



MATLAB 基础

翟晓雅

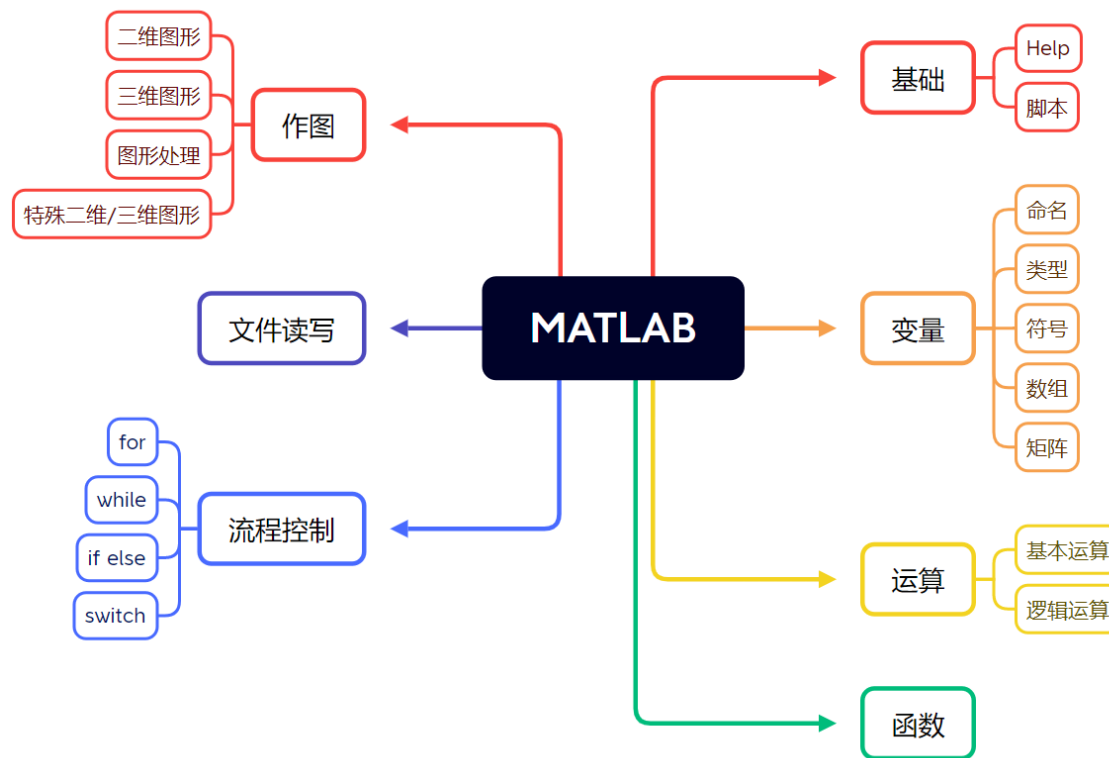
Email: xiaoyazhai@ustc.edu.cn

Homepage: <https://xiaoyazhai.github.io/>

本课件仅用于中科大教学目的，禁止在网络上传播分享！

MATLAB 基础

- 变量（数值（向量，矩阵），符号变量）
- 运算
- 函数
- 流程控制
- 文件读写
- 作图



1. MATLAB HELP 的使用

- HELP的进入
- 函数的查找
 - 按功能
 - 按名称
 - 按关键词
- 函数帮助文档的使用
- 以simple函数为例

2. MATLAB 执行程序

- 交互式命令
- 脚本 (M-Script)
- 函数 (Function)

其中，函数和脚本可以采用如下方法新建

1. 在MATLAB中，点击:File→New→M-file
2. 在编辑窗口中输入程序内容
3. 点File → Save，存盘，如果所编辑的文件为函数，M文件名必须与函数名一致.

2. MATLAB 脚本

- 脚本的建立

1. 在MATLAB中， 点击:File→New→M-file
2. 在编辑窗口中输入程序内容
3. 点File → Save， 存盘

- 执行脚本的两种方法

- 在命令行窗口输入脚本文件名
- 在M文件编辑器内， 单击 运行脚本程序

2. MATLAB 脚本-常用的标点符号

- “;”MATLAB的每条命令后，若为**逗号或无标点符号**，则显示命令的结果；若命令后为**分号**，则禁止显示结果.
- “%”后面所有文字为注释.
- “...”表示续行.
- “%%”较新版本的MATLAB支持，程序的功能段划分.

