

chap1. matlab简介

1. matlab安装指南

- 此部分参见上一个专题[matlab安装吐血指南（用过win系统一学就会）](#) - shi先森的文章 - 知乎

2. Matlab 的特点与功能

- Matlab 是一个交互式软件系统，输入一条命令，立即就可以得出该命令的结果
- Matlab 符号计算功能
- Matlab 的绘图功能：Matlab提供丰富的绘图命令，很方便实现数据的可视化
- Matlab 的编程功能：Matlab具有程序结构控制、函数调用、数据结构、输入输出、面向对象等程序语言特征，而且简单易学、编程效率高。通过Matlab 进行编程完成特定的任务
- Matlab 丰富的工具箱（toolbox）
- Matlab 的 Simulink 动态仿真集成环境

3. Matlab帮助系统

- help 显示指定命令的简短使用说明
- doc 以网页形式显示指定命令的帮助页
- lookfor 按指定的关键词查询与之相关的命令
- which 显示指定函数所在的目录

```
>> help eig
>> doc eig
>> lookfor inverse
>> which eig
cd、dir、more
```

4. Matlab 变量

变量命名原则

- 以字母或者下划线开头
- 后面可以跟 字母、数字 和 下划线
- 长度不超过 63 个字符（6.5 版本以前为 19 个）
- 变量名 区分字母的 大小 写

Matlab 语句的通常形式

```
变量 = 表达式
```

表达式是用运算符将有关运算量连接起来的式子，其结果被赋给赋值号“=”左边的变量

分号和续行符的作用

- 若不想在屏幕上输出结果，可以在语句最后加分号
- 如果语句很长，可用续行符“...”（三个点）续行；续行符的前面最好留一个空格

变量的查询

- who 显示工作空间中的所有变量
- whos 查看工作空间中变量的详细属性

系统预定义变量

- pi：圆周率\$pi\$
- inf, Inf：无穷大

- nan, NaN：Not-a-Number，一个不定值，如 0/0
- eps：浮点运算相对精度
- i, j：虚部单位
- 特殊变量 ans -- 应尽量避免给系统预定义变量重新赋值

数学运算符

- + 加法
- - 减法
- * 乘法
- / 和 \ 除法（右除和左除）
- ^ 幂运算
- 命令分隔符：逗号和分号

输出格式

- Matlab 以双精度执行所有的运算，运算结果可以在屏幕上输出，同时赋给指定变量；若无指定变量，则系统会自动将结果赋给变量“ans”
- Matlab 中数的输出格式可以通过 format 命令指定
- format 只改变变量的输出格式，但不会影响变量的值！

变量的存储

```
save 文件名 变量名列表
```

- 变量名列表中各变量之间用空格分隔

变量的读取

```
load mydata A x 从数据文件中提取指定变量
```

- 清除当前工作空间中的变量

```
clear A x 清除指定的变量
```

几个小技巧

- Matlab的命令记忆功能：上下箭头键(先输入命令的前几个字符，再按上下键缩小搜索范围)
- 命令补全功能：Tab 键
- 用 Esc 键 删除命令行

chap2. matlab矩阵数值计算

1. 矩阵的定义

Matlab 的操作对象是矩阵

- 定义矩阵：直接输入法

```
A = [1 2 3; 4 5 6; 7 8 9]
```

- 矩阵用方括号 “[]” 括起

```
mat = [元素]
```

- 矩阵同一行中的元素之间用 空格 或 逗号 分隔

```
mat1 = [1 2 3];  
mat2 = [1,2,3];
```

- 矩阵行与行之间用 分号 分开

```
mat3 =[ 1 2 3; 4, 5, 6];
```

- 直接输入法中，分号可以用 回车 代替

```
mat4 = [1 2 3
        4 5 6];
```

2. 矩阵赋值操作

- 矩阵A是一个1*2矩阵，第一个元素1，第二个元素2

```
A(1)=1;
A(2)=2;
```

- 大矩阵可以把小矩阵作为其元素

```
A = [A ; 3 4];
```

3. 矩阵元素提取引用操作

- 单个元素的引用：利用小括弧和元素所在的位置(下标)

```
A(1)%引用A的第一个元素
A(i): 向量 x 中的第 i 个元素
A(i,j): 矩阵 A 中的第 i 行, 第 j 列元素
```

