

Matlab简介 – 微分方程数值解

笔记本： 我的第一个笔记本

创建时间： 2017/5/9 15:10

更新时间： 2017/5/9 15:23

URL： <https://iclass.bupt.edu.cn/webapps/blackboard/content/listContent.jsp?course...>

Matlab

简介



作业&思考：A课程导论 第四节



第四节 Matlab入门

第二章 Matlab简介



MATLAB是矩阵实验室（Matrix Laboratory）的简称，是美国MathWorks 公司出品的商业数学软件，用于算法开发、数据可视化、数据分析、数值分析以及工程科学的矩阵数学运算的高级技术计算机语言和交互式环境。它是一种极其灵活的计算机体系，在解决工程技术问题方面，MATLAB比其它任何计算机语言都简单高效。

2.1 基础知识

工作界面

MATLAB的工作界面主要由菜单、工具栏、命令窗口、工作空间管理窗口、命令历史窗口和当前目录窗口组成。

MATLAB命令窗口是用来接受MATLAB命令的窗口。在命令窗口中直接输入命令，可以实现显示、清除、储存、调出、管理、计算和绘图等功能。MATLAB命令窗口中的符号“>>”为运算提示符，表示MATLAB 处于准备状态。当在提示符后输入一段程序或一段运算式后按回车键，MATLAB 会给出计算结果并将其保存在工作空间管理窗口中，然后再次进入准备状态。

工作空间管理窗口显示当前MATLAB的内存中使用的所有变量的变量名、变量的大小和变量的数据结构等信息，数据结构不同的变量对应着不同的图标。

命令历史窗口显示所有执行过的命令。在默认设置下，该窗口会保留自MATLAB 安装后使用过所有命令，并表明使用的时间。利用该窗口，一方面可以查看曾经执行过的命令；另一方面，可以重复利用原来输入的命令，这只需在命令历史窗口中直接双击某个命令，就可以执行该命令。

当前目录窗口显示当前目录下所有文件的文件名、文件类型和最后修改时间。

MATLAB为用户提供了非常完善的帮助系统，例如在线帮助、帮助窗口以及MATLAB 演示等。通过使用帮助菜单或在命令窗口中输入帮助命令，可以很容易地获得MATLAB 的帮助信息，进一步学习MATLAB。

变量与函数

1. 变量

变量是任何程序设计语言的基本要素之一，MATLAB语言当然也不例外，与一般常规的程序设计语言不同的是，MATLAB语言并不要求对所使用的变量进行事先声明，也不需要指定变量类型，它会自动根据赋予变量的值或对变量进行的操作来确定变量的类型并为其分配内存空间。在赋值过程中，如果变量已存在，MATLAB将使用新值代替旧值，并以新的变量类型代替旧的变量类型。

MATLAB中变量的命名规则是：

- 变量名区分大小写；
- 变量名的长度不超过31位，第31个字符之后的字符将被忽略；
- 变量名必须以字母开头，之后可以是任意字母、数字或下划线，变量名中不允许使用标点符号。



MATLAB中有一些预定义的变量，这些特殊的变量称为常量。

与其他程序设计语言相同，MATLAB语言中也存在变量作用域的问题，在未特殊说明的情况下，MATLAB语言将所识别的一切变量视为局部变量，即仅在其调用的函数内有效。若要定义全局变量，应对变量进行声明，即在该变量前加关键字global。

2. 数学运算符号

MATLAB中的数学运算符号有：+ 加法运算；- 减法运算；* 乘法运算；.* 点乘运算；./ 右除运算；.\ 左除运算；./ 点右除运算；.\ 点左除运算；^ 乘幂运算。

MATLAB中标点符号的含义是：

- 在命令窗口中输入一个MATLAB语句（语句的一般形式为：变量=表达式），如果语句后为逗号或无标点符号，则在命令窗口中显示该语句的计算结果；如果语句后为分号，MATLAB只进行计算，不在命令窗口中显示计算结果。如果要查看计算结果，只需要在命令窗口中输入变量名按回车键或打开工作空间双击选中的变量即可。
- 在MATLAB的命令窗口中输入一个表达式或利用MATLAB进行编程时，如果表达式太长，可以用续行符号“...”将其延续到下一行。
- 编写MATLAB程序时，通常利用符号“%”对程序或其中的语句进行注释。