3. 变量-命名规则

- 变量名由**字母**开头，后面可以是任意**字母、数字或下划线**。变量名中**不允许**使用标点符号。
- 不超过19个字符
- 区分大小写Temp ≠ temp
- 请判断如下的单个变量命名是否正确？
- 00var; var00; va r00; var_temp; _vartemp; var.temp

3. 变量-特殊变量

特殊变量	取 值
ans	用于结果的缺省变量名
pi	圆周率
eps	计算机的最小数, 和1相加时产生一个比1大的数
inf	无穷大, 如1/0
NaN	不定量, 如0/0
i, j	$i=j=\sqrt{-1}$
nargin	所用函数的输入变量数目
nargout	所用函数的输出变量数目
realmin	最小可用正实数
realmax	最大可用正实数

3. 变量-类型

数学建模中主要涉及到两种变量：

数值型（包括整型、双精度型、无符号整型等，矩阵为最主要形式，用于数值运算，可以为复数矩阵；还可以直接定义出向量、标量等）

符号变量（用于公式推导和解析解求解）

其他数据类型

字符串、多维数组、结构体、类与对象等

3. 变量-符号

符号变量，用于公式推导和求解解析解，
有两种等价定义方式：

方式1

- `syms x y real;`

方式2

- `x = sym('x','real'); y = sym('y','real');`

定义完符号型变量，就可以表达复杂函数

如： $F = x^2 + \exp(y)$;

3. 变量-符号

用subs函数代换符号型变量

- 单变量代换

```
syms a b;
```

```
subs(a + b, a, 4)
```

- 多变量代换

- `syms a b; subs(cos(a) + sin(b), {a, b}, {sym('alpha'), 2})`

请问下列语句的返回结果应该是什么？

```
syms x y; subs(x*y, {x, y}, {[0 1; -1 0], [1 -1; -2 1]})
```

Single Substitution

Replace a with 4 in this expression.

```
syms a b  
subs(a + b, a, 4)
```

Replace a*b with 5 in this expression.

```
subs(a*b^2, a*b, 5)
```

```
>> syms a b  
subs(cos(a) + sin(b), [a,b], [sym('alpha'),2])  
  
ans =  
  
sin(2) + cos(alpha)
```

3. 变量-符号

符号型变量的常用函数

- simple/ simplify 函数化简
- expand 函数展开
- collect 合并同类项

```
syms x y
coeffs_x = collect(x^2*y + y*x - x^2 - 2*x, x)
coeffs_y = collect(x^2*y + y*x - x^2 - 2*x, y)

coeffs_x =
(y - 1)*x^2 + (y - 2)*x
coeffs_y =
(x^2 + x)*y - x^2 - 2*x
```

```
syms x a b c
S = simplify(sin(x)^2 + cos(x)^2)
```

S = 1

```
S = simplify(exp(c*log(sqrt(a+b))))
```

S = $(a + b)^{c/2}$

```
syms x
M = [(x^2 + 5*x + 6)/(x + 2), sin(x)*sin(2*x) + cos(x)*cos(2*x);
      (exp(-x*1i)*1i)/2 - (exp(x*1i)*1i)/2, sqrt(16)];
S = simplify(M)
```

S =

$$\begin{pmatrix} x + 3 & \cos(x) \\ \sin(x) & 4 \end{pmatrix}$$

```
syms x
p = (x - 2)*(x - 4);
expand(p)
```

ans =

$$x^2 - 6x + 8$$

```
syms x y
expand(cos(x + y))
```

ans =

$$\cos(x)\cos(y) - \sin(x)\sin(y)$$

3. 变量-数组创建

`x=[a b c d e f]` %创建包含指定元素的行向量.

`x=first:last` %创建从first开始, 加1计数, 到last结束的行向量.

如`x=1:5;`

结果为 `x=[1,2,3,4,5]`

请问`x=1:0` 结果如何?

`x=first:increment:last` %创建从first开始, 加increment计数, 到last结束的行向量.

3. 变量-数组访问

访问单个元素: $x(i)$ 表示访问数组x的第i个元素.

访问一块元素: $x(a:b:c)$ 表示访问数组x的第a个元素开始, 以步长b到第c个元素(但不超过c), b可以为负数, b缺省时为1.

直接使用元素编址序号: $x([a \ b \ c \ d])$ 表示提取数组x的第a、b、c、d个元素构成一个新的数组 $[x(a) \ x(b) \ x(c) \ x(d)]$.

3. 变量-矩阵创建

生成双精度实数矩阵 $A = \begin{bmatrix} 1.1 & 0 & 0 \\ 0 & 1.2 & 0 \\ 0 & 0 & 1.3 \end{bmatrix}$

- `A=[1.1,0,0;0,1.2,0;0,0,1.3]`

说明：以**空格**或**逗号**分隔的元素指定的是**不同列**的元素，而以**分号**分隔的元素指定了**不同行**的元素。

- `A=[1.1,0,0;0,1.2,0;0,0,1.3]` %直接赋值
- `A=[1.1 0 0;0,1.2,0;0,0,1.3]` %直接赋值
- `A=[[1.1,0,0];[0,1.2,0];[0,0,1.3]]` %由几个更小的矩阵组合产生
- `A=[]; [1.1,0,0];[0,1.2,0];[0,0,1.3]]` %由空矩阵及更小的矩阵组合产生

3. 变量-矩阵创建

特殊矩阵

- 空矩阵 $A=[];$
- 单位矩阵 $A=\text{eye}(m,n)$
- 零矩阵 $A=\text{zeros}(n,m)$
- 对角矩阵 $A=\text{diag}([1,3,5,7,2])$
- 均匀分布随机数矩阵 $A=\text{rand}(n,m)$

