

『大一时』 Matlab笔记抄录

前述：本速记笔记不完整，并且是意识流的方式记载的

- 1、只是一些当时『不记得的单词』、『函数名』、『注意事项』的记录。
- 2、将纸质版上传至GitHub做电子版备份

一、基础语法

✓ matlab中%引导的是『注释行』

✓ matlab程序设计：

- 脚本文件
- 函数文件

✓ 脚本文件中：

输入：

```
a=input('.....'); %即显示' '中的文字或字符、数字，然后将你输入的赋给a
```

输出：

```
fprintf('.....%f\n',.....); %“\n”表示换行
```

函数文件中：

- 『函数文件名』必须与『函数名』相同
- M函数文件的第1个可执行必须是function

函数的『定义』格式：

```
function [输出变量列表] = 函数名(变量参数表)
```

例子有：

```
function z=ex3_2(a,b,c)
```

函数的『调用』格式： %即经函数中参数赋值

```
[输出变量列表] = 函数名(输入变量列表)
```

例子有：

```
z=ex3_2(1,2,1)
```

- 经常错的：
- `M = mod(X,Y)` 而不是 `M=mod(X/Y)`

二、函数名记忆

- 本质：matlab的所有命令基于『库函数』

(1) 三角函数类

```
asin %反正弦
acos
atan
acot
sinh %双曲正弦
cosh
tanh
coth
```

(2) 指数函数类

```
exp %e为底的指数
log %自然对数
log10 %10为底的对数
```

- `sqrt` 平方根
- `pow2` 2的幂

(3) 统计类

```
mean %平均值
median %中位数
```

(4) 取整方向类

```
floor %向 $-\infty$ 圆整
fix %向0圆整
ceil %向 $+\infty$ 圆整
round %向靠近整数圆整
sign %符号函数
```

- `mod` 模除取余
- `rem` 求余数

(5) 其他

```
abs %绝对值  
conj %复数共轭  
real %复数实部  
imag %复数虚部
```

(6) 其他常考函数

```
sort %排序  
length %个数  
sum %总和  
prod %总乘积  
cumprod %累计元素总乘积
```

- format格式

三、向量和矩阵

1、向量的创建：

(1) 冒号表达式

```
>>> x=1:0.5:Pi  
x=  
1.0000 1.5000 2.0000 2.5000 3.0000
```

(2) 函数法

- linspace

```
linspace(a,b,n) %n处如果没有，默认是100 a为第1元素，b为最后元素，n指n个元素  
>>> x=linspace(1,pi,5)  
x=  
1.0000 1.5354 2.0708 2.6062 3.1416
```

- logspace

```
logspace(a,b,n)
```

(3) 向量的访问

```
>>>k=[1,3,5];
>>>k(3)
    5
>>>k(1:3)    %提取从1到3的元素
    1 3 5
>>>k([1 3])  %提取1和3号位元素
    1 5
```

2、矩阵的创建:

(1) 函数生成法

```
rand(n)    %随机产生n*n的矩阵
rand(m,n)
eye(n)
eye(m,n)
ones(n)
ones(m,n)
diag(A)
diag(a)
magic(n)
```

(2) 拼接法

```
>>>x=1:2:5
    1 3 5
>>>y=rand(1,3)
    0.1419 0.4218 0.9157
>>>A=[x;y]    %注意是;号
    A=
    1 3 5
    0.1419 0.4218 0.9157
>>>B=[A,eye(2,2)]
    B=
    1.0000 3.0000 5.0000 1.0000 0
    0.1419 0.4218 0.9157 0 1.0000
```

- `repmat(A,M,N)` 复制矩阵A来构造新的MXN个A的大矩阵

四、零碎的笔记

```
num=get(handles.edit1, 'String');
%      句柄      Tag (标签)      匈牙利命名(String表示字符串)
```

```
str2num % 将字符串（string）转换（to）为双精度的数据（number）
```

```
set(handles.edit1, 'String', ' ');
```

```
open('test 14m.mdl'); %打开模型文件  
sim('test 14m.mdl'); %运行模型文件  
plot(tout,yout); %画粒子tout和yout的图 『我做的什么笔记。。。』
```

- memory

五、仿真部分

Initial adj.最初的 初值？初始化？
variable adj.变量的

- 1) tout 指time out 『比然会有』

%仿真过程中返回的时间变量，为1数组

%默认tout保留最后仿真为1000个数据 『列向量』

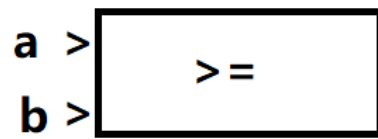
- 2) yout指的y out

采样点 out 『列向量』

- 3) simout 仿真输出模块，用于向matlab传递数据 『to workspace』

%该模块是将仿真模型中**某一个模块**的信号输出到matlab的工作空间（workspace）中，保存为数组和结构体类型

仿真Simulink中，『relational Operator』（关系比较）



如图，这个中，表示的意思是：

若 $a > b$

即：上比较下

六、GUI编程

```
bar n.条  
font n.字体  
dlg/dialog Box 对话框  
index v.索引  
default 默认值  
Data 数据  
Icon 图标  
options 选项
```

