NAG Toolbox

nag_eigen_real_gen_qu_svd (f02wd)

1 Purpose

nag_eigen_real_gen_qu_svd (f02wd) returns the Householder QU factorization of a real rectangular m by n $(m \geq n)$ matrix A. Further, on request or if A is not of full rank, part or all of the singular value decomposition of A is returned.

Note: This function is scheduled to be withdrawn, please see f02wd in Advice on Replacement Calls for Withdrawn/Superseded Routines..

2 Syntax

```
[a, b, svd, irank, z, sv, r, pt, work, ifail] = nag_eigen_real_gen_qu_svd(a,
wantb, b, tol, svd, wantr, wantpt, lwork, 'm', m, 'n', n)
```

```
[a, b, svd, irank, z, sv, r, pt, work, ifail] = f02wd(a, wantb, b, tol, svd,
wantr, wantpt, lwork, 'm', m, 'n', n)
```

3 Description

The real m by n $(m \geq n)$ matrix A is first factorized as

$$A = Q\begin{pmatrix} U \\ 0 \end{pmatrix},$$

where Q is an m by m orthogonal matrix and U is an n by n upper triangular matrix.

If either U is singular or **svd** is supplied as *true*, then the singular value decomposition (SVD) of U is obtained so that U is factorized as

$$U = RDP^{\mathrm{T}},$$

where R and P are n by n orthogonal matrices and D is the n by n diagonal matrix

$$D = \mathrm{diag}(sv_1, sv_2, \ldots, sv_n),$$

with $sv_1 \geq sv_2 \geq \cdots \geq sv_n \geq 0$.

Note that the SVD of A is then given by

$$A = Q_1\begin{pmatrix} D \\ 0 \end{pmatrix}P^{\mathrm{T}} \quad \text{where} \quad Q_1 = Q\begin{pmatrix} R & 0 \\ 0 & I \end{pmatrix},$$

the diagonal elements of D being the singular values of A.

The option to form a vector $Q^{\mathrm{T}}b$, or if appropriate $Q_1^{\mathrm{T}}b$, is also provided.

The rank of the matrix A, based upon a user-supplied argument **tol**, is also returned.

The QU factorization of A is obtained by Householder transformations. To obtain the SVD of U the matrix is first reduced to bidiagonal form by means of plane rotations and then the QR algorithm is used to obtain the SVD of the bidiagonal form.

4 References

Wilkinson J H (1978) Singular Value Decomposition – Basic Aspects *Numerical Software – Needs and Availability* (ed D A H Jacobs) Academic Press

5 Parameters

5.1 Compulsory Input Parameters

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

lda, the first dimension of the array, must satisfy the constraint $lda \geq$ **m**.

The leading m by n part of **a** must contain the matrix to be factorized.

2: **wantb** – LOGICAL

Must be *true* if $Q^{\mathrm{T}}b$ or $Q_1^{\mathrm{T}}b$ is required.

If on entry **wantb** $=$ *false*, **b** is not referenced.

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

If **wantb** is supplied as *true*, **b** must contain the m element vector b. Otherwise, **b** is not referenced.

4: **tol** – REAL (KIND=nag_wp)

Must specify a relative tolerance to be used to determine the rank of A. **tol** should be chosen as approximately the largest relative error in the elements of A. For example, if the elements of A are correct to about 4 significant figures, **tol** should be set to about 5×10^{-4}. See Section 9.3 for a description of how **tol** is used to determine rank.

If **tol** is outside the range $(\epsilon, 1.0)$, where ϵ is the ***machine precision***, the value ϵ is used in place of **tol**. For most problems this is unreasonably small.

5: **svd** – LOGICAL

Must be *true* if the singular values are to be found even if A is of full rank.

If before entry, **svd** $=$ *false* **and** A is determined to be of full rank, only the QU factorization of A is computed.

6: **wantr** – LOGICAL

Must be *true* if the orthogonal matrix R is required when the singular values are computed.

If on entry **wantr** $=$ *false*, **r** is not referenced.

7: **wantpt** – LOGICAL

Must be *true* if the orthogonal matrix P^{T} is required when the singular values are computed.

Note that if **svd** is returned as *true*, **pt** is referenced even if **wantpt** is supplied as *false*, but see argument **pt**.

8: **lwork** – INTEGER

The dimension of the array **work**.

Constraint: **lwork** $\geq 3 \times$ **n**.

5.2 Optional Input Parameters

1: **m** – INTEGER

Default: the dimension of the array **b** 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: **m** $\geq$ **n**.

2: **n** – INTEGER

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

n, the number of columns of the matrix A.

Constraint: $1 \leq \mathbf{n} \leq \mathbf{m}$.

