

NAG Library Routine Document

F08NAF (DGEEV)

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

F08NAF (DGEEV) computes the eigenvalues and, optionally, the left and/or right eigenvectors for an n by n real nonsymmetric matrix A .

2 Specification

```
SUBROUTINE F08NAF (JOBVL, JOBVR, N, A, LDA, WR, WI, VL, LDVL, VR, LDVR,
1                     WORK, LWORK, INFO)
    INTEGER          N, LDA, LDVL, LDVR, LWORK, INFO
    double precision A(LDA,*), WR(*), WI(*), VL(LDVL,*), VR(LDVR,*),
1                     WORK(max(1,LWORK))
    CHARACTER*1      JOBVL, JOBVR
```

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

3 Description

The right eigenvector v_j of A satisfies

$$Av_j = \lambda_j v_j$$

where λ_j is the j th eigenvalue of A . The left eigenvector u_j of A satisfies

$$u_j^H A = \lambda_j u_j^H$$

where u_j^H denotes the conjugate transpose of u_j .

The matrix A is first reduced to upper Hessenberg form by means of orthogonal similarity transformations, and the QR algorithm is then used to further reduce the matrix to upper quasi-triangular Schur form, T , with 1 by 1 and 2 by 2 blocks on the main diagonal. The eigenvalues are computed from T , the 2 by 2 blocks corresponding to complex conjugate pairs and, optionally, the eigenvectors of T are computed and backtransformed to the eigenvectors of 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 <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: JOBVL – CHARACTER*1 *Input*
On entry: if JOBVL = 'N', the left eigenvectors of A are not computed.
 If JOBVL = 'V', the left eigenvectors of A are computed.
Constraint: JOBVL = 'N' or 'V'.

- 2: JOBVR – CHARACTER*1 *Input*
On entry: if JOBVR = 'N', the right eigenvectors of A are not computed.
 If JOBVR = 'V', the right eigenvectors of A are computed.
Constraint: JOBVR = 'N' or 'V'.
- 3: N – INTEGER *Input*
On entry: n , the order of the matrix A .
Constraint: $N \geq 0$.
- 4: A(LDA,*) – **double precision** array *Input/Output*
Note: the second dimension of the array A must be at least $\max(1, N)$.
On entry: the n by n matrix A .
On exit: A has been overwritten.
- 5: LDA – INTEGER *Input*
On entry: the first dimension of the array A as declared in the (sub)program from which F08NAF (DGEEV) is called.
Constraint: $LDA \geq \max(1, N)$.
- 6: WR(*) – **double precision** array *Output*
 7: WI(*) – **double precision** array *Output*
Note: the dimension of the arrays WR and WI must be at least $\max(1, N)$.
On exit: WR and WI contain the real and imaginary parts, respectively, of the computed eigenvalues. Complex conjugate pairs of eigenvalues appear consecutively with the eigenvalue having the positive imaginary part first.
- 8: VL(LDVL,*) – **double precision** array *Output*
Note: the second dimension of the array VL must be at least $\max(1, N)$ if JOBVL = 'V', and at least 1 otherwise.
On exit: if JOBVL = 'V', the left eigenvectors u_j are stored one after another in the columns of VL, in the same order as their corresponding eigenvalues.
 If JOBVL = 'N', VL is not referenced.
 If the j th eigenvalue is real, then $u_j = \text{VL}(:, j)$, the j th column of VL.
 If the j th and $(j + 1)$ st eigenvalues form a complex conjugate pair, then $u_j = \text{VL}(:, j) + i \times \text{VL}(:, j + 1)$ and $u_{j+1} = \text{VL}(:, j) - i \times \text{VL}(:, j + 1)$.
- 9: LDVL – INTEGER *Input*
On entry: the first dimension of the array VL as declared in the (sub)program from which F08NAF (DGEEV) is called.
Constraints:
 if JOBVL = 'V', $LDVL \geq \max(1, N)$;
 otherwise $LDVL \geq 1$.