- 多个元素的引用：冒号的特殊用法

```
a:b:c %产生一个由等差序列组成的向量； a 是首项, b 是公差, c 确定最后一项；若 b=1, 则 b 可以省略。
x=1:2:5
x = 2:5
x = 10:-3 :2
```

- 矩阵元素的引用

```
A(i:j, m:n) %表示由矩阵 A 的第 i 到第 j 行和第 m 到第 n 列交叉线上的元素组成的子矩阵。
%可利用冒号提取矩阵 的整行或整列。
>> A(1, :)
>> A(:, 1:3)
>> A(:, :)
```

4. 建立矩阵进阶

- 利用函数建立数值矩阵：MATLAB提供了许多生成和操作矩阵的函数，可以利用它们去建立矩阵。例如: reshape函数和diag函数等。reshape函数用于建立数值矩阵。diag函数用于产生对角阵。
- 利用M文件建立矩阵：对于比较大且比较复杂的矩阵，可以为它专门建立一个M文件。其步骤为：
 - 第一步：使用编辑程序输入文件内容。
 - 第二步：把输入的内容以纯文本方式存盘(设文件名为mymatrix.m)。
 - 第三步：在MATLAB命令窗口中输入mymatrix，就会自动建立一个名为AM的矩阵，可供以后显示和调用。
- 利用M文件建立矩阵：对于比较大且比较复杂的矩阵可以它专门建立一个M文件。其步骤为：
 - 第一步：使用编辑程序输入文件内容。
 - 第二步：把输入的内容以纯文本方式存盘(设文件名为mymatrix.m)。
 - 第三步：在MATLAB命令窗口中输入mymatrix，就会自动建立一个名为AM的矩阵，可供以后显示和调用。
- 矩阵的基本运算

- （1）矩阵转置
- （2）矩阵加和减
- （3）矩阵乘法
- （4）矩阵除法 $A \backslash b = \text{inv}(A) * b$
- （5）矩阵的乘方 a^2
 - ** 矩阵函数**

help matfun

Matrix functions - numerical linear algebra. Matrix analysis. norm - Matrix or vector norm.

normest	- Estimate the matrix 2-norm.
rank	- Matrix rank.
det	- Determinant.
trace	- Sum of diagonal elements.
null	- Null space.
orth	- Orthogonalization.
rref	- Reduced row echelon form.
subspace	- Angle between two subspaces.

5. 建立矩阵的函数

eye(size(A))	产生与A矩阵同阶的单位矩阵
zeros(m,n)	产生0矩阵
ones(m,n)	产生1矩阵
rand (m,n)	产生随机元素的矩阵
Size(a)	返回包含两个元素的向量。
Length(a)	返回向量的长度。

6. 数组运算

（1）数组的加和减 （2）数组的乘和除 （3）数组的乘方

7. 举例

1、行列式计算；

```
det([1 2 ;3 4])
```

2、求解线性方程组；

```
x = A\b
```

3、解特征值问题

```
eig(A)
```

chap3. matlab图像可视化

基本要求

- （1）掌握图形窗口的创建与控制，以及图形窗口的基本操作；
- （2）熟练掌握二维和三维绘图基本的命令、线型控制；
- （3）初步掌握用特殊的图形来表现特殊数据的性质，如面积图、直方图、饼图等。
- （4）掌握坐标轴的控制和图形标注命令及其用法。

1. 二维平面图形与坐标系

1. 几个基本的绘图命令

2. 线性坐标曲线 plot 函数命令 plot 是 MATLAB 二维曲线绘图中最简单、最重要、使用最广泛的一个线性绘图函数。它可以生成线段、曲线和参数方程曲线的函数图形。命令格式:

```
plot(X,Y)
plot(x1,y1,x2,y2,...): 综合调用方式
```

3. 用命令 plot(x,y)绘制函数 $y=\cos(x)$ 在一个周期内的图形。

```
x=0:0.01:2*pi;
y=cos(x);
plot(x,y)
```

4. 在同一图形窗口中用命令 plot(x,y)绘出正弦余弦函数的图形。

```
x=0:0.01:2*pi;
y=[sin(x);cos(x)];
plot(x,y)
```