5.3 Output Parameters

1: **a**$(lda, \mathbf{n})$ – REAL (KIND=nag_wp) array

The leading m by n part of **a**, together with the n-element vector **z**, contains details of the Householder QU factorization.

Details of the storage of the QU factorization are given in Section 9.4.

2: **b**$(\mathbf{m})$ – REAL (KIND=nag_wp) array

Contains $Q_1^{\mathrm{T}}b$ if **svd** is returned as *true* and $Q^{\mathrm{T}}b$ if **svd** is returned as *false*.

3: **svd** – LOGICAL

Is returned as *false* if only the QU factorization of A has been obtained and is returned as *true* if the singular values of A have been obtained.

4: **irank** – INTEGER

Returns the rank of the matrix A. (It should be noted that it is possible for **irank** to be returned as n and **svd** to be returned as *true*, even if **svd** was supplied as *false*. This means that the matrix U only just failed the test for nonsingularity.)

5: **z**$(\mathbf{n})$ – REAL (KIND=nag_wp) array

The n-element vector **z** contains some details of the Householder transformations. See Section 9.4 for further information.

6: **sv**$(\mathbf{n})$ – REAL (KIND=nag_wp) array

If **svd** is returned as *true*, **sv** contains the n singular values of A arranged in descending order.

7: **r**$(ldr, \mathbf{n})$ – REAL (KIND=nag_wp) array

The first dimension, ldr, of the array **r** will be

if **wantr** $=$ *true*, $ldr = \mathbf{n}$;
otherwise $ldr = 1$.

The second dimension of the array **r** will be **n** if **wantr** $=$ *true* and 1 otherwise.

If **svd** is returned as *true* and **wantr** was supplied as *true*, the leading n by n part of **r** will contain the left-hand orthogonal matrix of the **svd** of U.

8: **pt**$(ldpt, \mathbf{n})$ – REAL (KIND=nag_wp) array

If **svd** is returned as *true* and **wantpt** was supplied as *true*, the leading n by n part of **pt** contains the orthogonal matrix P^{T}.

If **svd** is returned as *true*, but **wantpt** was supplied as *false*, the leading n by n part of **pt** is used for internal workspace.

9: **work**$(\mathbf{lwork})$ – REAL (KIND=nag_wp) array

If **svd** is returned as *false*, **work**(1) contains the condition number $\|U\|_E \|U^{-1}\|_E$ of the upper triangular matrix U.

If **svd** is returned as *true*, **work**(1) will contain the total number of iterations taken by the QR algorithm.

The rest of the array is used as workspace and so contains no meaningful information.

10: **ifail** – INTEGER

ifail $= 0$ unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

On entry, **n** < 1,
or **m** $<$ **n**,
or $lda <$ **m**,
or $ldr <$ **n** when **wantr** $= true$,
or $ldpt <$ **n**
or **lwork** $< 3 \times$ **n**.

(The function only checks ldr if **wantr** is supplied as *true*.)

ifail > 1

The QR algorithm has failed to converge to the singular values in $50 \times$ **n** iterations. In this case **sv**(1), **sv**$(2), \dots,$ **sv**$($**ifail** $- 1)$ may not have been correctly found and the remaining singular values may not be the smallest singular values. The matrix A has nevertheless been factorized as $A = Q_1 C P^{\mathrm{T}}$, where C is an upper bidiagonal matrix with **sv**(1), **sv**$(2), \dots,$ **sv**(n) as its diagonal elements and **work**(2), **work**$(3), \dots,$ **work**(n) as its superdiagonal elements.

This failure cannot occur if **svd** is returned as *false* and in any case is extremely rare.

ifail $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

ifail $= -399$

Your licence key may have expired or may not have been installed correctly.

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

The computed factors Q, U, R, D and P^{T} satisfy the relations

$$Q\begin{pmatrix} U \\ 0 \end{pmatrix} = A + E,$$

$$Q\begin{pmatrix} R & 0 \\ 0 & I \end{pmatrix}\begin{pmatrix} D \\ 0 \end{pmatrix} P^{\mathrm{T}} = A + F$$

where $\|E\|_2 \leq c_1 \epsilon \|A\|_2$, $\|F\|_2 \leq c_2 \epsilon \|A\|_2$,

ϵ being the ***machine precision*** and c_1 and c_2 are modest functions of m and n. Note that $\|A\|_2 = sv_1$.

8 Further Comments

8.1 Timing

The time taken by nag_eigen_real_gen_qu_svd (f02wd) to obtain the Householder QU factorization is approximately proportional to $n^2(3m - n)$.

The **additional** time taken to obtain the singular value decomposition is approximately proportional to n^3, where the constant of proportionality depends upon whether or not the orthogonal matrices R and P^{T} are required.

