



信阳师范学院  
数学与统计学院  
SCHOOL OF MATHEMATICS AND STATISTICS

## 第2章 矩阵分析与计算



讲授人：牛言涛



日期：2020年2月16日

# 数据与矩阵分析计算知识结构图



# 目录

## CONTENTS



MATLAB数组定义与创建



数组操作方法



MATLAB处理向量



矩阵分析与处理



矩阵分解



矩阵方程的MALAB求解



# 1 数组定义与大小

➤ 数组是MATLAB的基本构件，数组中的单个数据项称为元素。任何变量在MATLAB中都是以数组形式存储和运算。

— 包括：数值数组，字符数组，元胞数组，结构数组等

1、`sz = size(A)` %返回一个n维向量，包含了数组每一维的大小

2、`[rows, cols] = size(A)` %返回具有所要求数量的数组A的各维大小

3、`lt = length(A)` %返回数组的最大维值

```
>> load('BreastTissuedata.mat')
>> A = data;
>> sz = size(data)
sz =
    569    32
>> [rows, cols] = size(data)
rows =
    569
cols =
    32
>> lt = length(data)
lt =
    569
```

## 2 输入数据创建数组

- $A = [1,2,3,4; 5,6,7,8; 9,10,11,12]$  %使用 “;” 表示另起一行，行内元素以 “,” 分隔;
- $A = [1\ 2\ 3\ 4; 5\ 6\ 7\ 8; 9\ 10\ 11\ 12]$  %使用 “;” 表示另起一行，行内元素以 “空格” 分隔;
- $A = [1\ 2\ 3\ 4$   
5 6 7 8  
9 10 11 12] %另起一行表明一行结束，行内元素以 “空格” 分隔。

构建线性规划问题的约束系数矩阵;

定义常微分方程组函数

- 结构体是多维 MATLAB 数组，包含可按文本字段标志符访问的元素。结构体通过字段 (fields) 来对元素进行索引，在访问时只需通过点号来访问数据变量。

```
编辑器 - D:\demo\matlab2020file\structmat.m
structmat.m
1 %% 结构数组创建
2 %第一种形式创建
3 S.name = '乔峰';
4 S.score = 95;
5 S.grade = 'A+' ;
6 S(2).name = '虚竹';
7 S(2).score = 80;
8 S(2).grade = 'B' ;
9 %第二种形式创建
10 S(3) = struct('name','段誉','score',70,'grade','C');
11 %% 结构数组引用
12 S %输出结构数组的内容
13 elem1 = S(1) %结构体的索引，输出第一个结构体元素
14 elemname = S(2).name %输出第二个元素的name值
15 scores = [S.score] %取出结构体元素值，构成数组，进行运算
16 sm = mean(scores) %均值
```

```
命令行窗口
>> structmat
S =
    包含以下字段的 1×3 struct 数组:
        name
        score
        grade
elem1 =
    包含以下字段的 struct:

        name: '乔峰'
        score: 95
        grade: 'A+'
elemname =
    虚竹
scores =
        95        80        70
sm =
    81.6667
```

元胞数组是MATLAB的一种特殊数据类型，可以将元胞数组看做一种无所不包的通用矩阵，或者叫做**广义矩阵**。

组成元胞数组的元素可以是**任何一种数据类型的常数或者常量**，每一个元素也可以具有**不同的尺寸和内存占用空间**，每一个元素的**内容也可以完全不同**，所以元胞数组的元素叫做**元胞 (cell)**。

与一般的数值矩阵一样，元胞数组的内存空间也是**动态分配**的。

元胞数组的定义可以用两种方法来实现，一种是用赋值语句直接定义，另一种是由cell函数预分配存储空间，而后对元胞元素逐个赋值。

# 元胞数组示例

```
>> a=[1,3,5;2,4,6];  
>> b=[1,3;5,7;9,11];  
>> XC={a,b,[1,2,3,4;5,6,7,8]};'love',1:3,a*b} %第一种创建形式  
XC =
```

2×3 cell 数组

[2×3 double] [3×2 double] [2×4 double]

'love' [1×3 double] [2×2 double]

```
>> y=cell(2,3); %第二种创建形式
```

```
>> y{1,1}=a; y{1,2}=b; y{1,3}=[1,2,3,4;5,6,7,8];
```

```
>> y{2,1}='love'; y{2,2}=1:3; y{2,3}=a*b;
```

```
>> y
```

y =

2×3 cell 数组

[2×3 double] [3×2 double] [2×4 double]

'love' [1×3 double] [2×2 double]

