MATLAB Lecture 4 – Eigenvalues

特征值

Ref: MATLAB→Mathematics→Matrices and Linear Algebra →Solving Linear Systems of Equations

- **Vocabulary:**

coefficients 系数　　characteristic polynomial 特征多项式
root 根　　characteristic roots 特征根
eigenvalues 特征值　　eigenvector 特征向量
identity matrix 单位阵　　decomposition 分解
defective matrix 亏损矩阵（即不可对角化矩阵）
orthogonal matrix 正交阵　　unitary matrix 酉阵
nonsymmetric 非对称的　　symmetric 对称的
roundoff error 舍入误差　　diagonalize 对角化
conjugate transpose 共轭转置　　associate matrix 共轭转置矩阵
defective matrix 亏损矩阵（即不可对角化矩阵 not diagonalizable）
singular value decomposition 奇异值分解
Schur decomposition Schur 分解　　similar matrix 相似矩阵
geometric multiplicity 几何重数　　algebraic multiplicity 代数重数

- **Some functions**

eig　poly　roots　schur　svd

- **Eigenvalue & * singular value decomposition**

✧ **Eigenvalues**

An eigenvalue and eigenvector of a square matrix A are a scalar λ and a nonzero vector v that satisfy

$$Av = \lambda v$$

Eigenvalue Decomposition

With the eigenvalues on the diagonal of a diagonal matrix Λ and the corresponding eigenvectors forming the columns of a matrix V, we have

$$AV = V\Lambda$$

If V is nonsingular, this becomes the eigenvalue decomposition

$$A = V\Lambda V^{-1}$$

A good example is provided by the coefficient matrix of the ordinary differential equation

```
>> A =[ 0    -6    -1
        6     2   -16
       -5    20   -10];
>> lambda = eig(A)      % produces a column vector containing the eigenvalues.
lambda =
```

```
   -3.0710
   -2.4645+17.6008i
   -2.4645-17.6008i
>> [V,D] = eig(A)    % computes the eigenvectors and stores the eigenvalues in a …
                     diagonal matrix.
V =
  -0.8326          0.2003 - 0.1394i    0.2003 + 0.1394i
  -0.3553         -0.2110 - 0.6447i   -0.2110 + 0.6447i
  -0.4248         -0.6930             -0.6930
D =
  -3.0710                 0                  0
        0     -2.4645+17.6008i               0
        0                 0       -2.4645-17.6008i
```

The first eigenvector is real and the other two vectors are complex conjugates of each other. All three vectors are normalized to have Euclidean length, norm(v,2), equal to one.

The matrix V*D*inv(V), which can be written more succinctly as V*D/V, is within roundoff error of A. And, inv(V)*A*V, or V\A*V, is within roundoff error of D.

Defective Matrices

Some matrices do not have an eigenvector decomposition. These matrices are defective, or not diagonalizable. For example,

```
>> A = [ 6    12    19
        -9   -20   -33
         4     9    15 ];
>> [V,D] = eig(A)
V =
    -0.4741   -0.4082   -0.4082
     0.8127    0.8165    0.8165
    -0.3386   -0.4082   -0.4082
D =
    -1.0000         0         0
          0    1.0000         0
          0         0    1.0000
```

There is a double eigenvalue at $\lambda = 1$. The second and third columns of V are the same. For this matrix, a full set of linearly independent eigenvectors does not exist.

The optional Symbolic Math Toolbox extends the capabilities of MATLAB by connecting to Maple, a powerful computer algebra system. One of the functions provided by the toolbox computes the Jordan Canonical Form. This is appropriate for matrices like our example, which is 3-by-3 and has exactly known, integer elements.

```
>> [X,J] = jordan(A)
X =
    -1.7500    1.5000    2.7500
     3.0000   -3.0000   -3.0000
    -1.2500    1.5000    1.2500
```

```
J =
    -1     0     0
     0     1     1
     0     0     1
```

The Jordan Canonical Form is an important theoretical concept, but it is not a reliable computational tool for larger matrices, or for matrices whose elements are subject to roundoff errors and other uncertainties.

