Matlab语言及其应用

第2讲

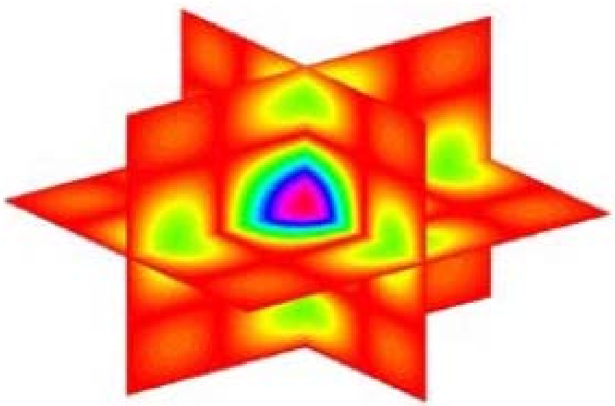
第2讲 矩阵与数组

Matlab数据类型非常丰富，除数值型、字符型等基本数据类型外，还有结构体、单元等更为复杂的数据类型。

各种数据类型都以矩阵形式存在，矩阵(数组)是Matlab最基本的数据对象，并且矩阵的运算是定义在复数域上的。

2 矩阵与数组

- 数组的概念
- 创建一维数组
- 创建二维数组
- 多维数组
- 数组元素的标识与索引
- 矩阵代数
- 数据统计与分析
- 数组运算



2.1 数组(array)的概念

数组定义：

按行(row)和列(column)顺序排列的实数或复数的有序集，被称为数组。

数组 (array)	大小(size)
$a=\begin{bmatrix}1 & 2\\ 3 & 4\\ 5 & 6\end{bmatrix}$	3×2
$b=\begin{bmatrix}1 & 2 & 3 & 4\end{bmatrix}$	1×4
$c=\begin{bmatrix}1\\ 2\\ 3\end{bmatrix}$	3×1

行向量

列向量

2.1 数组(array)的概念

- 数组的分类
  - 一维数组，也称为向量(vector)。
    - 行向量(row vector)、列向量(column vector)。
  - 二维数组(矩阵matrix)。
  - 多维数组。

* 有效矩阵：每行元素的个数必须相同，每列元素的个数也必须相同。

2.2 创建一维数组

- 第一种方法：使用方括号“[ ]”操作符

【例2-1】创建数组(行向量)a=[1 3 pi 3+5i]

```
>>a=[1 3 pi 3+5*i]  %or a=[1, 3, pi, 3+5*i]
a= 1.0000     3.0000     3.1416      3.0000 + 5.0000i
```

所有的向量元素必须在操作符“[ ]”之内；

向量元素间用空格或英文的逗点“,”分开。

2.2 创建一维数组

等差或等比数组

$$\begin{bmatrix}1 & 3 & 5 & 7 & 9\end{bmatrix}$$

$$\begin{bmatrix}1 & 10 & 100 & 1000 & 10000\end{bmatrix}$$

- 第二种方法：使用冒号“:”操作符

【例2-2】创建以1~10顺序排列整数为元素的行向量b。

```
>>b=1:10
```

b=1 2 3 4 5 6 7 8 9 10

2.2 创建一维数组（续）

【例2-3】键入并执行c=1:2:10和d=1:2:9

```
>> c=1:2:10
    c=1 3 5 7 9
>>d=1:2:9
    d= 1 3 5 7 9
```

利用冒号“:”操作符创建行向量的基本语法格式:

x=Start:Increment:End

- Start表示新向量x的第一个元素;
- 新向量x的最后一个元素不能大于End；
- Increment可正可负，若负，则必须Start>End；若正，则必须Start<End，否则创建的为空向量。
- 若Increment=1,则可简写为：x=Start:End。

2.2 创建一维数组（续）

- 第三种方法：利用函数linspace

函数linspace的基本语法

x= linspace(x1, x2, n)

