

NAG Library Routine Document

F08GBF (DSPEVX)

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

F08GBF (DSPEVX) computes selected eigenvalues and, optionally, eigenvectors of a real n by n symmetric matrix A in packed storage. Eigenvalues and eigenvectors can be selected by specifying either a range of values or a range of indices for the desired eigenvalues.

2 Specification

```
SUBROUTINE F08GBF (JOBZ, RANGE, UPLO, N, AP, VL, VU, IL, IU, ABSTOL, M, W,      &
                  Z, LDZ, WORK, IWORK, JFAIL, INFO)
INTEGER          N, IL, IU, M, LDZ, IWORK(5*N), JFAIL(*), INFO
REAL (KIND=nag_wp) AP(*), VL, VU, ABSTOL, W(N), Z(LDZ,*), WORK(8*N)
CHARACTER(1)     JOBZ, RANGE, UPLO
```

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

3 Description

The symmetric matrix A is first reduced to tridiagonal form, using orthogonal similarity transformations. The required eigenvalues and eigenvectors are then computed from the tridiagonal matrix; the method used depends upon whether all, or selected, eigenvalues and eigenvectors are required.

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>

Demmel J W and Kahan W (1990) Accurate singular values of bidiagonal matrices *SIAM J. Sci. Statist. Comput.* **11** 873–912

Golub G H and Van Loan C F (1996) *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

- 1: JOBZ – CHARACTER(1) *Input*
On entry: indicates whether eigenvectors are computed.
 JOBZ = 'N'
 Only eigenvalues are computed.
 JOBZ = 'V'
 Eigenvalues and eigenvectors are computed.
Constraint: JOBZ = 'N' or 'V'.
- 2: RANGE – CHARACTER(1) *Input*
On entry: if RANGE = 'A', all eigenvalues will be found.
 If RANGE = 'V', all eigenvalues in the half-open interval (VL, VU] will be found.

If RANGE = 'I', the ILth to IUth eigenvalues will be found.

Constraint: RANGE = 'A', 'V' or 'I'.

3: UPLO – CHARACTER(1) *Input*

On entry: if UPLO = 'U', the upper triangular part of A is stored.

If UPLO = 'L', the lower triangular part of A is stored.

Constraint: UPLO = 'U' or 'L'.

4: N – INTEGER *Input*

On entry: n , the order of the matrix A .

Constraint: $N \geq 0$.

5: AP(*) – REAL (KIND=nag_wp) array *Input/Output*

Note: the dimension of the array AP must be at least $\max(1, N \times (N + 1)/2)$.

On entry: the upper or lower triangle of the n by n symmetric matrix A , packed by columns.

More precisely,

if UPLO = 'U', the upper triangle of A must be stored with element A_{ij} in $AP(i + j(j - 1)/2)$ for $i \leq j$;

if UPLO = 'L', the lower triangle of A must be stored with element A_{ij} in $AP(i + (2n - j)(j - 1)/2)$ for $i \geq j$.

On exit: AP is overwritten by the values generated during the reduction to tridiagonal form. The elements of the diagonal and the off-diagonal of the tridiagonal matrix overwrite the corresponding elements of A .

6: VL – REAL (KIND=nag_wp) *Input*

7: VU – REAL (KIND=nag_wp) *Input*

On entry: if RANGE = 'V', the lower and upper bounds of the interval to be searched for eigenvalues.

If RANGE = 'A' or 'I', VL and VU are not referenced.

Constraint: if RANGE = 'V', $VL < VU$.

8: IL – INTEGER *Input*

9: IU – INTEGER *Input*

On entry: if RANGE = 'I', the indices (in ascending order) of the smallest and largest eigenvalues to be returned.

If RANGE = 'A' or 'V', IL and IU are not referenced.

Constraints:

if RANGE = 'I' and $N = 0$, $IL = 1$ and $IU = 0$;

if RANGE = 'I' and $N > 0$, $1 \leq IL \leq IU \leq N$.

10: ABSTOL – REAL (KIND=nag_wp) *Input*

On entry: the absolute error tolerance for the eigenvalues. An approximate eigenvalue is accepted as converged when it is determined to lie in an interval $[a, b]$ of width less than or equal to

$$ABSTOL + \epsilon \max(|a|, |b|),$$

where ϵ is the *machine precision*. If ABSTOL is less than or equal to zero, then $\epsilon \|T\|_1$ will be used in its place, where T is the tridiagonal matrix obtained by reducing A to tridiagonal form. Eigenvalues will be computed most accurately when ABSTOL is set to twice the underflow

threshold $2 \times X02AMF()$, not zero. If this routine returns with $INFO > 0$, indicating that some eigenvectors did not converge, try setting $ABSTOL$ to $2 \times X02AMF()$. See Demmel and Kahan (1990).

