



MATLAB程序设计与应用

第二章数据构造及其运算

重要内容

- 2.1 数据类型
- 2.2 一维数组
- 2.3 二维数组
- 2.4 高维数组
- 2.5 数组操作
- 2.6 数组运算与矩阵运算
- 2.7 多项式
- 2.8 关系运算、逻辑运算和运算符
- 2.9 字符串数组
- 2.10 细胞数组
- 2.11 构造体数组

2.1 引言

数据构造是程序设计的重要基础，使用合理的数据构造去描述问题，能够缩短程序代码、简化程序构造、便于程序维护。

在MATLAB里共有六种基本数据类型，每一种类型能够构成一维、二维和多维的数组。这六种是：

双精度型（**double**）：双精度数值类型，是最惯用的类型；

字符型（**char**）：字符数组，每个字符占16位；

稀疏型（**sparse**）：双精度稀疏矩阵，只存储矩阵中的非0元素；

细胞型（**cell**）：细胞，能够寄存任意类型数据

构造体（**struct**）：不同类型的数据集合

存储型（**storage**）：用于图像解决

2.1 引言

- 数组（**Array**）能够是一维的行（或列），也能够是二维或多维的。顾客能够操作整个数组，也能够操作数组中的某个或者某些元素。
- **MATLAB**会根据体现式的运算成果，自动拟定变量的类型和大小。变量的数据类型能够用下列函数来查看：
 - **isa (var, 'type')** %变量**var**的数据类型名称如果是**type**，则返回**1**，否则返回**0**
 - **class(var)** %返回变量**a**的数据类型名称
 - **whos var** %查看变量**var**的具体状况

2.2 一维数组

一、一维数组的创立

1. 逐个元素输入法
2. 冒号运算符法

【阐明】

冒号运算符的格式是：**startv : step : endv**

startv是初值，即数组的第一种元素值。

endv是终值，即数组的最后一种元素值。

step称为步长，即数组元素每次增加的值；

步长**step**能够省略不写，此时默认步长为1；

step可觉得负值，此时规定**startv > endv**。



2.2 一维数组

3、线性分隔法

【说明】

➤ **linspace**函数的调用格式为：**x=linspace(a, b, n)**

➤ 数组的第一个元素值为**a**，最后一个元素值为**b**，数组中共有**n**个元素，这**n**个元素线性均匀分布于**a**和**b**之间，即数组元素依次为

$$a + \frac{b-a}{n-1} * i, \quad i = 0, 1, \dots, n-1$$

4、对数分隔法

【说明】

➤ **logspace**函数的调用格式为：**x=logspace(a, b, n)**

➤ 数组的第一个元素值为 10^a ，最后一个元素值为 10^b ，数组中共有**n**个元素，这**n**个元素的以**10**为底的对数值均匀分布与**a**和**b**之间，即数组元素依次为

$$10^{a + \frac{b-a}{n-1} * i}, \quad i = 0, 1, \dots, n-1$$

2.2 一维数组

二、一维数组的访问

一维数组的访问遵照下列商定：

用下标方式访问数组元素，下标要用一对圆括号()引发来；

下标代表的是元素在数组中的位置序号，从1开始，最大值为数组中元素的个数；

下标能够是常量，也能够是变量；

能够访问数组中的单个元素，也能够访问数组中的某些元素，即数组的子数组。

例2.2.1

2.3 二维数组

一、二维数组的建立

1. 逐个输入数组元素值

如果数组内元素数量少，能够直接从键盘逐个输入元素的值，需要遵照的规则是：

整个数组必须用中括号“[]”括起来

数组的行与行之间用分号“;”分隔，或者用回车符分隔

每行之间的元素必须用逗号“,”或者空格分隔

分隔符必须是英文字符，即在英文状态下输入分号、

括号、**例2.3.1**方括号、逗号等

2.3 二维数组

2. 运用M文献

如果数组元素诸多，或者元素值要经常变化，我们能够采用M文献来输入和保存数组。

用M文献实现对数组 x 的输入和保存，办法以下：

- (1) 在现在目录下，用程序编辑器建立一种名为**MyData.m**的文献；
- (2) 在编辑器中输入 x 内容；
- (3) 保存**MyData.m**文献；
- (4) 在命令窗口键入**MyData**，就能够在内存中建立数组 x 并读入数组元素的值。