- 该函数生成一个由n个元素组成的行向量；
- x1为其第一个元素；
- x2为其最后一个元素；
- x1、x2之间元素的间隔=(x2-x1)/(n-1)。
- 如果忽略参数n，则系统默认生成100个元素的行向量。

【例2-4】键入并执行x= linspace(1,2,5)

x=1.0000 1.2500 1.5000 1.7500 2.0000

同学们可以在实验时察看x= linspace(1,2)执行结果。

2.2 创建一维数组（续）

- 第四种方法：利用函数logspace

  通过实验认识该函数的功能。

- 列向量的创建
  - 使用方括号“[ ]”操作符，使用分号“;”分割行。

  【例2-5】键入并执行x= [1; 2; 3]

  X=1
  2
  3

  - 使用冒号操作符

  【例2-6】键入并执行x= (1:3)’ % “ ’ ”表示矩阵的转置

2.3 创建二维数组

- **第一种方法：使用方括号“[ ]”操作符**

  使用规则

  - 数组元素必须在“[ ]”内键入；
  - 行与行之间须用分号“;” 间隔，也可以在分行处用回车键间隔；
  - 行内元素用空格或逗号“,”间隔。

- **第二种方法：函数方法**

  用于创建特殊矩阵。

2.3 创建二维数组（续）

【例3-1】键入并执行a2=[1 2 3;4 5 6;7 8 9]

```
a2=
    1  2  3
    4  5  6
    7  8  9
```

【例3-2】键入并执行a2=[1:3;4:6;7:9] %结果同上

【例3-3】由向量构成二维数组。

```
>>a=[1 2 3]; b=[2 3 4];
>>c=[a;b];
>>c1=[a b];
```

2.3 创建二维数组（续）

- **第二种方法：函数方法**

用于创建特殊矩阵：

如0矩阵、1矩阵、单位矩阵、随机矩阵等。

常用的函数有：

zeros：产生全0矩阵（0矩阵）。

ones：产生全1矩阵（1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。

2.3 创建二维数组（续）

利用函数创建随机矩阵

① 在区间[20,50]内均匀分布的5阶随机矩阵。

② 均值为0.6，方差为0.1的5阶正态分布随机矩阵。

命令如下：

```
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)
```

2.4 多维数组

- 多维数组的定义

在 MATLAB 的数据类型中，向量可视为一维数组，矩阵可视为二维数组，对于维数超过2的数组均可视为「多维数组」。

2.4 多维数组(续)

- 将两个二维（平面）数组叠在一起，就构成三维数组，**第三维称为「页」(Page)**，如下图所示：

页

(1,1)	(1,2)	(1,3)	(1,4)
(2,1)	(2,2)	(2,3)	(2,4)
(3,1)	(3,2)	(3,3)	(3,4)

页

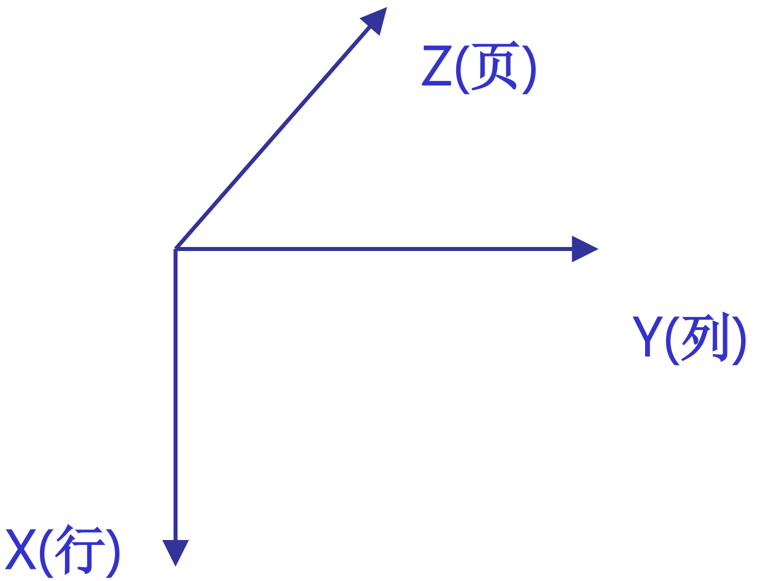
行

(1,1)	(1,2)	(1,3)	(1,4)
(2,1)	(2,2)	(2,3)	(2,4)
(3,1)	(3,2)	(3,3)	(3,4)

