

MATLAB统计分析与应用

——MATLAB基本操作

主讲人：谢中华

ssstudy.com
科学软件学习网

主要内容

- 变量的定义与数据类型
- 常用函数
- 数组运算
- MATLAB常用标点符号
- MATLAB常用快捷键和快捷命令

第一节 变量的定义与数据类型

一、变量的定义与赋值

1. 变量命名规则

- 可由任意的字母、数字或下划线组成，但必须以字母打头；
- 变量名区分字母大小写；
- 变量名不超过63个字符。

注：不要用内部函数名作为变量名。

2. 赋值语句

```
>> x = 1  
x =  
1  
>> y = 1+2+sqrt(9)  
y =  
6  
>> z = 'Hello World !!!'  
z =  
Hello World !!!
```

二、MATLAB中的常量

1. MATLAB中的特殊函数或常量列表

在变量名缺省的情况下，计算结果被赋给变量ans，ans是一个内部函数。

MATLAB中提供了一些特殊函数，它们的返回值是一些有用的常量。

MATLAB中的特殊函数或常量列表

特殊函数（常量）	意 义
ans	用于存储计算结果的默认变量名
pi	圆周率π (= 3.1415926...)
i或j	虚数单位 (sqrt(-1))
inf或Inf	无穷大 (∞) , 正数除以0的结果
NaN或nan	非数 (或不定量) , 0/0、inf/inf或inf-inf的结果
eps	浮点运算的相对精度, $\varepsilon = 2^{-52}$
realmin	最小的正浮点数 2^{-1022}
realmax	最大的正浮点数($2-\varepsilon$) 2^{1023}
version	MATLAB版本信息字符串, 例如7.14.0.739 (R2012a)

2. 清除变量和恢复内部函数

```
>> pi      % 查看圆周率的值  
>> pi = 1  % 对变量pi重新赋值  
>> clear pi % 清除变量pi  
>> pi  
ans = 3.1416
```

