第七讲 Matlab矩阵分析与处理

范 琛

华东理工大学 自动化系

2020.12.03

第七讲 主要内容

- 特殊矩阵
- 矩阵结构变换
- 矩阵求逆与线性方程组求解
- 矩阵求值
- 矩阵的特征值与特征向量
- 矩阵的超越函数

1、特殊矩阵

（1）通用的特殊矩阵

常用的产生通用特殊矩阵的函数有：

- zeros：产生全0矩阵(零矩阵)
- ones：产生全1矩阵(幺矩阵)
- eye：产生单位矩阵
- rand：产生0～1间均匀分布的随机矩阵
- randn：产生均值为0，方差为1的标准正态分布随机矩阵

1、特殊矩阵

（1）通用的特殊矩阵

- zeros(m)建立一个m×m零矩阵；
- zeros(m,n)建立一个m×n零矩阵。
- 可以用zeros(size(A))建立一个与矩阵A同样大小零矩阵。

1、特殊矩阵

（1）通用的特殊矩阵

```
>> zeros(3)
ans =

     0     0     0
     0     0     0
     0     0     0

>> zeros(3,2)
ans =

     0     0
     0     0
     0     0
```

```
>> A=[1 2 3;4 5 6]

A =

     1     2     3
     4     5     6

>> zeros(size(A))

ans =

     0     0     0
     0     0     0
```

1、特殊矩阵

- 在区间[20,50]内均匀分布的5阶随机矩阵
- 均值为0.6、方差为0.1的5阶正态分布随机矩阵

```
>> x=20+(50-20)*rand(5)

x =

   48.5039   42.8629   38.4630   32.1712   21.7367
   26.9342   33.6940   43.7581   48.0641   30.5860
   38.2053   20.5551   47.6544   47.5071   44.3950
   34.5795   44.6422   42.1462   32.3081   20.2958
   46.7390   33.3411   25.2880   46.8095   24.1667
```

```
>> y=0.6+sqrt(0.1)*randn(5)

y =

    0.4632    0.9766    0.5410    0.6360    0.6931
    0.0733    0.9760    0.8295    0.9373    0.1775
    0.6396    0.5881    0.4140    0.6187    0.8259
    0.6910    0.7035    1.2904    0.5698    1.1134
    0.2375    0.6552    0.5569    0.3368    0.3812
```

1、特殊矩阵

范得蒙德矩阵

- 范得蒙德(Vandermonde)矩阵最后一列全为1，倒数第二列为一个指定的向量，其他各列是其后列与倒数第二列的点乘积。
- 可以用一个指定向量生成一个范得蒙德矩阵。
- vander(V)生成以向量V为基础向量的范得蒙德矩阵。

```
>> A=vander([1; 2; 3; 5])
A =
     1     1     1     1
     8     4     2     1
    27     9     3     1
   125    25     5     1
```

1、特殊矩阵

希尔伯特矩阵

- 希尔伯特矩阵的每个元素是 $$h_{ij} = \frac{1}{i+j-1}$$
- 生成希尔伯特矩阵的函数是 hilb(n)；
- 希尔伯特矩阵是一种数学变换矩阵，正定，且高度病态（即，任何一个元素发生一点变动，整个矩阵的行列式的值和逆矩阵都会发生巨大变化），病态程度和阶数相关。
- 希尔伯特矩阵是一个条件数很差的矩阵，使用一般方法求逆会因为原始数据的微小扰动而产生不可靠的计算结果；
- MATLAB中，有一个专门求n阶的希尔伯特矩阵的逆函数 invhilb(n) 。

1、特殊矩阵

希尔伯特矩阵

例如，求4阶希尔伯特矩阵及其逆矩阵

```
>> format rat   %以有理形式输出
>> H=hilb(4)
H =
      1        1/2       1/3       1/4
      1/2      1/3       1/4       1/5
      1/3      1/4       1/5       1/6
      1/4      1/5       1/6       1/7
>> H=invhilb(4)
H =
     16       -120       240      -140
   -120       1200     -2700      1680
    240      -2700      6480     -4200
   -140       1680     -4200      2800
>> format short  %恢复默认输出格式
```