列

2.4 多维数组(续)

- 三维数组，可对应至一个X-Y-Z三维立体坐标，如下图所示：



2.4 多维数组(续)

- 多维数组的建立
  - 建立一个简单的多维数组，可直接由MATLAB 命令视窗内输入（使用“[ ]”操作符）
  - 例：由两个相同大小二维数组创建三维数组

```
A(:, :, 1) = [1 0 2 5; 4 1 8 7; 3 2 6 3];
A(:, :, 2) = [3 5 4 1; 2 6 2 1; 4 2 3 0]
```

```
A(:,:,1) =
    1    0    2    5
    4    1    8    7
    3    2    6    3
```

```
A(:,:,2) =
    3    5    4    1
    2    6    2    1
    4    2    3    0
```

2.4 多维数组(续)

执行命令：whos A，得到如下结果：

Name	Size	Bytes	Class
A	3x4x2	192	double array

Grand total is 24 elements using 192 bytes

2.5 数组元素的标识与索引

- **数组元素的标识**

数组中的任何一个数都被称为这个数组的元素，由其所在的行和列标识，这个标识也称为数组元素的下标或索引。Matlab将标量视为1×1的数组。

对m行、n列的2维数组a：

计为m×n的数组a;

*行标识、列标识均从1开始;

行标识从上到下递增;

列标识从左到右递增。

2.5 数组元素的标识与索引

- 数组元素的标识
  - “全下标（index）”标识

每一维对应一个下标。

- 如对于二维数组，用“行下标和列下标”标识数组的元素，a(2,3)就表示二维数组a的“第2行第3列”的元素。
- 对于一维数组，用一个下标即可，b(2)表示一维数组b的第2个元素，无论b是行向量还是列向量。