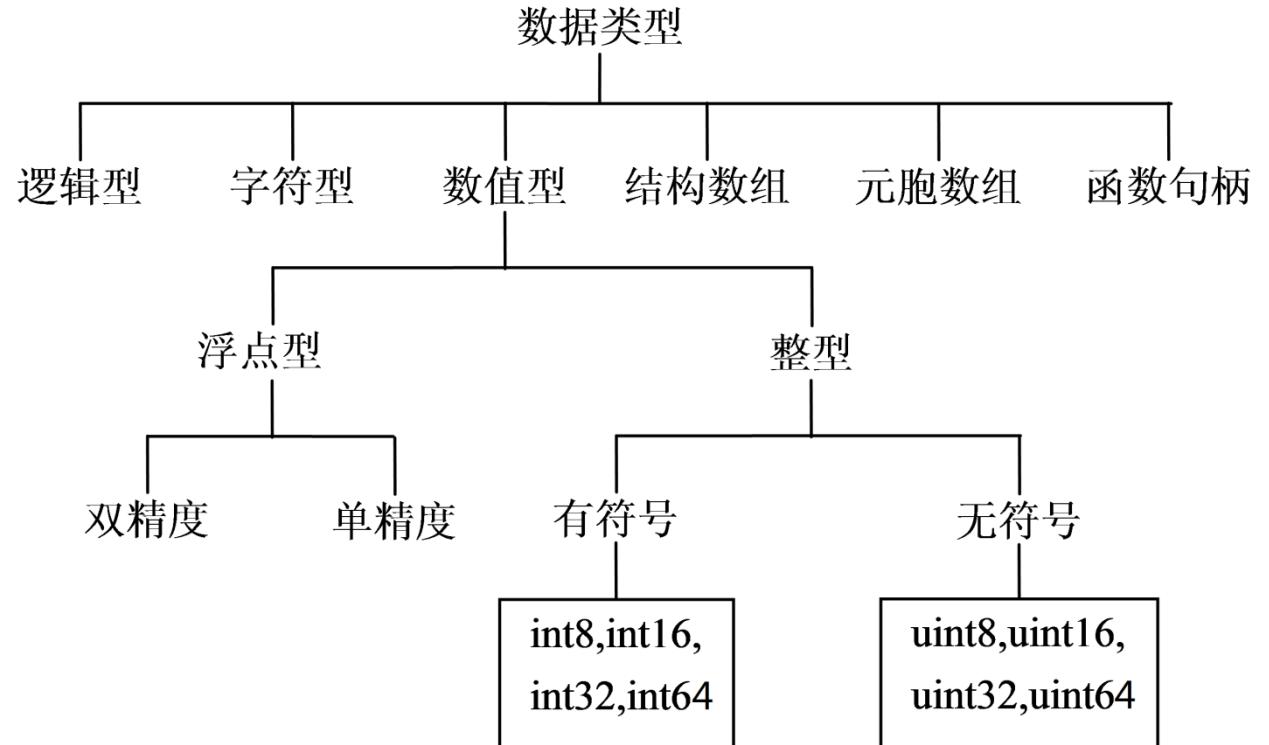
思考：如果用户对clear进行赋值，则clear函数失效，此时怎么清除变量呢？

三、MATLAB中的关键字

作为一种编程语言，MATLAB中为编程保留了一些关键字：break、case、catch、classdef、continue、else、elseif、end、for、function、global、if、otherwise、parfor、persistent、return、spmd、switch、try、while，这些关键字在程序编辑窗口中会以蓝色显示，它们是不能作为变量名的，否则会出现错误。

四、数据类型

MATLAB中有15种基本的数据类型，有逻辑型、字符型、整型、浮点型、结构数组、元胞数组以及函数句柄等。其中整型又分为有符号整型和无符号整型，8位整型、16位整型、32位整型和64位整型，浮点型又分为单精度浮点型和双精度浮点型。具体可以通过MATLAB中自带的isa函数查看。



五、数据输出格式

MATLAB中数值型数据的输出格式可以通

过 **format** 命令指定

格 式	说 明
format short	固定短格式，4位小数。例 3.1416.
format long	固定长格式，14至15位小数（双精度）；7位小数（单精度）。例 3.141592653589793.
format short e	浮点短格式，4位小数。例 3.1416e+000.
format long e	浮点长格式，14至15位小数（双精度）；7位小数（单精度）。例 3.141592653589793e+000.
format short g	最好的固定或浮点短格式，4位小数。例 3.1416.
format long g	最好的固定或浮点长格式，14至15位小数（双精度）；7位小数（单精度）。例 3.14159265358979.
format short eng	科学计数法短格式，4位小数，3位指数。例 3.1416e+000.
format long eng	科学计数法长格式，16位有效数字，3位指数。例 3.14159265358979e+000.
format +	以“+”号显示
format bank	固定的美元和美分格式。例 3.14.
format hex	十六进制格式。例 400921fb54442d18.
format rat	分式格式，分子分母取尽可能小的整数。例 355/113.

续表：

格 式	说 明
format compact	压缩格式（或紧凑格式），不显示空白行，比较紧凑。例 >> format compact >> pi ans = 3.141592653589793
format loose	自由格式（或宽松格式），显示空白行，比较宽松。例 >> format loose >> pi ans = 3.141592653589793

第二节 常用函数

函数名	说 明	函数名	说 明
abs	绝对值或复数的模	sqrt	平方根函数
exp	指数函数	log	自然对数
log2	以 2 为底的对数	log10	以 10 为底的对数
round	四舍五入到最接近的整数	ceil	向正无穷方向取整
floor	向负无穷方向取整	fix	向 0 零方向取整
rem	求余函数	mod	取模函数
sin	正弦函数	cos	余弦函数
tan	正切函数	cot	余切函数
asin	反正弦函数	acos	反余弦函数
atan	反正切函数	acot	反余切函数
real	求复数实部	imag	求复数虚部
angle	求相位角	conj	求共轭复数
mean	求均值	std	求标准差
max	求最大值	min	求最小值
var	求方差	cov	求协方差
corrcoef	求相关系数	range	求极差
sign	符号函数	plot	画线图