3. 函数

MATLAB语言中最基本最重要的成分是函数，一个函数由函数名、输入变量和输出变量组成。同一个函数，不同数目的输入变量和不同数目的输出变量，均代表不同的含义。

这不仅丰富了MATLAB的函数功能，而且大大减少了需要的磁盘空间，使得MATLAB编写的程序简单而且高效。

MATLAB的函数，按照用途，可以分为三类：标量函数、向量函数和矩阵函数。

- 标量函数：标量函数作用于标量，一般用于简单的数值计算。常用的标量函数有三角函数、指数对数函数、复数函数、截断函数和求余函数等。
当一个标量函数作用于向量或矩阵时，是这个标量函数作用于这个向量或矩阵的每一个元素，这个功能将大大方便我们处理成批的数据。
- 向量函数：MATLAB中有些函数只有当它们作用于行向量或列向量时才有意义，称为向量函数。常用的向量函数：max(最大值)、min(最小值)、sum(和)、length(长度)、mean(平均值)、median(中数)、prod(乘积)、sort(从小到大排列)。
当一个向量函数作用于一个矩阵时会产生一个行向量，这个行向量的每个元素是向量函数作用于矩阵相应列向量的结果。
- 矩阵函数：MATLAB中有大量的矩阵函数，从其作用来看，可以分为构造矩阵的函数和进行矩阵计算的函数。

矩阵与数组

MATLAB中最基本的数据对象就是数组或矩阵，标量可看作是 1×1 的矩阵，向量可看作是 $1 \times n$ 或 $n \times 1$ 的矩阵，一维数组是向量，二维数组便是矩阵，还有三维甚至更高维的数组。标量运算是数学的基础，然而，当需要对多个数执行同样的运算时，采用数组或矩阵运算将非常简洁和方便。

1. 创建矩阵



对于较小较简单的矩阵,从键盘上直接输入矩阵是最常用的数值矩阵创建方法.用这种方法输入矩阵时注意以下三点:

- 整个输入矩阵以方括号“[]”为其首尾;
- 矩阵的元素必须以逗号“,”或空格分隔;
- 矩阵的行与行之间必须用分号“;”或回车键隔离.

数值矩阵的创建还可由其他方法实现.如:利用MATLAB函数和语句创建数值矩阵;利用m文件创建数值矩阵;从其他文件获取数值矩阵.对于一维数组,有两种简单的创建方式,例如,

```
A = 0:0.1:1;%从0到1,增量为0.1.  
B = linspace(0,5,100);%100个从0到5的等间隔数.
```

上述数组创建形式所得到的数组的元素之间是线性分隔的特殊情况,当需要对数分隔的数组时,MATLAB提供了函数logspace,例如,

```
C = logspace(first,last,n)%数组元素个数为n.
```

也可以通过M文件建立大矩阵.当矩阵的规模比较大,直接输入法就显得笨拙,出现差错也不易修改.为了解决此问题,可以通过M文件输入矩阵.M文件是一种可以在MATLAB环境中运行的文本文件,分为命令文件和函数文件两种.这里是用命令M文件来建立大型矩阵.从菜单栏的File中选择New,再选择M-file命令,打开MATLAB Editor窗口,按格式把所要输入的矩阵写入一文本文件中,并将此文件以m为扩展名,即为M文件.在MATLAB命令窗口中输入此M文件名,运行后则把M文件中的大型矩阵输入到MATLAB的内存中.

