

附录二 Matlab 在线性代数中的应用

§1 向量组的线性相关性

求列向量组 A 的一个最大线性无关组可用命令 `rref(A)` 将 A 化成行最简形, 其中单位向量对应的列向量即为最大线性无关组所含向量, 其它列向量的坐标即为其对应向量用最大线性无关组线性表示的系数。

例 1 求下列矩阵列向量组的一个最大无关组。

$$A = \begin{bmatrix} 1 & -2 & -1 & 0 & 2 \\ -2 & 4 & 2 & 6 & -6 \\ 2 & -1 & 0 & 2 & 3 \\ 3 & 3 & 3 & 3 & 4 \end{bmatrix}$$

解 编写 M 文件 `ex1.m` 如下:

```
format rat
a=[1,-2,-1,0,2;-2,4,2,6,-6;2,-1,0,2,3;3,3,3,3,4];
b=rref(a)
```

求得

$$b = \begin{bmatrix} 1 & 0 & 1/3 & 0 & 16/3 \\ 0 & 1 & 2/3 & 0 & -1/9 \\ 0 & 0 & 0 & 1 & -1/3 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

记矩阵 A 的五个列向量依次为 $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5$, 则 $\alpha_1, \alpha_2, \alpha_4$ 是列向量组的一个最大无关组。且有

$$\alpha_3 = \frac{1}{3}\alpha_1 + \frac{2}{3}\alpha_2, \quad \alpha_5 = \frac{16}{3}\alpha_1 - \frac{1}{9}\alpha_2 - \frac{1}{3}\alpha_4.$$

$$\text{例 2 设 } A = [a_1, a_2, a_3] = \begin{bmatrix} 2 & 2 & -1 \\ 2 & -1 & 2 \\ -1 & 2 & 2 \end{bmatrix}, \quad B = [b_1, b_2] = \begin{bmatrix} 1 & 4 \\ 0 & 3 \\ -4 & 2 \end{bmatrix},$$

验证 a_1, a_2, a_3 是 R^3 的一个基, 并把 b_1, b_2 用这个基线性表示。

解 编写 M 文件 `ex2.m` 如下:

```
format rat
a=[2,2,-1;2,-1,2;-1,2,2];b=[1,4;0,3;-4,2];
c=rref([a,b])
```

求得

$$c = \begin{bmatrix} 1 & 0 & 0 & 2/3 & 4/3 \\ 0 & 1 & 0 & -2/3 & 1 \\ 0 & 0 & 1 & -1 & 2/3 \end{bmatrix}$$

说明 a_1, a_2, a_3 是 R^3 的一个基, 且有 $b_1 = \frac{2}{3}a_1 - \frac{2}{3}a_2 - a_3$, $b_2 = \frac{4}{3}a_1 + a_2 + \frac{2}{3}a_3$ 。

§2 齐次线性方程组

在 Matlab 中, 函数 `null` 用来求解零空间, 即满足 $Ax = 0$ 的解空间, 实际上是求

出解空间的一组基（基础解系）。

格式：

$z = \text{null}(A)$ %z 的列向量为方程组的正交规范基，满足 $z^T z = E$ 。

$z = \text{null}(A, 'r')$ %z 的列向量是方程 $Ax = 0$ 的有理基。

例 3 求方程组的通解

$$\begin{cases} x_1 + 2x_2 + 2x_3 + x_4 = 0 \\ 2x_1 + x_2 - 2x_3 - 2x_4 = 0 \\ x_1 - x_2 - 4x_3 - 3x_4 = 0 \end{cases}$$

解 编写程序如下：

```
format rat
a=[1, 2, 2, 1; 2, 1, -2, -2; 1, -1, -4, -3]
b=null(a, 'r') %求有理基
syms k1 k2
x=k1*b(:, 1)+k2*b(:, 2) %写出方程组的通解
```

求得基础解系为

$$(2, -2, 1, 0)^T, (5/3, -4/3, 0, 1)^T$$

通解为 $k_1(2, -2, 1, 0)^T + k_2(5/3, -4/3, 0, 1)^T$, $k_1, k_2 \in R$ 。

§ 3 非齐次线性方程组

Matlab 中解非齐次线性方程组可以使用“\”。虽然表面上只是一个简简单单的符号，而它的内部却包含许许多多的自适应算法，如对超定方程用最小二乘法，对欠定方程它将给出范数最小的一个解，解三对角阵方程组时用追赶法等。

另外求解欠定方程组可以使用求矩阵 A 的行最简形命令 **rref(A)**，求出所有的基础解系。

例 4 求超定方程组

$$\begin{cases} 2x_1 + 4x_2 = 11 \\ 3x_1 - 5x_2 = 3 \\ x_1 + 2x_2 = 6 \\ 2x_1 + x_2 = 7 \end{cases}$$

解 编写程序如下：

```
a=[2,4;3,-5;1,2;2,1];
b=[11;3;6;7];
solution=a\b
```

求得最小二乘解为 $x = (3.0403, 1.2418)^T$ 。

上面解超定方程组的“\”可以用伪逆命令 **pinv** 代替，且 **pinv** 的使用范围比“\”更加广泛，**pinv** 也给出最小二乘解或最小范数解。