【例2.2-1】常用函数的用法举例

```
>> x = [1 -1.65 2.2 -3.1];
>> y1 = abs(x)
y1 =
    1.0000  1.6500  2.2000  3.1000
>> y2 = sin(x)
y2 =
    0.8415  -0.9969  0.8085  -0.0416
>> y3 = round(x)
y3 =
    1   -2    2   -3
>> y4 = floor(x)
y4 =
    1   -2    2   -4
```

```
>> y5 = ceil(x)  
y5 =  
    1   -1    3   -3
```

```
>> y6 = min(x)  
y6 =  
-3.1000
```

```
>> y7 = mean(x)  
y7 =  
-0.3875
```

```
>> y8 = range(x)  
y8 =  
5.3000
```

```
>> y9 = sign(x)  
y9 =    1   -1    1   -1
```

第三节 数组运算

一、定义向量

1. 逐个输入向量元素

$x = [x_1, x_2, x_3, \dots]$ % 定义行向量

$x = [x_1; x_2; x_3; \dots]$ % 定义列向量

【例2.3-1】 定义行向量 $x = [1 \ 0 \ 2 \ -3 \ 5]$ 。

```
>> x = [1,0,2,-3, 5] % 定义行向量
```

```
x = 1 0 2 -3 5
```

注：行向量各元素之间用逗号或空格分隔；

列向量各元素之间用分号分隔。

【例2.3-2】定义列向量 $y = [-1 \ 10 \ 3 \ -2 \ 7]^T$ 。

% 方式一

```
>> y = [-1; 10; 3; -2; 7] % 定义列向量
```

% 方式一

```
>> y = [-1 10 3 -2 7]' % 行向量转置定  
义列向量
```

注：Matlab中 x' 表示 x 的转置。

2. 规模化定义向量

【例2.3-3】通过冒号运算符构造等间隔向量。

$x = \text{初值} : \text{步长} : \text{终值}$

>> $x = 1:2:10$

$x =$

1 3 5 7 9

>> $y = 1:10$

$y =$

1 2 3 4 5 6 7 8 9 10

注：步长为1时可以省略。

【例2.3-4】调用linspace函数生成等间隔向量。

调用格式：

$x = \text{linspace}(\text{初值}, \text{终值}, \text{向量长度})$

```
>> x = linspace(1, 10, 10)
```

$x =$

1 2 3 4 5 6 7 8 9 10

二、定义矩阵

1. 按行方式输入矩阵元素

【例2.3-5】定义矩阵 $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ 。

```
>> A = [1, 2, 3; 4 5 6; 7 8, 9]
```

A =

1	2	3
4	5	6
7	8	9

2. 矩阵与向量的互相转换

➤ 矩阵转为向量

`x = A(:)` % 矩阵转为列向量

【例2.3-6】 将矩阵 $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ 转为向量。

```
>> A = [1, 2, 3; 4, 5, 6; 7, 8, 9];
```

```
>> x = A(:)
```

➤ 向量转为矩阵

`A = reshape(x, [m, n]) % 将向量x转为m行n列的矩阵`

行n列的矩阵

【例2.3-7】 定义长度为18的向量，将其转为3行6列的矩阵。

```
>> x = 1:18 ;
```

```
>> A = reshape(x, [3, 6])
```

3. 访问矩阵元素

➤ 双下标访问

$x = A(i, j)$ % 访问矩阵A的第i行第j列的元素

➤ 单下标访问

$x = A(k)$ % 访问矩阵A的第k个元素

注：单下标访问时相当于访问A所转成的向量的元素。

【例2.3-8】利用行标、列标和冒号运算符提取矩阵元素。

```
>> A = [1 2 3; 4 5 6; 7 8 9];
```

```
>> y1 = A(1, 2)
```

```
y1 = 2
```

```
>> y2 = A(2:3, 1:2)
```

```
y2 =
```

```
 4 5
```

```
 7 8
```

```
>> y3 = A(3:6)
```

```
y3 = 7 2 5 8
```

```
>> y4 = A(:, 1:2) % 提取A的前两列元素
```

```
y4 =
```

```
1 2
```

```
4 5
```

```
7 8
```

```
>> y5 = A(1, :) % 提取A的第一行元素
```

```
y5 =
```

```
1 2 3
```