例2.3.2

2.3 二维数组

二、二维数组的访问

二维数组的访问遵照下列商定：

用下标方式访问数组元素，下标要用一对圆形括号()引发来；
用双下标方式访问数组元素，格式为(r,c)，其中r为二维数组的行下标，c为二维数组的列下标，下标之间用逗号分隔；

用单下标方式访问二维数组，二维数组的单下标是按照列优先规则排序的，即二维数组被看作是从第一列开始从左到右依次将各列首位连接而成的一维数组，单下标表达元素在这个一维数组中的位置；

单下标和双下标含有对应关系，其值能够通过

ind2sub和**sub2ind**函数进行转换；
 The MathWorks

能够访问一维数组的某个元素及其子数组。可

例2.3.3



2.3 二维数组

【阐明】有关空数组

某一维长度为0的数组称为空数组；

空数组用[]表达，表达数组中没有元素，但能够表达计算成果为“空”；

仅仅能用`isempty`函数对的判断数组与否为空；

能够通过给数组元素赋值空数组来缩小数组的大小；

尽量不要用空数组参加逻辑运算和关系运算；

例2.3.4

2.4 高维数组

一、高维数组的创立

能够采用下列办法创立高维数组：

直接通过全下标方式进行元素赋值；

用低维数组合成高维数组；

用数组生成函数（**ones/zeros/rand**等）生成高维数组；

用数组操作函数（**repmat/reshape**等）构造高维数组。

例2.4.1

2.4 高维数组

二、多维数组的访问

对于高维数组的访问，有下列商定：

能够通过全下标方式访问。对于三维数组来说，第一维下标称为“行下标”，第二维下标称为“列下标”，第三维下标普通称为“页下标。”

能够通过单下标方式访问。高维数组的单下标是按照后维优先的次序排列的，对于三维数组来说，先排列“页”，页内先排列“列”，列内再排列“行”，即第1行第1列第1页的元素单下标为1，然后先变化行下标，再变化列下标，最后变化页下标。

数组的维数通过 `ndims` 函数获取。

例2.4.2



洛阳师范学院

2.5 数组操作

一、原则数组的生成

数学中定义了诸多原则数组或者矩阵，如全1数组、全零数组、对角阵等，在Matlab中有对应的函数用来生成这些原则数组。

1. ones

功效：生成全1数组，即数组中的元素都为1。

格式： $Y = \text{ones}(n)$

生成 $n \times n$ 的全1矩阵

$Y = \text{ones}(m_1, m_2, \dots, m_k)$

生成

$m_1 \times m_2 \times \dots \times m_k$ 的全1数组

$Y = \text{ones}(\text{size}(A))$

生成和数组A同样尺寸的

2.5 数组操作

3. rand

功效：生成均匀分布随机数组。

格式：**Y=rand('state',v)** 设立随机发生器的初始状态为v

其它同ones函数

4. randn

功效：产生正态分布随机数组。

格式：同rand函数

5. magic

功效：产生魔方矩阵,不合用于高维数组。

格式：**M=magic(n)** 产生n×n的魔方矩阵

2.5 数组操作

6. eye

功效：产生单位矩阵，即主对角线元素都为1而其它元素都为0的二维数组

格式： $Y = \text{eye}(n)$

$Y = \text{eye}(n,m)$

$Y = \text{eye}(\text{size}(A))$

$Y = \text{eye}(m,n,\text{classname})$

阐明：**classname**是字符串，表达元素的数据类型名称，可取'double', 'single', 'int8', 'uint8', 'int16', 'uint16', 'int32', 'uint32', 'int64', 'uint64'。

2.5 数组操作

7. diag

功效：产生对角阵，即矩阵的某个对角线元素不全为0，其它元素为0。

格式： $X = \text{diag}(v,k)$

生成 $\text{length}(v)+k$ 阶方阵，并在第 k 条对角线放置元素 v

$X = \text{diag}(v)$

生成 $\text{length}(v)$ 阶方阵，并在主对角线放置元素 v

$v = \text{diag}(X,k)$

返回方阵 X 的第 k 条对角线元素构成的列向量

$v = \text{diag}(X)$

返回方阵 X 的主对角线元素构成的列向量

阐明： v 为行向量，表达对角线元素； k 为对角线位置， $k=0$ 表达主对角线， $k>0$ 表达在主对角线上方的第 k 条对角线， $k<0$ 表达在主对角线下方的第 $(-k)$ 条对角线。

