



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17日

# 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. 数组的寻址

函数表达式	函数功能
A(k)	单一索引向量k来寻找A的子数组，如果k是逻辑数组，则用来寻找A的子数组，k的维数和A的维数必须一致
A(:, j)	返回二维矩阵A中第j列列向量
A(i, :)	返回二维矩阵A中第i行行向量
A(:, j:k)	返回二维矩阵A中第j列到第k列，列向量组成的子矩阵
A(i:k, :)	返回二维矩阵A中第i行到第k行，行向量组成的子矩阵
A(i:k, j:m)	返回A中第i到第k行、第j到第m列组成的子矩阵
A(:)	将二维矩阵A中的每列合并成一个列向量
A(j:k)	返回一个行向量，其元素为A(:)中第j个元素到第k个元素

```
>> A = magic(5) %生成一个5阶魔方矩阵
A =
    17    24     1     8    15
    23     5     7    14    16
     4     6    13    20    22
    10    12    19    21     3
    11    18    25     2     9
>> ind = [3 8 6 7 10, 15, 17, 25]; %设定取值索引位置
>> subA = A(ind)
subA =
     4     6    24     5    18    25    14     9
>> k = logical([1 0 1 0 0 1 0 0 1 0 1]); %逻辑向量
>> subA2 = A(k)
subA2 =
    17     4    24    12     1
>> A(:, 2) %取矩阵所有行的第二列
ans =
    24
     5
     6
    12
    18
>> A(3, :) %取矩阵第三行的所有列
ans =
     4     6    13    20    22
```

# 1. 数组的寻址



```
>> A = magic(8) %生成一个8阶魔方矩阵
```

```
A =  
    64     2     3    61    60     6     7    57  
     9    55    54    12    13    51    50    16  
    17    47    46    20    21    43    42    24  
    40    26    27    37    36    30    31    33  
    32    34    35    29    28    38    39    25  
    41    23    22    44    45    19    18    48  
    49    15    14    52    53    11    10    56  
     8    58    59     5     4    62    63     1
```

```
>> subA3 = A(:, 2:5) %返回二维矩阵A中第j列到第k列，列向量组成的子矩阵
```

```
subA3 =  
     2     3    61    60  
    55    54    12    13  
    47    46    20    21  
    26    27    37    36  
    34    35    29    28  
    23    22    44    45  
    15    14    52    53  
    58    59     5     4
```

```
>> subA4 = A(2:5, :) %返回二维矩阵A中第2行到第5行，行向量组成的子矩阵
```

```
subA4 =  
     9    55    54    12    13    51    50    16  
    17    47    46    20    21    43    42    24  
    40    26    27    37    36    30    31    33  
    32    34    35    29    28    38    39    25
```

```
>> subA5 = A(2:5, 2:6) %返回A中第i到第k行、第j到第m列组成的子矩阵
```

```
subA5 =  
    55    54    12    13    51  
    47    46    20    21    43  
    26    27    37    36    30  
    34    35    29    28    38
```

```
>> subA6 = A(15:25) %返回一个行向量，其元素为A(:)中第j个元素到第k个元素
```

```
subA6 =  
    15    58     3    54    46    27    35    22    14    59    61
```

```
>> A(:) %将二维矩阵A中的每列合并成一个列向量
```

```
ans =  
    64  
     9  
    17  
    40  
    32  
    41  
    49  
     8  
     2  
    55  
    47  
    26  
    34  
    23  
    15  
    58
```