4. 矩阵拼接

```
B = repmat(A, [m,n]) % 将矩阵A拼接为大矩阵
```

【例2.3-9】 通过矩阵拼接定义新的矩阵。

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

```
>> B = repmat(A, [2, 2])
```

```
B =
```

1	2	3	1	2	3
4	5	6	4	5	6
1	2	3	1	2	3
4	5	6	4	5	6

5. 定义字符矩阵

【例2.3-10】 定义字符型矩阵。

```
>> C = ['abc'; 'def'; 'ghi']
```

```
C =
```

```
abc
```

```
def
```

```
ghi
```

```
>> size(C) % 查看矩阵行数和列数
```

```
ans = 3 3
```

6. 定义复数矩阵

【例2. 3-11】定义复数矩阵。

```
>> x = 2i+5  
x =  
    5.0000 + 2.0000i  
>> A1 = [1 2 3; 4 5 6]*i+7  
A1 =  
    7.0000 + 1.0000i  7.0000 + 2.0000i  7.0000 + 3.0000i  
    7.0000 + 4.0000i  7.0000 + 5.0000i  7.0000 + 6.0000i  
>> a = [1 2; 3 4];  
>> b = [5 6; 7 8];  
>> A2 = complex(a,b)  
A2 =  
    1.0000 + 5.0000i  2.0000 + 6.0000i  
    3.0000 + 7.0000i  4.0000 + 8.0000i
```

7. 定义符号矩阵

【例2. 3-12】定义符号矩阵。

```
>> syms a b c d % 定义符号变量  
>> A1 = [a b; c d] % 用符号变量定义符号矩阵  
A1 = [ a, b]  
      [ c, d]  
  
>> A2 = [1 2 3; 4 5 6];  
>> A2 = sym(A2) % 把数值矩阵转为符号矩阵  
A2 = [ 1, 2, 3]  
      [ 4, 5, 6]
```

【例2.3-12续】定义符号矩阵。

```
>> A3 = sym('a%d%d', [3 4]) % 定义符
```

号矩阵

```
A3 =
```

```
[ a11, a12, a13, a14]
```

```
[ a21, a22, a23, a24]
```

```
[ a31, a32, a33, a34]
```

三、特殊矩阵

- 零矩阵 : zeros
- 一矩阵 : ones
- 单位阵 : eye
- 对角阵 : diag
- 随机阵 : rand
- 魔方阵 : magic

【例2.3-13】生成特殊矩阵。

```
>> A = zeros(3)  
>> B = ones(3,5)  
>> C = eye(3,5)  
>> D = diag([1 2 3])  
>> E = diag(D)  
>> F = rand(3)  
>> G = magic(3)
```

四、高维数组

【例2.3-14】通过直接赋值的方式定义3维数组。

```
>> x(1:2, 1:2, 1)=[1 2; 3 4];
```

```
>> x(1:2, 1:2, 2)=[5 6; 7 8];
```

```
x(:,:,1) =
```

```
1 2
```

```
3 4
```

```
x(:,:,2) =
```

```
5 6
```

```
7 8
```

【例2.3-15】利用cat函数定义3维数组。

```
>> A1 = [1 2; 3 4];  
>> A2 = [5 6; 7 8];  
>> A = cat(3, A1, A2)  
A(:,:,1) =  
     1     2  
     3     4  
A(:,:,2) =  
     5     6  
     7     8
```

【例2. 3-16】利用reshape函数定义3维数组。

```
>> x = reshape(1:12, [2, 2, 3])
```

```
x(:,:,1) =
```

```
1 3
```

```
2 4
```

```
x(:,:,2) =
```

```
5 7
```

```
6 8
```

```
x(:,:,3) =
```

```
9 11
```

```
10 12
```

【例2.3-17】利用repmat函数定义3维数组。

```
>> x = repmat([1 2; 3 4], [1 1 2])
```

```
x(:,:,1) =
```

```
1 2
```

```
3 4
```

```
x(:,:,2) =
```

```
1 2
```

```
3 4
```