8.2 General Remarks

Singular vectors associated with a zero or multiple singular value, are not uniquely determined, even in exact arithmetic, and very different results may be obtained if they are computed on different machines.

This function is called by the least squares function nag_linsys_real_gen_solve (f04jg).

8.3 Determining the Rank of A

Following the QU factorization of A, if **svd** is supplied as *false*, then the condition number of U given by

$$C(U) = \|U\|_F \left\|U^{-1}\right\|_F$$

is found, where $\|.\|_F$ denotes the Frobenius norm, and if $C(U)$ is such that

$$C(U) \times \textbf{tol} > 1.0$$

then U is regarded as singular and the singular values of A are computed. If this test is not satisfied, then the rank of A is set to n. Note that if **svd** is supplied as *true* then this test is omitted.

When the singular values are computed, then the rank of A, r, is returned as the largest integer such that

$$sv_r > \textbf{tol} \times sv_1,$$

unless $sv_1 = 0$ in which case r is returned as zero. That is, singular values which satisfy $sv_i \leq \textbf{tol} \times sv_1$ are regarded as negligible because relative perturbations of order **tol** can make such singular values zero.

8.4 Storage Details of the QU Factorization

The kth Householder transformation matrix, T_k, used in the QU factorization is chosen to introduce the zeros into the kth column and has the form

$$T_k = I - 2\begin{pmatrix} 0 \\ u \end{pmatrix}\begin{pmatrix} 0 & u^{\mathrm{T}} \end{pmatrix}, \quad u^{\mathrm{T}}u = 1,$$

where u is an $(m - k + 1)$ element vector.

In place of u the function actually computes the vector z given by

$$z = 2u_1u.$$

The first element of z is stored in $\mathbf{z}(k)$ and the remaining elements of z are overwritten on the subdiagonal elements of the kth column of **a**. The upper triangular matrix U is overwritten on the n by n upper triangular part of **a**.

9 Example

This example obtains the rank and the singular value decomposition of the 6 by 4 matrix A given by

$$A = \begin{pmatrix} 22.25 & 31.75 & -38.25 & 65.50 \\ 20.00 & 26.75 & 28.50 & -26.50 \\ -15.25 & 24.25 & 27.75 & 18.50 \\ 27.25 & 10.00 & 3.00 & 2.00 \\ -17.25 & -30.75 & 11.25 & 7.50 \\ 17.25 & 30.75 & -11.25 & -7.50 \end{pmatrix}$$

the value **tol** to be taken as 5×10^{-4}.

9.1 Program Text

```
function f02wd_example

fprintf('f02wd example results\n\n');

a = [ 22.25, 31.75,-38.25, 65.5;
      20.00, 26.75, 28.50,-26.5;
     -15.25, 24.25, 27.75, 18.5;
      27.25, 10.00,  3.00, 2.00;
     -17.25,-30.75, 11.25, 7.50;
      17.25, 30.75,-11.25,-7.50];

% Compute rank and SVD of A
wantb = false;
b     = zeros(6,1);
tol   = 0.0005;
svd   = true;
wantr = true;
wantpt = true;
lwork = nag_int(24);

[a, b, svd, irank, z, sv, r, pt, work, ifail] = ...
f02wd( ...
       a, wantb, b, tol, svd, wantr, wantpt, lwork);

fprintf('Rank of A is %2d\n\n', irank);
disp('Details of QU factorization');
disp(a);

disp('Vector z');
disp(z');

disp('Matrix R');
disp(r);

disp('Singular values');
disp(sv');

disp('Matrix P');
disp(pt');
```

9.2 Program Results

```
f02wd example results

Rank of A is  4

Details of QU factorization
  -49.6519  -44.4092   20.3542   -8.8818
    0.4028  -48.2767   -9.5887  -20.3761
   -0.3071    0.8369   52.9270  -48.8806
    0.5488   -0.3907   -0.8364  -50.6742
   -0.3474   -0.2585   -0.1851    0.6321
    0.3474    0.2585    0.1851   -0.6321
```

```
Vector z
    1.4481    1.1153    1.4817    1.4482

Matrix R
   -0.5639    0.6344    0.4229    0.3172
   -0.3512    0.3951   -0.6791   -0.5093
   -0.6403   -0.5692    0.3095   -0.4126
   -0.3855   -0.3427   -0.5140    0.6854

Singular values
   91.0000   68.2500   45.5000   22.7500

Matrix P
    0.3077   -0.4615   -0.4615   -0.6923
    0.4615   -0.6923    0.3077    0.4615
   -0.4615   -0.3077    0.6923   -0.4615
    0.6923    0.4615    0.4615   -0.3077
```