# 1117 课堂内容

## 矩阵运算的补充和多项式的使用

### 一、矩阵运算的补充

tips : 默认列优先

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

```
A = 3×3
     1     2     3
     4     5     6
     7     8     9
```

```
% 元素按列进行求积
prod(A)
```

```
ans = 1×3
    28    80   162
```

```
prod(A,2) % 算行的积
```

```
ans = 3×1
     6
   120
   504
```

```
% 求累和
cumsum(A)
```

```
ans = 3×3
     1     2     3
     5     7     9
    12    15    18
```

```
cumsum(A,2)
```

```
ans = 3×3
     1     3     6
     4     9    15
     7    15    24
```

```
% 求累积
cumprod(A)
```

```
ans = 3×3
     1     2     3
     4    10    18
    28    80   162
```

```
cumprod(A,2)
```

```
ans = 3×3
     1     2     6
     4    20   120
     7    56   504
```

```
% 求平均值
```

```
mean(A)
```

```
ans = 1×3
     4     5     6
```

```
mean(A,2)
```

```
ans = 3×1
     2
     5
     8
```

```
% 求中值
median(A)
```

```
ans = 1×3
     4     5     6
```

```
median(A,2)
```

```
ans = 3×1
     2
     5
     8
```

```
% 求标准差
std(A)
```

```
ans = 1×3
     3     3     3
```

```
std(A,0,2)
```

```
ans = 3×1
     1
     1
     1
```

```
% 元素的差分
A = [1,2,3;4,5,6;7,8,9]
```

```
A = 3×3
     1     2     3
     4     5     6
     7     8     9
```

```
diff(A)
```

```
ans = 2×3
     3     3     3
     3     3     3
```

```
diff(A,1,1)
```

```
ans = 2×3
     3     3     3
     3     3     3
```

```
diff(A,1,2)
```

```
ans = 3×2
     1     1
     1     1
     1     1
```

```
% 相关系数
% R = corrcoef(A)返回 A 的相关系数的矩阵, 其中 A 的列表示随机变量
% , 行表示观测值
% R = corroef(A,B)返回两个随机变量 A 和 B 之间的系数
```

## 二、稀疏矩阵

**稀疏矩阵存储效率对比**

```
tic
A = sprand(2000,3000,0.1);
toc
```

```
历时 0.064197 秒。
```

```


tic
B = full(A);
toc
```

```
历时 0.017871 秒。
```

**稀疏矩阵的生成**

```
% S = sparse(A)
% 通过挤出任何零元素将满矩阵转换为稀疏格式。
% 如果矩阵包含许多 0, 将矩阵转换位稀疏存储空间可以节省内存
% S = sparse(m,n)
% 生成 mxn 全 0 稀疏矩阵
```

```
% S = sparese(i,j,v)
% 根据三元组生成稀疏矩阵 S
D = sparse(1:10,1:10,1:10)
```

```
D =
   (1,1)        1
   (2,2)        2
   (3,3)        3
   (4,4)        4
   (5,5)        5
   (6,6)        6
   (7,7)        7
   (8,8)        8
   (9,9)        9
  (10,10)      10
```

```
% S = sparse(i,j,v,m,n)
% 将 S 的大小指定为 mxn
% S = sparse(i,j,v)指定行下标为 v
F = sparse(1:10,1:10,3)
```

```
F =
   (1,1)        3
   (2,2)        3
   (3,3)        3
   (4,4)        3
   (5,5)        3
   (6,6)        3
   (7,7)        3
   (8,8)        3
   (9,9)        3
  (10,10)       3
```

**稀疏矩阵的转换**

```
A = [1,0,0,0;0,5,0,0;2,0,0,6];
S = sparse(A)
```

```
S =
   (1,1)        1
   (3,1)        2
   (2,2)        5
   (3,4)        6
```

```
B = full(S)
```

```
B = 3×4
     1     0     0     0
     0     5     0     0
     2     0     0     6
```

```
C = full(sparse(1:5,1:5,1:5))
```

```
C = 5×5
     1     0     0     0     0
     0     2     0     0     0
     0     0     3     0     0
     0     0     0     4     0
     0     0     0     0     5
```