a=

17	24	1	8	15
23	5	7	14	16
4	6	13	20	22
10	12	19	2	9

a(3, 4)=20 row is first

2.5 数组元素的标识与索引

数组元素的标识

- **“单下标”（linear index）标识**

  所谓“单下标”标识就是用一个下标来表明元素在数组的位置。

  - 对于二维数组，“单下标”编号：设想把二维数组的所有列，按先后顺序首尾相接排成“一维长列”，然后自上往下对元素位置执行编号。

- **两种“下标”标识的变换：sub2ind、ind2sub**

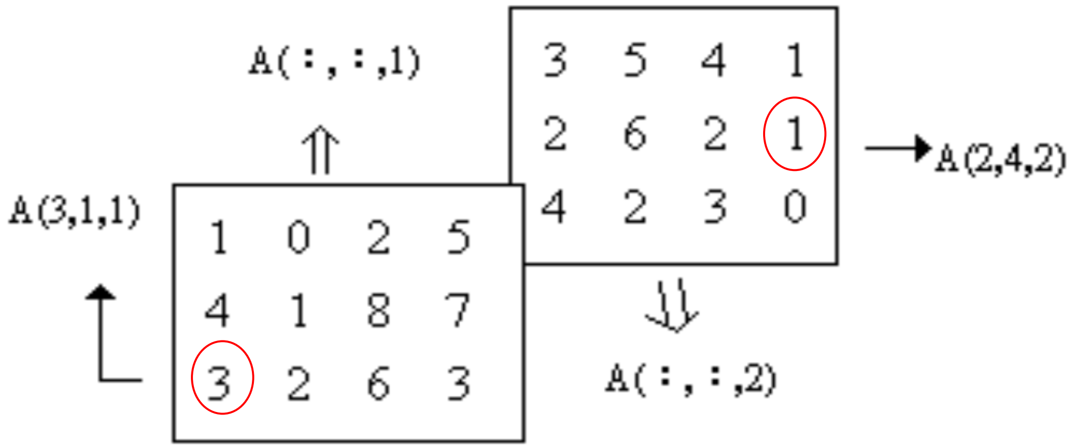
a=

17	24	1	8	15
23	5	7	14	16
4	6	13	20	22
10	12	19	2	9

a(15)=34

2.5 数组元素的标识与索引

- 三维数组元素的标识：可以(行、列、页)来确定。
- 以维数为3×4×2的三维数组为例，其标识方式如下图所示：



- *数组A是三维数组，其中A(: ,: ,1)代表第一页的二维数组，A(: ,: ,2)代表第二页的二维数组。*

2.5 数组元素的标识与索引

■ 全下标到单下标的转换

【例4-5】 sub2ind函数-双下标转换为单下标

```
>>A = [17 24 1 8; 2 22 7 14; 4 6 13 20];
>>A(:,:,2) = A - 10
>>A(2,1,2)
>>sub2ind(size(A),2,1,2)
>>A(14)
```

2.5 数组元素的标识与索引

- 单下标到全下标的转换

【例4-6】 ind2sub函数-双下标转换为单下标

```
>>b = zeros(3);
>>b(:) = 1:9
>>IND = [3 4 5 6]
>>[I,J] = ind2sub(size(b),IND)
```

2.5 数组元素的标识与索引

【例4-1】单下标的使用

```
>>a=zeros(2, 5);
>>a(:)=-4:5
  a =
     -4   -2    0    2    4
     -3   -1    1    3    5
```

注意数组的排列顺序。

2.5 数组元素的标识与索引

- 元素与子数组的索引与赋值

【例4-3】一维数组元素与子数组的索引与赋值

```
>>a=linspace(1,10,5)
    a =
        1.0000    3.2500    5.5000    7.7500   10.0000
>>a(3)      %索引a的第3个元素
    ans =
        5.5000
>>a([1 2 5])   %索引a的第1、2、5个元素组成的子数组
    ans =
        1.0000    3.2500   10.0000
```

2.5 数组元素的标识与索引

```
>>a(1:3)   %索引前3个元素组成的子数组
  ans =
    1.0000   3.2500   5.5000
>>a(3:-1:1)  %由前3个元素倒序构成的子数组
  ans =
    5.5000   3.2500   1.0000
>>a(3:end)
  ans =
    5.5000   7.7500   10.0000
>>a(3:end-1)
  ans =
    5.5000   7.7500
```

- %第3个及其后所有元素构成的子数组
- 函数end作为参数使用，返回最后一个元素的下标

2.5 数组元素的标识与索引

```
>>a([1 2 3 5 5 3 2 1])
  ans =
     1.0000    3.2500    5.5000   10.0000   10.0000
      5.5000    3.2500    1.0000
```

- 数组元素可以被任意重复访问，构成长度大于原数组的新数组。

```
>>a(6)
  ??? Index exceeds matrix dimensions.
```

- 下标值超出了数组的维数，导致错误

```
>>a(2.1)
  ??? Subscript indices must either be real positive
      integers or logicals.
```

- 下标值只能取正整数或逻辑值

2.5 数组元素的标识与索引

```
>>a(3)=0          %修改数组a的第3元素值为0
    a =
        1.0000    3.2500    0    7.7500   10.0000
>>a([2 5])=[1 1]
    a =
        1.0000    1.0000    0    7.7500   1.0000
```

- 可以修改指定数组元素的值
- 一次可以修改多个数组元素的值
- 要修改的数组元素的个数应与送入数组的元素个数相同

2.5 数组元素的标识与索引

【例4-3】二维数组元素与子数组的索引与赋值

```
>>a=zeros(2, 4) %创建2x4的全0数组
    a =
        0     0     0     0
        0     0     0     0
>>a(:)=1:8
    a =
        1     3     5     7
        2     4     6     8
>>a([2 5 8])    %单下标方式索引多个元素
    ans =
        2     5     8
```