五、定义元胞数组

【例2.3-18】直接赋值定义元胞数组。

```
>> c1 = {[1 2; 3 4], 'xiezhh', 10; [5 6 7], ...
    ['abc';'def'], 'I LOVE MATLAB'}
```

c1 =

[2x2 double] 'xiezhh' [10]

[1x3 double] [2x3 char] 'I LOVE MATLAB'

【例2.3-19】 利用cell函数定义元胞数组。

调用格式：

```
c = cell(n) >> c2 = cell(2,4)
```

`c = cell(m, n)` `c2 =`

`c = cell([m, n])`

```
c = cell(m, n, p,...)    >> c{2, 3} = [1 2 3]
```

c = cell([m n p ...]) c2 =

```
c = cell(size(A))
```

【例2.3-20】元胞数组的访问。

访问元胞数组C的第i行第j列的元胞，用命令C(i, j)，注意用的是圆括号；访问元胞数组C的第i行第j列的元胞里的元素，用命令C{i, j}，注意用的是花括号。celldisp函数可以显示元胞数组里的所有内容。

```
>> c = {[1 2], 'xie', 'xiezhh'; 'MATLAB', ...
[3 4; 5 6], 'I LOVE MATLAB'}
```

c =

```
[1x2 double]    'xie'        'xiezhh'  
'MATLAB'       [2x2 double]   'I LOVE MATLAB'
```

```
>> c(2, 2)
```

ans =

```
[2x2 double]
```

```
>> c{2, 2}
```

ans =

```
3 4  
5 6
```

```
>> c = {[1 2], 'xiezh'; 'MATLAB', [3 4; 5 6]};
```

```
>> celldisp(c)
```

c{1,1} =

1 2

c{2,1} =

MATLAB

c{1,2} =

xiezh

c{2,2} =

3 4

5 6

六、定义结构体数组

【例2.3-21】直接赋值定义结构体数组。

```
>> struct1(1).name = 'xiezh';  
>> struct1(2).name = 'heping';  
>> struct1(1).age = 31;  
>> struct1(2).age = 22;  
>> struct1  
struct1 =  
1x2 struct array with fields:  
    name  
    age
```

【例2.3-22】 利用struct函数定义结构体数组。

调用格式：

```
s = struct('field1', values1, 'field2', values2, ...)
```

```
s = struct('field1', {}, 'field2', {}, ...)
```

```
>> struct2 = struct('name', {'xiezhh', 'heping'},  
'age',{31, 22})
```

```
struct2 =
```

```
1x2 struct array with fields:
```

```
    name
```

```
    age
```

```
>> struct2(1).name
```

```
ans =
```

```
xiezhh
```

七、几种数组的转换

- mat2cell , 将矩阵分块，转为元胞数组
- cell2mat , 将元胞数组转为矩阵
- num2cell , 将数值型数组转为元胞数组
- cell2struct , 将元胞数组转为结构数组
- struct2cell , 将结构数组转为元胞数组
- num2str , 将数值型数组转为字符型数组
- str2num , 将字符型数组转为数值型数组

【例2.3-23】数组转换函数示例。

```
>> A1 = rand(60,50);  
>> B1 = mat2cell(A1, [10 20 30], [25 25])  
>> C1 = cell2mat(B1);  
>> isequal(A1,C1)  
>> A2 = [1 2 3 4;5 6 7 8;9 10 11 12];  
>> B2 = num2cell(A2)  
>> C = {'Heping', 'Tianjin', 22; 'Xiezhh', 'Xingyang', 31}  
>> fields = {'Name', 'Address', 'Age'};  
>> S = cell2struct(C, fields, 2)  
>> CS = struct2cell(S)  
>> isequal(C,CS')
```

八、矩阵的算术运算

1. 矩阵的加减

【例2.3-24】矩阵的加减运算。

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

```
>> B = [5 6; 7 8];
```

```
>> C = A+B
```

```
C =
```

```
6 8
```

```
10 12
```

```
>> D = A-B
```

```
D =
```

```
-4 -4
```

```
-4 -4
```

2. 矩阵的乘法

矩阵的乘法包括 乘 (A^*B) 和 点乘 ($A.^*B$) 两种。其中 A^*B 是通常意义上的乘法，要求A的列数等于B的行数。而 $A.^*B$ 则表示同型矩阵A和B的对应元素相乘。