## 1 矩阵补充

### 稀疏矩阵

#### 特定的生成函数

| 函数 | 说明 |
| --- | --- |
| S = speye(m, n) | 创建单位稀疏矩阵 |
| S = spones(X) | 创建非零元素为1的稀疏矩阵 |
| S = sprand(X) | 创建非零元素为均匀分布随机数的稀疏矩阵 |
| S = sprandn(X) | 创建非零元素为高斯分布随机数的稀疏矩阵 |
| S = sprandsym(X) | 创建非零元素为高斯分布随机数的对阵稀疏矩阵 |
| S = spdiags(X) | 创建对角稀疏矩阵 |
| S = spalloc(X) | 为稀疏矩阵分配空间 |

```
>> A = speye(3, 4)

A =

   (1,1)        1
   (2,2)        1
   (3,3)        1
```

```
>> B = full(A)

B =

     1     0     0     0
     0     1     0     0
     0     0     1     0
```

```
>> S = sprand(A)

S =

   (1,1)        0.9134
   (2,2)        0.6324
   (3,3)        0.0975
```

16

| 函数 | 说明 |
| --- | --- |
| n = nnz(S) | 返回稀疏矩阵中所有非零元素单元个数 |
| s = nonzeros(S) | 返回一个包含所有非零元素的列向量 |
| n = nzmax(S) | 返回稀疏矩阵中所有非零元素的储存空间 |
| spy(S) | 稀疏矩阵图形化，绘制非零元素分布图形 |

**稀疏矩阵的运算原**则

多个矩阵输入时，如果其中至少有一个矩阵是满矩阵，那么大部分函数的输出结果是满矩阵

对于矩阵的加减乘除运算，只要有一个是满矩阵，那么数据结果都是满矩阵

稀疏矩阵的数乘和幂都是稀疏矩阵、

## 三、矩阵分解

### 矩阵分解

矩阵分解是矩阵运算中一个重要的概念，如求矩阵的特征值和特征向量、矩阵的秩等重要参数时都要用到矩阵分解

直接定义：将矩阵拆解为数个矩阵的乘积

一些常见分解：

- ✓ 三角分解法是将原正方 (square) 矩阵分解成一个上三角形矩阵或是排列(permuted) 的上三角形矩阵和一个 下三角形矩阵，这样的分解法又称为LU分解法；它的用途主要在简化一个大矩阵的行列式值的计算过程。
- ✓ QR分解法是将矩阵分解成一个正规正交矩阵与上三角形矩阵,所以称为QR分解法,与此正规正交矩阵的通用符号Q有关。
- ✓ 奇异值分解 (singular value decomposition,SVD) 是另一种正交矩阵分解法；SVD是最可靠的分解法，但是它比QR 分解法要花上近十倍的计算时间。

20

**三角分解(LU 分解)**

或者称为高斯消去法；

将一个任意方阵 A 分解为一个下三角矩阵 L 和一个上三角矩阵 U 的乘积

即 A = LU

```
A = [1,2,3;4,5,6;7,8,9];
[L,U] = lu(A);
L,U
```

```
L = 3×3
    0.1429    1.0000         0
    0.5714    0.5000    1.0000
    1.0000         0         0
U = 3×3
    7.0000    8.0000    9.0000
         0    0.8571    1.7143
         0         0   -0.0000
```

**正交分解(QR 分解)**

对于非奇异矩阵 A(nxn)，存在正交矩阵 Q 和上三角矩阵 R 使得

A = QxR， QR 的分解是唯一的。

```
A = [1,2,3;4,5,6;7,8,9];
[Q,R]=qr(A);
Q,R
```

```
Q = 3×3
   -0.1231    0.9045    0.4082
   -0.4924    0.3015   -0.8165
   -0.8616   -0.3015    0.4082
R = 3×3
   -8.1240   -9.6011  -11.0782
         0    0.9045    1.8091
         0         0   -0.0000
```

**奇异值分解(SVD 分解)**

利用 mxn 矩阵 A 的特征值和奇异值进行分解

```
A = [1,2,3;4,5,6;7,8,9];
[U,S,V] = svd(A);
U,S,V
```

```
U = 3×3
   -0.2148    0.8872    0.4082
   -0.5206    0.2496   -0.8165
   -0.8263   -0.3879    0.4082
S = 3×3
   16.8481         0         0
         0    1.0684         0
         0         0    0.0000
V = 3×3
   -0.4797   -0.7767    0.4082
   -0.5724   -0.0757   -0.8165
   -0.6651    0.6253    0.4082
```

**特征分解**

对于 N 阶方阵 A，特征值为 λ1....λN，有 AV=VA

如果 V 线性无关，则 V 可逆，此时$A = V\Lambda V^{-1}$称为 A 的特征解，对角阵$\Lambda$称为 A 的标准型。

```
A = [1,2,3;4,5,6;7,8,9];
[V,D] = eig(A);
V,D
```

```
V = 3×3
   -0.2320   -0.7858    0.4082
   -0.5253   -0.0868   -0.8165
   -0.8187    0.6123    0.4082
D = 3×3
   16.1168         0         0
         0   -1.1168         0
         0         0   -0.0000
```