3. 变量-矩阵创建

创建复数矩阵

- $A = \begin{bmatrix} 1 + 2i & 2 + 3i \\ 3 + 4i & 4 + 5i \end{bmatrix}$

方法1, 直接赋值

- $A = [1+2i, 2+3i; 3+4i, 4+5i]$

方法2, 用complex函数

- $c = [1, 2; 3, 4]; d = [2, 3; 4, 5]; A = \text{complex}(c, d)$

3. 变量-矩阵元素访问

创建复数矩阵

- $A = \begin{bmatrix} 1 + 2i & 2 + 3i \\ 3 + 4i & 4 + 5i \end{bmatrix}$

方法1, 直接赋值

- $A = [1+2i, 2+3i; 3+4i, 4+5i]$

方法2, 用complex函数

- $c = [1, 2; 3, 4]; d = [2, 3; 4, 5]; A = \text{complex}(c, d)$

3. 变量-矩阵元素访问

- 以逆序提取矩阵A的第j1~j2列，构成新矩阵： $A(:,j2:-1:j1)$
- 删除A的第i1~i2行，构成新矩阵： $A(i1:i2,:)=[]$
- 删除A的第j1~j2列，可以怎么做？
- 将矩阵A和B拼接成新矩阵： $[A \ B]$; $[A; B]$

4. 运算-基本运算

+	加法运算，适用于两个数或两个同阶矩阵相加.
—	减法运算
*	乘法运算
.*	点乘运算
/	除法运算
./	点除运算
^	乘幂运算
.^	点乘幂运算
\	反斜杠表示左除.

$A=[1,2,3;4,5,6;7,8,0]; B=A.^A$

$A=[1,2,3;4,5,6;7,8,0]; B=A.*A$

B =

1	4	27
256	3125	46656
823543	16777216	1

B =

1	4	9
16	25	36
49	64	0

4. 运算-逻辑运算

- 逻辑变量：
 - 当前版本有逻辑变量
 - 对 double 变量来说，非 0 表示逻辑 1
- 逻辑运算（相应元素间的运算）
 - 与运算 $A \& B$
 - 或运算 $A | B$
 - 非运算 $\sim A$
 - 异或运算 $\text{xor}(A, B)$
- 比较关系
 - $>$, \geq , $<$, \leq , $==$, $\sim =$, $\text{find}()$, $\text{all}()$, $\text{any}()$

5. 函数

函 数	名 称	函 数	名 称
$\sin(x)$	正弦函数	$\text{asin}(x)$	反正弦函数
$\cos(x)$	余弦函数	$\text{acos}(x)$	反余弦函数
$\tan(x)$	正切函数	$\text{atan}(x)$	反正切函数
$\text{abs}(x)$	绝对值	$\text{max}(x)$	最大值
$\text{min}(x)$	最小值	$\text{sum}(x)$	元素的总和
$\text{sqrt}(x)$	开平方	$\text{exp}(x)$	以 e 为底的指数
$\log(x)$	自然对数	$\log_{10}(x)$	以 10 为底的对数
$\text{sign}(x)$	符号函数	$\text{fix}(x)$	取整

5. 函数

- 函数文件是文件名后缀为M的文件
- 第一行必须是function开始，格式为：
 - function 输出变量=函数名（输入变量）
- 函数值的获得必须通过具体的运算实现，并赋给因变量.
- 注意：函数的文件名必须与函数名一致.

5. 函数实例

- 例：定义函数 $f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$
- 1. 建立M文件：myfun.m

```
function f=myfun(x)
    f = 100*(x(2)-x(1)^2)^2+(1-x(1))^2
```
- 2. 使用myfun函数
- 如，要计算 $f(1, 2)$ ，只需在MATLAB命令窗口键入命令：

```
x=[1 2]
myfun(x)
```


6. 流程控制-for 循环

- for循环允许一组命令以固定的和预定的次数重复
- for循环标准格式
for x=array
 循环结构体
end

- 例 $S = \sum_{i=1}^{100} i$
s=0;
for i=1:100
 s=s+i;
end
s

6. 流程控制-while 循环

- 与for循环以**固定次数**求一组命令相反，while循环以**不定的次数**求一组语句的值.
- 只要在表达式(expression)里的所有元素为真，就执行while和end语句之间的循环结构体.

while循环的标准格式

```
while (expression)
```

```
    循环结构体
```

```
end
```

6. 流程控制-while 循环

- 例 设银行年利率为11.25%，将10000元钱存入银行，问多长时间会连本带利翻一番？

```
money=10000;
```

```
years=0;
```

```
while money<20000
```

```
    years=years+1;
```

```
    money=money*(1+11.25/100);
```

```
end
```

6. 流程控制-转移结构

- 转移结构的标准格式

```
if (condition1)
    {commands1}
elseif (condition2)
    {commands2}
elseif .....
else
    {commands}
end
```

例 用循环求解 $\sum_{i=1}^m i > 1000$, 求最小的 m .

```
s=0;
for i=1:10000
    s=s+i;
    if s>10000,
        break;
    end
end
```

6. 流程控制-开关结构

- 开关结构的标准格式

```
switch switch_expression
    case case_expression statements
    case case_expression statements
    :
    otherwise statements
end
```

例

```
mynumber = input('Enter a number:');
switch mynumber
    case -1 disp('negative one');
    case 0 disp('zero');
    case 1 disp('positive one');
    otherwise disp('other value');
end
```