1、特殊矩阵

```
>> toeplitz(1:6)
ans =
     1     2     3     4     5     6
     2     1     2     3     4     5
     3     2     1     2     3     4
     4     3     2     1     2     3
     5     4     3     2     1     2
     6     5     4     3     2     1
```

托普利兹矩阵

- 托普利兹(Toeplitz)矩阵除第一行第一列外，其他每个元素都与左上角的元素相同。
- 生成托普利兹矩阵的函数是toeplitz(x,y)，它生成一个以x为第一列，y为第一行的托普利兹矩阵。
- 这里x, y均为向量，两者不必等长。
- toeplitz(x)用向量x生成一个对称的托普利兹矩阵。

1、特殊矩阵

伴随矩阵

$$p(x)=a_n x^n + a_{n-1}x^{n-1}+\cdots+a_1 x+a_0$$

$$A=\begin{bmatrix} -\frac{a_{n-1}}{a_n} & -\frac{a_{n-2}}{a_n} & -\frac{a_{n-3}}{a_n} & \cdots & -\frac{a_1}{a_n} & -\frac{a_0}{a_n} \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

称矩阵A为多项式的伴随矩阵。

例如，为了求多项式的x^3-7x+6的伴随矩阵：

```
>> p=[1,0,-7,6];
>> compan(p)

ans =

    0    7   -6
    1    0    0
    0    1    0
```

生成伴随矩阵的函数是compan(p)，其中p是一个多项式的系数向量，高次幂系数排在前，低次幂排在后。

1、特殊矩阵

帕斯卡矩阵

- 二次项$(x+y)^n$展开后的系数随n的增大组成一个三角形表，称为杨辉三角形。
- 由杨辉三角形表组成的矩阵称为帕斯卡(Pascal)矩阵。
- 函数pascal(n)生成一个n阶帕斯卡矩阵。

例2-7 求$(x+y)^5$的展开式

```
>> pascal(6)
ans =

     1     1     1     1     1     1
     1     2     3     4     5     6
     1     3     6    10    15    21
     1     4    10    20    35    56
     1     5    15    35    70   126
     1     6    21    56   126   252
```

2、矩阵结构变换

（1）对角阵

只有对角线上有非0元素的矩阵称为对角矩阵；

对角线上的元素相等的对角矩阵称为数量矩阵；

对角线上的元素都为1的对角矩阵称为单位矩阵。

提取矩阵的对角线元素

- 设A为m×n矩阵，diag(A)函数用于提取矩阵A主对角线元素，产生一个具有min(m,n)个元素的列向量。
- diag(A)函数还有一种形式diag(A,k)，其功能是提取第k条对角线的元素。

```
>> A=[1 2 3;4 5 6]
A =
    1    2    3
    4    5    6

>> D=diag(A)
D =
    1
    5

>> D1=diag(A,1)
D1 =
    2
    6

>> D2=diag(A,-1)
D2 =
    4
```

2、矩阵结构变换

构造对角矩阵

- 设V为具有m个元素的向量，diag(V)将产生一个m×m对角矩阵，其主对角线元素即为向量V的元素。
- diag(V,k)，其功能是产生一个n×n(n=m+|k|)对角阵，其第k条对角线的元素即为向量V的元素。

```
>> diag ([1,2,-1,4])

ans =
    1    0    0    0
    0    2    0    0
    0    0   -1    0
    0    0    0    4
```

```
>> diag(1:3,-1)

ans =
    0    0    0    0
    1    0    0    0
    0    2    0    0
    0    0    3    0
```

2、矩阵结构变换

A =

17	0	1	0	15
23	5	7	14	16
4	0	13	0	22
10	12	19	21	3
11	1	25	2	19

```
>> D=diag(1:5);
```

```
>> D*A
```