## 四、多项式的创建和使用

在 matlab 中，提供了 poly2sym 函数实现多项式的构造

```
% r = poly2sym(p) p 为多项式的系数向量
% r = poly2sym(p,v) p 为多项式的系数向量, v 为其变量
a = poly2sym([1,2,3,4])
```

a = $x^3 + 2x^2 + 3x + 4$

```
t = sym('t')
```

t = $t$

```
b = poly2sym([1,2,3,4],t)
```

b = $t^3 + 2t^2 + 3t + 4$

**使用**

```
a = [1,2,4,7];
b = [1,4,6,12];
d = a+b;
poly2sym(d)
```

ans = $2x^3 + 6x^2 + 10x + 19$

```
c = [1,2,-6,10,0,99];
d
```

```
d = 1×4
     2     6    10    19
```

```
e2 = c + [0,0,d] % 注意：这里要补齐首 0
```

```
e2 = 1×6
     1     2    -4    16    10   118
```

```
% 不补 0 的话运算会有误
```

**多项式乘法**

```
% c = conv(a,b) % 执行两个向量的卷积运算
u = [1,0,1];
v = [2,7];
w = conv(u,v);
poly2sym(w)
```

ans = $2x^3 + 7x^2 + 2x + 7$

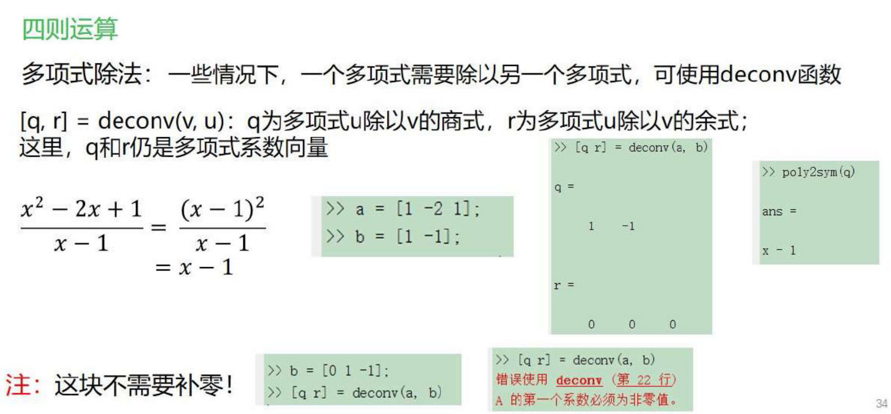
```
% c = conv(a,b,shape) 按照参数 shape 返回卷积运算
% 可选：
% full:完整卷积：即默认值
% same:返回部分卷积, 大小与向量 a 大小相等
% valid:只返回无填充 0 部分的卷积
u = [-1,2,3,-2,0,1,2];
v = [2,4,-1,-1];
conv(u,v)
```

```
ans = 1×10
    -2     0    15     7   -13     1    10     7    -3    -2
```

```
conv(u,v,'same')
```

```
ans = 1×7
    15     7   -13     1    10     7    -3
```

**多项式除法**




```
% [q,r]
% q 为多项式 u/v 的商式, r 为 u/v 的余式
% q 和 r 仍是多项式系数向量
a = [1,-2,1];
b = [1,-1];
```

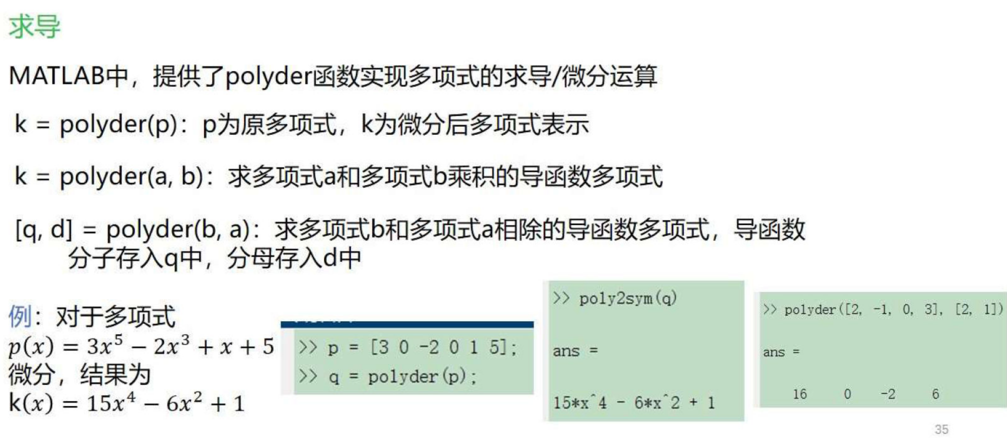
```
[q,r] = deconv(a,b)
```

```
q = 1×2
     1    -1
r = 1×3
     0     0     0
```

```
poly2sym(q)
```