例 5 用最小二乘解法解方程组

$$\begin{cases} x_1 + x_2 = 1 \\ x_1 + x_3 = 2 \\ x_1 + x_2 + x_3 = 0 \\ x_1 + 2x_2 - x_3 = -1 \end{cases}$$

解 编写程序如下:

```
format rat
a=[1,1,0;1,0,1;1,1,1;1,2,-1];
b=[1;2;0;-1];
x1=a\b %这里\和 pinv 是等价的
x2=pinv(a)*b
求得最小二乘解为
```

$$x_1 = \frac{17}{6}, \quad x_2 = -\frac{13}{6}, \quad x_3 = -\frac{4}{6}。$$

例 6 求解方程组

$$\begin{cases} x_1 - x_2 - x_3 + x_4 = 0 \\ x_1 - x_2 + x_3 - 3x_4 = 1 \\ x_1 - x_2 - 2x_3 + x_4 = -1/2 \end{cases}$$

解 编写程序如下:

```
format rat
a=[1,-1,-1,1,0;1,-1,1,-3,1;1,-1,-2,3,-1/2];
b=rref(a)
```

求得:

$$b = \begin{bmatrix} 1 & -1 & 0 & -1 & 1/2 \\ 0 & 0 & 1 & -2 & 1/2 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

故方程组有解, 并有

$$\begin{cases} x_1 = x_2 + x_4 + \frac{1}{2} \\ x_3 = 2x_4 + \frac{1}{2} \end{cases}$$

因而方程组的通解为

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = k_1 \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} + k_2 \begin{bmatrix} 1 \\ 0 \\ 2 \\ 1 \end{bmatrix} + \begin{bmatrix} 1/2 \\ 0 \\ 1/2 \\ 0 \end{bmatrix}$$

求解非齐次线性方程组需要先判断方程组是否有解, 若有解, 再去求通解。因此, 步骤为:

第一步: 判断 $Ax = b$ 是否有解, 若有解则进行第二步;

第二步: 求 $Ax = b$ 的一个特解;

第三步: 求 $Ax = 0$ 的通解;

第四步：写出 $Ax = b$ 的通解。

例7 求解下列方程组

$$\begin{cases} 2x_1 + x_2 - 5x_3 + x_4 = 8 \\ x_1 - 3x_2 - 6x_4 = 9 \\ 2x_2 - x_3 + 2x_4 = -5 \\ x_1 + 4x_2 - 7x_3 + 6x_4 = 0 \end{cases}$$

解 编写程序如下：

```
clc,clear
format rat
a=[2,1,-5,1;1,-3,0,-6;0,2,-1,2;1,4,-7,6];
b=[8;9;-5;0];
R_a=rank(a),R_B=rank([a,b])
n=size(a,2)
if R_a==R_B & R_a==n
    fprintf('方程组有唯一解\n')
    x=a\b
elseif R_a==R_B & R_a<n
    fprintf('方程组有无穷多解\n')
    x=a\b    %求非齐次方程组的特解
    xt=null(a,'r')    %求齐次方程的基础解系
else
    fprintf('方程组无解\n方程组的最小二乘解为\n')
    x=a\b
end
```

求得唯一解为 $x = (3, -4, -1, 1)^T$ 。

例8 求方程组的解

$$\begin{cases} x_1 + x_2 - 3x_3 - x_4 = 1 \\ 3x_1 - x_2 - 3x_3 + 4x_4 = 4 \\ x_1 + 5x_2 - 9x_3 - 8x_4 = 0 \end{cases}$$

解 编写程序如下：

```
clc,clear
format rat
a=[1,1,-3,-1;3,-1,-3,4;1,5,-9,-8];
b=[1;4;0];
R_a=rank(a),R_B=rank([a,b])
n=size(a,2)
if R_a==R_B & R_a==n
    fprintf('方程组有唯一解\n')
    x=a\b
elseif R_a==R_B & R_a<n
    fprintf('方程组有无穷多解\n')
    x=a\b    %求非齐次方程组的特解
    xt=null(a,'r')    %求齐次方程的基础解系
```

```

else
    fprintf('方程组无解\n 方程组的最小二乘解为\n')
    x=a\b
end

```

求得方程组的通解为

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = k_1 \begin{bmatrix} 3/2 \\ 3/2 \\ 1 \\ 0 \end{bmatrix} + k_2 \begin{bmatrix} -3/4 \\ 7/4 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -8/15 \\ 3/5 \end{bmatrix}, \quad k_1, k_2 \in R$$

例 9 设有线性方程组

$$\begin{cases} (1+\lambda)x_1 + x_2 + x_3 = 0 \\ x_1 + (1+\lambda)x_2 + x_3 = 3 \\ x_1 + x_2 + (1+\lambda)x_3 = \lambda \end{cases}$$