7. 常见文件的读写

- xlsread
- xlswrite
- save
- Load

8. MATLAB 作图

- 二维图形
- 三维图形
- 图形处理
- 特殊二维/三维图形

8. MATLAB 作图

- MATLAB作图是通过描点、连线来实现的，故在画一个曲线图形之前，必须先取得该图形上的一系列的点的坐标（即横坐标和纵坐标），然后将该点集的坐标传给MATLAB函数画图。

命令为：

plot(X,Y,S)

线型

X, Y是向量, 分别表示点集的横坐标和纵坐标

•y	黄色	•	点	-	连线
•m	洋红	o	圈	:	短虚线
•c	蓝绿色	x	x-符号		
-.	长短线	r	红色		
+	加号	--	长虚线		

plot(x,y)——画实线

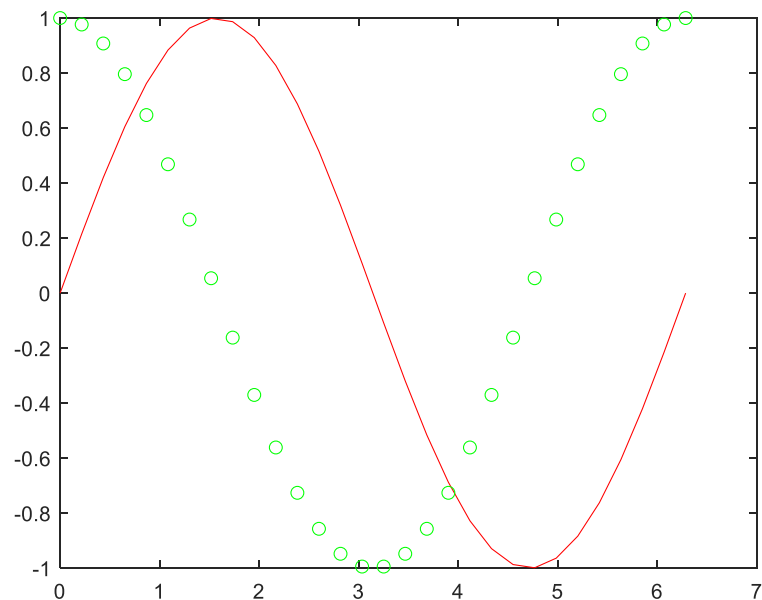
plot(x,y1,s1,x,y2,s2,.....,x,yn,sn)

——将多条线画在一起

8. MATLAB 作图

解 例 在 $[0, 2\pi]$ 用红线画 $\sin x$, 用绿圈画 $\cos x$.

```
x=linspace(0,2*pi,30);  
y=sin(x);  
z=cos(x);  
plot(x,y,'r',x,z,'go')
```



8. MATLAB 作图-符号函数(显函数、隐函数和参数方程)画图

- (1) `ezplot` (新版matlab不推荐使用)
- `ezplot('f(x)', [a, b])`
- 表示在 $a < x < b$ 绘制显函数 $f=f(x)$ 的函数图.
- `ezplot('f(x, y)', [xmin, xmax, ymin, ymax])`
- 表示在区间 $xmin < x < xmax$ 和 $ymin < y < ymax$ 绘制隐函数 $f(x, y)=0$ 的函数图.
- `ezplot('x(t)', 'y(t)', [tmin, tmax])`
- 表示在区间 $tmin < t < tmax$ 绘制参数方程 $x=x(t), y=y(t)$ 的函数图.

8. MATLAB 作图

解 输入命令

```
ezplot('sin(x)', [0, pi])
```

例 在 $[0, 2\pi]$ 上画 $x = \cos^3 t$, $y = \sin^3 t$ 星形图.

解 输入命令

```
ezplot('cos(t)^3', 'sin(t)^3', [0, 2*pi])
```

例 在 $[-2, 0.5]$, $[0, 2]$ 上画隐函数 $e^x + \sin(xy) = 0$ 的图.

解 输入命令

```
ezplot('exp(x)+sin(x*y)', [-2, 0.5, 0, 2])
```

8. MATLAB 作图

- `fplot('fun',lims)`

表示绘制字符串`fun`指定的函数在`lims=[xmin,xmax]`的图形。

- 注意：
- [1] `fun`必须是M文件的函数名或是独立变量为`x`的字符串。
- [2] `fplot`函数不能画参数方程和隐函数图形，但在一个图上可以画多个图形。

8. MATLAB 作图

例 在 $[-1, 2]$ 上画 $y = e^{2x} + \sin(3x^2)$ 的图形.

解 先建M文件myfun1.m:

```
function Y=myfun1(x)
Y=exp(2*x)+sin(3*x.^2)
```

再输入命令:

```
fplot('myfun1', [-1, 2])
```

例 在 $[-2, 2]$ 范围内绘制函数 \tanh 的图形.