%获取指定元胞大小，用小括号“()”

```
>> XC(1,2)
```

ans =

cell

[3×2 double]

%获取指定元胞的内容，用大括号“{}”

```
>> XC{1,2}
```

ans =

1 3

5 7

9 11

%进一步获取指定元胞的内容

```
>> XC{1,2}(2,2)
```

ans =

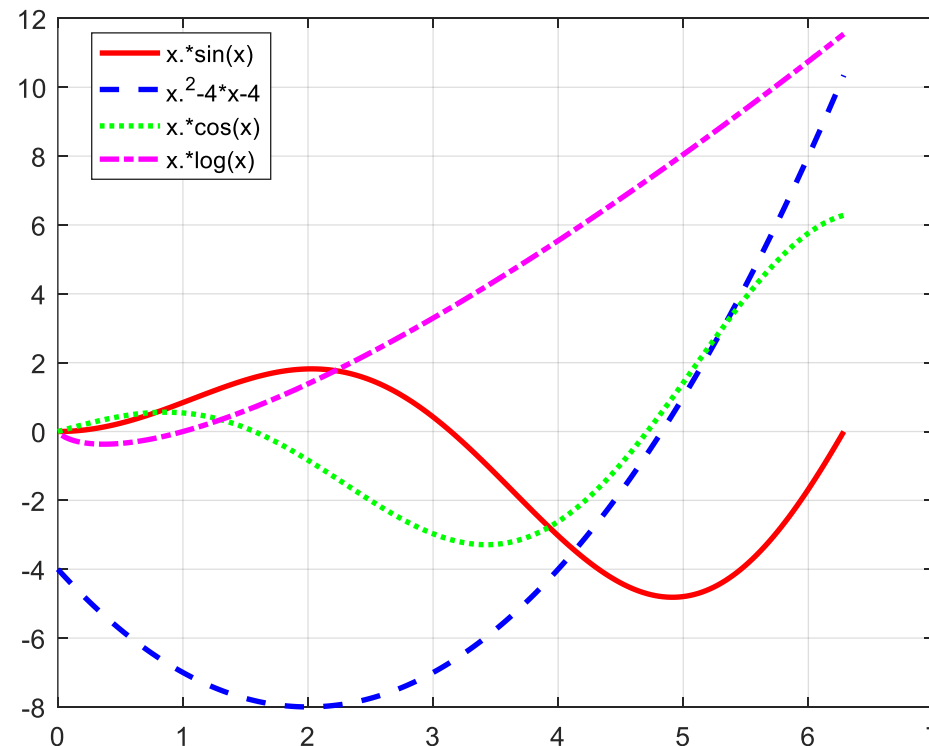
7



序号	函数名	说明
1	cell	生成元胞数组
2	cellfun	对元胞数组变量中的元素使用函数fun
3	celldisp	显示元胞数组变量的内容
4	cellplot	用图形显示元胞数组变量的内容
5	num2cell	将数值数组转换为元胞数组
6	deal	输入/输出处理， cell中、结构体中的多个房间分别赋给一个矩阵； 或者把n个矩阵通过一个语句就赋值到对应的n个新矩阵
7	cell2struct	将元胞数组转换为结构数组
8	struct2cell	将结构数组转换为元胞数组
9	iscell	判断是否为元胞数组变量
10	reshape	改变元胞数组的结构

# 元胞数组示例

```
x = 0:pi/100:2*pi; %x从[0, 2pi]之间等分100份
y(1,:) = x.*sin(x); %第一个函数, 且计算值
y(2,:) = x.^2-4*x-4;
y(3,:) = x.*cos(x);
y(4,:) = x.*log(x);
colors = {'r','b','g','m'}; %颜色元胞数组
linestyle = {'-','--',':','-.'}; %线形元胞数组
led = {'x.*sin(x)','x.^2-4*x-4','x.*cos(x)','x.*log(x)'}; %图例元胞数组
for i = 1:4 %循环绘制4条曲线
    plot(x,y(i,:),strcat(colors{i},linestyle{i}),'LineWidth',2);
    hold on
end
legend(led,'Location','best') %添加图例
grid on; hold off
```



