# 第二讲 **MATLAB**的数值计算

—— matlab 具有出色的数值计算能力，占据世界上数值计算软件的主导地位

# 数值运算的功能

- 创建矩阵
- 矩阵运算
- 多项式运算
- 线性方程组
- 数值统计
- 线性插值
- 函数优化
- 微分方程的数值解

# 一、命令行的基本操作

## 1. 创建矩阵的方法

- 直接输入法

规则：

① 矩阵元素必须用[ ]括住

② 矩阵元素必须用逗号或空格分隔

③ 在[ ]内矩阵的行与行之间必须用分号分隔

# 矩阵元素

矩阵元素可以是任何matlab表达式，可以是实数，也可以是复数，复数可用特殊函数I，j 输入

a=[1 2 3;4 5 6]

x=[2 pi/2;sqrt(3) 3+5i]

# 符号的作用

- 逗号和分号的作用
  - ♣逗号和分号可作为指令间的分隔符，matlab允许多条语句在同一行出现。
  - ♣分号如果出现在指令后，屏幕上将不显示结果。

注意：只要是赋过值的变量，不管是否在屏幕上显示过，都存储在工作空间中，以后可随时显示或调用。变量名尽可能不要重复，否则会覆盖 。

当一个指令或矩阵太长时，可用•••续行

- 冒号的作用

❶用于生成等间隔的向量，默认间隔为1。

❷用于选出矩阵指定行、列及元素。

❸循环语句

# 2.用matlab函数创建矩阵

- 空阵 [ ] — matlab允许输入空阵，当一项操作无结果时，返回空阵。
- rand —— 随机矩阵
- eye —— 单位矩阵
- zeros ——全部元素都为0的矩阵
- ones ——全部元素都为1的矩阵

还有伴随矩阵、稀疏矩阵、魔方矩阵、对角矩阵、范德蒙等矩阵的创建，就不一一介绍了。

注意：matlab严格区分大小写字母，因此a与A是两个不同的变量。

matlab函数名必须小写。

# 3. 矩阵的修改

- ① 直接修改

  可用↑键找到所要修改的矩阵，用←键移动到要修改的矩阵元素上即可修改。

  ② 指令修改

  可以用A(*,*)= * 来修改。

例如

**a=[1 2 0;3 0 5;7 8 9]**

| | | | |
|---|---|---|---|
| **a =** | **1** | **2** | **0** |
| | **3** | **0** | **5** |
| | **7** | **8** | **9** |

**a(3,3)=0**

| | | | |
|---|---|---|---|
| **a =** | **1** | **2** | **0** |
| | **3** | **0** | **5** |
| | **7** | **8** | **0** |

- 还可以用函数subs修改，matlab6.0还可用find函数修改。

# 二、数据的保存与获取

- 把matlab工作空间中一些有用的数据长久保存下来的方法是生成mat数据文件。

  ①save —— 将工作空间中所有的变量存到matlab.mat文件中。

默认文件名

②save data——将工作空间中所有的变量存到data.mat文件中。

③save data a b ——将工作空间中a和b变量存到data.mat文件中。

下次运行matlab时即可用load指令调用已生成的mat文件。

①load ———
②load data ———
③load data a b ———

即可恢复保存过的所有变量

mat文件是标准的二进制文件，还可以ASCII码形式保存。

# 三、矩阵运算

## 1. 矩阵加、减（＋,－）运算

规则：

① 相加、减的两矩阵必须有相同的行和列两矩阵对应元素相加减。

② 允许参与运算的两矩阵之一是标量。标量与矩阵的所有元素分别进行加减操作。

## 2. 矩阵乘（*）运算

规则：

- A矩阵的列数必须等于B矩阵的行数
- 标量可与任何矩阵相乘。

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

c =14

32

23

d=[-1;0;2];f=pi*d

f = -3.1416

0

6.2832

矩阵除的运算在线性代数中没有，有矩阵逆的运算，在matlab中有两种矩阵除运算

# 3. 矩阵乘方—— a^n,a^p,p^a

a ^ p —— a 自乘p次幂

方阵　　>1的整数

对于p的其它值,计算将涉及特征值和特征向量，如果p是矩阵，a是标量a^p使用特征值和特征向量自乘到p次幂；如a,p都是矩阵，a^p则无意义。

a=[1,2,3;4,5,6;7,8,9];a^2

ans =30    36    42

66    81    96

102  126  150

※当一个方阵有复数特征值或负实特征值时，非整数幂是复数阵。

a^0.5

ans =

| | | |
|---|---|---|
| 0.4498 + 0.7623i | 0.5526 + 0.2068i | 0.6555 -0.3487i |
| 1.0185 + 0.0842i | 1.2515 + 0.0228i | 1.4844 - 0.0385i |
| 1.5873 - 0.5940i | 1.9503 - 0.1611i | 2.3134 + 0.2717i |