# 1. 数组的寻址

函数表达式	函数功能
A([j1,j2 ...])	返回一个行向量，其元素为A(:)中第j1,j2...个元素
A(:, [j1,j2...])	返回矩阵A中的第j1,j2...列的列向量组成的子矩阵
A([i1,i2...], :)	返回矩阵A中的第i1,i2...行的行向量组成的子矩阵
A([i1,i2...], [j1,j2...])	返回矩阵A的第j1,j2...列、第i1,i2...行的元素
A(i,j) = k	设置二维数组的元素数值
A(:,j) = k	改变二维数组的一列元素数值
reshape(A,m,n)	通过reshape函数生成新的数组（对原数组的抽取是按照列抽取的）
B(:j) = []	通过空赋值语句删除数组元素
repmat(A,m,n)	矩阵的复制，把A矩阵复制成m行n列，每一个元素为A矩阵

```
>> A([5:5:50]) %返回一个行向量，其元素为A(:)中第j1, j2...个元素
ans =
    32     55     15     27     61     44     21     4     38     50

>> A(:, [3, 6, 8]) %返回矩阵A中的第j1, j2...列的列向量组成的子矩阵
ans =
     3     6     57
    54    51    16
    46    43    24
    27    30    33
    35    38    25
    22    19    48
    14    11    56
    59    62     1

>> A([3, 6, 8], :) %返回矩阵A中的第i1, i2...行的行向量组成的子矩阵
ans =
    17    47    46    20    21    43    42    24
    41    23    22    44    45    19    18    48
     8    58    59     5     4    62    63     1

>> A([3, 6, 8], [3, 6, 8]) %返回矩阵A的第j1, j2...列、第i1, i2...行的元素
ans =
    46    43    24
    22    19    48
    59    62     1
```

# 1. 数组的寻址

```
>> A = magic(4) %生成一个4阶魔方矩阵
```

```
A =  
    16     2     3    13  
     5    11    10     8  
     9     7     6    12  
     4    14    15     1
```

```
>> A(3,3) = 100 %设置矩阵A第三行第三列的值为100
```

```
A =  
    16     2     3    13  
     5    11    10     8  
     9     7    100    12  
     4    14    15     1
```

```
>> A(:,4) = 100 %改变二维数组的一列元素数值
```

```
A =  
    16     2     3    100  
     5    11    10    100  
     9     7    100    100  
     4    14    15    100
```

```
>> B = reshape(A, 2, 8) %通过reshape函数生成新的数组（对原数组的抽取是按照列抽取的）
```

```
B =  
    16     9     2     7     3    100    100    100  
     5     4    11    14    10    15    100    100
```

```
>> B(:, [2, 4, 6, 8]) = [] %删除矩阵第2、4、6、8列的值
```

```
B =  
    16     2     3    100  
     5    11    10    100
```

```
>> A = magic(4) %生成一个4阶魔方矩阵
```

```
A =  
    16     2     3    13  
     5    11    10     8  
     9     7     6    12  
     4    14    15     1
```

```
>> C = repmat(A, 3, 2) %矩阵的复制，把A矩阵复制成m行n列，每一个元素为A矩阵
```

```
C =  
    16     2     3    13    16     2     3    13  
     5    11    10     8     5    11    10     8  
     9     7     6    12     9     7     6    12  
     4    14    15     1     4    14    15     1  
    16     2     3    13    16     2     3    13  
     5    11    10     8     5    11    10     8  
     9     7     6    12     9     7     6    12  
     4    14    15     1     4    14    15     1
```