MATLAB为方便编程和运算,提供了一些常用矩阵的生成指令,即可以用MATLAB的函数来建立全零矩阵、全1矩阵、单位矩阵、均匀分布(标准正态分布)随机矩阵、对角矩阵和上、下三角矩阵等特殊矩阵.矩阵元素全是0或1的矩阵由zeros或ones来建立,单位矩阵由eye建立,均匀分布随机矩阵和标准正态分布随机矩阵分别由rand和randn来建立.例如,

- eye(n):生成 $n \times n$ 的单位矩阵;
- ones(n):生成 $n \times n$ 全1矩阵;
- zeros(n):生成 $n \times n$ 零矩阵;
- eye(m,n):生成 $m \times n$ 标准型矩阵;
- ones(m,n):生成 $m \times n$ 全1矩阵;
- zeros(m,n):生成 $m \times n$ 零矩阵;

11

- eye(size(A)):生成与A同型的标准型矩阵;
- ones(size(A)):生成与A同型的全1矩阵;
- zeros(size(A)):生成与A同型的零矩阵;

其中指令size(A)给出矩阵A的行数和列数.

2. 矩阵元素的标识

矩阵的元素、子矩阵可以通过标量、向量、冒号的标识来援引和赋值.

I 矩阵元素的标识方式 $A(n_i, n_j)$.

n_i, n_j 都是标量.若它们不是整数,则在调用格式中会自动圆整到最临近整数. n_i 指定元素的行位置, n_j 指定元素的列位置.

II 子矩阵的序号向量标识方式 $A(v, w)$.

v, w 是向量, v, w 中的任意一个可以是冒号“:”,表示取全部行(在 v 位置)或全部列(在 w 位置). v, w 中所用序号必须大于等于1且小于等于矩阵的行列数.

从一个矩阵中取出若干行(列)构成新矩阵称为剪裁.冒号“:”是非常重要的剪裁工具.将几个矩阵接在一起称为拼接,左右拼接行数要相同,上下拼接列数要相同.例如,

```
A = [1 2 3; 4 5 6; 7 8 9];  
B = A(1:2,[1 3]);%A的第1,2行,第1,3列  
C = [B,zeros(2,1)];  
D = [C;eye(2),ones(2,1)]
```

最后输出

D =



```

1 3 0
4 6 0
1 0 1
0 1 1

```

- $A(:)$: 逐列提取A中的所有元素作为一个列向量;
- $A(i)$: 把A看作列向量提取其中第i个元素;
- $A(r, c)$: 提取A中, 由索引向量r定义的行, 和由索引向量c 定义的列所构成的A 的子数组;
- $A(r, :)$: 提取A中由索引向量r定义的行, 和全部列所构成的A的子数组;
- $A(:, c)$: 提取A中由全部行, 和由索引向量c定义的列所构成的A 的子数组。

3. 矩阵运算

矩阵运算的指令和意义如下:

12

- A' : 矩阵A的共轭转置矩阵, 当A是实矩阵时, A' 是A 的转置矩阵。
- $A + B$: 两个同型矩阵A与B相加。
- $A - B$: 两个同型矩阵A与B相减。
- $A * B$: 矩阵A与矩阵B相乘, 要求A的列数等于B的行数。
- $s + B$: 标量和矩阵相加 (MATLAB约定的特殊运算, 等于s 加B 的每一个分量)。
- $s - B$ 和 $B - s$: 标量和矩阵相减 (MATLAB 约定的特殊运算, 含意同上)。
- $s * A$: 数与矩阵A相乘。



数组可以看成特殊矩阵即一行n列的矩阵, 矩阵运算的指令和含意同样适用于数组运算。如果在运算符前加“.”, 含意将有所不同。

- $A .* B$: 同维数组或同型矩阵对应元素相乘;
- $A ./ B$: A的元素被B的元素对应除;
- $A .\wedge s$: A的每个元素s 次方;
- $p .\wedge A$: 以p为底, 分别以A的元素为指数求幂。

图形功能

在进行数值计算的过程中, 为了从直观上认识计算结果, 可以通过MATLAB 的图形功能将计算结果图形化。MATLAB是通过描点、连线来作图的, 因此, 在作二维图形和三维图形之前, 必须先取得该图形上一系列点的坐标, 然后利用MATLAB函数作图。