# 4. 矩阵的其它运算

- inv —— 矩阵求逆
- det —— 行列式的值
- eig —— 矩阵的特征值
- diag —— 对角矩阵
- ’ —— 矩阵转置
- sqrt —— 矩阵开方

# 5.矩阵的一些特殊操作

- 矩阵的变维

  a=[1:12];b=reshape(a,3,4)

  c=zeros(3,4);c(:)=a(:)

- 矩阵的变向

  rot90:旋转; fliplr:上翻; flipud:下翻

- 矩阵的抽取

  diag:抽取主对角线;tril: 抽取主下三角;

  triu:抽取主上三角

- 矩阵的扩展

# 关系运算

| 关系符号 | 意义 |
| --- | --- |
| < | 小于 |
| <= | 小于或等于 |
| > | 大于 |
| >= | 大于或等于 |
| == | 等于 |
| ~= | 不等于 |

## 5. 矩阵的数组运算

数组运算指元素对元素的算术运算，与通常意义上的由符号表示的线性代数矩阵运算不同

1. 数组加减(.+,.-)

a.+b
a.- b

对应元素相加减（与矩阵加减等效）

2. 数组乘除(.*，./，.\）

a.∗b —— a，b两数组必须有相同的行和列两数组相应元素相乘。

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

```
a=[1 2 3;4 5 6;7 8  9];
b=[2 4 6;1 3 5;7 9 10];

a*b
ans =
    25     37     46
    55     85    109
    85    133    172
```

$\left.\begin{array}{l}a./b=b.\backslash a\\ a.\backslash b=b./a\end{array}\right\}$ —— 给出a,b对应元素间的商.

a./b=b.\a — 都是a的元素被b的对应元素除

a.\b=b./a — 都是a的元素被b的对应元素除

例: a=[1 2 3];b=[4 5 6]; c1=a.\b; c2=b./a

c1 = 4.0000    2.5000    2.0000

c2 = 4.0000    2.5000    2.0000

3. 数组乘方(.^) — 元素对元素的幂

例:

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

z=a.^2

z =

| | | |
|---|---|---|
| 1.00 | 4.00 | 9.00 |

z=a.^b

z =

| | | |
|---|---|---|
| 1.00 | 32.00 | 729.00 |

# 四、 多项式运算

matlab语言把多项式表达成一个行向量，该向量中的元素是按多项式降幂排列的。

$f(x)=a_n x^n+a_{n-1}x^{n-1}+\ldots\ldots+loa_0$

可用行向量 $p=[a_n\ a_{n-1}\ \ldots\ldots\ a_1\ +a_0]$表示

1. poly —— 产生特征多项式系数向量

- <u>特征多项式一定是n+1维的</u>
- <u>特征多项式第一个元素一定是1</u>

例:a=[1 2 3;4 5 6;7 8 0];

p=poly(a)

p =1.00   -6.00   -72.00   -27.00

p是多项式$p(x)=x^3-6x^2-72x-27$的matlab描述方法，我们可用：

p1=poly2str(p,‘x’) — 函数文件，显示数学多项式的形式

p1 =x^3 - 6 x^2 - 72 x - 27

## 2.roots —— 求多项式的根

a=[1 2 3;4 5 6;7 8 0];p=poly(a)

p =

1.00 -6.00 -72.00 -27.00

r=roots(p)

r = 12.12

-5.73 ——显然 r是矩阵a的特征值

-0.39

当然我们可用poly令其返回多项式形式

p2=poly(r)

p2 =

1.00 -6.00 -72.00 -27.00

- matlab规定多项式系数向量用行向量表示，一组根用列向量表示。

# 3.conv,convs多项式乘运算

例:a(x)=$x^2$+2x+3; b(x)=$4x^2$+5x+6;

c = ($x^2$+2x+3)($4x^2$+5x+6)

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

c=conv(a,b)=conv([1 2 3],[4 5 6])

c = 4.00 13.00 28.00 27.00 18.00

p=poly2str(c,'x')

p = 4 x^4 + 13 x^3 + 28 x^2 + 27 x + 18

# 4.deconv多项式除运算

a=[1 2 3];

c = [4.00  13.00  28.00   27.00  18.00]

d=deconv(c,a)

d =4.00        5.00        6.00

[d,r]=deconv(c,a)