2.5 数组操作

二、数组操作

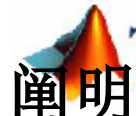
对数组的操作涉及数组的扩展、收缩、重排、元素交换和子数组访问等。数组操作能够通过两种方式实现，一种是通过Matlab提供的运算符（逗号，分号，括号等）来实现，另外一种是使用Matlab提供的数组操作函数。下面介绍惯用的数组操作函数。

1、cat

功效：把大小相似的若干数组，沿着指定维的方向，串接成新数组。

格式： $C = \text{cat}(\text{dim}, A, B)$

$C = \text{cat}(\text{dim}, A1, A2, A3, A4, \dots)$



The MathWorks

阐明： $A, B, A1, A2$ 等为被串接数组，规定这些数



洛阳师范学院

2.5 数组操作

2、fliplr

功效：沿着垂直中线，左右（**Left-Right**）对称交换数组元素（不超出2维）

格式： $B = \text{fliplr}(A)$

3、flipud

功效：沿着水平中线，上下（**Up-Down**）对称交换数组元素（不超出2维）

格式： $B = \text{flipud}(A)$

4、rot90

功效：逆时针旋转二维数组。

格式： $B = \text{rot90}(A)$

逆时针旋转矩阵90度

$B = \text{rot90}(A,k)$

逆时针旋转矩阵90*k度

2.5 数组操作

5、 repmat

功效：按指定维上的数目，分块铺放指定数组。

格式： $B = \text{repmat}(A,m,n)$ 沿着第1维铺放m个A，第2维铺放n个A

$B = \text{repmat}(A,[m\ n])$

$B = \text{repmat}(A,[m\ n\ p\dots])$

6、 reshape

功效：在总元素不变的前提下，重新安排数组各个维的长度，形成新数组。

格式： $B = \text{reshape}(A,m,n)$

$B = \text{reshape}(A,m,n,p,\dots)$

$B = \text{reshape}(A,[m\ n\ p\ \dots])$

$B = \text{reshape}(A,\dots,[\],\dots)$

阐明：**A**是待重新安排的数组；**m**，**n**，**p**等是新数组各个维的长度；**[]**表达自动计算某个维的长度而无需顾客指定。

2.5 数组操作

7、tril

功效：提取矩阵的下三角元素，生成下三角阵。

格式： $L = \text{tril}(X)$

$L = \text{tril}(X,k)$

阐明： X 为待提取的矩阵； k 为三角阵的分界限位置，含义同diag

8、triu

功效：提取矩阵的上三角元素，生成上三角阵。

格式： $L = \text{triu}(X)$

$L = \text{triu}(X,k)$

阐明： X 为待提取的矩阵； k 为三角阵的分界限位置，
含义同diag函数。

2.6 数组运算与矩阵运算

一、数组运算

Matlab定义了数组运算，数组运算是指对数组中的每个元素进行相似的运算。数组运算能够通过**Matlab**提供的运算符和数组运算函数实现。

1.用数组运算符进行数组运算

A+B 数组加法运算

A- B 数组减法运算

A.*B 数组相乘，**A**和**B**相似位置元素的乘积
作为成果数组的元素

A./B 数组相除，**A**和**B**相似位置元素相除作
为成果数组的元素



2.6 数组运算与矩阵运算

- A.\B** 一定与**A./B**相似
- A.^p** 数组各元素求**p**次幂
- A#B** **A**、**B**数组对应元素间进行关系运算，**#**代表关系运算符
- A@B** **A**、**B**数组对应元素间进行逻辑运算，**@**代表逻辑运算符
- A.'** 数组转置，非共轭转置
- s◎A** 标量**s**与数组**A**运算，**s**与**A**的每个元素进行运算，
◎代表某个运算符

2.6 数组运算与矩阵运算

2.数组运算函数

➤ 三角函数:

sin, cos, asin, asinh, asec, sect, tan, atan等

➤ 指数对数函数:

exp 指数函数

log 自然对数函数

log10 以10为底的对数函数

log2 以2为底的对数函数

pow2 2的幂函数

sqrt 平方根函数

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/728116104004006131>