## 2. 数组的连接与翻转

函数	说明
<code>C = [A;B]</code>	用于垂直方向连接具有相同列数的数组A和B。
<code>D = [A B]</code>	<code>D=[A,B]</code> 用于水平方向连接具有相同行数的数组。
<code>cat(1,A,B)</code>	<code>cat(1,A,B)</code> 相同于 <code>[A;B]</code>
<code>cat(2,A,B)</code>	<code>cat(2,A,B)</code> 相同于 <code>[A B]</code>
<code>horzcat(A,B)</code>	用于水平方向连接数组A和B
<code>vertcat(A,B)</code>	用于垂直方向连接数组A和B
<code>rot90(A)</code>	按逆时针将矩阵A旋转90°，形成矩阵B。
<code>rot90(A,k)</code>	逆时针旋转 $k \times 90^\circ$ ，k可为负数，顺时针旋转。
<code>fliplr(A)</code>	把矩阵左右翻转形成矩阵B。
<code>flipud(A)</code>	把矩阵从上到下的方向翻转矩阵，即沿水平轴翻转，形成B。
<code>flipdim(A,dim)</code>	沿指定维数dim翻转成B，dim=1相当于flipud，dim=2相当于fliplr。

```
>> A = magic(4) %生成一个4阶魔方矩阵
A =
    16     2     3    13
     5    11    10     8
     9     7     6    12
     4    14    15     1

>> B = round(randn(4)) %对4阶N(0,1)四舍五入取整
B =
     1     0     1     1
     0    -1    -1     0
     1     3     0     1
    -2     1     0    -2

>> C = [A;B] %垂直方向连接，等价于cat(1,A,B)，等价于vertcat(A,B)
C =
    16     2     3    13
     5    11    10     8
     9     7     6    12
     4    14    15     1
     1     0     1     1
     0    -1    -1     0
     1     3     0     1
    -2     1     0    -2

>> D = [A,B] %水平方向连接，等价于cat(2,A,B)，等价于horzcat(A,B)
D =
    16     2     3    13     1     0     1     1
     5    11    10     8     0    -1    -1     0
     9     7     6    12     1     3     0     1
     4    14    15     1    -2     1     0    -2
```

### 3. 去除重复元素与排序

序号	函数	说明
1	C = unique(A)	去除矩阵A中重复的元素，返回无重复元素的A到新变量C，C以向量形式存在，并按从小到大的顺序排列A中无重复的元素
2	C = unique(A,'rows')	去除矩阵A中的重复行，返回新矩阵C。
3	[B,m,n] = unique(A, 'rows')	也返回索引向量m和n，使得B = A(M)和A= B(N)。m的每一个元素是最大的下标，使得B = A(M)。对于行的组合，B = A(M,:)和 A = B(N,:)。
4	[Y,I] = sort(X,DIM,MODE)	Y表示对X排序后的结果，I表示Y中对应元素原来在X中的下标。 mode的默认值是'ascend'升序，'descend'为降序；DIM的默认值是1，如果X是矩阵，则默认对矩阵的各个列进行升序排列，即sort(X,1)与sort(X)等效，sort(X,2)表示对矩阵的各行中的元素按照升序排列。
5	[Y,I]=sortrows(X,Colnum)	sortrows函数可以使用矩阵的某列值的大小对矩阵行进行排序，就像excel 中按照某列排序并扩展选定区域一样。 X是待排序的矩阵，Colnum 是列的序号，指定按照第几列进行排序，整数表示按照升序进行排序，负数表示按照降序进行排序。



### 3. 去除重复元素与排序

```
>> A = round(randn(6)) %对6阶N(0, 1)四舍五入取整
A =
     0     1    -1     1    -2     1
    -1     1    -1     1    -2    -1
    -1     0     1     0     0     0
     1     1    -2     0    -1     0
     0     1     0    -2     0     1
     0     0    -2     1     1     1

>> B = unique(A) %去除重复元素，并从小到大排序成列向量
B =
    -2
    -1
     0
     1

>> A(3, :) = A(1, :); %矩阵第3行赋值为矩阵第一行元素
>> A(5, :) = A(1, :); %矩阵第5行赋值为矩阵第一行元素
>> C = unique(A, 'rows') %去除重复行
C =
    -1     1    -1     1    -2    -1
     0     0    -2     1     1     1
     0     1    -1     1    -2     1
     1     1    -2     0    -1     0
```

```
>> [D, m, n] = unique(A, 'rows') %去除重复行，并赋值不同变量
D =
    -1     1    -1     1    -2    -1
     0     0    -2     1     1     1
     0     1    -1     1    -2     1
     1     1    -2     0    -1     0

m =
     2
     6
     1
     4

n =
     3
     1
     3
     4
     3
     2
```