1. 二维图形

• plot命令

绘制二维图形最常用的命令是plot。对于不同形式的输入, 该函数可以实现不同的功能。

当plot函数仅有一个输入变量时, 如plot(X)表示: 如果X为实向量, 则以X 的索引坐标作为横坐标, 以X的各元素作为纵坐标绘制图形。如果X为复向量, 则以X 的实部作为横坐标, 虚部作为纵坐标绘制图形。如果X为实数矩阵, 则绘制X 的列向量对其坐标索引的图形。

plot函数有两个输入变量时, 如plot(X, Y) 表示: 当X和Y为向量时, X 和Y 的维数必须相同, 而且同时为行向量或同时为列向量。此时以第一个向量的分量为横坐标, 第二个向量的分量为纵坐标绘制图形。

当plot函数有三个输入变量时, 可用命令plot(X, Y, 'Linespec')。在想绘制不同的线型、颜色、标识等的图形时, 可以调用此形式, 第3 个输入变量为图形显示属性的设置选项: 线型、颜色、标识。

线型: — 实线; : 点线; - - 虚点线; -- 虚线;

颜色: y 黄; m 紫; c 青; r 红; g 绿; b 蓝; w 白; k 黑;

13

在绘图过程中,经常会遇到在已存在的一张图中添加新的曲线,这就要求保持已存在的图形, MATLAB语言中实现该功能的函数是hold.

hold on 启动图形保持功能,此后绘制的图形将添加到当前的图形窗口中,并自动调整坐标轴的范围;

hold off 关闭图形保持功能,新绘制图形将覆盖原图形.

hold 在hold on和hold off之间切换.

• 子图绘制

在绘图过程中,经常需要将几个图形在同一图形窗口中表示出来,但又不在同一个坐标系中绘制,此时要用到函数subplot.

4. 矩阵函数

命令rank(A)给出矩阵A的秩.

命令det(A)给出方阵A的行列式.

命令inv(A)给出方阵A的逆矩阵,如果A不可逆,则inv(A)给出的矩阵的元素都是Inf.

命令rref可以求出矩阵的行简化阶梯形.

命令trac(A)给出方阵的迹.

命令norm可以求矩阵范数.

命令eig求出方阵A的特征值和相应的特征向量.

5. 线性方程组求解

线性方程的求解分为两类:一类是方程组唯一解或求特解;一类是方程组求无穷解即通解.若求线性方程组 $Ax = b$ 的唯一解,常见的基本求解方法有以下几种:

- 左除法: $x = A \setminus b$.
- 求逆法: $x = inv(A) * b$.

15

- 用linsolve函数求解: $x = linsolve(A, b)$.
- 用solve函数求解.
- 用LU分解和Doolittle方法.
- 用Jacobi迭代法.





2.2 程序设计

MATLAB作为一种广泛应用的科学计算软件，不仅具有强大的数值计算、符号计算能力和丰富的作图功能，还可以像FORTRAN、C等其他计算机高级语言一样进行控制流的程序设计，即编制一种以m 为扩展名的文本文件，简称M文件，实现各种复杂的运算。

MATLAB程序设计语言属于第四代编程语言，其编程方式相当简洁，编程效率比常用Fortran、C、C++ 等语言要高的多，如语言简单、可读性强、调试容易及调用方便等。

M文件

MATLAB有两种工作方式，一种是在命令窗口中输入命令，MATLAB会立即执行并显示结果；另一种就是把所要执行的一系列命令编制成文件，MATLAB 会自动地按照文件中的命令和语句，按顺序执行并显示结果，这种命令集成文件必须以m 为扩展名，这就是M 文件。M文件有两种形式，命令M 文件和函数M 文件。

1. 命令M文件

命令M文件就是命令行的简单叠加，MATLAB 会自动按顺序执行文件中的命令。

命令M文件在运行过程中可以调用MATLAB工作域内所有的数据，而且，所产生的所有变量均为全局变量。

2. 函数M文件

为了实现计算中的参数传递和函数调用的问题，需要用到函数M文件。函数M文件的第一句以function 语句为引导。函数文件在MATLAB中应用十分广泛，MATLAB所提供的绝大多数功能函数都是由函数文件实现的。