# 3 利用函数创建矩阵

函数	说明
<code>A_zeros = zeros(m,n)</code>	产生填充值为0的m行n列的数组。
<code>A_ones = ones(m,n)</code>	产生填充值为1的m行n列的数组。
<code>A_eye = eye(m,n)</code>	生成m行n列的单位矩阵。
<code>A_rand = rand(m,n)</code>	产生取值为0~1之间的随机数的数组（均匀分布）。
<code>A_randn = randn(m,n)</code>	产生均值0、方差1的随机数的数组（正态分布）。
<code>V_A = diag(A)</code>	A为数组，A的对角线上元素作为向量返回。 <code>diag(A,k)</code> ，默认k=0，k>0上第k条对角线，k<0下
<code>A_diag = diag(V_A)</code>	V_A为向量，返回一条对象线为V_A的方阵。
<code>A_magic = magic(m)</code>	产生一个填充1到m <sup>2</sup> 之间数字的数组，行、列、对角线元素加起来等于相同的数。
<code>A = [1 2 3 4; linspace(0,5,4); 1:2:7; 3:6]</code>	通过一维数组的方法创建二维数组。

### 3 利用函数创建矩阵

- **Hankel (汉克尔) 矩阵**: 是对称矩阵, 其特点是**每条反对角线**上所有元素都相同。  
在数字信号处理、数值计算、系统控制等领域均有广泛的应用。
- 调用格式:
  - **$H = \text{hankel}(c,r)$** : 首先将H矩阵**第一列**的各个元素定义为c向量, 将**最后一行**各个元素定义为r。
- 如生成Hankel矩阵

$$H_1 = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{bmatrix}, \quad H_2 = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 0 \\ 3 & 0 & 0 \end{bmatrix}$$

```
>> c = [1:3]; %取第一列
>> r = [3:9]; %取最后一行
>> H = hankel(c,r)
H =
     1     2     3     4     5     6     7
     2     3     4     5     6     7     8
     3     4     5     6     7     8     9
>> H2 = hankel(c)
H2 =
     1     2     3
     2     3     0
     3     0     0
```

### 3 利用函数创建矩阵

- **Hilbert (希尔伯特) 矩阵及Hilbert逆矩阵**: 特殊矩阵, 是一种数学变换矩阵, 正定, 且高度病态 (即, 任何一个元素发生一点变动, 整个矩阵的行列式的值和逆矩阵都会发生巨大变化), 病态程度和阶数相关。它的第 $(i, j)$ 元素的值满足 $h_{i,j} = 1/(i + j - 1)$

$$H = \begin{bmatrix} 1 & 1/2 & 1/3 & \dots & 1/n \\ 1/2 & 1/3 & 1/4 & \dots & 1/(n+1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1/n & 1/(n+1) & 1/(n+2) & \dots & 1/(2n-1) \end{bmatrix}$$

- 调用格式:  $A = \text{hilb}(n)$ ;  $B = \text{invhilb}(n)$  %Huilbert逆
- 由于Hilbert矩阵本身接近奇异的性质, 所以在处理矩阵时建议采用符号运算工具箱, 而采用数值解时应该检验结果的正确性。

### 3 利用函数创建矩阵

- Vandermonde (范德蒙) 矩阵 (各列为几何级数的矩阵)**：假设有一个序列 $\mathbf{c}$ ，其各个元素满足 $\{c_1, c_2, \dots, c_n\}$ ，则可以写出一个矩阵，其第 $(i, j)$ 元素满足

$$V = \begin{bmatrix} c_1^{n-1} & c_1^{n-2} & \dots & c_1 & 1 \\ c_2^{n-1} & c_2^{n-2} & \dots & c_2 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ c_n^{n-1} & c_n^{n-2} & \dots & c_n & 1 \end{bmatrix}$$

$$V = \begin{bmatrix} 1 & 1 & \dots & 1 \\ c_1 & c_2 & \dots & c_n \\ c_1^2 & c_2^2 & \dots & c_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ c_1^{n-1} & c_2^{n-1} & \dots & c_n^{n-1} \end{bmatrix}$$

- 调用格式： $\mathbf{V} = \text{vander}(\mathbf{c})$ ，%已知向量 $\mathbf{c} = [c_1, c_2, \dots, c_n]$

```
>> c = [1,2,3,4,5];  
>> V = vander(c)
```

行列式值  $V_n = \prod_{1 \leq j < i \leq n} (x_i - x_j)$

### 3 利用函数创建矩阵

- **相伴矩阵（友矩阵）**：矩阵标准形理论中一类重要的矩阵，是数域F上首项系数为1的多项式所对应的特定形式n阶矩阵。友矩阵的特征根多项式是首一多项式。

假设有一个首一化的多项式  $p(s) = s^n + a_1s^{n-1} + a_2s^{n-2} + \dots + a_{n-1}s + a_n$ ，则相伴矩阵为

$$A_c = \begin{bmatrix} -a_1 & -a_2 & \dots & -a_{n-1} & -a_n \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}$$