🔔注意元素的排列顺序

2.5 数组元素的标识与索引

```
>> a ([2 5 8]) =[10 20 30]
   a =
      1    3   20    7
     10    4    6   30
>>a (:,[2 3])=ones(2)  %双下标方式索引并修改
   a =
      1    1    1    7
     10    1    1   30
```

- 二维数组可以“单下标”方式或“全下标”方式访问、赋值；
- “单下标”方式赋值时，等号两边涉及的元素个数必须相等；
- “全下标”方式赋值时，等号右边数组的大小必须等于原数组中涉及元素构成的子数组的大小。

2.5 数组元素的标识与索引

```
>>a(:,end)
ans =
     7
    30

>>a(:,end-1)
ans =
     1
     1

>>a(:, end:-1:3)
ans =
     7     1
    30     1

>>a(end,:)
ans =
    10     1     1    30

>>a(end,[2:4])
ans =
     1     1    30

>>a([4 6])=6:7
a =
     1     1     1     7
    10     6     7    30

>>a(end,[2:end-1])
```

What is the result?

2.6 矩阵代数

- 有一维是0的数组即为空数组
- 空数组不占据存储空间
- 最简单的空数组：0 x 0的矩阵
- 复杂的空数组：0 x 5 or 10 x 0

例如：>>a=[]; b=zeros(0,5);

查看空数组：>>a, b   % or whos a b

* 空数组并非全0数组

2.6 矩阵代数

- 数组维数的减小
  - 删除数组的某列和行

```
>>a = ones(4), a(:,2)=[]
```

  - 删除(2-D、3-D)数组的单个元素
    - 使用“全下标”方式，不能删除单个元素

```
>>a(1, 2)=[]   %系统会警告信息
```

    - 使用“单下标”可以删除单个元素

```
>>a(2:4)=[]   %数组a将变为向量
```

  - 使用“[]”同样可以减小<u>字符数组</u>的维数

2.6 矩阵代数

- 矩阵的转置(A'和A.')
  - 实数矩阵的转置

```
>>a =[ 1, 2, 3 ; 4, 5, 6];
>>b=a'
 b =   1    4
       2    5
       3    6
```

```
>>b=a.'
 b =   1    4
       2    5
       3    6
```

2.6 矩阵代数

■ 复数矩阵的转置

```
>>a =[ 1, 2+i, 3 ; 4, 5-i, 6];
>>b=a'
 b = 1.0000              4.0000
     2.0000 - 1.0000i    5.0000 + 1.0000i
     3.0000              6.0000

>>b=a.'
 b = 1.0000              4.0000
     2.0000 + 1.0000i    5.0000 - 1.0000i
     3.0000              6.0000
```

2.6 矩阵代数

【例3-5】 reshape函数

```
>>a=-4:4
  a=
    -4 -3 -2 -1 0 1 2 3 4
>>b=reshape(a, 3, 3)
  b=
  -4  -1  2
  -3   0  3
  -2   1  4
```

- ☞ 数组元素的排列顺序，从上到下按列排列，先排第一列，然后第二列，…
- ☞ 要求数组的元素总数不变。

2.6 矩阵代数

矩阵操作函数**size、length**

```
>>A=[1 4 7 10;
     2 5 8 11;
     3 6 9 12]
%测定矩阵A的行列数
>>size(A)
%测定矩阵A的长度
  即行列数中的最大值
>>length(A)
```

```
A =
    1    4    7    10
    2    5    8    11
    3    6    9    12
ans =
    3    4
ans =
    4
```