5. plot 函数可以设置曲线的线段类型、定点标记和线段颜色。调用格式: **plot(x,y,s)**, **s** 为类型说明参数, 是字符串。s 字符串可以是三种类型的符号之一, 也可以是线型与颜色和定点标记与颜色的组合; 如果没有 s 参数, plot 将使用缺省设置(实线, 前七种颜色顺序着色)绘制曲线; 在当前坐标系中绘图时, 每调入一次绘图函数, MATLAB将擦掉坐标系中已有的图形对象。可以用 hold on 命令在一个坐标系中增加新的图形对象。注意MATLAB会根据新图形的大小, 重新改变坐标系的比例。

用不同的线型和标注来绘制两条曲线。

```
t1=0:0.1:2*pi;
t2=0:0.1:6;
y1=sin(t1);
y2=sqrt(t2);
plot(t1,y1,'hb',t2,y2,'-g')
```

- 图形窗口的分割 有时需要在一个图形窗口中显示几幅图, 以便对几个函数进行直观、便捷的比较。由于每个绘图命令在绘制数据图像时都会将已有图形覆盖掉, 而用 hold 命令不能实现同时显示几个不同坐标尺寸下的图形, 用 figure 命令再创窗口又很难同时比较由不同的数据绘得的图像。实现在同一个窗口中同时显示多个图像的命令subplot。使用格式为:

```
subplot(m, n, i)
```

其含义为: 把图形窗口分割为 **m** 行 **n** 列子窗口, 然后选定第 **i** 个窗口为当前窗口。subplot 命令不仅用于二维图形, 对三维图形一样适用。其本质是将 figure 窗口分为几个区域, 再在每个区域内分别绘图。

- 用 subplot 函数把两种不同的图形综合在一个图形窗口中。

```
subplot(2,2,1)
t=0:0.1:2*pi;
y=sin(t);
semilogx(t,y)
grid on
subplot(2,2,2)
t=0:0.1:4*pi;
y=sin(t);
plot(t,y)
subplot(2,2,3)
x=1:0.01:5;
y=exp(x);
plot(x,y,x,y,'semilogx','plot')
subplot(2,2,4)
x=1:0.1:10;
y=sqrt(x);
```

```
plot(x,y,':rd')
```

- 1. 坐标系的调整 实现坐标系的调整的命令是 axis 函数。调用格式为: **** axis([xmin,xmax,ymin,ymax,zmin,zmax])**** 坐标的最小值 (xmin,ymin,zmin) 必须小于相应的最大值 (xmax,ymax,zmax), 否则会出错。自动坐标系与用 axis 函数调整后的坐标系的比较。

```
subplot(2,1,1)
t=0:0.1:4*pi;
y=sin(t);
plot(t,y)
subplot(2,1,2)
t=0:0.1:4*pi;
y=sin(t);
plot(t,y)
axis([0,max(t),min(y),max(y)])
```

2. 三维绘图

- 三维曲线绘图命令 三维函数 plot3主要用来表现单参数的三维曲线, 与二维绘图函数 plot 相比, 只多了第三维数据。其调用格式为:

```
plot3(X1,Y1,Z1,s1,X2,Y2,Z2,s2,...)
```

参数的含义如下: X_n 、 Y_n 、 Z_n : 第一到三维数据, 是尺寸相等的向量/矩阵; s 、 s_1 、 s_2 : 是字符串, 用来设置线型、颜色、数据点标记。x、y、z 是向量时, plot3 命令的使用

```
t=0:0.1:8*pi;
plot3(sin(t),cos(t),t)
title('绘制螺旋线') %用命令 title 对图形主题进行标注
xlabel('sin(t)')
ylabel('cos(t)')
zlabel('t')
%命令 zlabel 用来指定 z 轴的数据名称
grid on
```

x、y、z 都是矩阵时, plot3 命令的使用

```
[X,Y]=meshgrid(-pi:0.1:pi);
Z=sin(X)+cos(Y);
plot3(X,Y,Z)
```