余数

c除a后的整数

# 5.多项式微分

matlab提供了polyder函数多项式的微分。

命令格式：

polyder(p): 求p的微分

polyder(a,b): 求多项式a,b乘积的微分

[p,q]=polyder(a,b): 求多项式a,b商的微分

例：a=[1 2 3 4 5]; poly2str(a,'x')

ans = x^4 + 2 x^3 + 3 x^2 + 4 x + 5

b=polyder(a)

b = 4 6 6 4

poly2str(b,'x')

ans =4 x^3 + 6 x^2 + 6 x + 4

# 五、代数方程组求解

matlab中有两种除运算左除和右除。

对于方程ax+b，a 为$a_{n\times m}$矩阵，有三种情况：

① 当n=m时，此方程成为“恰定”方程

② 当n>m时，此方程成为“超定”方程

③ 当n<m时，此方程成为“欠定”方程

matlab定义的除运算可以很方便地解上述三种方程

## 1.恰定方程组的解

方程ax+b(a为非奇异)

$x=a^{-1}b$

矩阵逆

两种解:

- x=inv(a)*b — 采用求逆运算解方程
- x=a\b — 采用左除运算解方程

例: $\begin{cases} x_1+2x_2=8 \\ 2x_1+3x_2=13 \end{cases} \longrightarrow \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 8 \\ 13 \end{bmatrix}$

**a x = b**

方程ax=b

a=[1 2;2 3];b=[8;13];

①x=inv(a)*b

x =

2.00

3.00

② x=a\b

x =

2.00

3.00

## 2.超定方程组的解

方程 ax=b ,m<n时此时不存在唯一解。

方程解 (a ' a)x=a ' b

$x=(a'a)^{-1} a' b$ —— 求逆法

x=a\b —— matlab用最小二乘法找一个准确地基本解。

例: $\begin{cases} x_1+2x_2=1 \\ 2x_1+3x_2=2 \\ 3x_1+4x_2=3 \end{cases}$ $\longrightarrow$ $\begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$

**a x = b**

a=[1 2;2 3;3 4];b=[1;2;3];

解1 x=a\b 解2 x=inv(a'*a) * a' * b

```
x =
      1.00
         0
```

```
x =
      1.00
      0.00
```

## 3.欠定方程组的解

当方程数少于未知量个数时,即不定情况,有无穷多个解存在。

matlab可求出两个解:

- 用除法求的解x是具有最多零元素的解
- 是具有最小长度或范数的解，这个解是基于伪逆pinv求得的。

$$\begin{cases} x_1+2x_2+3x_3=1 \\ 2x_1+3x_2+4x_3=2 \end{cases} \longrightarrow \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

**a x = b**

a=[1 2 3;2 3 4];b=[1;2];

| x=a\b | x=pinv(a)*b |
|---|---|
| x = | x = |
| 1.00 | 0.83 |
| 0 | 0.33 |
| 0 | -0.17 |

# 六、微分方程求解

- 微分方程求解的仿真算法有多种，常用的有Euler（欧拉法）、Runge Kutta(龙格-库塔法。

$y_{n+1} = y_n + \frac{1}{2}k_1 + \frac{1}{2}k_2$

- Euler法称一步法，用于一阶微分方程

$$\frac{dy}{dt} = f(x, y), y(x_0) = y_0$$

- 当给定仿真步长时：

$$\frac{dy}{dt} \approx \frac{y_{n+1} - y_n}{x_{n+1} - x_n} = \frac{y_{n+1} - y_n}{h}$$

所以

$$\begin{cases} y_{n+1} = y_n + h\, f(x_n, y_n) \quad n=0,1,2\ldots \\ y(x_0)=y_0 \end{cases}$$

- Runge Kutta法

$$
\begin{cases}
y_{n+1} = y_n + \dfrac{1}{2}k_1 + \dfrac{1}{2}k_2 \\
k_1 = hf(x_n, y_n) \\
k_2 = hf(x_n + h, y_n + k)
\end{cases}
$$

龙格-库塔法：实际上取两点斜率的平均 斜率来计算的，其精度高于欧拉算法 。

龙格-库塔法：ode23　ode45

例：$\ddot{x}+(x^2-1)\dot{x}+x=0$

为方便令$x_1=x$，$x_2=\dot{x}$分别对$x_1,x_2$求一阶导数，整理后写成一阶微分方程组形式

$$\begin{cases} \dot{x}_1=x_2 \\ \dot{x}_2=x_2(1-x_1^2)-x_1 \end{cases}$$

1. 建立m文件
2. 解微分方程

建立m文件

```
function xdot=wf(t,x)
xdot=zeros(2,1)
xdot(1)=x(2)
xdot(2)=x(2)*(1-x(1)^2)-x(1)
```