ans =

17	0	1	0	15
46	10	14	28	32
12	0	39	0	66
40	48	76	84	12
55	5	125	10	95

```
>> A*D
```

ans =

17	0	3	0	75
23	10	21	56	80
4	0	39	0	110
10	24	57	84	15
11	2	75	8	95

D*A，对A的每行乘以一个指定常数；

A*D，对A的每列乘以一个指定常数；

2、矩阵结构变换

（2）三角矩阵

三角矩阵分为上三角阵和下三角阵：

- 上三角矩阵的函数是triu(A)，另一种形式triu(A,k)，其功能是求矩阵A的第k条对角线以上的元素。
- 下三角矩阵的函数是tril(A)和tril(A,k) 。

```
A =
    1    2    3    4    5
   11   22   33   44   55
   11   12   13   14   15
   21   22   23   24   25
   31   32   33   34   35
```

```
>> triu(A)

ans =
    1    2    3    4    5
    0   22   33   44   55
    0    0   13   14   15
    0    0    0   24   25
    0    0    0    0   35
```

```
>> triu(A,2)

ans =

    0    0    3    4    5
    0    0    0   44   55
    0    0    0    0   15
    0    0    0    0    0
    0    0    0    0    0
```

2、矩阵结构变换

（3）矩阵的转置

转置运算符是单撇号(‘)

A =

1	2	3	4	5
11	22	33	44	55
11	12	13	14	15
21	22	23	24	25
31	32	33	34	35

>> A'

ans =

1	11	11	21	31
2	22	12	22	32
3	33	13	23	33
4	44	14	24	34
5	55	15	25	35

2、矩阵结构变换

（4）矩阵的旋转

利用函数rot90(A,k)将矩阵A按逆时针方向旋转90^{o}的k倍，当k为1时可省略。

```
A =

    1    2    3    4    5
   11   22   33   44   55
   11   12   13   14   15
   21   22   23   24   25
   31   32   33   34   35
```

```
>> rot90(A)

ans =
    5   55   15   25   35
    4   44   14   24   34
    3   33   13   23   33
    2   22   12   22   32
    1   11   11   21   31
```

```
>> rot90(A,4)

ans =
    1    2    3    4    5
   11   22   33   44   55
   11   12   13   14   15
   21   22   23   24   25
   31   32   33   34   35
```

2、矩阵结构变换

（5）矩阵的翻转

- 对矩阵实施左右翻转是将原矩阵的第一列和最后一列调换，第二列和倒数第二列调换，…，依次类推，函数是fliplr(A)。
- 矩阵的上下翻转函数是flipud(A)。

```
A =

  1    2    3    4    5
  6    7    8    9    0
 21   22   23   24   25
 31   32   33   34   35
 41   42   43   44   45
```

```
>> fliplr(A)

ans =
  5    4    3    2    1
  0    9    8    7    6
 25   24   23   22   21
 35   34   33   32   31
 45   44   43   42   41
```

```
>> flipud(A)

ans =
 41   42   43   44   45
 31   32   33   34   35
 21   22   23   24   25
  6    7    8    9    0
  1    2    3    4    5
```

3、矩阵求逆与线性方程组求解

（1）矩阵的逆

- 对于一个方阵A，如果存在一个与其同阶的方阵B，使得：A·B=B·A=I (I为单位矩阵)，则称B为A的逆矩阵，当然，A也是B的逆矩阵。
- 求方阵A的逆矩阵可调用函数inv(A)。

```
A =
    1   -1    1
    5   -4    3
    2    1    1
>> B=inv(A)
B =
  -1.4000    0.4000    0.2000
   0.2000   -0.2000    0.4000
   2.6000   -0.6000    0.2000
```

```
>> A*B

ans =

    1.0000   -0.0000    0.0000
    0.0000    1.0000    0.0000
    0.0000   -0.0000    1.0000
```

```
>> B*A

ans =

    1.0000    0.0000    0.0000
   -0.0000    1.0000         0
   -0.0000    0.0000    1.0000
```

3、矩阵求逆与线性方程组求解

（2）矩阵的伪逆

- 如果矩阵A不是一个方阵，或者A是一个非满秩的方阵时，矩阵A没有逆矩阵，但可以找到一个与A的转置矩阵A'同型的矩阵B，使得：A·B·A=A，B·A·B=B，此时称矩阵B为矩阵A的伪逆，也称为广义逆矩阵。
- 在MATLAB中，求一个矩阵伪逆的函数是pinv(A)。

```
A =
     3     1     1     1
     1     3     1     1
     1     1     3     1
>> B=pinv(A)
B =
    0.3929   -0.1071   -0.1071
   -0.1071    0.3929   -0.1071
   -0.1071   -0.1071    0.3929
    0.0357    0.0357    0.0357
```

```
>> A*B*A
ans =
    3.0000    1.0000    1.0000    1.0000
    1.0000    3.0000    1.0000    1.0000
    1.0000    1.0000    3.0000    1.0000

>> B*A*B
ans =
    0.3929   -0.1071   -0.1071
   -0.1071    0.3929   -0.1071
   -0.1071   -0.1071    0.3929
```