【例2.3-25】矩阵的乘法。

```
>> A = [1 2 3; 4 5 6];  
>> B = [1 1 1 1; 2 2 2 2; 3 3 3 3];  
>> C = A*B  
C = 14 14 14 14  
      32 32 32 32  
>> D = [1 1 1; 2 2 2];  
>> E = A.*D  
E =  
    1   2   3  
    8  10  12
```

3. 矩阵的除法

矩阵的除法包括左除 ($A \setminus B$)、右除

(A / B) 和点除 ($A ./ B$) 三种。一般情况下，

$x = A \setminus b$ 是方程组 $A^*x = b$ 的解，而 $x = b / A$

是方程组 $x^*A = b$ 的解， $x = A ./ B$ 表示同型

矩阵 A 和 B 对应元素相除

【例2.3-26】矩阵的除法。

```
>> A = [2 3 8; 1 -2 -4; -5 3 1];  
>> b = [-5; 3; 2];  
>> x = A\b  
x =  
1  
3  
-2  
>> B = A;  
>> C = A./B  
C =  
1 1 1  
1 1 1  
1 1 1
```

4. 矩阵的乘方 (\wedge) 与点乘方($.^{\wedge}$)

➤ 矩阵的乘方要求矩阵必须是方阵，有以下3种情况：

- (1) 矩阵A为方阵， x 为正整数， A^x 表示矩阵A自乘 x 次；
- (2) 矩阵A为方阵， x 为负整数， A^x 表示矩阵 A^{-1} 自乘- x 次；
- (3) 矩阵A为方阵， x 为分数，例如 $x = m/n$ ， A^x 表示矩阵A先自乘 m 次，然后对结果矩阵开 n 次方。

➤ 矩阵的点乘方不要求矩阵为方阵，有以下2种情况：

- (1) A为矩阵， x 为标量， $A.^x$ 表示对矩阵A中的每一个元素求 x 次方；
- (2) A和x为同型矩阵， $A.^x$ 表示对矩阵A中的每一个元素求x中对应元素次方。

【例2.3-27】矩阵乘方与点乘方。

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

```
>> B = A ^ 2
```

```
B =
```

```
7 10
```

```
15 22
```

```
>> C = A .^ 2
```

```
C =
```

```
1 4
```

```
9 16
```

```
>> D = A .^ A
```

```
D =
```

```
1 4
```

```
27 256
```

九、矩阵的关系运算

矩阵的关系运算是通过比较两个同型矩阵的对应元素的大小关系，或者比较一个矩阵的各元素与某一标量之间的大小关系，返回一个逻辑矩阵（1表示真，0表示假）。关系运算的运算符有： $<$ （小于）、 \leq （小于或等于）、 $>$ （大于）、 \geq （大于或等于）、 $=$ （等于）、 \neq （不等于）6种。

【例2.3-28】矩阵的关系运算。

```
>> A = [1 2; 3 4];
>> B = [2 2; 2 2];
>> C1 = A > B
C1 =
    0 0
    1 1
>> C2 = A ~= B
C2 =
    1 0
    1 1
>> C3 = A >= 2
C3 =
    0 1
    1 1
```

十、矩阵的逻辑运算

矩阵的逻辑运算包括：

- 逻辑“或”运算，运算符为“|”. $A | B$

表示同型矩阵A和B的或运算，若A和B的对应元素至少有一个非0，则相应的结果元素值为1，否则为0；

- 逻辑“与”运算，运算符为“&”。 $A \& B$ 表示同型矩阵A和B的与运算，若A和B的对应元素均非0，则相应的结果元素值为1，否则为0；
- 逻辑“非”运算，运算符为“~”。 $\sim A$ 表示矩阵A的非运算，若A的元素值为0，则相应的结果元素值为1，否则为0；

- 逻辑“异或”运算。 $\text{xor}(A, B)$ 表示同型矩阵
A和B的异或运算，若A和B的对应元素均为0或
均非0，则相应的结果元素值为0，否则为1.
- 先决与运算，运算符“`&&`”。`A && B`表
示当A为真时，才执行A和B的逻辑与运算
- 先决或运算，运算符“`||`”。`A || B`表示当A
为真时，不用再执行A和B的逻辑或运算