2.6 矩阵代数

矩阵操作函数**triu**

```
>>A=[1 2 3;5 4 6;9 7 8]
%取上三角矩阵
>>E=triu(A)
%提取矩阵A的第k条对角线以上的元素
>>F=triu(A,1)
% k>0对应主对角线以上，
  k<0对应主对角线以下
```

```
A =
     1     2     3
     5     4     6
     9     7     8
E =
     1     2     3
     0     4     6
     0     0     8
F =
     0     2     3
     0     0     6
     0     0     0
```

2.6 矩阵代数

矩阵操作函数**tril**

```
>>A=[1 2 3;5 4 6;9 7 8]
%取下三角矩阵
>>E=tril(A)
%提取矩阵A的第k条对角线以下的元素
>>F=tril(A,-1)
% k>0对应主对角线以上，
  k<0对应主对角线以下
```

```
A =
     1     2     3
     5     4     6
     9     7     8
E =
     1     0     0
     5     4     0
     9     7     8
F =
     0     0     0
     5     0     0
     9     7     0
```

2.6 矩阵代数

提取矩阵对角线元素

（1）diag(A)提取矩阵的对角线元素

例如：

```
>>A = [1,2,3;4,5,6];
>>D = diag(A)
D =
      1
      5
```

（2）diag(A,k)提取第k条对角线的元素。

例如：

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

2.6 矩阵代数

构造对角矩阵

如果V是一个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。

例如：

```
>> diag([1,2,-1,4])
 ans =
   1    0    0    0
   0    2    0    0
   0    0   -1    0
   0    0    0    4
```

例如：

```
>>diag(1:3,-1)
ans =
   0    0    0    0
   1    0    0    0
   0    2    0    0
   0    0    3    0
```

2.6 矩阵代数

建立一个5×5矩阵A，然后将A的第一行元素乘以1，第二行乘以2，…第五行乘以5。

解：用一个对角矩阵左乘一个矩阵时，相当于用对角阵的第一个元素乘以该矩阵的第一行，依次类推。

```
>>A = ones(5);
>>D = diag(1:5);
>>D * A
ans =
    1    1    1    1    1
    2    2    2    2    2
    3    3    3    3    3
    4    4    4    4    4
    5    5    5    5    5
```

2.6 矩阵代数

在Matlab中，使用函数det(A)计算行列式值。

例如：

```
>>A = rand(4)
  A =
     0.9501    0.7621    0.6154    0.4057
     0.2311    0.4565    0.7919    0.9355
     0.6068    0.0185    0.9218    0.9169
     0.4860    0.8214    0.7382    0.4103
>>B = det(A)
  B =
    -0.1520
```

2.6 矩阵代数

矩阵的逆

求方阵A的逆矩阵调用函数inv(A)。

$$A \cdot B = B \cdot A = I$$

例5.18 求方阵A的逆矩阵，且验证。

```
>>A = [1,-1,1;5,-4,3;2,1,1];
>>B = inv(A);
>>A*B
ans =
   1.0000        0         0
  -0.0000   1.0000         0
  -0.0000        0    1.0000
```

2.6 矩阵代数

将包含n个未知数，由n个方程构成的线性方程组表示为：

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \cdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases}$$

其矩阵表示形式为： $Ax = b$

其中：

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}, x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

$$x = A^{-1}b$$

2.6 矩阵代数

矩阵的特征值和特征向量

[V,D]=eig(A)：返回矩阵V和D，V矩阵的各列为矩阵A的特征向量，D矩阵为对角矩阵，对角线上的各元素为矩阵A的特征值，与特征向量一一对应；

```
>>a = [ 3, 2, 3 ; 4, 8, 6 ; 7, 5, 9 ];
>>[v,d]=eig(a)
```

v =

-0.2615	-0.7029	0.1291
-0.6520	-0.1777	-0.8195
-0.7117	0.6887	0.5584

d =

16.1528	0	0
0	0.5660	0
0	0	3.2811

2.7 数组运算

- MATLAB数组支持线性代数中所有的矩阵运算。
- 建立特有的数组运算符，如：“.*”、“./”等。

MATLAB数组运算符列表

运算	运算符	含义说明
加	+	相应元素相加
减	-	相应元素相减
乘	*	矩阵乘法
点乘	.*	相应元素相乘
幂	^	矩阵幂运算
点幂	.^	相应元素进行幂运算
左除或右除	\或/	矩阵左除或右除
左点除或右点除	.\或./	A的元素被B的对应元素除

