

## 5.3 Eigenvalues of an Unsymmetric Matrix

### A. Purpose

Compute all eigenvalues of a real  $N \times N$  unsymmetric matrix  $A$ . Some or all of the eigenvalues may be complex.

### B. Usage

#### B.1 Program Prototype, Single Precision

**REAL** A(LDA,  $\geq N$ ) [LDA $\geq N$ ], **VR**( $\geq N$ ), **VI**( $\geq N$ )

**INTEGER** LDA, N, **IFLAG**( $\geq N$ )

Assign values to A( $\cdot$ ), LDA, and N.

**CALL SEVUN(A, LDA, N, VR, VI, IFLAG)**

Results are returned in VR( $\cdot$ ), VI( $\cdot$ ), IFLAG(1). The contents of A( $\cdot$ ) will be modified.

#### B.2 Argument Definitions

**A**( $\cdot$ ), **LDA**, **N** A( $\cdot$ ) is [inout], LDA and N are [in]. On entry A( $\cdot$ ) must contain the  $N \times N$  matrix  $A$  whose eigenvalues are to be computed. The integer LDA is the dimension of the first subscript of the array A( $\cdot$ ). Require  $LDA \geq N$ . On return the contents of A( $\cdot$ ) will be modified.

**VR**( $\cdot$ ), **VI**( $\cdot$ ) [out] The subroutine will store the  $J^{th}$  eigenvalue in VR( $J$ ) and VI( $J$ ),  $J = 1, \dots, N$ . The real part is stored in VR( $J$ ), and the imaginary part in VI( $J$ ). If the  $J^{th}$  eigenvalue is real VI( $J$ ) will be zero. The eigenvalues will be sorted so that  $VR(1) \leq VR(2) \leq \dots \leq VR(N)$ , and if  $VR(J) = VR(J+1)$  for some  $J$  then  $|VI(J)| \leq |VI(J+1)|$ .

Complex eigenvalues will occur in conjugate pairs. Such pairs will be stored in adjacent locations with the eigenvalue having positive imaginary part preceding its conjugate partner.

**IFLAG**( $\cdot$ ) [out, scratch] The N-array IFLAG( $\cdot$ ) will be used as INTEGER working space. In addition, the first location, IFLAG(1), will be used to pass information back to the user as follows:

- = 1 If successful and all eigenvalues are real.
- = 2 If successful and some eigenvalues are complex.

See Section E for use of IFLAG(1) in error conditions.

#### B.3 Modifications for Double Precision

Change SEVUN to DEVUN, and the REAL type statement to DOUBLE PRECISION.

### C. Examples and Remarks

**Example 1:** Define

$$A = \begin{bmatrix} -954 & -464 & -2088 \\ 792 & 387 & 1728 \\ 264 & 128 & 579 \end{bmatrix}.$$

The eigenvalues of  $A$  are  $\lambda_1 = 3$ ,  $\lambda_2 = 3$ ,  $\lambda_3 = 6$ . This example was constructed by using the matrix called  $A^{-1}$  in Example 3.1, page 29, of [1] as the matrix of eigenvectors. It illustrates the case of a double eigenvalue having a full set of eigenvectors.

**Example 2:** Define

$$B = \begin{bmatrix} 4 & 1 & 1 \\ 2 & 4 & 1 \\ 0 & 1 & 4 \end{bmatrix}.$$

This is Example 5.2 of page 82 of [1]. The eigenvalues are  $\lambda_1 = 3$ ,  $\lambda_2 = 3$ , and  $\lambda_3 = 6$ . The matrix is defective in that there is only a one-dimensional space of eigenvectors associated with the eigenvalue 3.

**Example 3:** Define

$$C = \begin{bmatrix} 8 & -1 & -5 \\ -4 & 4 & -2 \\ 18 & -5 & -7 \end{bmatrix}.$$

This is Example 5.4 of page 84 of [1]. The eigenvalues are  $\lambda_1 = 1$ ,  $\lambda_2 = 2 + 4i$ , and  $\lambda_3 = 2 - 4i$ .

The demonstration program DRSEVUN below applies SEVUN to compute eigenvalues for the above three matrices. Results are in the file ODSEVUN.

### D. Functional Description

Given an  $N \times N$  real unsymmetric matrix  $A$  there exists an  $N \times N$  nonsingular matrix  $C$  such that the matrix

$$U = C^{-1}AC$$

is  $N \times N$  upper triangular. The matrices  $C$  and  $U$  may be complex. The diagonal elements of  $U$  are called the eigenvalues of  $A$ . This set of  $N$  numbers is uniquely determined by  $A$  although  $C$  and  $U$  are not unique. Note that  $\lambda$  is an eigenvalue of  $A$  if and only if  $A - \lambda I$  is singular.

This subroutine SEVUN was developed using the subroutines BALANC, ELMHES, and HQR from the EISPACK package of eigenvalue-eigenvector subroutines, [2]. The Fortran subroutines in EISPACK are based directly on the earlier set of Algol procedures described in [3].

Subroutine SEVUN first calls SEVBH which consists of the two EISPACK subroutines, BALANC and ELMHES.

