NAG Toolbox for MATLAB

f08sc

1 Purpose

f08sc computes all the eigenvalues and, optionally, the eigenvectors of a real generalized symmetric-definite eigenproblem, of the form

$$Az = \lambda Bz, \qquad ABz = \lambda z \qquad \text{or} \qquad BAz = \lambda z,$$

where A and B are symmetric and B is also positive-definite. If eigenvectors are desired, it uses a divide-and-conquer algorithm.

2 Syntax

```
[a, b, w, info] = f08sc(itype, jobz, uplo, a, b, 'n', n)
```

3 Description

f08sc first performs a Cholesky factorization of the matrix B as $B = U^{\mathrm{T}}U$, when **uplo** = 'U' or $B = LL^{\mathrm{T}}$, when **uplo** = 'L'. The generalized problem is then reduced to a standard symmetric eigenvalue problem

$$Cx = \lambda x,$$

which is solved for the eigenvalues and, optionally, the eigenvectors; the eigenvectors are then backtransformed to give the eigenvectors of the original problem.

For the problem $Az = \lambda Bz$, the eigenvectors are normalized so that the matrix of eigenvectors, z, satisfies

$$Z^{\mathrm{T}}AZ = \Lambda \qquad \text{and} \qquad Z^{\mathrm{T}}BZ = I,$$

where Λ is the diagonal matrix whose diagonal elements are the eigenvalues. For the problem $ABz = \lambda z$ we correspondingly have

$$Z^{-1}AZ^{-\mathrm{T}} = \Lambda \qquad \text{and} \qquad Z^{\mathrm{T}}BZ = I,$$

and for $BAz = \lambda z$ we have

$$Z^{\mathrm{T}}AZ = \Lambda \qquad \text{and} \qquad Z^{\mathrm{T}}B^{-1}Z = I.$$

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 URL: 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

5.1 Compulsory Input Parameters

1: **itype – int32 scalar**

Specifies the problem type to be solved.

itype $= 1$

$Az = \lambda Bz.$

itype $= 2$

$ABz = \lambda z.$

itype $= 3$

$BAz = \lambda z.$

2: **jobz – string**

If **jobz** = 'N', compute eigenvalues only.

If **jobz** = 'V', compute eigenvalues and eigenvectors.

Constraint: **jobz** = 'N' or 'V'.

3: **uplo – string**

If **uplo** = 'U', the upper triangles of A and B are stored.

If **uplo** = 'L', the lower triangles of A and B are stored.

Constraint: **uplo** = 'U' or 'L'.

4: **a(lda,$*$) – double array**

The first dimension of the array **a** must be at least $\max(1, \mathbf{n})$

The second dimension of the array must be at least $\max(1, \mathbf{n})$

The n by n symmetric matrix A.

If **uplo** = 'U', the upper triangular part of A must be stored and the elements of the array below the diagonal are not referenced.

If **uplo** = 'L', the lower triangular part of A must be stored and the elements of the array above the diagonal are not referenced.

5: **b(ldb,$*$) – double array**

The first dimension of the array **b** must be at least $\max(1, \mathbf{n})$

The second dimension of the array must be at least $\max(1, \mathbf{n})$

The symmetric matrix B:

if **uplo** = 'U', the leading n by n upper triangular part of **b** contains the upper triangular part of the matrix B;
if **uplo** = 'L', the leading n by n lower triangular part of **b** contains the lower triangular part of the matrix B.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The first dimension of the arrays **a**, **b** and the second dimension of the arrays **a**, **b**. (An error is raised if these dimensions are not equal.)

n, the order of the matrices A and B.

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

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldb, work, lwork, iwork, liwork

5.4 Output Parameters

1: **a(lda,$*$) – double array**

The first dimension of the array **a** must be at least $\max(1,\mathbf{n})$

The second dimension of the array must be at least $\max(1,\mathbf{n})$

If **jobz** = 'V', then if **info** = 0, **a** contains the matrix Z of eigenvectors. The eigenvectors are normalized as follows:

if **itype** = 1 or 2, $Z^{\mathrm{T}}BZ = I$;
if **itype** = 3, $Z^{\mathrm{T}}B^{-1}Z = I$.

If **jobz** = 'N', the upper triangle (if **uplo** = 'U') or the lower triangle (if **uplo** = 'L') of **a**, including the diagonal, is destroyed.

2: **b(ldb,$*$) – double array**

The first dimension of the array **b** must be at least $\max(1,\mathbf{n})$

The second dimension of the array must be at least $\max(1,\mathbf{n})$

If **info** $\leq$ **n**, the part of **b** containing the matrix contains the triangular factor U or L from the Cholesky factorization **b** $= U^{\mathrm{T}}U$ or **b** $= LL^{\mathrm{T}}$.

3: **w($*$) – double array**

Note: the dimension of the array **w** must be at least $\max(1,\mathbf{n})$.

If **info** = 0, the eigenvalues in ascending order.

4: **info – int32 scalar**

info = 0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

info $= -i$

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

1: **itype**, 2: **jobz**, 3: **uplo**, 4: **n**, 5: **a**, 6: **lda**, 7: **b**, 8: **ldb**, 9: **w**, 10: **work**, 11: **lwork**, 12: **iwork**, 13: **liwork**, 14: **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 > 0

f07fd or f08fc returned an error code:

$\leq$ **n** if **info** $= i$, f08fc failed to converge; i off-diagonal elements of an intermediate tridiagonal form did not converge to zero;

$>$ **n** if **info** $=$ **n** $+ i$, for $1 \leq i \leq$ **n**, then the leading minor of order i of B is not positive-definite. The factorization of B could not be completed and no eigenvalues or eigenvectors were computed.

7 Accuracy

If B is ill-conditioned with respect to inversion, then the error bounds for the computed eigenvalues and vectors may be large, although when the diagonal elements of B differ widely in magnitude the eigenvalues and eigenvectors may be less sensitive than the condition of B would suggest. See Section 4.10 of Anderson *et al.* 1999 for details of the error bounds.

The example program below illustrates the computation of approximate error bounds.

8 Further Comments

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

The complex analogue of this function is f08sq.

9 Example

```
itype = int32(2);
jobz = 'Vectors';
uplo = 'Upper';
a = [0.24, 0.39, 0.42, -0.16;
     0, -0.11, 0.79, 0.63;
     0, 0, -0.25, 0.48;
     0, 0, 0, -0.03];
b = [4.16, -3.12, 0.56, -0.1;
     0, 5.03, -0.83, 1.09;
     0, 0, 0.76, 0.34;
     0, 0, 0, 1.18];
[aOut, bOut, w, info] = f08sc(itype, jobz, uplo, a, b)
```

```
aOut =
    0.0356   -0.1039   -0.7459    0.1909
   -0.3809    0.4322   -0.7845    0.3540
    0.2943    1.5644   -0.7144    0.5665
    0.3186   -1.0647    1.1184    0.3859
bOut =
    2.0396   -1.5297    0.2746   -0.0490
         0    1.6401   -0.2500    0.6189
         0         0    0.7887    0.6443
         0         0         0    0.6161
w =
   -3.5411
   -0.3347
    0.2983
    2.2544
info =
           0
```