给定区间、初始值;求解微分方程

```
t0=0; tf=20; x0=[0 0.25]';
[t,x]=ode23('wf', t0, tf, x0)
plot(t,x), figure(2),plot(x(:,1),x(:,2))
```

命令格式:

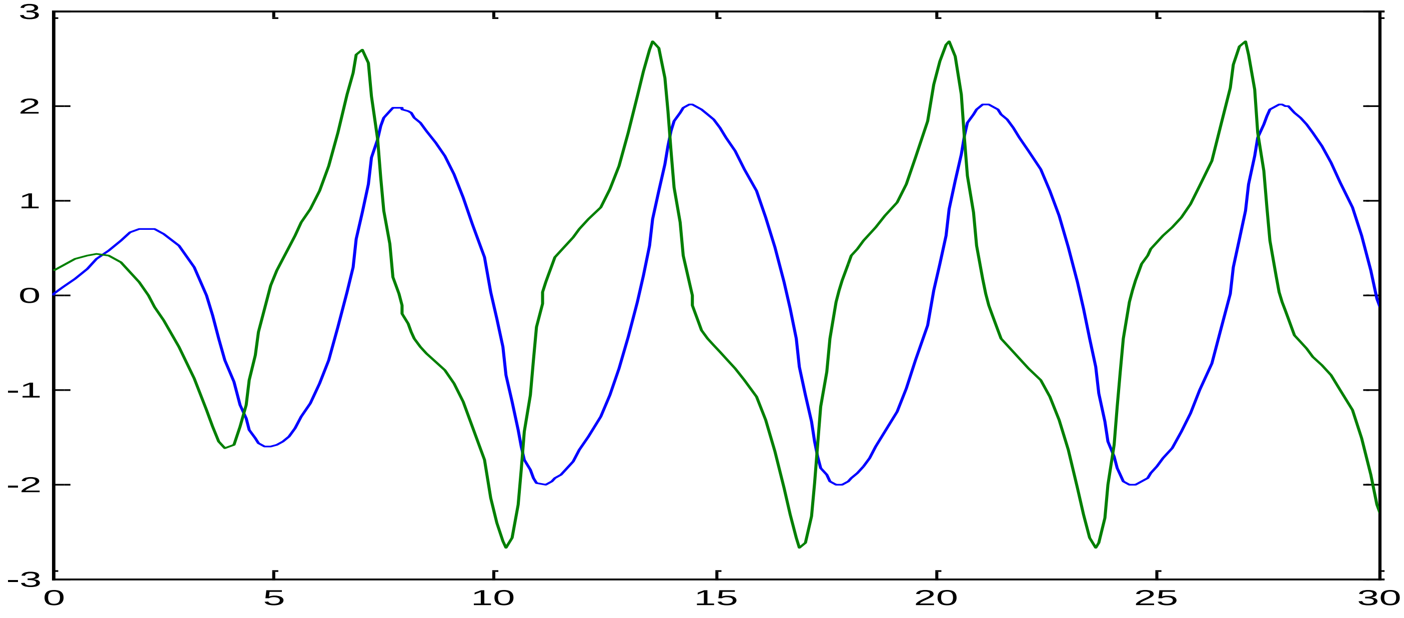
[T,Y] = ODE23(ODEFUN,TSPAN,Y0)