解 `fplot('tanh', [-2, 2])`

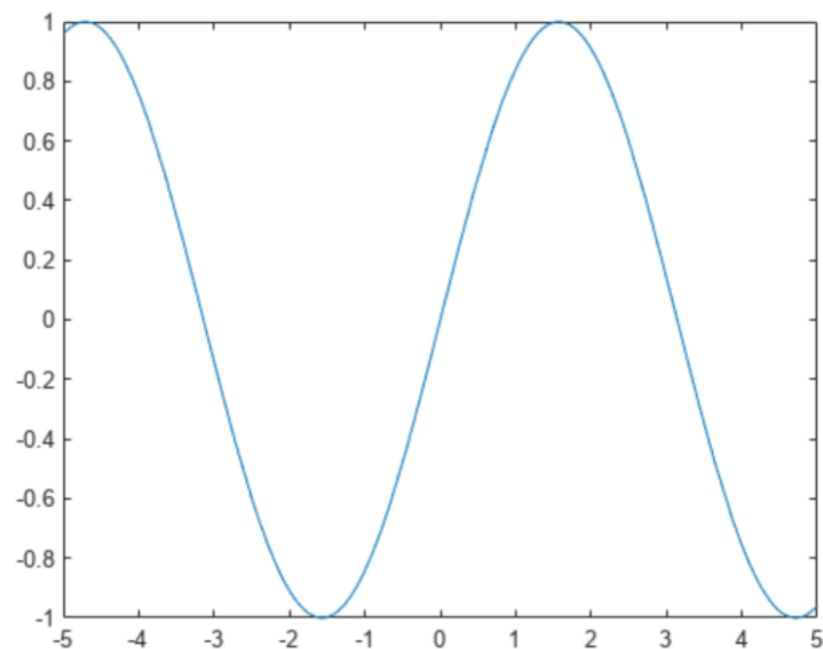
例 x 、 y 的取值范围都在 $[-2\pi, 2\pi]$,
画函数 $\tanh(x)$, $\sin(x)$, $\cos(x)$ 的图形.

解 输入命令:

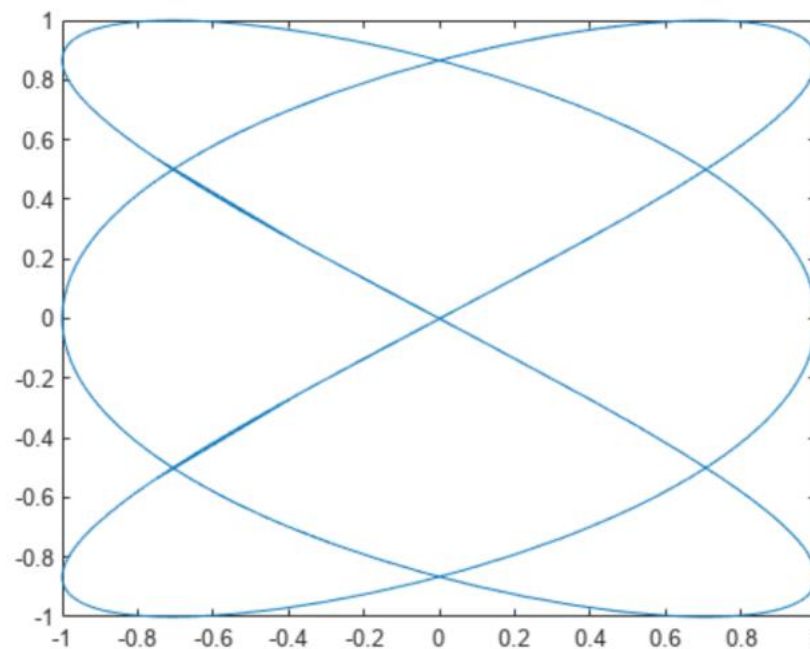
```
fplot('[tanh(x), sin(x), cos(x)]', 2*pi*[-1 1 -1 1])
```

8. MATLAB 作图

```
fplot(@(x) sin(x))
```



```
xt = @(t) cos(3*t);  
yt = @(t) sin(2*t);  
fplot(xt,yt)
```



8. MATLAB 作图

- 对数转换有双对数坐标转换和单轴对数坐标转换两种.用`loglog`函数可以实现双对数坐标转换,用`semilogx`和`semilogy`函数可以实现单轴对数坐标转换.