2.7 数组运算(续)

【例5-1】数组加减

```
>>a=zeros(2, 3);
>>a(:)=1:6;
>>b=a+2.5
b =
        3.5000    5.5000    7.5000
        4.5000    6.5000    8.5000
>>c=b-a
   c =
        2.5000    2.5000    2.5000
        2.5000    2.5000    2.5000
```

2.7 数组运算(续)

- 矩阵乘法

(1)矩阵与矩阵相乘

```
>>a=[ 1 2 3; 4 5 6; 7 8 0 ];
>>b=[ 1; 2; 3 ];
>>c=a*b
```

(2)矩阵与常数相乘

```
>> a=[ -1; 0; 2 ];
>> b=pi*a
```

(3)矩阵的幂 注意矩阵必须是方阵

```
>> a=[ 1 2 3; 4 5 6; 7 8 0 ];
>> b=a^5
```

2.7 数组运算(续)

- **矩阵除法**

A/B — 矩阵右除，相当于 A*inv(B)

A\B — 矩阵左除，相当于 inv(A)*B

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

命令如下：

```
A = [1,2,3;1,4,9;1,8,27];
b = [5,-2,6]';
x = A\b   % x = inv(A)*b
```

2.7 数组运算(续)

- 数组除法

(1)矩阵与矩阵对应元素相乘

C(i,j) = A(i,j)*B(i,j)

```
>>a=[ 1 2 3; 4 5 6; 7 8  9 ];
>>b=[ 2 4 6; 1 3 5; 7 9 10 ];
>>c=a*b
c =
     2      8     18
     4     15     30
    49     72     90
```

```
>>c=a.*b
c =
    25     37     46
    55     85    109
    85    133    172
```

2.7 数组运算(续)

- 数组除法 🔔注意矩阵的size必须一致

(1)矩阵与矩阵对应元素相除

C=A./B —— 数组右除C(i,j) = A(i,j)/B(i,j)

C=A.\B —— 数组左除C(i,j) = B(i,j)/A(i,j)

(2)矩阵的点幂

A.^n —— A的每个元素自乘n次

A.^p —— 对A各元素分别求非整数幂

例： >>A = [1 2 3]; B = [4 5 6];

>>X = A.^2

>>X = A.^B

2.7 数组运算(续)

【例5-2】点幂“.^”举例

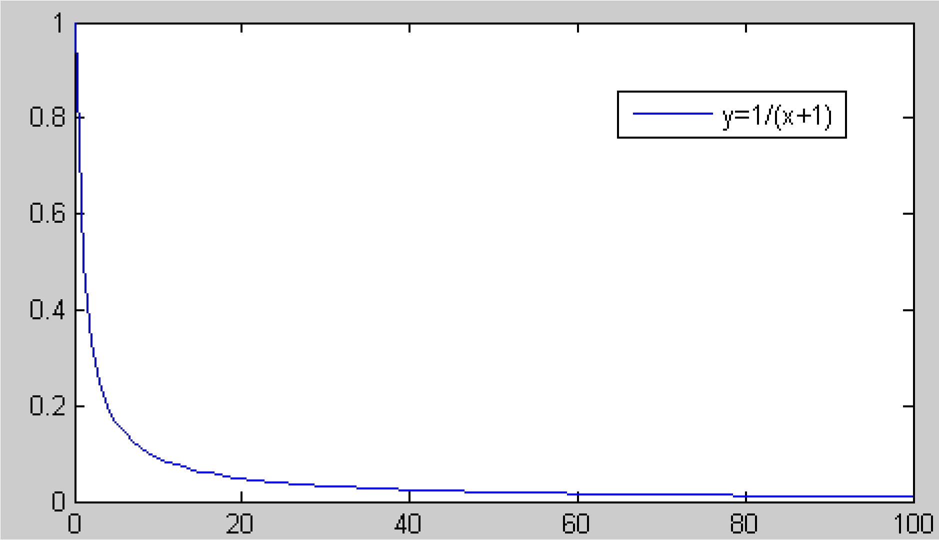
```
>>a=1:6
  a =
      1   2   3   4   5  6
>>b=[1 3 5;2 4 6]
  b =
      1     3     5
      2     4     6
```

```
>>a=a.^2
a =
   1   4   9   16   25  36
>>b=b.^2
b =
   1     9     25
   4     16    36
```