BALANC applies similarity permutations to isolate eigenvalues available by inspection, if any. It then applies diagonal similarity scaling to balance the size of the matrix elements

$$B = D^{-1}P^TAPD.$$

ELMHES reduces  $B$  to upper Hessenberg form using stabilized elementary transformations

$$H = G^{-1}BG.$$

The remainder of SEVUN consists of a minor modification of HQR which applies the QR algorithm to  $H$ . This is an iterative process that reduces  $H$  to a real nearly-upper-triangular matrix  $R$

$$R = Q^THQ.$$

The matrix  $R$  has a mixture of single elements and  $2 \times 2$  blocks on its diagonal and is otherwise upper triangular.

The eigenvalues of  $A$  are the single diagonal elements of  $R$  along with the eigenvalues of the  $2 \times 2$  blocks on the diagonal of  $R$ . These latter eigenvalues are computed by direct formulas.

SEVUN reorders the eigenvalues to achieve the ordering described previously in Section B.

## References

1. R. T. Gregory and D. L. Karney, **A Collection of Matrices for Testing Computational Algorithms**, J. Wiley and Sons, New York (1969) 153 pages.
2. B. T. Smith, J. M. Boyle, B. S. Garbow, Y. Ikebe, V. C. Klema, and C. B. Moler, **Matrix Eigensystem Routines — EISPACK Guide**, *Lecture Notes in Computer Science 6*, Springer Verlag, Berlin (1974) 387 pages.
3. J. H. Wilkinson and C. Reinsch, **Handbook for Automatic Computation, Vol. II. Linear Algebra**, Springer Verlag, Berlin (1971) 439 pages.

## E. Error Procedures and Restrictions

If  $N \leq 0$  or if there is convergence failure in the QR algorithm the error processing subroutine ERMSG of Chapter 19.2 will be called with an error level 0 to print an error message. Upon return, IFLAG(1) = 3 or 4 to indicate  $N \leq 0$  or convergence failure, respectively.

In these error conditions all computed eigenvalues should be regarded as invalid.

## F. Supporting Information

The source language is ANSI Fortran 77.

### Entry

### Required Files

**DEVUN** AMACH, DEVBH, DEVUN, ERFIN, ERMSG

**SEVUN** AMACH, ERFIN, ERMSG, SEVBH, SEVUN

The EISPACK package of Fortran subroutines was acquired at JPL from Argonne National Laboratories where it was developed with financial support from the AEC and the NSF. The subroutine SEVUN was written by F. T. Krogh, JPL, October 1991.

## DRSEVUN

```

c      program DRSEVUN
c>> 1994-10-19 DRSEVUN Krogh Changes to use M77CON
c>> 1992-04-23 DRSEVUN CLL Declaring all variables.
c>> 1992-03-04 DRSEVUN Krogh Initial version.
c      Demonstrate unsymmetric eigenvalue subroutine SEVUN.
c
c-----S replaces "?": DR?EVUN, ?EVUN, ?VECP
c
c      integer I, KASE, LDA, N
c      parameter (LDA = 3)
c      integer IFLAG(LDA)
c      real          A(LDA,LDA,3), VR(LDA), VI(LDA)
c
c      data (A(1,I,1), I=1,3) / -954.0e0, -464.0e0, -2088.0e0 /
c      data (A(2,I,1), I=1,3) / 792.0e0, 387.0e0, 1728.0e0 /
c      data (A(3,I,1), I=1,3) / 264.0e0, 128.0e0, 579.0e0 /
c
c      data (A(1,I,2), I=1,3) / 4.0e0, 1.0e0, 1.0e0 /

```

```

data (A(2,I,2), I=1,3) / 2.0e0, 4.0e0, 1.0e0 /
data (A(3,I,2), I=1,3) / 0.0e0, 1.0e0, 4.0e0 /
c
data (A(1,I,3), I=1,3) / 8.0e0, -1.0e0, -5.0e0 /
data (A(2,I,3), I=1,3) / -4.0e0, 4.0e0, -2.0e0 /
data (A(3,I,3), I=1,3) / 18.0e0, -5.0e0, -7.0e0 /
data N / LDA /
c


---


print*, 'DRSEVUN.. Demo driver for SEVUN.'
do 10 KASE = 1, 3
  print '(/a,i2)', ' Beginning Case =', KASE
  call SEVUN(A(1, 1, KASE), LDA, N, VR, VI, IFLAG)
  print '(a,i2)', ' IFLAG(1) =', IFLAG(1)
  if (IFLAG(1) .le. 2) then
    call SVECP(VR, N, ' Real part of the eigenvalues')
    call SVECP(VI, N, ' Imaginary part of the eigenvalues')
  else
    print '(/a)', ' Failure in SEVUN.'
  end if
10 continue
stop
end

```

## ODSEVUN

DRSEVUN.. Demo driver for SEVUN.

```

Beginning Case = 1
IFLAG(1) = 1
Real part of the eigenvalues
  1 TO   3   3.000000   3.000000   6.000000
Imaginary part of the eigenvalues
  1 TO   3   0.000000   0.000000   0.000000

Beginning Case = 2
IFLAG(1) = 1
Real part of the eigenvalues
  1 TO   3   3.000000   3.000000   6.000000
Imaginary part of the eigenvalues
  1 TO   3   0.000000   0.000000   0.000000

Beginning Case = 3
IFLAG(1) = 2
Real part of the eigenvalues
  1 TO   3   0.9999997   2.000000   2.000000
Imaginary part of the eigenvalues
  1 TO   3   0.000000   4.000000  -4.000000

```