3、矩阵求逆与线性方程组求解

（3）用求逆矩阵的方法解线性方程组

$$\begin{cases} x+2y+3z=5 \\ x+4y+9z=-2 \\ x+8y+27z=6 \end{cases}$$

```
A =
     1     2     3
     1     4     9
     1     8    27

>> b=[5 -2 6]'
b =
     5
    -2
     6
```

```
>> x=inv(A)*b
x =
   23.0000
  -14.5000
    3.6667

>> X=A\b
X =
   23.0000
  -14.5000
    3.6667
```

4、矩阵求值

（1）方阵的行列式

把一个方阵看作一个行列式，并对其按行列式的规则求值，这个值就称为矩阵所对应的行列式的值。在MATLAB中，求方阵A所对应的行列式的值的函数是det(A)。

```
>> A=rand(5)
A =
    0.9501    0.7621    0.6154    0.4057    0.0579
    0.2311    0.4565    0.7919    0.9355    0.3529
    0.6068    0.0185    0.9218    0.9169    0.8132
    0.4860    0.8214    0.7382    0.4103    0.0099
    0.8913    0.4447    0.1763    0.8936    0.1389

>> B=det(A)
B =
   -0.0071
```

4、矩阵求值

（2）矩阵的秩

矩阵线性无关的行数与列数称为矩阵的秩。在MATLAB中，求矩阵秩的函数是rank(A)。

```
>> A=rand(5)
A =
    0.9501    0.7621    0.6154    0.4057    0.0579
    0.2311    0.4565    0.7919    0.9355    0.3529
    0.6068    0.0185    0.9218    0.9169    0.8132
    0.4860    0.8214    0.7382    0.4103    0.0099
    0.8913    0.4447    0.1763    0.8936    0.1389

>> B=det(A)
B =
  -0.0071
```

```
>> rank(A)

ans =

     5
```

4、矩阵求值

（3）矩阵的迹

矩阵的迹等于矩阵的对角线元素之和，也等于矩阵的特征值之和。在MATLAB中，求矩阵的迹的函数是trace(A)。

```
A =
    2     2     3
    4     5    -6
    7     8     9

>> trace(A)

ans =
   16
```

4、矩阵的特征值与特征向量

矩阵的特征值与特征向量

E=eig(A)：求矩阵A的全部特征值，构成向量E。

[V,D]=eig(A)：求矩阵A的全部特征值，构成对角阵D，并求A的特征向量构成V的列向量。

```
A =

    1.0000    1.0000    0.5000
    1.0000    1.0000    0.2500
    0.5000    0.2500    2.0000
```

```
>> [V,D]=eig(A)
V =
    0.7212    0.4443    0.5315
   -0.6863    0.5621    0.4615
   -0.0937   -0.6976    0.7103

D =
   -0.0166         0         0
         0    1.4801         0
         0         0    2.5365
```

一个矩阵的特征向量有无穷多个，eig函数只找出其中的n个，其他的特征向量均可由这n个特征向量的线性组合表示。

4、矩阵的特征值与特征向量

矩阵的特征值与特征向量

求解方程

$$3x^5-7x^4+5x^2+2x-18=0$$

用求特征值的方法解方程

```
>> p=[3 -7 0 5 2 -18];
>> A=compan(p);
>> x1=eig(A)

x1 =
   2.1837
   1.0000 + 1.0000i
   1.0000 - 1.0000i
  -0.9252 + 0.7197i
  -0.9252 - 0.7197i
```

直接求方程的根

```
>> x2=roots(p)

x2 =
   2.1837
   1.0000 + 1.0000i
   1.0000 - 1.0000i
  -0.9252 + 0.7197i
  -0.9252 - 0.7197i
```