函数M文件执行之后，只保留最后结果，不保留中间过程，所定义的变量也仅在函数内部起作用，并随调用的结束而被清除。

函数M文件的第一行有特殊的要求，其形式必须是：

function [因变量列表]=函数名(自变量列表)

注意：一般要求函数M文件的文件名必须与函数名相同，这样才能保证调用成功。这是由于在调用函数时，MATLAB先去搜索与函数名相同的文件，然后再执行文件。

3. 子函数和私有目录

MATLAB中可定义子函数，用来扩充函数的功能。一个M文件可以包含一个以上的函数，其中有一个主函数，其他为子函数。这些子函数只能被同一个文件中的函数（主函数或其他子函数）调用，但是不能被其他文件的函数调用。在一个M 文件中，主函数必须出现在最上方，其后可接上任意数目的子函数，而且子函数的次序可随意。同一文件的主函数、子函数的工作空间都是彼此独立的，各函数的信息可通过输入输出变量、全局变量或跨空间指令传递。

MATLAB中可以在某一目录中建立一个自己命名的私有目录来存放相关函数。当M 文件中需要调用某一个函数时，MATLAB是按照以下顺序来搜索的：检查此函数是否是子函数；检查此函数是否为私有目录的函数；从所设定的搜索路径上搜索此函数。搜索过程中，只要找到与第一个文件名相符的函数就会立即取用而停止搜索。

关系和逻辑运算

除了传统的数学运算，MATLAB支持关系和逻辑运算。一个重要的应用是控制基于真/假命题的一系列MATLAB命令的流程或执行次序。作为所有关系和逻辑表达式的输，MATLAB 把任何



非零数值当作真，把零当作假。所有关系和逻辑表达式的输出，对于真，输出为1；对于假，输出为0。

1. 关系运算符

MATLAB关系运算符能用来比较两个同样大小的矩阵，或用来比较一个矩阵和一个标量，后一种情况是，标量和矩阵中的每一个元素相比较，结果与矩阵大小一样。

- 运算符“<”：小于；
- 运算符“>”：大于；
- 运算符“<= ”：小于等于；

- 运算符“>= ”: 大于等于;
- 运算符“== ”: 等于;
- 运算符“~= ”: 不等于.

2. 逻辑运算符

- 运算符“& ”: 逻辑与, 当运算双方的对应元素都为非0 时, 结果为1; 否则为0;
- 运算符“| ”: 逻辑或, 当运算双方的对应元素有一非0 时, 结果为1; 否则为0;
- 运算符“~ ”: 逻辑非, 当运算数组上对应位置上的值为0 时, 结果为1; 否则为0.

在算术运算、比较运算和逻辑运算中, 它们的优先关系先后为: 算术运算、比较运算、逻辑与或非运算.

循环和条件语句

1. 循环语句

MATLAB中有for循环和while循环两种语句用于有规律的重复计算.

(1). for循环

for循环的最大特点是它的循环次数是预先设定好的. for循环语句的一般形式如下:

```
for 循环变量=数组
    循环体
end
```

for循环应注意: for循环语句一定要有end 作为结束标志; 循环体中的分号可防止中结果的输出; MATLAB并不要求循环条件中的数组是等间距的, 循环次数由循环条件中数组的列数决定; 为提高运算速度, 应尽量提高程序的向量化程度, 采用矩阵运算, 避免使用循环语句, 必须使用for循环时, 在循环语句前应预先分配数组; for 循环可以多重嵌套; 循环语句书写成锯齿形将增加可读性.

(2). while循环

while循环的一般形式是:

18

```
while 表达式
    循环体
end
```

与for循环固定循环次数不同, while循环的次数是不固定的. 在while 循环中, 只要达式的值是真, 循环体就会被执行. 通常表达式给出的是一个标量值, 也可以是数组或矩阵, 如果是后者, 则要求所有的元素都必须为真. 另外, while语句的循环条件可以是一个逻辑判断语句, 因此, 它的适用范围更广.