- 三维曲面绘图命令 为了绘制定义在平面区域 $D=[x_0,x_m] \times [y_0,y_n]$ 上的三维曲面 $z=f(x,y)$, 首先将 $[x_0,x_m]$ 在 x 方向分成 m 份, 将 $[y_0,y_n]$ 在 y 方向分成 n 份, 由各划点分别作平行于坐标轴的直线, 将区域 D 分成 $m \times n$ 个小矩形; 对于每个小矩形, 计算出网格点的函数值, 决定出空间中四个顶点 ($x_i, y_i, f(x_i, y_i)$), 连接四个顶点得到一个空间的四边形片; 所有四边形片连在一起构成函数 $z=f(x,y)$ 定义在区域 D 上的空间网格曲面。因此, 三维曲面绘图命令可分为平面网格点的生成、在平面网格基础上绘制三维网格及对三维表面进行处理三个步骤。
- 平面网格点的生成 函数命令 meshgrid 用来生成 x-y 平面上的网格点矩阵。调用形式为:

```
[X,Y]=meshgrid(x,y)
[X,Y]=meshgrid(x) 等价于 [X,Y]=meshgrid(x,x)
```

参数含义如下: **x**: 是区间 $[x_0, x_m]$ 上分划的向量; **y**: 是区间 $[y_0, y_n]$ 上分划的向量; **X**, **Y**: 输出变量矩阵, 矩阵 **X** 的行向量都是向量 **x**, 矩阵 **Y** 的列向量都是向量 **y**。函数 meshgrid 将由两个向量决定的区域转换为对应的网格点矩阵。

- 三维网格命令 mesh 利用函数 mesh 生成网格曲面。调用格式为:

```
mesh(X, Y, Z, C): X、Y、Z、C 是同维数的矩阵, X、Y、Z 对应空间上的网格点, 网格线颜色由C决定;
mesh(X, Y, Z): 相当于上面的 C=Z 的情况;
mesh(x, y, Z, C): x 和 y 是向量, Z 和 C 是同维数的矩阵, 网格曲面的网格顶点是 ( x(j), y(i), Z(i,j) ), 网格线的颜色由矩阵 C 决定;
mesh(x, y, Z): 相当于上面的 C=Z 的情况;
mesh(Z, C): 等价于 mesh(x, y, Z, C), 此时向量 x=1:n, 向量 y=1:m;
mesh(Z): 相当于上面的 C=Z 的情况
mesh(...,'PropertyName',PropertyValue,...): 给函数 mesh 设置曲面属性。
```

- 三维表面命令 surf 函数 surf 可实现对网格曲面片进行着色，将网格曲面转化为实曲面。surf 命令的调用格式与 mesh 相同。利用三维网格表面命令 surf 绘制图形。

```
z=peaks;          %绘制山峰的图像，将函数值赋予变量z
surf(z)           %对山峰的图像进行着色处理
shading interp    %函数 shading 改变着色方式
```

- 柱面的表达cylinder cylinder命令中，柱面的轴线定义为 z 轴，只要给出母线的描述就可完成一个柱面。调用格式为：

```
[X,Y,Z] = cylinder(R,N);
[X,Y,Z] = cylinder(R): 缺省值 N=20;
[X,Y,Z] = cylinder: 缺省值 N=20, R=[1, 1]。
```

R：是一描述柱面母线的向量； N：是旋转柱面上的分割线条数； [X,Y,Z]：是返回的x,y,z坐标向量。绘制一个柱面。

```
t=pi:0.01:3*pi;
r=sin(t)+t;
cylinder(r,30)
shading interp
```

- 球面的表达sphere 调用格式为：

```
[X,Y,Z]=sphere(N): 产生一个 ( N+1) × ( N+1) 的矩阵，然后用函数 surf 命令绘制一个单位的球面，N 为设置分割线的条数；
[X,Y,Z] = sphere: 缺省 N = 20。
```

画一个球面。

```
[X,Y,Z]=sphere;
surf(X,Y,Z)
```

- 特殊图形绘制

为了将抽象的数据表达得更形象，除了绘制二维、三维图形外，还要用到直方图、面积图、饼图等特殊图形。

图像标注修饰

- 图形标注

坐标轴和图形标题标注；标注坐标轴 x、y 和 z 的命令函数为 xlabel、ylabel 和 zlabel，调用格式为：

```
xlabel('text')
xlabel('text','Property1',PropertyValue1,'Property2',PropertyValue2,...)
H = xlabel(...) 返回坐标轴标注的句柄。
```