5、矩阵的超越函数

Matlab提供了一些直接作用于矩阵的超越函数，这些函数都在上述内部函数名之后缀以m，并规定输入参数必须是方阵。

5、矩阵的超越函数

（1）矩阵平方根sqrtm

- 若A为实对称正定矩阵或Hermitian正定阵，则一定能算出它的平方根；
- 若矩阵A含有负的特征值，则其平方根会得到一个复矩阵；

```
A =
     4     2
     3     6
>> B=sqrtm(A)
B =
    1.9171    0.4652
    0.6978    2.3823
>> B*B
ans =
    4.0000    2.0000
    3.0000    6.0000
```

```
A =
     4     9
    16    25
>> eig(A)
ans =
   -1.4452
   30.4452
>> B=sqrtm(A)
B =
   0.9421 + 0.9969i   1.5572 - 0.3393i
   2.7683 - 0.6032i   4.5756 + 0.2053i
```

5、矩阵的超越函数

（2）矩阵对数logm

（3）矩阵指数expm

```
A =

    4     9
    1     5

>> L=logm(A)

L =
   1.0639    2.4308
   0.2701    1.3340

>> B=expm(L)

B =
   4.0000    9.0000
   1.0000    5.0000
```

5、矩阵的超越函数

（4）通用矩阵函数funm

funm(A,'fun')用来计算直接作用于矩阵A的由'fun'指定的超越函数值。

fun函数可以用于exp、log，但不能用于sqrt。

```
>> A=[2 -1;1 0]

A =
     2    -1
     1     0

>> funm(A,'exp')

ans =
    5.4366   -2.7183
    2.7183   -0.0000

>> expm(A)

ans =
    5.4366   -2.7183
    2.7183    0.0000
```

第八讲 主要内容

- 数据统计处理
- 数据插值
- 曲线拟合
- 多项式计算
- 数值微积分
- 非线性方程与最优化问题求解
- 常微分方程的数值求解

第八讲 主要内容

- 数据统计处理
- 数据插值
- 曲线拟合
- 多项式计算
- 数值微积分
- 非线性方程与最优化问题求解
- 常微分方程的数值求解

1、数据统计处理

(1) 最大值和最小值--向量

- y=max(X)：返回向量X的最大值存入y，如果X中包含复数元素，则按模取最大值
- [y,I]=max(X)：返回向量X的最大值存入y，最大值的序号存入I，如果X中包含复数元素，则按模取最大值
- 求向量X的最小值的函数是min(X)，用法和max(X)完全相同

1、数据统计处理

例：向量x的最大值

```
x=[-43,72,9,16,23,47];
y=max(x)        %求向量x中的最大值
[y,l]=max(x)   %求向量x中的最大值及其该元素的位置
```

1、数据统计处理

(1) 最大值和最小值--矩阵

- max(A)：返回一个行向量，向量的第i个元素是矩阵A的第i列上的最大值
- [Y,U]=max(A)：返回行向量Y和U，Y向量记录A的每列的最大值，U向量记录每列最大值的行号
- max(A,[],dim)：dim取1或2。dim取1时，该函数和max(A)完全相同；dim取2时，该函数返回一个列向量，其第i个元素是A矩阵的第i行上的最大值
- 求最小值的函数是min，其用法和max完全相同

1、数据统计处理

- 求整个矩阵的最大元素？
- 求整个矩阵的最小元素？

max(max(A))或者max(A(:))

min(min(A))或者min(A(:))

1、数据统计处理

例：

$$A=\begin{bmatrix} 13 & -56 & 78 \\ 25 & 63 & -235 \\ 78 & 25 & 563 \\ 1 & 0 & -1 \end{bmatrix}$$

A=[13,-56,78;25,63,-235;78,25,563;1,0,-1]

max(A,[],2) %求每行最大元素

min(A,[],2) %求每行最小元素

max(A) %求每列最大元素

min(A) %求每列最小元素

max(max(A)) %求整个矩阵的最大元素

min(min(A)) %求整个矩阵的最小元素

1、数据统计处理

练习：

（1）A = [-2+2i 4+i -1-3i], 求max(A)

（2）A = [1.7 1.2 1.5; 1.3 1.6 1.99], 求M = max(A,[],2)

（3）A = [1 9 -2; 8 4 -5]，求[M,I] = max(A)