【例2.3-29】矩阵的逻辑运算。

```
>> A = [0 0 1 2];
>> B = [0 -2 0 1];
>> C1 = A | B
C1 =
    0   1   1   1
>> C2 = A & B
C2 =
    0   0   0   1
>> C3 = ~ A
C3 =
    1   1   0   0
>> C4 = xor(A, B)
C4 =
    0   1   1   0
```

十一、运算符的优先级

优先级	运算符						
1	()						
2	!	'	.^	^			
3	代数正	代数负	~				
4	.*	.\	./	*	\	/	
5	+	-					
6	:						
7	<	>	==	>=	<=	~=	
8	&						
9							
10	&&						
11							

十二、矩阵的其他常用运算

1. 矩阵的转置与共轭转置

矩阵的转置包括转置 (A') 和共轭转置 (\bar{A}') 两种。对于实矩阵，两种转置是相同的。

【例2.3-30】矩阵的转置。

```
>> A = [1 2 3; 4 5 6; 7 8 9]
```

```
A =
```

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

```
>> B = A' % 或B = A.'
```

```
B =
```

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

【例2.3 -31】复矩阵的共轭转置。

```
>> A = [1 2 ;3 4]*i+1
```

```
A = 1.0000 + 1.0000i 1.0000 + 2.0000i  
      1.0000 + 3.0000i 1.0000 + 4.0000i
```

```
>> B = A'
```

```
B = 1.0000 - 1.0000i 1.0000 - 3.0000i  
      1.0000 - 2.0000i 1.0000 - 4.0000i
```

2. 矩阵的翻转

【例2.3-32】矩阵的翻转。

```
>> A = [1 2 3; 4 5 6; 7 8 9];
```

```
>> B1 = flipud(A)
```

```
B1 =
```

7	8	9
4	5	6
1	2	3

```
>> B2 = fliplr(A)
```

```
B2 =
```

3	2	1
6	5	4
9	8	7

3. 调用det函数计算方阵的行列式

`d = det(A) % 计算方阵A的行列式`

【例2.3-33】计算数值矩阵 $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ 的行列式。

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

```
>> d1 = det(A)
```

```
d1 = -2
```

【例2.3-34】 计算数值矩阵 $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ 的行列式。

```
>> A = [1 2 3; 4 5 6; 7 8 9];
```

```
>> d2 = det(A)
```

```
d1 = 6.6613e-16
```

思考：题中矩阵 A 的行列式为什么不等于0 ?

【例2.3-35】计算符号矩阵的行列式（3阶
行列式的对角线法则）。

```
>> A = sym('a%d%d', [3,3])  
>> det(A)  
  
ans =  
  
a11*a22*a33 - a11*a23*a32 -  
a12*a21*a33 + a12*a23*a31 +  
a13*a21*a32 - a13*a22*a31
```

4. 逆矩阵与广义伪逆矩阵

【例2.3-36】逆矩阵 (inv) 与广义伪逆矩阵 (pinv)。

```
>> A = [1 2; 3 4];
>> Ai = inv(A)
Ai =
-2.0000  1.0000
 1.5000 -0.5000
>> syms a b c d
>> B = [a b; c d];
>> Bi = inv(B)
Bi =
[ d/(a*d-b*c), -b/(a*d-b*c)]
[ -c/(a*d-b*c), a/(a*d-b*c)]
```

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

```
>> Cpi = pinv(C)
```

```
Cpi =
```

```
-0.9444 0.4444
```

```
-0.1111 0.1111
```

```
0.7222 -0.2222
```

```
>> D = C * Cpi * C
```

```
D =
```

```
1.0000 2.0000 3.0000
```

```
4.0000 5.0000 6.0000
```

5. 方阵的特征值与特征向量

【例2.3-37】方阵的特征值与特征向量。

```
>> A = [5 0 4; 3 1 6; 0 2 3];
>> d = eig(A)
d =
-1.0000
3.0000
7.0000
>> [V, D] = eig(A)
V =
-0.2857  0.8944  0.6667
-0.8571  0.0000  0.6667
 0.4286 -0.4472  0.3333
D =
-1.0000      0      0
      0  3.0000      0
      0      0  7.0000
```

```
>> [Vs, Ds] = eig(sym(A))
```