`loglog(Y)` 表示 x 、 y 坐标都是对数坐标系

`semilogx(Y)` 表示 x 坐标轴是对数坐标系

`semilogy(...)` 表示 y 坐标轴是对数坐标系

`plotyy` 有两个 y 坐标轴，一个在左边，一个在右边

8. MATLAB 作图

`plot3(x,y,z,s)`

n维向量，分别表示曲线上点集的横坐标、纵坐标、函数值

指定颜色、线形等

`plot3(x,y,z)`

其中 x , y , z 是都是 $m \times n$ 矩阵，其对应的每一列表示一条曲线。

例 在区间 $[0, 10\pi]$ 画出参数曲线 $x=\sin t, y=\cos t,$

$z=t.$

解

```
t=0:pi/50:10*pi;
```

```
plot3(sin(t),cos(t),t)
```

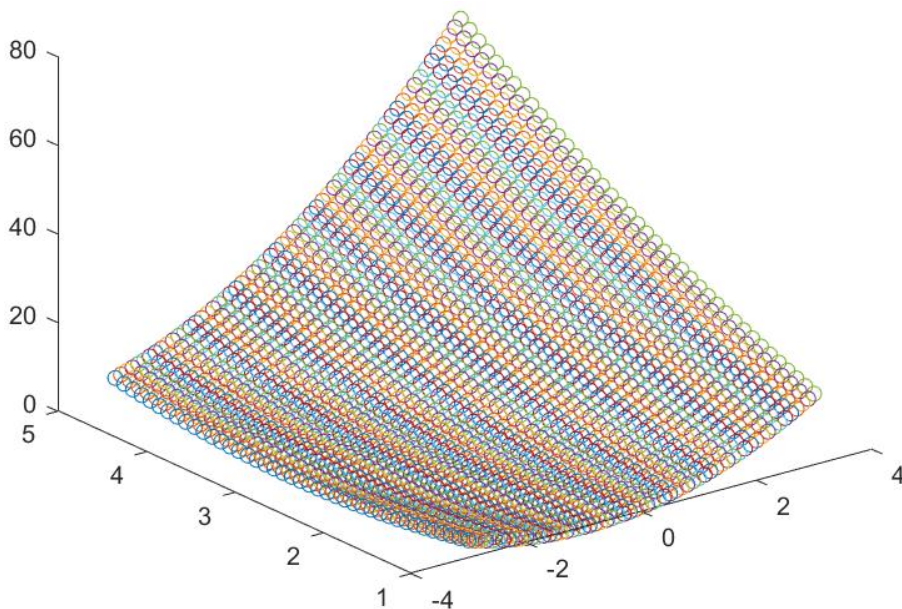
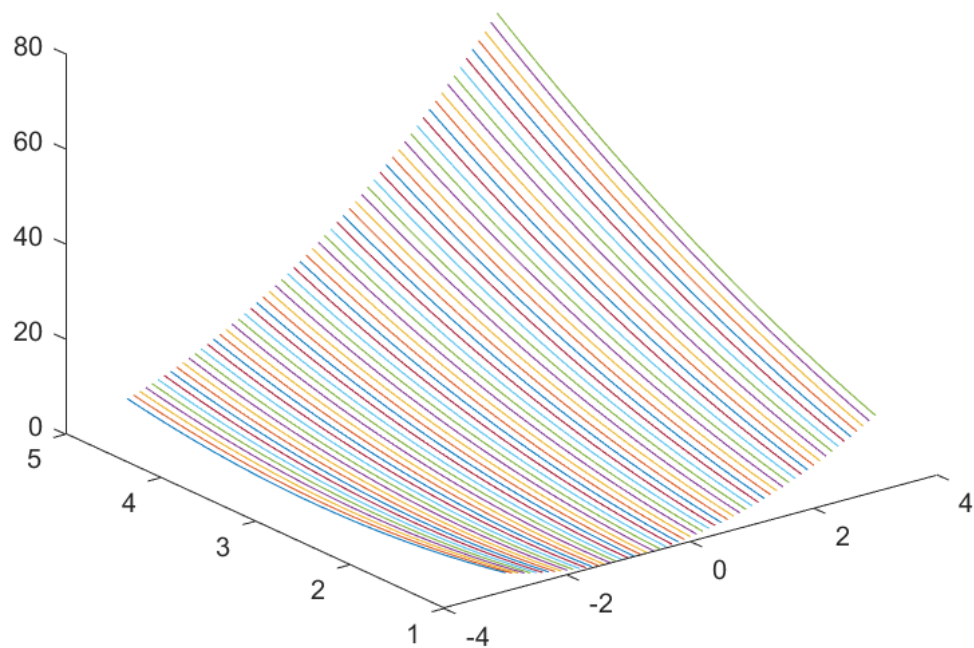
```
rotate3d %旋转
```


8. MATLAB 作图

例 画多条曲线观察函数 $z = (x+y)^2$ 。

解 `x=-3:0.1:3;y=1:0.1:5;`
`[X,Y]=meshgrid(x,y);`
`Z=(X+Y).^2;`
`plot3(X,Y,Z)`

(这里`meshgrid(x,y)`的作用是产生一个以向量`x`为行、向量`y`为列的矩阵)