其中，'text'是要添加的标注文本。'Property'是文本的属性名，'PropertyValue'是属性值（所用字体、大小、标注角度等）。图形加标题的函数为 title，其调用格式与坐标轴标注类似。

- 图例的标注 legend命令实现不同图例的说明。其调用格式为：

```
legend(string1,string2,string3, ...)
legend(string1,string2,string3,...,Pos)
```

按顺序把字符串添加到相应的曲线线型符号之后；Pos对图例的位置作出设置和调整：

```
0 = 自动把图例置于最佳位置（和图中曲线重复最少）；
1 = 置于图形窗口的右上角（缺省值）；
2 = 置于图形窗口的左上角；
3 = 置于图形窗口的左下角；
4 = 置于图形窗口的右下角；
-1 = 置于图形窗口的右侧（外部）。
```

- 控制分格线 对二维和三维图形都适用。有三种用法：

grid on: 打开分格线控制开关, 以后绘制的图形都带有分格线;
grid off: 关闭分格线控制开关, 以后绘制的图形都不带分格线;
grid: 用于实现分格线绘制切换。

- 举例

绘制图形, 并用函数 xlabel、title 和 legend 命令进行标注。

```
t=0:0.1:4*pi; y=sin(t); y1=cos(t);
plot(t,y, ':',t,y1,'r*')
xlabel('x 轴 (0--4\pi)','fontsize',12,'fontweight','bold')
ylabel('y 轴','fontsize',12,'fontweight','bold')
title('绘制正弦波和余弦波 Pos=1','fontsize',10,'fontweight','bold','fontangle','italic')
text(pi,0,'\leftarrowsin(\pi)=0')
text(pi,-1,'\leftarrowcos(\pi)=-1')
text(pi/2,0.9,['\uparrowsin(\pi/2)=' ,num2str(sin(pi/2))])
text(0,-0.6,['绘图日期: ',date])
text(0,-0.8,['MATLAB 版本: ',version])
legend('正弦波','余弦波')
figure(2)
plot(t,y, ':',t,y1,'r*')
title('绘制正弦波和余弦波 Pos=0','fontsize',10,'fontweight','bold','fontangle','italic')
legend('正弦波','余弦波',0)
grid on
figure(3)
plot(t,y, ':',t,y1,'r*')
title('绘制正弦波和余弦波 Pos=-1','fontsize',10,'fontweight','bold','fontangle','italic')
```

chap4. matlab符号运算

1. 符号对象的建立: sym 和 syms

```
符号变量 = sym(A)
syms 符号变量1 符号变量2 ... 符号变量n
syms a b c
```

Matlab 符号运算采用的运算符和基本函数, 在形状、名称和使用上, 都与数值计算中的运算符和基本函数完全相同

- 查找符号表达式中的符号变量

```
findsym(expr) %按字母顺序列出符号表达式 expr 中的所有符号变量
findsym(expr, N)%按顺序列出 expr 中离 x 最近的 N 个符号变量
```

- 用给定的数据替换符号表达式中的指定的符号变量

```
subs(f,x,a)
```

用 a 替换字符函数 f 中的字符变量 x a 可以是 数/数值变量/表达式 或 字符变量/表达式。若 x 是一个由多个字符变量组成的数组或矩阵, 则 a 应该具有与 x 相同的形状的数组或矩阵

- 因式分解

```
syms x;
f=x^6+1;
factor(f)
```

- 函数展开

```
syms x;
f=(x+1)^6;
```



```
expand(f)
```

- 合并同类项

```
collect(f,v): 按指定变量 v 进行合并  
collect(f): 按默认变量进行合并
```

- 函数简化

```
[How,y]=simple(f): y 为 f 的最简短形式, How 中记录的为简化过程中使用的方法。
```

- 计算极限

```
limit(f,x,a): 计算  
limit(f,a): 当默认变量趋向于 a 时的极限  
limit(f): 计算 a=0 时的极限  
limit(f,x,a,'right'): 计算右极限  
limit(f,x,a,'left'): 计算左极限
```

- 计算导数

```
g=diff(f,v): 求符号表达式 f 关于 v 的导数  
g=diff(f): 求符号表达式 f 关于默认变量的导数  
g=diff(f,v,n): 求 f 关于 v 的 n 阶导数
```