m为D在A中的线性索引值，n为A在D中的线性索引值，如A中第一行值在D中标号为第3行。

### 3. 去除重复元素与排序



```
>> A = magic(5) %生成一个5阶魔方矩阵
```

```
A =  
    17    24     1     8    15  
    23     5     7    14    16  
     4     6    13    20    22  
    10    12    19    21     3  
    11    18    25     2     9
```

```
>> [Y, I] = sort(A) %Y表示对X排序后的结果, I 表示Y中对应元素原来在X中的下标。
```

```
Y =  
     4     5     1     2     3  
    10     6     7     8     9  
    11    12    13    14    15  
    17    18    19    20    16  
    23    24    25    21    22
```

```
I =  
     3     2     1     5     4  
     4     3     2     1     5  
     5     4     3     2     1  
     1     5     4     3     2  
     2     1     5     4     3
```

```
>> [tmp, idx] = sort(A(:, 4)) %对矩阵A第4列排序
```

```
tmp =  
     2  
     8  
    14  
    20  
    21  
idx =  
     5  
     1  
     2  
     3  
     4
```

```
>> [AS, idx] = sort(sort(A), 2) %先对矩阵A进行按列排序, 然后再按照行排序
```

```
AS =  
     1     2     3     4     5  
     6     7     8     9    10  
    11    12    13    14    15  
    16    17    18    19    20  
    21    22    23    24    25
```

```
idx =  
     3     4     5     1     2  
     2     3     4     5     1  
     1     2     3     4     5  
     5     1     2     3     4  
     4     5     1     2     3
```

```
>> [YS, J] = sortrows(A, 2) %使用矩阵的第2列值的大小对矩阵行进行排序
```

```
YS =  
    23     5     7    14    16  
     4     6    13    20    22  
    10    12    19    21     3  
    11    18    25     2     9  
    17    24     1     8    15
```

```
J =  
     2  
     3  
     4  
     5  
     1
```

# 4. 子数组搜索

序号	函数示例	说明
1	<code>ind = find(X)</code>	找出矩阵X中的所有非零元素，并将这些元素的线性索引值（linear indices：按列）返回到向量ind中。 如果X是一个行向量，则ind是一个行向量；否则，ind是一个列向量。如果X不含非零元素或是一个空矩阵，则ind是一个空矩阵。
2	<code>ind = find(X, k, 'first')</code>	找到前k个不为0的线性索引值。k必须是一个正数，但是它可以是任何数字数值类型。
3	<code>ind = find(X, k, 'last')</code>	找到后k个不为零元素的线性索引值。
4	<code>[row,col] = find(X, ...)</code>	返回矩阵X中非零元素的行和列的索引值。这个语法对于处理稀疏矩阵尤其有用。
5	<code>[row,col,v] = find(X, ...)</code>	返回X中非零元素的一个列或行向量v，同时返回行和列的索引值。如果X是一个逻辑表示，则v是一个逻辑矩阵。输出向量v包含通过评估X表示得到的逻辑矩阵的非零元素。
6	<code>[Y,I] = max(X,[ ],DIM)</code> <code>[Y,I] = min(X,[ ],DIM)</code>	返回X中有dim（1为列，2为行）指定的维数范围中的最大值。返回行向量Y和I，Y向量记录X的每列的最大值，I向量记录每列最大值的行号。

## 4. 子数组搜索

```
>> A = magic(5) %生成一个5阶魔方矩阵
A =
    17     24     1     8    15
    23     5     7    14    16
     4     6    13    20    22
    10    12    19    21     3
    11    18    25     2     9

>> [rows,cols] = find(A >= 21) %按照双索引值搜索大于等于21的元素
rows =
     2
     1
     5
     4
     3
cols =
     1
     2
     3
     4
     5

>> k = find(A >= 21) %按照单一索引值来搜索
k =
     2
     6
    15
    19
    23
```

```
>> fir_max = find(A >= 21,1) %搜索第一个大于21的元素
fir_max =
     2

>> last_max = find(A >= 21,2,'last') %搜索后两个数值大于21的元素
last_max =
    19
    23

>> [max_x,ind_x] = max(A) %搜索每一列的最大值
max_x =
    23    24    25    21    22
ind_x =
     2     1     5     4     3

>> [min_x,ind_x] = min(A) %搜索每一列的最小值
min_x =
     4     5     1     2     3
ind_x =
     3     2     1     5     4
```



---

# 感谢聆听

---