问 λ 取何值时，此方程组有唯一解？

解 编写如下程序：

```

clc, clear
syms k
a=[1+k, 1, 1; 1, 1+k, 1; 1, 1, 1+k];
D=det(a); %求系数矩阵的行列式
y=factor(D) %系数矩阵的行列式进行因式分解
root=solve(y)
b=[0; 3; k];
AB=[a, b];
m=length(root);
for n=1:m
    fprintf('k='), disp(root(n))
    fprintf('行最简型为\n')
    s{n}=rref(subs(AB, root(n)));
    disp(s{n}) %显示细胞数组的元素
    if rank(subs(a, root(n)))~=rank(subs(AB, root(n)))
        fprintf('方程组无解\n');
    else
        fprintf('方程组有无穷多解\n');
    end
    fprintf('*****\n')
end

```

求得当 $\lambda \neq -3$ 且 $\lambda \neq 0$ 时，方程组有唯一解；当 $\lambda = -3$ 时，方程组有无穷多解；当 $\lambda = 0$ 时，方程组无解。

§ 4 相似矩阵及二次型

有时我们需要精确的特征值和特征向量，就须利用 Matlab 的符号运算功能。

在 Matlab 中创建符号矩阵和创建数值矩阵的形式很相似，只不过要用到符号定义

函数 sym。下面介绍使用此函数创建符号矩阵的几种形式。

3.1 使用 sym 函数直接创建符号矩阵

此方法和直接创建数值矩阵的方法几乎完全相同。矩阵元素可以是符号表达式，各符号表达式的长度可以不同，矩阵元素之间可用空格或逗号分隔。

例如：

```
x=sym(' [a+sin(d),b;1/c,d] ');
y=det(x)
```

求得 $y=(d*c*a+d*c*\sin(d)-b)/c$

3.2 将数值矩阵转化为符号矩阵

在 Matlab 中，数值矩阵不能直接参与符号运算，必须先转化为符号矩阵。

例如：

```
a=[2/3,sqrt(2);3,1]
a= 0.6667    1.4142
    3.0000    1.0000
```

```
b=sym(a)
```

```
b=[ 2/3, sqrt(2)]
   [ 3,      1]
```

3.3 符号矩阵的索引和修改

Matlab 的符号矩阵索引和修改同数值矩阵的索引和修改完全相同。

例如：对上例中的矩阵 b 进行修改

```
b(2,2)='log(9)'
```

3.4 举例

例 10 求一个正交变换 $x = Py$ ，把二次型

$$f = 2x_1x_2 + 2x_1x_3 - 2x_1x_4 - 2x_2x_3 + 2x_2x_4 + 2x_3x_4$$

化为标准形。

解：二次型的矩阵为

$$A = \begin{bmatrix} 0 & 1 & 1 & -1 \\ 1 & 0 & -1 & 1 \\ 1 & -1 & 0 & 1 \\ -1 & 1 & 1 & 0 \end{bmatrix}$$

由

```
A=[0,1,1,-1;1,0,-1,1;-1,1,0,1;-1,1,1,0];
[P,D]=eig(A)
```

求得

```
P= 0.7887    0.2113    0.5000   -0.2887
    0.2113    0.7887   -0.5000    0.2887
    0.5774   -0.5774   -0.5000    0.2887
         0         0    0.5000    0.8660
D= 1.0000         0         0         0
         0    1.0000         0         0
         0         0   -3.0000         0
         0         0         0    1.0000
```

P 就是所求的正交矩阵，使得 $P^TAP=D$ ，令 $x = Py$ ，其中 $x = [x_1 \cdots x_4]^T$ ，

$y = [y_1 \ \cdots \ y_4]^T$, 化简后的二次型为 $g = y_1^2 + y_2^2 - 3y_3^2 + y_4^2$ 。

上面求得的正交矩阵 P 是数值解, 下面我们求正交矩阵的精确解。

由

```
a=sym(' [0,1,1,-1;1,0,-1,1;1,-1,0,1;-1,1,1,0] ');
```

```
[v,d]=eig(a)
```

求得

$$v = \begin{bmatrix} 1 & -1 & 1 & 1 \\ 1 & 0 & 0 & -1 \\ 0 & 0 & 1 & -1 \\ 0 & 1 & 0 & 1 \end{bmatrix} \quad d = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -3 \end{bmatrix}$$

即求得矩阵 A 的特征值为 1、1、1、3, 对应的特征向量分别是矩阵 v 的第 1、2、3、4 列。再把对应于特征值 1 的 3 个特征向量正交化、单位化, 我们就容易求出正交矩阵 P 。

习题

1. 求解矩阵方程

$$X \begin{bmatrix} 2 & 1 & -1 \\ 2 & 1 & 0 \\ 1 & -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 3 \\ 4 & 3 & 2 \end{bmatrix}$$

2. 求下列向量组的秩, 并求一个最大无关组。

$$a_1^T = (1, 2, 1, 3), \quad a_2^T = (4, -1, -5, -6), \quad a_3^T = (1, -3, -4, -7)。$$

3. 已知

$$A = \begin{bmatrix} 1 & 2 \\ 0 & 0 \\ 2 & 4 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}$$

求 $Ax = b$ 的最小范数解。

4. 求一个正交变换 $x = Py$, 把二次型 $f = 2x_1^2 + 3x_2^2 + 3x_3^2 + 4x_2x_3$ 变换为标准形。