解：（1）ans = 4.0000 + 1.0000i

（2）M = [1.7; 1.99]

（3）M = [8, 9, -2]; I = [2, 1, 1]

1、数据统计处理

(1) 最大值和最小值-两个向量或矩阵对应元素的比较

- 函数max和min还能对两个同型的向量或矩阵进行比较，调用格式为：
- U=max(A,B)：A,B是两个同型的向量或矩阵，结果U是与A,B同型的向量或矩阵，U的每个元素等于A,B对应元素的较大者
- U=max(A,n)：n是一个标量，结果U是与A同型的向量或矩阵，U的每个元素等于A对应元素和n中的较大者
- min函数的用法和max完全相同

1、数据统计处理

例： $x = \begin{bmatrix} 4 & 5 & 6 \\ 1 & 4 & 8 \end{bmatrix} \qquad y = \begin{bmatrix} 1 & 7 & 5 \\ 4 & 5 & 7 \end{bmatrix}$

```
x=[4,5,6;1,4,8]
y=[1,7,5;4,5,7]
p=max(x,y)
f=4.5
P=max(x,f)
```

```
p =
    4     7     6
    4     5     8

P =
    4.5000    5.0000    6.0000
    4.5000    4.5000    8.0000
```

1、数据统计处理

(2) 求平均值和中值—向量

- X是一个向量
- mean(X)：返回向量X的算术平均值
- median(X)：返回向量X的中值
- 当数据序列为奇数个时，是位于序列中间的值
- 当数据序列为偶数个时，是中间两个数的均值

1、数据统计处理

例：求向量y的平均值和中值

```
y=[9 -2 5 6 7 12];
mean(y)
median(y)
```

1、数据统计处理

(2) 求平均值和中值—矩阵

- mean(A)：返回一个行向量，其第i个元素是A的第i列的算术平均值
- median(A)：返回一个行向量，其第i个元素是A的第i列的中值

1、数据统计处理

(2) 求平均值和中值—矩阵

- mean(A,dim)：当dim为1时，该函数等同于mean(A)；当dim为2时，返回一个列向量，其第i个元素是A的第i行的算术平均值
- median(A,dim)：当dim为1时，该函数等同于median(A)；当dim为2时，返回一个列向量，其第i个元素是A的第i行的中值

1、数据统计处理

(3) 矩阵元素求和与求积

设X是一个向量，A是一个矩阵，函数的调用格式为：

- sum(X)：返回向量X各元素的和
- prod(X)：返回向量X各元素的乘积
- sum(A)：返回一个行向量，其第i个元素是A的第i列的元素和
- prod(A)：返回一个行向量，其第i个元素是A的第i列的元素乘积

1、数据统计处理

- sum(A,dim)：当dim为1时，该函数等同于sum(A)；当dim为2时，返回一个列向量，其第i个元素是A的第i行的各元素之和
- prod(A,dim)：当dim为1时，该函数等同于prod(A)；当dim为2时，返回一个列向量，其第i个元素是A的第i行的各元素乘积

1、数据统计处理

(4) 矩阵元素累加和与累乘积

设X是一个向量 $X=(x_1, x_2, \cdots, x_n)$

- 向量X累加和向量：$V=(\sum_{i=1}^{1} x_i, \sum_{i=1}^{2} x_i, \cdots, \sum_{i=1}^{n} x_i)$
- 向量X累乘积向量：$W=(\prod_{i=1}^{1} x_i, \prod_{i=1}^{2} x_i, \cdots \prod_{i=1}^{n} x_i)$

1、数据统计处理

(4) 矩阵元素累加和与累乘积

设X是一个向量，A是一个矩阵，函数的调用格式为：

- cumsum(X)：返回向量X累加和向量
- cumprod(X)：返回向量X累乘积向量
- cumsum(A)：返回一个矩阵，其第i列是A的第i列的累加和向量
- cumprod(A)：返回一个矩阵，其第i列是A的第i列的累乘积向量
- cumsum(A,dim) , cumprod(A,dim)

1、数据统计处理

程序如下：

```
A=[1 2 3;4 5 6]
cumsum(A)
cumprod(A)
```

结果如下：

```
A =

     1     2     3
     4     5     6

ans =

     1     2     3
     5     7     9

ans =

     1     2     3
     4    10    18
```

