\ -0.54 \\ -1.68 \\ 0.82 \end{pmatrix},$$

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

$$B = \begin{pmatrix} 1.0 & 0 & -1.0 & 0 \\ 0 & 1.0 & 0 & -1.0 \end{pmatrix}$$

and

$$d = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

The constraints $Bx = d$ correspond to $x_1 = x_3$ and $x_2 = x_4$.

9.1 Program Text

```
function f08za_example

fprintf('f08za example results\n\n');

% Minimize ||c - Ax|| given Bx=d
a = [-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];
c = [-1.50; -2.14;  1.23; -0.54;  -1.68;  0.82];
b = [ 1,     0,    -1,     0;
      0,     1,     0,    -1];
d = [ 0;
      0];

%
[~, ~, resid, ~, x, info] = f08za( ...
                                   a, b, c, d);

sqres = norm(resid(3:6),2);
disp('Constrained least-squares solution');
disp(x);
disp('Square root of the residual sum of squares');
disp(sqres);
```

9.2 Program Results

```
f08za example results

Constrained least-squares solution
    0.4890
    0.9975
    0.4890
    0.9975

Square root of the residual sum of squares
    0.0251
```