- 10: VR(LDVR,*) – **double precision** array *Output*
- Note:** the second dimension of the array VR must be at least $\max(1, N)$ if JOBVR = 'V', and at least 1 otherwise.
- On exit:* if JOBVR = 'V', the right eigenvectors v_j are stored one after another in the columns of VR, in the same order as their corresponding eigenvalues.
- If JOBVR = 'N', VR is not referenced.
- If the j th eigenvalue is real, then $v_j = \text{VR}(:, j)$, the j th column of VR.
- If the j th and $(j + 1)$ st eigenvalues form a complex conjugate pair, then $v_j = \text{VR}(:, j) + i \times \text{VR}(:, j + 1)$ and $v_{j+1} = \text{VR}(:, j) - i \times \text{VR}(:, j + 1)$.
- 11: LDVR – INTEGER *Input*
- On entry:* the first dimension of the array VR as declared in the (sub)program from which F08NAF (DGEEV) is called.
- Constraints:*
- if JOBVR = 'V', $\text{LDVR} \geq \max(1, N)$;
otherwise $\text{LDVR} \geq 1$.
- 12: WORK($\max(1, \text{LWORK})$) – **double precision** array *Workspace*
- On exit:* if INFO = 0, WORK(1) contains the minimum value of LWORK required for optimal performance.
- 13: LWORK – INTEGER *Input*
- On entry:* the dimension of the array WORK as declared in the (sub)program from which F08NAF (DGEEV) 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 must generally be larger than the minimum, say, $4 \times N + nb \times N$, where nb is the optimal **block size** of F08NEF (DGEHRD).
- Constraints:*
- if JOBVL = 'V' or JOBVR = 'V', $\text{LWORK} \geq 4 \times N$;
otherwise $\text{LWORK} \geq \max(1, 3 \times N)$.
- 14: 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 = $-i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

INFO > 0

If INFO = i , the QR algorithm failed to compute all the eigenvalues, and no eigenvectors have been computed; elements $i + 1 : N$ of WRWI contain eigenvalues which have converged.

7 Accuracy

The computed eigenvalues and eigenvectors are exact for a nearby matrix $(A + E)$, where

$$\|E\|_2 = O(\epsilon)\|A\|_2,$$

and ϵ is the *machine precision*. See Section 4.8 of Anderson *et al.* (1999) for further details.

8 Further Comments

Each eigenvector is normalized to have Euclidean norm equal to unity and the element of largest absolute value real and positive.

The total number of floating-point operations is proportional to n^3 .

The complex analogue of this routine is F08NNF (ZGEEV).

9 Example

This example finds all the eigenvalues and right eigenvectors of the matrix

$$A = \begin{pmatrix} 0.35 & 0.45 & -0.14 & -0.17 \\ 0.09 & 0.07 & -0.54 & 0.35 \\ -0.44 & -0.33 & -0.03 & 0.17 \\ 0.25 & -0.32 & -0.13 & 0.11 \end{pmatrix}.$$

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

```
*      F08NAF Example Program Text
*      Mark 21 Release.  NAG Copyright 2004
*      .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NB, NMAX
      PARAMETER        (NB=64,NMAX=10)
      INTEGER          LDA, LDVR, LWORK
      PARAMETER        (LDA=NMAX,LDVR=NMAX,LWORK=(2+NB)*NMAX)
*      .. Local Scalars ..
      COMPLEX *16      EIG
      INTEGER          I, INFO, J, LWKOPT, N
*      .. Local Arrays ..
      DOUBLE PRECISION A(LDA,NMAX), DUMMY(1,1), VR(LDVR,NMAX), WI(NMAX),
+      WORK(LWORK), WR(NMAX)
*      .. External Subroutines ..
      EXTERNAL         DGEEV
*      .. Intrinsic Functions ..
      INTRINSIC        CMPLX, INT
*      .. Executable Statements ..
      WRITE (NOUT,*) 'F08NAF Example Program Results'
*      Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      IF (N.LE.NMAX) THEN
*
*          Read the matrix A from data file
*
*          READ (NIN,*) ((A(I,J),J=1,N),I=1,N)
*
*          Compute the eigenvalues and right eigenvectors of A
*
*          CALL DGEEV('No left vectors','Vectors (right)',N,A,LDA,WR,WI,
+          DUMMY,1,VR,LDVR,WORK,LWORK,INFO)
*          LWKOPT = INT(WORK(1))
*

