



Werkzeuge zur numerischen Behandlung von Eigenwertproblemen

	Bibliotheken	Abgeschlossene Programme
symbolisch		Maple Mathematica MuPad
numerisch	IMSL NAG LAPACK / BLAS EISPACK Numerical Recipes	Matlab Scilab Octave

Matlab

Mit Hilfe von Matlab können Eigenwertaufgaben sehr einfach gelöst werden. Dabei wird sowohl das spezielle als auch das allgemeine Eigenwertproblem unterstützt. Die Lösung wird durch die in Abschnitt 4.6 dargestellten Methoden generiert.

```
A = [ 4 -1 1; 9 -8 9; 11 -11 12 ]  
[ ev, ew ] = eig(A)
```

```
B = [ 1 0 0; 0 1 0; 0 0 1 ]  
[ ev, ew ] = eig(A, B)
```

```
m1 = 600  
m2 = 40  
d1 = 2400  
k1 = 15000  
k2 = 160000  
A = [ -d1/m1 d1/m1 -k1/m1 k1/m1; d1/m2 -d1/m2 k1/m2 -(k1+k2)/m2; ...  
      1 0 0 0; 0 1 0 0 ]  
[ ev, ew ] = eig(A)
```



Maple

Mit Maple können Eigenwerte und Eigenvektoren symbolisch und numerisch bestimmt werden. Dabei wird sowohl das spezielle als auch das allgemeine Eigenwertproblem unterstützt.

```
with(linalg);
A := matrix(3, 3, [ [ 4, -1, 1 ], [ 9, -8, 9 ], [ 11, -11, 12 ] ]);
eigenvals(A);

eigenvects(A);

charpoly(A, 'x');
det(evalm(A - x));
fsolve("");

B := matrix(3, 3, [ [ 1, 0, 0 ], [ 0, 1, 0 ], [ 0, 0, 1 ] ]);
eigenvals(A, B);

m1 := 600.0;
m2 := 40.0;
d1 := 2400.0;
k1 := 15000.0;
k2 := 160000.0;
A := matrix(4, 4,
    [ [ -d1/m1, d1/m1, -k1/m1, k1/m1 ],
      [ d1/m2, -d1/m2, k1/m2, -(k1+k2)/m2 ],
      [ 1.0, 0.0, 0.0, 0.0 ],
      [ 0.0, 1.0, 0.0, 0.0 ] ]);
eigenvals(A);
eigenvects(A);

A := matrix(3, 3, [ [ a11, a21, a31 ], [ a12, a22, a32 ],
    [ a13, a23, a33 ] ] );
eigenvals(A);
eigenvects(A);
```



LAPACK/BLAS

Als Teil der Methoden zur Behandlung von Linearen Algebra Problemen enthält die Bibliothek LAPACK auch leistungsfähige Routinen zur Behandlung von Eigenwertproblemen. Dabei werden symmetrische/hermitesche und nichtsymmetrische sowie speziell vorkonditionierte (Bsp. Schursche Form) Matrizen reell und komplex unterstützt.

NAME

DGEEVX - compute for an N-by-N real nonsymmetric matrix A, the eigenvalues and, optionally, the left and/or right eigenvectors

SYNOPSIS

```
SUBROUTINE DGEEVX( BALANC, JOBVL, JOBVR, SENSE, N, A, LDA,  
                  WR, WI, VL, LDVL, VR, LDVR, ILO, IHI,  
                  SCALE, ABNRM, RCONDE, RCONDV, WORK,  
                  LWORK, IWORK, INFO )
```

PURPOSE

DGEEVX computes for an N-by-N real nonsymmetric matrix A, the eigenvalues and, optionally, the left and/or right eigenvectors.

Optionally also, it computes a balancing transformation to improve the conditioning of the eigenvalues and eigenvectors (ILO, IHI, SCALE, and ABNRM), reciprocal condition numbers for the eigenvalues (RCONDE), and reciprocal condition numbers for the right eigenvectors (RCONDV).

The left eigenvectors of A are the same as the right eigenvectors of A^*T . If $u(j)$ and $v(j)$ are the left and right eigenvectors, respectively, corresponding to the eigenvalue $\lambda(j)$, then $(u(j)^*T) * A = \lambda(j) * (u(j)^*T)$ and $A * v(j) = \lambda(j) * v(j)$.

The computed eigenvectors are normalized to have Euclidean norm equal to 1 and largest component real.

Balancing a matrix means permuting the rows and columns to make it more nearly upper triangular, and applying a diagonal similarity transformation $D * A * D^{*-1}$, where D is a diagonal matrix, to make its rows and columns closer in norm and the condition numbers of its eigenvalues and eigenvectors smaller. The computed reciprocal condition numbers correspond to the balanced matrix. Permuting rows and columns will not change the condition numbers (in exact arithmetic) but diagonal scaling will. For further explanation of balancing, see section 4.10.2 of the LAPACK Users' Guide.