ans = $x - 1$

## 求导




```
p = [3,0,-2,0,1,5];
poly2sym(p)
```

ans = $3\,x^5 - 2\,x^3 + x + 5$

```
q = polyder(p)
```

```
q = 1×5
    15     0    -6     0     1
```

```
poly2sym(q)
```

ans = $15\,x^4 - 6\,x^2 + 1$

## 积分

## 积分

MATLAB中，提供了polyint函数实现多项式的积分运算

q = polyint(p,k)：返回以向量p为系数的多项式积分，积分的常数项为k

例：对于多项式

$p(x)=2x$^2+$x-5$积分，结果为

$q(x)=2/3\ x$^3+1/2 $x$^2$-5x+6$

```
>> q = polyint(p, k)

q =

    0.6667    0.5000   -5.0000    6.0000

>> q = polyint(p)

q =

    0.6667    0.5000   -5.0000         0
```

```
p = [2,1,-5];
poly2sym(p)
```

ans = $2\,x^2+x-5$

```
% k 是积分常数
k = 1;
q = polyint(p,k);
poly2sym(q)
```

ans =

$$\frac{2\,x^3}{3}+\frac{x^2}{2}-5\,x+1$$

### 多项式求值

MATLAB提供了两种求多项式值的函数：polyval和polyvalm

包含两个参数多项式系数向量p和自变量x

$\boxed{\text{前者是代数代数多项式求值，后者是矩阵多项式求值}}$

- polyval 函数：求代数多项式的值

```
>> help polyval
 polyval - 多项式计算
    此 MATLAB 函数 计算多项式 p 在 x 的每个点处的值。参数 p 是长度为 n+1 的向量，其元素
    是 n 次多项式的系数（降幂排序）。

    y = polyval(p,x)
    [y,delta] = polyval(p,x,S)
    y = polyval(p,x,[],mu)
    [y,delta] = polyval(p,x,S,mu)
```

```
A = [1,6,0,0,-9];
poly2sym(A)
```

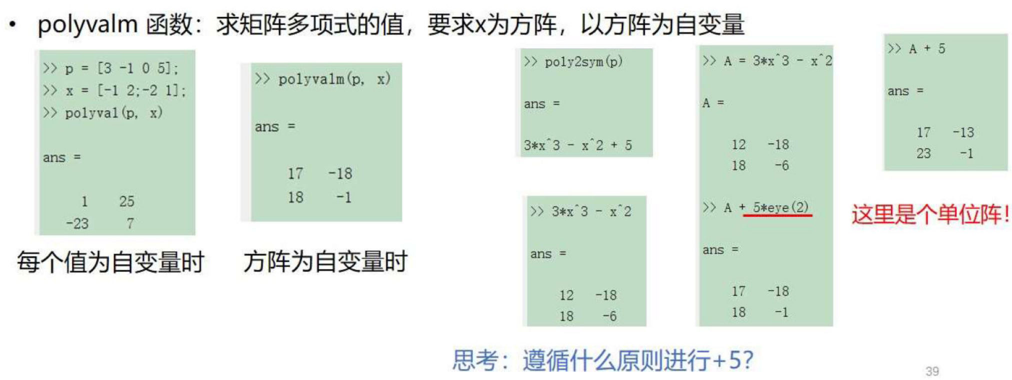
ans = $x^4+6\,x^3-9$

```
% x 是变量的值
x = 1.3;
y1 = polyval(A,x)
```

```
y1 = 7.0381
```

```
% B 是变量矩阵
B = [-1,1.3,1.5;2,-1.8,-1.6];
y2 = polyval(A,B)
```

```
y2 = 2×3
  -14.0000    7.0381   16.3125
   55.0000  -33.4944  -27.0224
```




```
% 求矩阵多项式的值，要求 x 为方阵，以方阵为自变量
p = [3,-1,0,5];
poly2sym(p)
```

ans = $3\,x^3-x^2+5$

```
x = [-1,2;-2,1];
polyval(p,x)
```

```
ans = 2×2
     1    25
   -23     7
```

```
polyvalm(p,x)
```

```
ans = 2×2
    17   -18
    18    -1
```

**求根**

```
% 求根
p = [1,2,1];
poly2sym(p)
```

ans = $x^2 + 2\,x + 1$

```
r = roots(p)
```

```
r = 2×1
    -1
    -1
```

```
% 如果知道了多项式的全部根，可以用 poly 函数建立该多项式
poly(r)
```

```
ans = 1×3
     1     2     1
```