```

```

      IF (INFO.EQ.0) THEN
*
*       Print solution
*
      DO 20 J = 1, N
        WRITE (NOUT,*)
        IF (WI(J).EQ.0.0D0) THEN
          WRITE (NOUT,99999) 'Eigenvalue( ', J, ' ) = ', WR(J)
        ELSE
          EIG = CMPLX(WR(J),WI(J),KIND=KIND(A))
          WRITE (NOUT,99998) 'Eigenvalue( ', J, ' ) = ', EIG
        END IF
        WRITE (NOUT,*)
        WRITE (NOUT,99997) 'Eigenvector( ', J, ' )'
        IF (WI(J).EQ.0.0D0) THEN
          WRITE (NOUT,99996) (VR(I,J),I=1,N)
        ELSE IF (WI(J).GT.0.0D0) THEN
          WRITE (NOUT,99995) (VR(I,J),VR(I,J+1),I=1,N)
        ELSE
          WRITE (NOUT,99995) (VR(I,J-1),-VR(I,J),I=1,N)
        END IF
20      CONTINUE
      ELSE
        WRITE (NOUT,*)
        WRITE (NOUT,99994) 'Failure in DGEEV.  INFO = ', INFO
      END IF
*
*       Print workspace information
*
      IF (LWORK.LT.LWKOPT) THEN
        WRITE (NOUT,*)
        WRITE (NOUT,99993) 'Optimum workspace required = ', LWKOPT,
+       'Workspace provided      = ', LWORK
      END IF
      ELSE
        WRITE (NOUT,*)
        WRITE (NOUT,*) 'NMAX too small'
      END IF
*
99999 FORMAT (1X,A,I2,A,1P,E11.4)
99998 FORMAT (1X,A,I2,A,'( ',1P,E11.4,', ',1P,E11.4,')')
99997 FORMAT (1X,A,I2,A)
99996 FORMAT (1X,1P,E11.4)
99995 FORMAT (1X,'( ',1P,E11.4,', ',1P,E11.4,')')
99994 FORMAT (1X,A,I4)
99993 FORMAT (1X,A,I5,/1X,A,I5)
      END

```

9.2 Program Data

F08NAF Example Program Data

```

4                               :Value of N

0.35   0.45  -0.14  -0.17
0.09   0.07  -0.54   0.35
-0.44  -0.33  -0.03   0.17
0.25  -0.32  -0.13   0.11 :End of matrix A

```

9.3 Program Results

F08NAF Example Program Results

Eigenvalue(1) = 7.9948E-01

```

Eigenvector( 1)
-6.5509E-01
-5.2363E-01
5.3622E-01
-9.5607E-02

```

```
Eigenvalue( 2) = (-9.9412E-02, 4.0079E-01)
```

```
Eigenvector( 2)  
(-1.9330E-01, 2.5463E-01)  
( 2.5186E-01,-5.2240E-01)  
( 9.7182E-02,-3.0838E-01)  
( 6.7595E-01, 0.0000E+00)
```

```
Eigenvalue( 3) = (-9.9412E-02,-4.0079E-01)
```

```
Eigenvector( 3)  
(-1.9330E-01,-2.5463E-01)  
( 2.5186E-01, 5.2240E-01)  
( 9.7182E-02, 3.0838E-01)  
( 6.7595E-01, 0.0000E+00)
```

```
Eigenvalue( 4) = -1.0066E-01
```

```
Eigenvector( 4)  
1.2533E-01  
3.3202E-01  
5.9384E-01  
7.2209E-01
```