Vs =

[2, 1, -2]

[2, 3, 0]

[1, -3/2, 1]

Ds =

[7, 0, 0]

[0, -1, 0]

[0, 0, 3]

6. 矩阵的迹和矩阵的秩

【例2.3-38】矩阵的迹和矩阵的秩。

```
>> A = [1 2 3; 4 5 6; 7 8 9];
```

```
>> t = trace(A)
```

```
t =
```

```
15
```

```
>> r = rank(A)
```

```
r =
```

```
2
```

第四节 MATLAB常用标点符号

名称	标 点	功能说明
空格		数组元素或输入量之间的分隔符
逗号	,	数组元素或输入量之间的分隔符
黑点	.	小数点；结构体数组的字段标识符；点运算标识符。
分号	;	定义数组时，作为行间分隔符；用在某条命令的“结尾”，不显示计算结果
冒号	:	作为冒号运算符，用来生成一维数组；作为数组单下标引用时，表示将数组按列拉长为长向量；作为数组多下标引用时，表示该维上的所有元素
注释号	%	注释内容引导符
单引号对	''	字符串标记符
圆括号	()	用来访问数组元素；用来标记运算作用域；定义函数时用来标记输入变量列表
方括号	[]	用来定义数组；定义函数时用来标记输出变量列表
花括号	{}	用来定义或访问元胞数组；用来标记图形对象中的特殊字符
下连符	_	作为变量、函数或文件名中的连字符；图形对象中下脚标前导符
续行号	...	由三个以上连续黑点构成。它把其下的命令行看作该行的延续，以构成一个“较长”的完整命令
“At”号	@	放在函数名前，形成函数句柄；匿名函数前导符；放在目录名前，形成“用户对象”类目录。

第五节 MATLAB常用快捷键和快捷命令

一、 MATLAB常用快捷键

快捷键	说 明	用在何处
方向键↑	调出历史命令中的前一个命令	命令窗口 (Command Window)
方向键↓	调出历史命令中的后一个命令	
Tab 键	输入命令的前几个字符，然后按 Tab 键，会弹出前面包含这几个字符的所有命令，方便查找所需命令	
Ctrl+C	中断程序的运行，用于耗时过长程序的紧急中断	
Tab 键或 Ctrl+]	增加缩进（对多行有效）	程序编辑窗口 (Editor)
Ctrl+[减少缩进（对多行有效）	
Ctrl+I	自动缩进（即自动排版，对多行有效）	
Ctrl+R	注释（对多行有效）	
Ctrl+T	去掉注释（对多行有效）	
F12 键	设置或清除断点	
F5 键	运行程序	

二、 MATLAB常用快捷命令

快捷命令	说 明	快捷命令	说 明
help	查找 Matlab 函数的帮助	cd	返回或设置当前工作路径
lookfor	按关键词查找帮助	dir	列出指定路径的文件清单
doc	查看帮助页面	whos	列出工作空间窗口的变量清单
clc	清除命令窗口中的内容	class	查看变量类型
clear	清除内存变量	which	查找文件所在路径
clf	清空当前图形窗口	what	列出当前路径下的文件清单
cla	清空当前坐标系	open	打开指定文件
edit	新建一个空白的程序编辑窗口	type	显示 M 文件的内容
save	保存变量	more	使显示内容分页显示
load	载入变量	exit/quit	退出 MATLAB

【例2.5-1】 MATLAB常用快捷命令举例。

```
>> A = [1 2 3;4 5 6;7 8 9];  
>> B = 100;  
>> Str = 'Hellow World !!!';  
>> C = cell(2,3);  
>> S = struct('name', {'heping','xiezhh'}, 'age',{22, 31});  
>> syms a b c d  
>> D = [a b;c d];  
>> whos  
>> save xiezhh.mat A B Str  
>> clear A B Str  
>> load xiezhh.mat  
>> which sin  
>> open sqrt
```

Thank You