8. MATLAB 作图

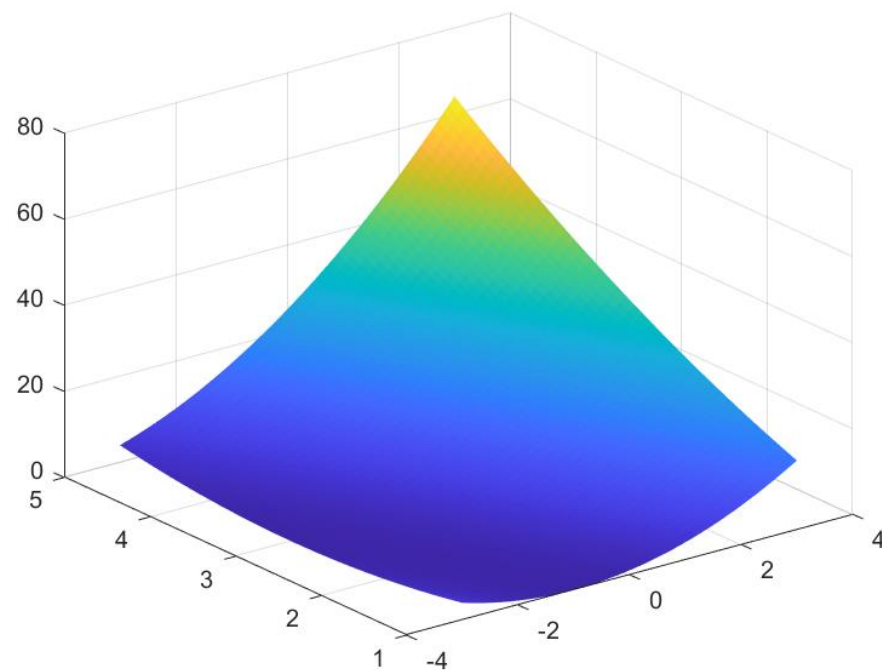
- **surf(x,y,z)** 画出数据点 (x, y, z) 表示的曲面

数据矩阵. 分别表示
数据点的横坐标、纵
坐标、函数值

例 画函数 $z = (x+y)^2$ 的图形.

解

```
x=-3:0.1:3;  
y=1:0.1:5;  
[X,Y]=meshgrid(x,y);  
Z=(X+Y).^2;  
surf(X,Y,Z)  
shading flat           %将当前图形变得平滑
```



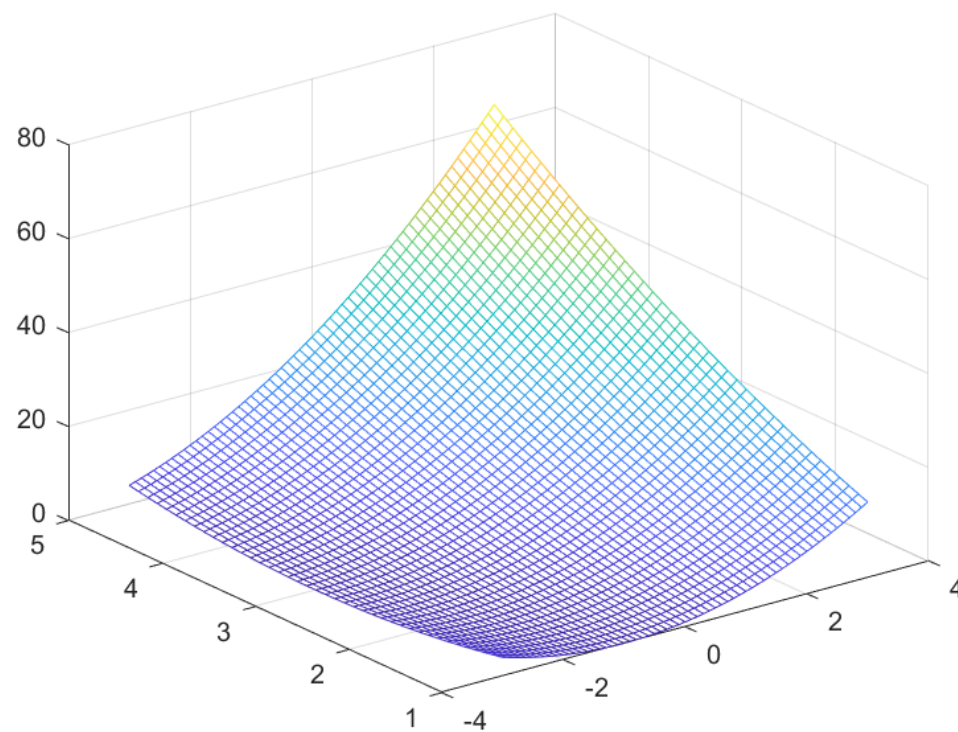
8. MATLAB 作图

- **mesh(x,y,z)** 画网格曲面

数据矩阵. 分别表示
数据点的横坐标、纵
坐标、函数值

例 画出曲面 $z = (x+y)^2$ 在不同视角的网格图.