- 调用格式：  $A = \text{compan}(a)$ ，%a为降幂排列的系数向量
- 考虑一个多项式  $p(s) = 2s^4 + 4s^2 + 5s + 6$ ，写出相伴矩阵。

```
>> P = [2 0 4 5 6];  
>> A = compan(P)  
A =  
      0   -2.0000   -2.5000   -3.0000  
  1.0000      0      0      0  
      0   1.0000      0      0  
      0      0   1.0000      0  
  
>> p = poly(A)  
p =  
      1.0000      0.0000      2.0000      2.5000      3.0000  
  
>> po = 2*poly2sym(p)  
po =  
2*x^4 + x^3/562949953421312 + 4*x^2 + 5*x + 6
```

### 3 利用函数创建矩阵

- 稀疏矩阵:  $A = \text{sparse}(i,j,s,m,n,nzmax)$ 
  - 其中 $i, j, s$ 都是向量,  $i, j$ 分别为非零元素的行下标和列下标构成的列向量,  $i(k)$ 和 $j(k)$ 作为两个下标标记非零元素在矩阵中的位置, 这个位置上的元素值为 $s(k)$ .
  - $m, n$ 表明 $A$  是  $m \times n$  阶矩阵 (可省略);
  - $nzmax$  用于指定 $A$  中非零元素所用存储空间大小 (可省略)。最简单的使用方式是只输入非零元的数据以及各非零元的行指标和列指标。

```
>> i=[1 3 1 2 1 4]; j=[1 1 2 2 3 4]; s = [1 2 3 4 5 6];
```

```
>> S = sparse(i,j,s,10,10)
```

```
>> A = full(S) %显示矩阵形式
```



### 3 利用函数创建矩阵

- 稀疏带状矩阵的创建  $S = \text{spdiags}(B, d, m, n)$ 
  - 其中 $m$ 和 $n$ 分别是矩阵的行数和列数； $d$ 是长度为 $p$ 的整数向量，它指定矩阵 $S$ 的对角线位置； $B$ 是全元素矩阵，用来给定 $S$ 对角线位置上的元素，行数为 $\min(m, n)$ ，列数为 $p$ 。

```
>> e = ones(10,1); % 写成n*1的形式  
>> A = spdiags([e,4*e,e],[-1,0,1],10,10) % 头一  
个大括号分别对应是3个不同非零对角线元素，  
按从下往上排列；第二个大括号对应于每个对  
角线相对于中央对角线的位置，后面的两个数  
就是要构造的系数矩阵的行列数  
>> SA = full(A)
```

```
>> e = [1:10]';  
>> A = spdiags([e, 4*e, e], [-1, 0, 1], 10, 10);  
>> SA = full(A)  
SA =  
     4     2     0     0     0     0     0     0     0     0  
     1     8     3     0     0     0     0     0     0     0  
     0     2    12     4     0     0     0     0     0     0  
     0     0     3    16     5     0     0     0     0     0  
     0     0     0     4    20     6     0     0     0     0  
     0     0     0     0     5    24     7     0     0     0  
     0     0     0     0     0     6    28     8     0     0  
     0     0     0     0     0     0     7    32     9     0  
     0     0     0     0     0     0     0     8    36    10  
     0     0     0     0     0     0     0     0     9    40
```

## 4 从外部文件导入数据

Excel文件

A

`[num,txt] = xlsread(filename,sheet,xlRange)` %num表示数值数据, txt表示文本数据;

CSV文件或  
dat文件

B

`M = csvread(filename,R1,C1,[R1 C1 R2 C2])` %要求文件中数据都是数值型数据;

txt文本文件

C

`[A,B,C,...] = textread(filename,format,N)` %通过help命令查看说明。format:  
%d%s%u%f%q%c...

```
>> [num,txt] = xlsread('D:\ data_apriori.xls ');
>> txt = txt(:,7:10);
>> [num,txt] = xlsread('D:\data_apriori.xls',1,'A2:J931'); %读取指定范围
>> M = csvread('D:\ data_income.csv',1,0); %csvread读取的数据只能为数值类型
>> [a1,a2,a3,a4,a5,a6,a7] = textread('D:\apriori.txt','%s%s%s%s%s%s%s'); %按文件数据格式, 确定是否指定分隔符, 读取为一列
>> [a1,a2,a3,a4,a5,a6,a7] = textread('D:\apriori.txt','%s%s%s%s%s%s%s','delimiter',' ');
>> apriori = [a1,a2,a3,a4,a5,a6,a7];
```



---

# 感谢聆听

---