句柄的概念

Matlab中图形句柄

- 1) 句柄是matlab语言中独有的参数，相当于C语言中的指针
- 2) 句柄
 - 函数柄
 - 对象柄
 - 图形柄

图形句柄就指1个图形，在生成图形时间时得到1代号

比如语句：

```
h=plot(x,y) h即1个图形句柄
```

在后来的某个地方就可以用h代表这个图

比如：

```
set(h,.....) %对该图形进行再设置
```

- gcf 返回当前Figure对象的句柄值

七、符号计算

```
a=sym('sin(2)') %创建符号常量      sym('常量')
>>a=
    sin(2)
a=sym('sin(2)') %sym可以把数值转换成某格式的符号常量
>>a=
4095111552621091/450359962730496
```

sym(常量,常量) %参数

- d返回最接近的十进制的数值
- f返回最接近的浮点表示
- fi返回最接近的『有理数』
- e带有机浮点误差的有理值；『默认的』

sym 变量 参数 %创建符号变量

syms 变量1 变量2 参数 %创建完符号变量后，若在某表达式中全用到该变量，则该表达式则为符号表达式了！『若不限定，则参数可省』

sym('表达式') %创建符号表达式

sym('矩阵') %创建符号矩阵

symbol 符号

```
B=('[a b ; c d]') %创建字符串矩阵
```

```
>>B=
```

```
 [a b ; c d]
```

digits(n) %设定默认的精度，n为所期望的有效倍数

digits函数可以改变默认的有效位数来改变精度

vpa(s,n) %将s表示为n位有效位数的符号对象

PS: s可以是数值对象，亦可为符号对象

```

x1=2/3      %数值型
>>x1=
    0.6667
>>x2=sym(2/3)  %有理数型
x2=
    2/3
>>x3=vpa('2/3',32)  %VPA型
x3=
    0.6666』6666』6666』6666』6666』6666』6666』6666』6666』
    %注意，上面的』是不会打印出来的，只是我用来数数的

```

数值对象：

- sym 有理数型符号对象
- vpa VPA型符号对象

以上2者，如果用double就能转换为『数值对象』

```

>>a1=sym('3*sqrt(5)+pi')
a1=
    3*sqrt(5)+pi
>>b1=double(a1) %N=double(s)  %s为符号对象，N为数值变量
b1=
    9.8498
>>b3=eval(a1)  %函数eval()可以得到符号变量的数值结果
b3=
    9.8498

```

八、『科研图形绘制』

```

plot(t,y)
set(gca,'YTick',-1:0.5:1); %将该图的y轴变为，以-1到1每小格0.5
text(pi,0,'\leftarrow sin(x)', 'FontSize', 18);
text(pi,0,'\leftarrow sin(x)', 'Color', 'r');

```

- bar 条状图
- hist 直方图 『统计』

```

%隐函数绘图
syms x y
f=x^3+y^3+1;
expplot(f);

```



```
x=[0 0.2 0.8 1 0.5 0]
Y=[1 0 0 1 1.8 1]
patch(X,Y,'r','Linewidth',2) %填充颜色为red，边框的字的宽度为2
```

(1) 绘图的各种属性

```
xlabel(sin(x)) %给x轴加上名称 似ylabel、zlabel figure
title('名称') %给图形取名称
text(pi,0,'\leftarrow sin(x)') %在点(pi,0)处加上<-sinx
%转义字符 \leftarrow \rightarrow \pi \sigma
grid on/off %...
linspace(0,pi,12) %首为0，末位pi，总共12个数据点（等分）
subplot(2,2,1) %将区域分为2行2列，4个区域，将图画在第1区
axis([0 pi -1 1])%规定了x,y轴的最小、最大值
axis([xmin xmax ymin ymax])
hold on %在该图上继续画
legend('三角函数'); %给某线命名... 『--三角函数』
```

(2) plot

```
plot(x,y1,'b-',x,y2,'r--');
legend('sin(x)','cos(x)');
plot(x1,y1,'r.','Linewidth',1);
```

(3) plotyy

- plotyy 双y轴
- plotyy(x,y1,x,y2)

```
plotyy(x,y1,x,y2);
h=plotyy(x,y1,x,y2);
set(get(h(1),'Children'),'Linewidth',3);
set(get(h(2),'YLabel'),'String','快衰'); %加粗黑体
title('衰变曲线','color','r','FontSize',18);
```

(4) 直方图

```
x=-2:2
y=[3,5,2,4,1
3,4,5,2,1
5,4,3,2,5]; %直方图
bar(x,y,'stacked'); %堆叠起来：需为行向量
colormap(cool);
ylabe(c'\sigma Y');
```