Schur Decomposition in MATLAB Matrix Computations

The MATLAB advanced matrix computations do not require eigenvalue decompositions. They are based, instead, on the Schur decomposition

$$A = USU'$$

where U is an orthogonal matrix and S is a block upper triangular matrix with 1-by-1 and 2-by-2 blocks on the diagonal. The eigenvalues are revealed by the diagonal elements and blocks of S, while the columns of U provide a basis with much better numerical properties than a set of eigenvectors. The Schur decomposition of our defective example is

```
>> [U,S] = schur(A)
U =
   -0.4741    0.6648    0.5774
    0.8127    0.0782    0.5774
   -0.3386   -0.7430    0.5774
S =
   -1.0000   20.7846  -44.6948
         0    1.0000   -0.6096
         0         0    1.0000
```

The double eigenvalue is contained in the lower 2-by-2 block of S.

✧ *Singular Value Decomposition

A singular value and corresponding singular vectors of a rectangular matrix A are a scalar σ and a pair of vectors u and v that satisfy

$$Av = \sigma u,\ A'u = \sigma v$$

With the singular values on the diagonal of a diagonal matrix Σ and the corresponding singular vectors forming the columns of two orthogonal matrices U and V, we have

$$AV = U\Sigma,\ A'U = V\Sigma$$

Since U and V are orthogonal, this becomes the singular value decomposition

$$A = U\Sigma V'$$

The full singular value decomposition of an m-by-n matrix involves an m-by-m U, an m-by-n Σ, and an n-by-n V. In other words, U and V are both square and Σ is the same size as A. If A has many more rows than columns, the resulting U can be quite large, but most of its columns are multiplied by zeros in Σ. In this situation, the economy sized decomposition saves both time and storage by producing an m-by-n U, an n-by-n Σ and the same V.

The eigenvalue decomposition is the appropriate tool for analyzing a matrix when it represents a mapping from a vector space into itself, as it does for an ordinary differential equation. On the other hand, the singular value decomposition is the appropriate tool for analyzing a mapping from one vector space into another vector space, possibly with a different dimension. Most systems of simultaneous linear equations fall into this second category.

If *A* is square, symmetric, and positive definite, then its eigenvalue and singular value decompositions are the same. But, as *A* departs from symmetry and positive definiteness, the difference between the two decompositions increases. In particular, the singular value decomposition of a real matrix is always real, but the eigenvalue decomposition of a real, nonsymmetric matrix might be complex.

```
>> A =[9    4
       6    8
       2    7];
>> [U,S,V] = svd(A)     %the full singular value decomposition
U =
    -0.6105    0.7174    0.3355
    -0.6646   -0.2336   -0.7098
    -0.4308   -0.6563    0.6194
S =
   14.9359         0
         0    5.1883
         0         0
V =
    -0.6925    0.7214
    -0.7214   -0.6925
```

You can verify that *U*S*V'* is equal to *A* to within roundoff error.

```
>> [U,S,V] = svd(A,0)     %the economy size decomposition is only slightly smaller.
U =
    -0.6105    0.7174
    -0.6646   -0.2336
    -0.4308   -0.6563
S =
   14.9359         0
         0    5.1883
V =
    -0.6925    0.7214
    -0.7214   -0.6925
```

Again, *U*S*V'* is equal to *A* to within roundoff error.

✧ MATLAB

```
>> A = round(10*randn(3))
```

```
A =

    -4      3     12
   -17    -11      0
     1     12      3
>> p_A = poly(A)      % computes the coefficients of the characteristic polynomial of A
ans =
  1.0e+003 *
    0.0010    0.0120    0.0380    2.0310
>> r=roots(p_A)
r =
 -16.8781
   2.4391 +10.6951i
   2.4391 -10.6951i
```

We usually use eig to compute the eigenvalues of a matrix directly.