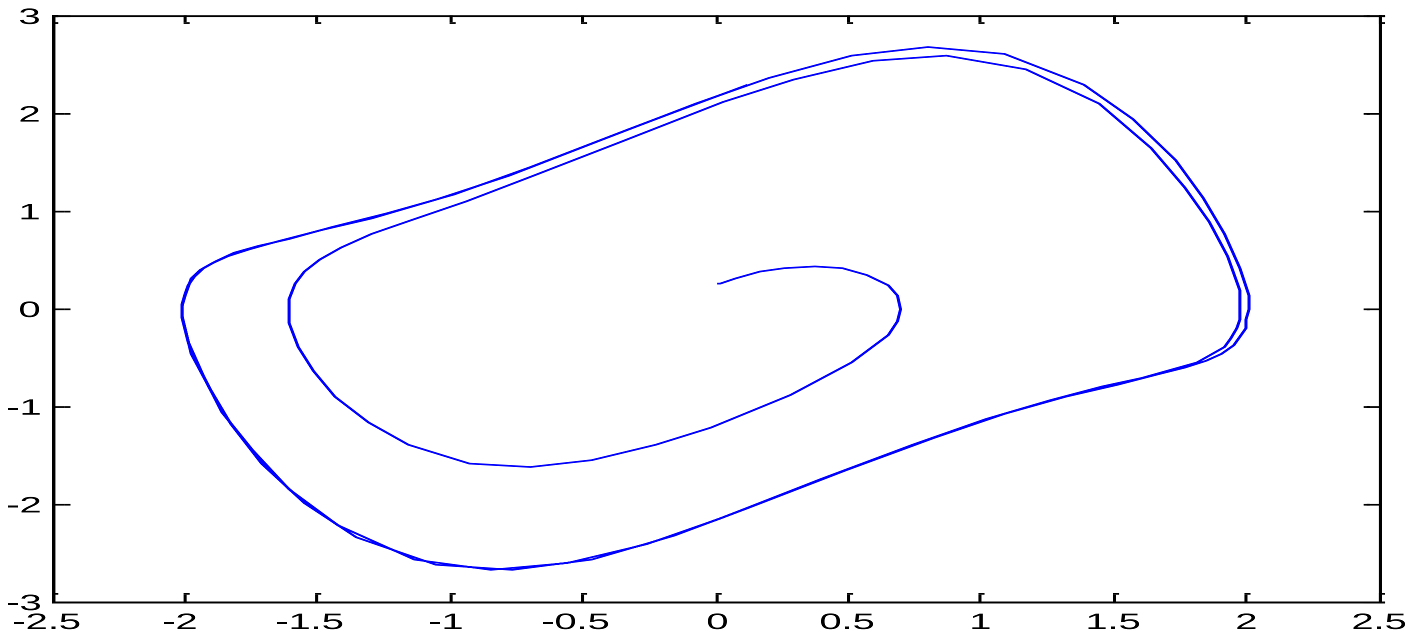
建立m文件

```
function dxdt=wf(t,x)
dxdt=[x(2);x(2)*(1-x(1)^2)-x(1)];
```

求解微分方程

```
[t,x]=ode23(@wf,[0 30],[0 0.25]);
plot(t,x);
figure(2)
plot(x(:,1),x(:,2))
```


3
2
1
0
-1
-2
-3
0
5
10
15
20
25
30


3
2
1
0
-1
-2
-3
-2.5
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
2.5

七、函数优化

寻优函数：

fmin —— 单变量函数

fmins —— 多变量函数

无约束条件

constr —— 有约束条件

例1：f(x)=‘$x^2+3x+2$’在[-5  5]区间的最小值

f=fmin('x^2+3*x+2',-5,5)

例2：$f(x)=100(x_2-x_1^2)^2+(a-x_1)^2$在$x_1=a$, $x_2=a^2$处有最小值

```
function f=xun(x,a)
f=100*(x(2)-x(1).^2).^2+(a-x(1)).^2;
x=fmins('xun',[0,0],[ ],[ ],sqrt(2))
```

# 八、数据分析与插值函数

**max** —— 各列最大值

**mean** —— 各列平均值

**sum** —— 各列求和

**std** —— 各列标准差

**var** —— 各列方差

**sort** —— 各列递增排序

# 九、拟合与插值

1. 多项式拟合

```
x0=0:0.1:1;
y0=[-.447 1.978 3.11 5.25 5.02 4.66 4.01
    4.58 3.45 5.35 9.22];
p=polyfit(x0,y0,3)
p = 56.6915  -87.1174   40.0070   -0.9043
xx=0:0.01:1;yy=polyval(p,xx);
plot(xx,yy,'-b',x0,y0,'or')
```

## 2.插值

- 插值的定义——是对某些集合给定的数据点之间函数的估值方法。
- 当不能很快地求出所需中间点的函数时，插值是一个非常有价值的工具。
- Matlab提供了一维、二维、 三次样条等许多插值选择

table1 ——
table2 ——
intep1 ——
interp2 ——
spline ——

插值函数

- 利用已知点确定未知点
- 粗糙—— 精确
- 集合大的—— 简化的

# 小　　结

本节介绍了matlab语言的数值运算功能，通过学习应该掌握:

- 如何创建矩阵、修改矩阵
- 符号的用法
- 矩阵及数组运算
- 多项式运算
- 线性方程组与微分运算