- help bar 和 barh
- 去了解'stacked'和'grouped'

(5) 画三维曲面图

```
[x,y]=meshgrid(-4:0.2:4, -4:0.2:4); %x,y是栅格点的坐标
z=exp(-0.5.*(x.^2 + y.^2 ));
surf(x,y,z);      %画三维曲面图
contour(x,y,z); %画二维（平面）等高线
contour3(x,y,z);  %画三维（空间）等高线
```

(6) 图像类代码

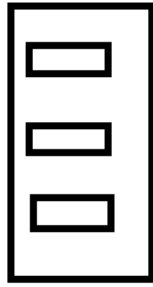
```
syms x y
f=x^2+y^2-1;
ezplot(f,[-1,1]); %告诉的
```

```
[AX,H1,H2]=plotyy(x,y1,x,y2);
set(H1,'Linewidth',1.5);
set(H2,'Linewidth',1.5);
```

- 下面的代码是画那种『对数左边』的，比如x坐标是10-2、10-1、100啥的

```
x=1/100:0.001:10;
y1=log(x)+x;
y2=x+5;
semilogx(x,y1);
hold on;
semilogx(x,y2,'r');
```

```
x=[48 12 10 6];
pie(x);
Legend( {'优秀', '良好', '中等', '及格', '不及格'}, ...
        'Location', 'west Out Side');
```



类似于上面的图形，放在西边外面

(7) 饼状图

```
x=[4 8 12 10 6]
explode=[ 1 0 1 1 0 ]; %1为出来
pie3( x, explode, {'优秀', '良好', '中等', '及格', '不及格'} );
%将它放在图上
```

(8) 『拟合图像』

```
time=[...略...];
temp=[...略...];
scatter(time , temp , '*', 'Linewidth', 2); %画散点图
p=polyfit(time, temp, 5);
%polyfit将time,temp拟合成5次方的多项式，形成1个相应的多项式向量积
y=polyval(p,time);
%算出p代表的多项式在各点x处的值
f=poly2sym(p); %将p所代表的-->转化-->符号函数
```

```
x=9;
y=subs(f,x);
hold on;
plot(x,y,'rs');
ezplot(f, [0 9] );
legend('\bf散点图', '\bf poly', '预测点' );
```

- subplot(2,2,[2,4]); 合并2,4区

```
x=linspace(-3*pi, 3*pi, 151);
y=sin(x)./x;
plot(x,y,'Linewidth',8, 'color', 'r');
%gcf是当前窗口的句柄
set( gcf, 'color', 'y'); %将当前底图颜色弄成黄色
set(gca,'YGrid','on'); %当前窗口的坐标轴句柄
```

```
x=-2:.5:2;
y=randn(10,1); %产生正态分布随机数10个，为列向量
h=figure;
set(h,'Name','直方图和条形图');%设置弹出的框的最左上的为该名
subplot(2,3,[1,4]);
hist(y,x);
[count,centers]=hist(y,x);
%在y中统计每个区间的数据个数count向量，centers为中心

subplot(2,3,[2 5] );
bar(centers,count); %条形图
subplot(2,3,[3 6]);
y=[end]=[]; %使x,y长度一致

helpdlg('请理解'); %对话框
```

九、函数类-零碎代码

```
function result=main_a(a)
%将数据分割赋值
b=a(1);
c=a(2);
d=a(3);
e=a(4);
add_result=add_1(b,c,d,e);
sub_result=sub_1(b,c,d,e);
```

```
mul_result=mul_1(b,c,d,e);
div_result=div_1(b,c,d,e);
result=add_result+sub_result+mul_result+div_result;
```

%% 子函数

```
function add_result=add_1(b,c,d,e)
    add_result=b+c+d+e;
function sub_result=sub_1(b,c,d,e)
    sub_result=b-c-d-e;
function mul_result=mul_1(b,c,d,e)
    mul_result=b*c*d*e;
function div_result=div_1(b,c,d,e)
    div_result=b*c/d*e;
```

```
i3=num2str(ii^3);
i4=num2str(ii^4);
i_34=sort( [i3 i4] ); %两数合并后升序排列
    '0 1 2 3 4 5 6 7 8 9'
disp( ['这数是' num2str(ii) ] );
disp( ['它的3次方是' i3, '4次方是' i4 ] );
%告诉了我们disp似fprintf的输出??
%1) 这个数是18
%2) 它的3次方是5832,4次方是104976
```

- `isequal(i_34, '0123456789');`

十、GUI类-零碎代码

```
set(hh, "UIContextMenu", handles.kkk );
```

```
get(handles.popupmenu1, 'value'); %弹出菜单
%获得弹出菜单，选了第n行
%下面是『清楚方式』
%1)set(handles.edit1, 'string', '');
%2)cls %清楚图像
```

- 关于:
- 『滑动条』、『复选框』的笔记，自己看不懂自己的速记笔记。。。