1、数据统计处理

(5) 标准方差

- 对于向量X，std(X)返回一个标准方差
- 对于矩阵A，std(A)返回一个行向量，它的各个元素便是矩阵A各列或各行的标准方差

$$S_1=\sqrt{\frac{1}{N-1}\sum_{i=1}^{n}(x_i-\overline{x})^2}\text{ , }S_2=\sqrt{\frac{1}{N}\sum_{i=1}^{n}(x_i-\overline{x})^2}\text{ , }\overline{x}=\frac{1}{N}\sum_{i=1}^{n}x_i$$

1、数据统计处理

- std函数的一般调用格式为：Y=std(A,flag,dim)
- dim取1或2：当dim=1时，求各列元素的标准方差；当dim=2时，则求各行元素的标准方差
- flag取0或1：当flag=0时，按S1所列公式计算标准方差，当flag=1时，按S2所列公式计算标准方差
- 缺省flag=0，dim=1

$$S_1=\sqrt{\frac{1}{N-1}\sum_{i=1}^{n}(x_i-\overline{x})^2}\text{，}S_2=\sqrt{\frac{1}{N}\sum_{i=1}^{n}(x_i-\overline{x})^2}\text{，}\overline{x}=\frac{1}{N}\sum_{i=1}^{n}x_i$$

1、数据统计处理

例：对二维矩阵x，从不同维方向求出其标准方差

```
>> x=[4 5 6;1 4 8]
x =
     4     5     6
     1     4     8
>> y1=std(x,0,1)
y1 =
    2.1213    0.7071    1.4142
>> y2=std(x,1,1)
y2 =
    1.5000    0.5000    1.0000
```

```
>> y3=std(x,0,2)
y3 =
    1.0000
    3.5119
>> y4=std(x,1,2)
y4 =
    0.8165
    2.8674
```

1、数据统计处理

(6) 相关系数 $$r=\frac{\sum(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum(x_i-\bar{x})^2\sum(y_i-\bar{y})^2}}$$

- corrcoef(X)：返回从矩阵X形成的一个相关系数矩阵，它把矩阵X的每列作为一个变量，然后求它们的相关系数
- corrcoef(X,Y)：在这里，X,Y是向量，它们与corrcoef([X,Y])的作用一样

1、数据统计处理

例：生成满足正态分布的10000*5的随机矩阵，然后求各列元素的均值和标准方差，再求这5列随机数据的相关系数矩阵

```
X=randn(10000,5);
M=mean(X)
D=std(X)
R=corrcoef(X)
```

```
M =
    0.0011    0.0066    0.0009    0.0264    0.0101
D =
    1.0011    1.0036    1.0049    1.0058    1.0061
R =
    1.0000    0.0119    0.0051   -0.0114   -0.0011
    0.0119    1.0000    0.0093   -0.0012    0.0071
    0.0051    0.0093    1.0000    0.0048    0.0095
   -0.0114   -0.0012    0.0048    1.0000   -0.0017
   -0.0011    0.0071    0.0095   -0.0017    1.0000
```

1、数据统计处理

(7) 元素排序

- 排序函数sort(A)返回一个对X中的元素按升序排列的新向量
- [Y,I]=sort(A,dim,mode)
- 其中dim指明对A的列还是行进行排序。若dim=1，则按列排；若dim=2，则按行排
- mode指明升序还是降序，若ascend则按升序，若取descend，则按降序
- Y是排序后的矩阵，而I记录Y中的元素在A中位置

1、数据统计处理

例：对矩阵排序

$$A=\begin{bmatrix} 1 & -8 & 5 \\ 4 & 12 & 6 \\ 13 & 7 & -13 \end{bmatrix}$$

```
A=[1 -8 5;4 12 6;13 7 -13]
sort(A)
sort(A,2,'descend')
[X,I]=sort(A)
```

```
ans =
     1    -8   -13
     4     7     5
    13    12     6
ans =
     5     1    -8
    12     6     4
    13     7   -13
```

```
X =
     1    -8   -13
     4     7     5
    13    12     6
I =
     1     1     3
     2     3     1
     3     2     2
```