2. 条件语句

在程序中如果需要根据一定条件来执行不同的操作时就需要用到条件语句. MATLAB 中有两种条件语句: if语句和switch语句.

(1). if语句的一般形式是:

```
if 表达式
    语句1
else
    语句2
end
```

如果表达式为真, 则执行语句1, 如果表达式为假, 则执行语句2.

(2). switch语句

switch语句根据表达式的值来执行相应的语句, 此语句与C语言中的选择语句具有相同的功能, 它通常用于条件较多而且较单一的情况. 一般形式是:

```
switch 表达式
case value1
    语句1
case value2
    语句2
...
otherwise
    语句n
end
```


表达式是一个标量或者字符串，将表达式值依次和各个case指令后面的检测值进行比较，当比较结果为真时，MATLAB执行后面的一组命令，然后跳出switch结构。如果所有的结果都为，则执行otherwise后的命令，当然otherwise指令也可以不存在。

程序编写注意事项

尽管MATLAB语言已将绝大多数的操作集化到功能强大的函数中，但由于MATLAB语言是一种解释性语言，所以有时MATLAB程序的执行速度不是很理想，这里将给出加快MATLAB程序执行速度的一些建议。

1. 以矩阵作为操作主体

19



尽量避免使用循环：循环运算经常被认为是MATLAB编程的瓶颈问题，是MATLAB语言中的最大弱点，在程序设计时，应当尽可能避免循环运算。由于矩阵是MATLAB语言的核心，所以，在MATLAB编程过程中应当强调对矩阵本身整体的运算，避免对矩阵元素的操作。应当注意的是，绝大多数的循环运算是可以转化为向量运算的。

在必须使用多重循环的情况下，如果两个循环执行的次数不同，则建议在循环的外环执行循环次数少的，内环执行循环次数多的。这样也可以显著提高速度。

2. 数据的预定义

虽然在MATLAB语言中没有规定变量使用时必须预先定义，但是对于未定义的变量，如果操作中出现越界赋值时，系统将不得不对变量进行扩充，这样的操作大大降低了程序运行的效率，所以，对于可能出现变量维数不断扩大的问题，应当预先估计变量可能出现的最小维数，进行预定义。建议在定义大矩阵时，首先用MATLAB的函数，如zeros或ones先进行定维，然后再进行赋值处理，这样会显著减少所需的时间的。

3. 优先考虑MATLAB的函数

矩阵运算应该尽量采用MATLAB的函数，因为内在函数是由更底层的编程语言C构造的，其执行速度快于使用循环的矩阵运算。

2.3 求解微分方程（组）的解析解和数值解

MATLAB求解析解

MATLAB提供了dsolve函数用于计算常微分方程的符号解，其使用格式：

$S = \text{dsolve}(' \text{方程1}', ' \text{方程2}', \dots, ' \text{初始条件1}', ' \text{初始条件2}', \dots, ' \text{自变量}')$;

方程用字符串表示，自变量缺省值为t，导数用D表示，2阶导数用D2表示，以此类推。S返回解析解。在方程组情形，S为一个结构体数组，它的每个域存放方程组的每一个解。

MATLAB求数值解

Matlab提供了多种求解器ode23、ode45、ode113、ode15s、ode23s、ode23t、ode23tb。

基本使用格式：

$[tout, Yout] = \text{solver}(\text{odefun}, tspan, y0, options)$

这里solver代表各种求解器，如ode45。变量说明：odefun一般为用M文件编写的ode函数，须返回一阶常微分方程组表达式右边的函数值。故odefun函数的返回值一般是列向量，其最简单的编写格式为：

$F = \text{odefun}(t, y)$

其中，t为标量（自变量），代表迭代点（迭代的当前时刻，如t0、t1等）；因计算t时刻的函数表达式F需要知道该时刻的y值，需输入列向量y；列向量F为返回值，代表t时刻微分方程组右边的函数值。有了向量F后，若再给出初始条件，根据方程式就可迭代出方程的解。求解器在计算时将不断地在各个时刻调用odefun 的值。

20