- 11: M – INTEGER *Output*
On exit: the total number of eigenvalues found. $0 \leq M \leq N$.
 If $RANGE = 'A'$, $M = N$.
 If $RANGE = 'I'$, $M = IU - IL + 1$.
- 12: $W(N)$ – REAL (KIND=nag_wp) array *Output*
On exit: the selected eigenvalues in ascending order.
- 13: $Z(LDZ,*)$ – REAL (KIND=nag_wp) array *Output*
Note: the second dimension of the array Z must be at least $\max(1, M)$ if $JOBZ = 'V'$, and at least 1 otherwise.
On exit: if $JOBZ = 'V'$, then
 if $INFO = 0$, the first M columns of Z contain the orthonormal eigenvectors of the matrix A corresponding to the selected eigenvalues, with the i th column of Z holding the eigenvector associated with $W(i)$;
 if an eigenvector fails to converge ($INFO > 0$), then that column of Z contains the latest approximation to the eigenvector, and the index of the eigenvector is returned in $JFAIL$.
 If $JOBZ = 'N'$, Z is not referenced.
Note: you must ensure that at least $\max(1, M)$ columns are supplied in the array Z ; if $RANGE = 'V'$, the exact value of M is not known in advance and an upper bound of at least N must be used.
- 14: LDZ – INTEGER *Input*
On entry: the first dimension of the array Z as declared in the (sub)program from which F08GBF (DSPEVX) is called.
Constraints:
 if $JOBZ = 'V'$, $LDZ \geq \max(1, N)$;
 otherwise $LDZ \geq 1$.
- 15: $WORK(8 \times N)$ – REAL (KIND=nag_wp) array *Workspace*
- 16: $IWORK(5 \times N)$ – INTEGER array *Workspace*
- 17: $JFAIL(*)$ – INTEGER array *Output*
Note: the dimension of the array $JFAIL$ must be at least $\max(1, N)$.
On exit: if $JOBZ = 'V'$, then
 if $INFO = 0$, the first M elements of $JFAIL$ are zero;
 if $INFO > 0$, $JFAIL$ contains the indices of the eigenvectors that failed to converge.
 If $JOBZ = 'N'$, $JFAIL$ is not referenced.
- 18: $INFO$ – 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 , then i eigenvectors failed to converge. Their indices are stored in array JFAIL. Please see ABSTOL.

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 4.7 of Anderson *et al.* (1999) for further details.

8 Further Comments

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

The complex analogue of this routine is F08GPF (ZHPEVX).

9 Example

This example finds the eigenvalues in the half-open interval $(-1, 1]$, and the corresponding eigenvectors, of the symmetric matrix

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 2 & 3 & 4 \\ 3 & 3 & 3 & 4 \\ 4 & 4 & 4 & 4 \end{pmatrix}.$$

9.1 Program Text

```

Program f08gbfe

!      F08GBF Example Program Text

!      Mark 24 Release. NAG Copyright 2012.

!      .. Use Statements ..
Use nag_library, Only: dspevx, nag_wp, x04caf
!      .. Implicit None Statement ..
Implicit None
!      .. Parameters ..
Real (Kind=nag_wp), Parameter      :: zero = 0.0E+0_nag_wp
Integer, Parameter                 :: nin = 5, nout = 6
Character (1), Parameter           :: uplo = 'U'
!      .. Local Scalars ..
Real (Kind=nag_wp)                 :: abstol, vl, vu
Integer                             :: i, ifail, il, info, iu, j, ldz, m, n
!      .. Local Arrays ..
Real (Kind=nag_wp), Allocatable    :: ap(:), w(:), work(:), z(:, :)
Integer, Allocatable                :: iwork(:), jfail(:)
!      .. Executable Statements ..
Write (nout,*) 'F08GBF Example Program Results'
Write (nout,*)
!      Skip heading in data file
Read (nin,*)

```

```

Read (nin,*) n
ldz = n
m = n
Allocate (ap((n*(n+1))/2),w(n),work(8*n),z(ldz,m),iwork(5*n),jfail(n))

! Read the lower and upper bounds of the interval to be searched,
! and read the upper or lower triangular part of the matrix A
! from data file

Read (nin,*) vl, vu
If (uplo=='U') Then
  Read (nin,*)((ap(i+(j*(j-1))/2),j=i,n),i=1,n)
Else If (uplo=='L') Then
  Read (nin,*)((ap(i+((2*n-j)*(j-1))/2),j=1,i),i=1,n)
End If

! Set the absolute error tolerance for eigenvalues. With ABSTOL
! set to zero, the default value is used instead

abstol = zero

! Solve the symmetric eigenvalue problem
! The NAG name equivalent of dspevx is f08gbf
Call dspevx('Vectors','Values in range',uplo,n,ap,vl,vu,il,iu,abstol,m, &
  w,z,ldz,work,iwork,jfail,info)

If (info>=0) Then

!   Print solution

  Write (nout,99999) 'Number of eigenvalues found =', m
  Write (nout,*)
  Write (nout,*) 'Eigenvalues'
  Write (nout,99998) w(1:m)
  Flush (nout)

!   Normalize the eigenvectors
  Do i = 1, m
    z(1:n,i) = z(1:n,i)/z(1,i)
  End Do

!   ifail: behaviour on error exit
!   =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
  ifail = 0
  Call x04caf('General',' ',n,m,z,ldz,'Selected eigenvectors',ifail)

  If (info>0) Then
    Write (nout,99999) 'INFO eigenvectors failed to converge, INFO =', &
      info
    Write (nout,*) 'Indices of eigenvectors that did not converge'
    Write (nout,99997) jfail(1:m)
  End If
Else
  Write (nout,99999) 'Failure in DSPEVX. INFO =', info
End If

99999 Format (1X,A,I5)
99998 Format (3X,(8F8.4))
99997 Format (3X,(8I8))
End Program f08gbfe

```

9.2 Program Data

F08GBF Example Program Data

```
4 :Value of N
-1.0 1.0 :Values of VL and VU

1.0 2.0 3.0 4.0
    2.0 3.0 4.0
        3.0 4.0
            4.0 :End of matrix A
```

9.3 Program Results

F08GBF Example Program Results

Number of eigenvalues found = 2

Eigenvalues

-0.5146 -0.2943

Selected eigenvectors

	1	2
1	1.0000	1.0000
2	-0.9431	-2.3976
3	-1.0537	2.3508
4	0.8831	-0.8879