2.7 数组运算(续)

【例5-2】画出y=1/(x+1)的函数曲线，x∈[0, 100]。

```
x=0:100;
y=1./(x+1);
```



2.7 数组运算(续)

【例5-2】生成一个信号:

x=sin(2*pi*t)+sin(4*pi*t)

```
t = [0:199]./100;     %采样时间点
% 生成信号
x = sin(2*pi*t) + sin(4*pi*t);
```

2.8 数据统计与分析

- 圆整

fix(x)：向0取整

```
>> fix(1.9)

ans =

     1
```

```
>> fix(-1.9)

ans =

    -1
```

floor(x)：向-∞取整

```
>> floor(1.9)

ans =

     1
```

```
>> floor(-1.9)

ans =

    -2
```

2.8 数据统计与分析(续)

- 圆整

ceil(x)： 向+∞取整

```
>> ceil(1.9)

ans =

    2
```

```
>> ceil(-1.9)

ans =

    -1
```

round(x)： 向最近的整数取整，即四舍五入

```
>> round(1.9)

ans =

    2
```

```
>> round(-1.9)

ans =

    -2
```

2.8 数据统计与分析(续)

- 求模和求余

注意：

mod(x,y)的结果和y的符号相同

mod(x,y)： 求模

mod(x,y) =x - n.*y, n = floor(x./y)

```
>> mod(8.2,-3)
ans =
    -0.8000
>> mod(8.2,3)
ans =
    2.2000
```

```
>> mod(-8.2,-3)
ans =
    -2.2000
>> mod(-8.2,3)
ans =
    0.8000
```

2.8 数据统计与分析(续)

- 求模和求余

注意：

rem(x,y)的结果和x的符号相同

rem(x,y)：求余数

rem(x,y) =x - n.*y, n = fix(x./y)

```
>> rem(8.2,-3)
    ans =
          2.2000
>> rem(8.2,3)
    ans =
          2.2000
```

```
>> rem(-8.2,-3)
    ans =
          -2.2000
>> rem(-8.2,3)
    ans =
          -2.2000
```

2.8 数据统计与分析(续)

- 符号函数与求绝对值

sign(x)

x>0，则返回值为1

x<0，则返回值为-1

x=0，则返回为0

- 求绝对值

abs(x)

2.8 数据统计与分析(续)

- 最大值和最小值

求最大值和最小值的函数分别为max和min。

(1) 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；

(2) [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；

(3) 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。

min函数和max函数用法相同。

2.8 数据统计与分析(续)

【例7-2】求矩阵的最大值

```
>>x=[-43,72,9; 16,23,47];
>>y=max(x)    %求矩阵x中每列的最大值
y =
    16  72  47
>>[y,l]=max(x) %求矩阵x中每列的最大值及其该元素的位置
y =
   16  72  47
l =
    2  1  2
>>max(x, [],1), max(x, [],2) %求矩阵中每行的最大值
```

2.8 数据统计与分析(续)

- 求和

1、向量求和

sum(X)：返回向量X各元素的和。

2、矩阵求和

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。

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。

2.8 数据统计与分析(续)

- **求均值**

1、向量求均值

mean(X)：返回向量X的算术平均值。

2、矩阵求均值

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。

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。