- 计算积分

```
int(f,v,a,b): 计算定积分  
int(f,a,b): 计算关于默认变量的定积分  
int(f,v): 计算不定积分  
int(f): 计算关于默认变量的不定积分
```

- 符号求和

```
symsum(f,v,a,b)
```

- 代数方程求解 solve(f,v): 求方程关于指定自变量的解, f 可以用字符串表示的方程、符号表达式或符号方程; solve 也可解方程组(包含非线性); 得不到解析解时, 给出数值解。

- 微分方程求解

```
y=dsolve('eq1','eq2', ... , 'cond1','cond2', ... , 'v')
```

其中 y 为输出的解, eq1、eq2、... 为微分方程, cond1、cond2、... 为初值条件, v 为自变量

chap5. matlab 程序设计

基本要求:

- (1)学会 Matlab 的M文件的编写方法, 包括命令文件和函数文件。了解M件的调试命令和调试方法;
- (2) 熟练掌握 Matlab 的程序结构(顺序结构、循环结构和分支结构)与流程控制语句 . Outline
- M-file(M文件);
- 数据的输入和输出;
- 程序设计;
- 函数文件;
- 全局变量和局部变量

1.M文件

用MATLAB语言编写的程序, 称为M文件。 M文件有两类: 命令文件和函数文件。 命令文件: 没有输入参数, 也不返回输出参数。 函数文件: 可以输

入参数，也可返回输出参数。

- input函数：用于向计算机输入一个参数。调用格式：A=input(提示信息，选项)；注：'s'选项，则允许用户输入一个字符串。例如想输入一个人的姓名，可采用命令

```
xm=input('What''s your name:', 's')
```

- pause函数：暂停程序的执行。调用格式：pause(延迟秒数) 注：如果省略延迟时间，直接使用pause，则将暂停程序，直到用户按任一键后程序继续执行。
- disp函数：命令窗口输出函数。调用格式：disp(输出项) 注：输出项为字符串或矩阵。例如

```
A='Hello,MATLAB';  
disp(A)
```

输出为：Hello,MATLAB

2.程序设计

- for语句：

格式：

```
for 循环变量=表达式1:表达式2:表达式3  
    循环体语句  
end
```

注：其中表达式1的值为循环变量的初值，表达式2的值为步长，表达式3的值为循环变量的终值。步长为1时，表达式2可以省略

- while语句

格式为：

```
while (条件)  
    循环体语句  
end
```

【例7】根据例3.6中求y的表达式，求：(1)y<3时的最大n值。(2)与(1)的n值对应的y值。

- while语句

格式为：while (条件) 循环体语句 end 【例7】根据例3.6中求y的表达式，求：(1)y<3时的最大n值。(2)与(1)的n值对应的y值。

3. 函数文件

是另一种形式的M文件，每一个函数文件都定义一个函数。事实上，MATLAB提供的标准函数大部分都是由函数文件定义的

- 函数文件由function语句引导，其格式为：

```
function 输出形参表=函数名(输入形参表)  
    注释说明部分  
    函数体
```

注：其中函数名的命名规则与变量名相同。输入形参为函数的输入参数，输出形参为函数的输出参数。当输出形参多于1个时，则应该用方括号括起来。

- 【例】编写函数文件求小于任意自然数n的Fibonacci数列各项。

```
function f=ffib(n)  
    %用于求Fibonacci数列的函数文件  
    %f=ffib(n)  
    %1999年9月30日编
```

```
f=[1,1];  
i=1;  
while f(i)+f(i+1)<n  
    f(i+2)=f(i)+f(i+1);  
    i=i+1;  
end
```

- 函数调用

[输出实参表]=函数名(输入实参表)

4.全局变量和局部变量

在MATLAB中，全局变量用命令global定义。函数文件的内部变量是局部的，与其他函数文件及MATLAB工作空间相互隔离。但是，如果在若干函数中，都把某一变量定义为全局变量，那么这些函数将公用这一个变量。全局变量的作用域是整个MATLAB工作空间，即全程有效。所有的函数都可以对它进行存取和修改。因此，定义全局变量是函数间传递信息的一种手段。