解 `x=-3:0.1:3; y=1:0.1:5;`
`[X,Y]=meshgrid(x,y);`
`Z=(X+Y).^2;`
`mesh(X,Y,Z)`



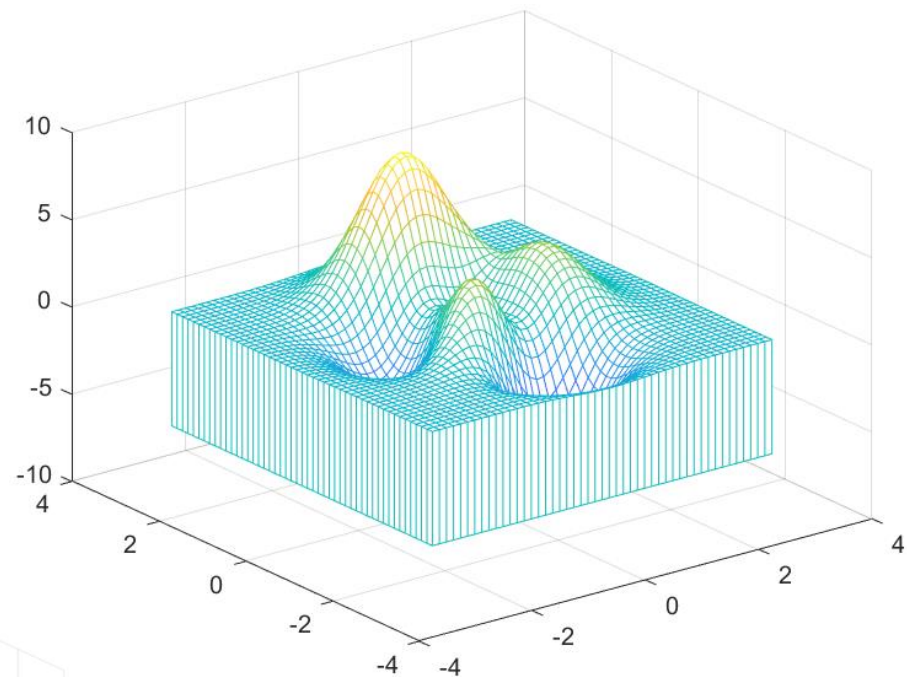
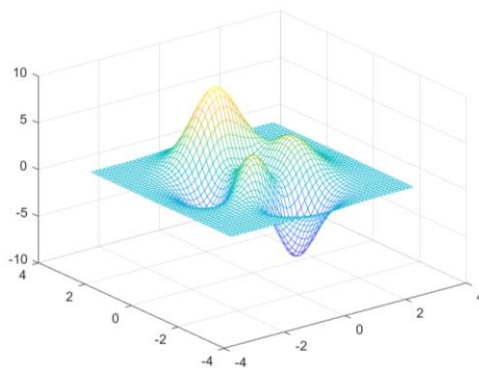
8. MATLAB 作图

- **meshz(X,Y,Z)** 在网格周围画一个curtain图(如,参考平面)

例 绘peaks的网格图

解 输入命令:

```
[X,Y]=meshgrid(-3:.125:3);  
Z=peaks(X,Y);  
meshz(X,Y,Z)
```



8. MATLAB 作图



8. MATLAB 作图

在图形上加格栅、图例和标注

(1) **GRID ON**: 加格栅在当前图上

GRID OFF: 删除格栅

(2) **hh = xlabel(string)**: 在当前图形的x轴上加图例string

hh = ylabel(string): 在当前图形的y轴上加图例string

hh = zlabel(string): 在当前图形的z轴上加图例string

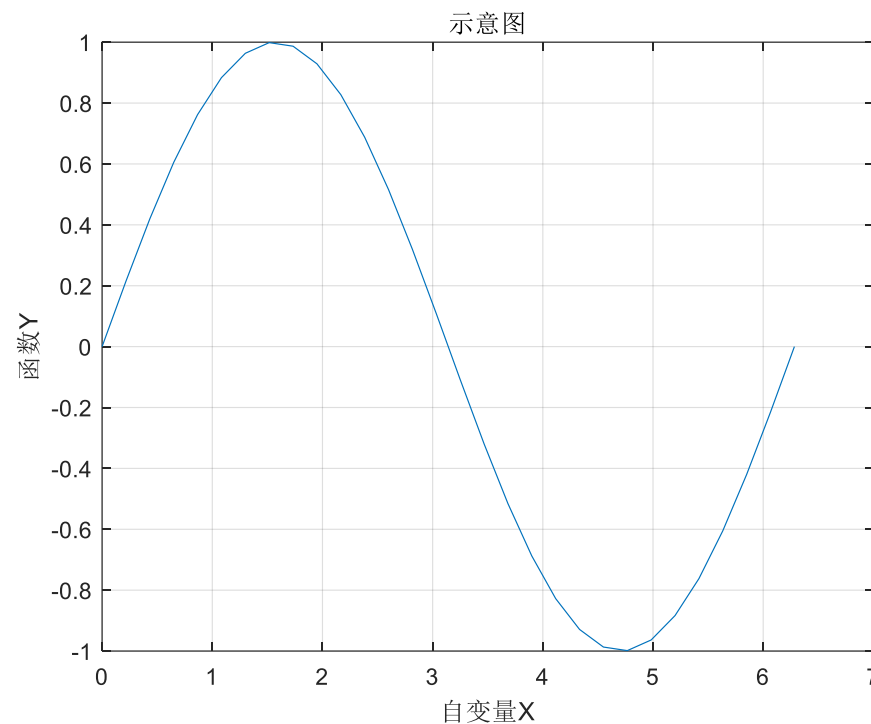
hh = title(string): 在当前图形的顶端上加图例string

8. MATLAB 作图

例 在区间 $[0, 2\pi]$ 画 $\sin(x)$ 的图形，并加注图例“自变量 x ”、“函数 y ”、“示意图”，并加格栅。

解

```
x=linspace(0,2*pi,30);  
y=sin(x);  
plot(x,y)  
xlabel('自变量x')  
ylabel('函数Y')  
title('示意图')  
grid on
```



8. MATLAB 作图

- `hh = gtext('string')`

命令`gtext('string')`用鼠标放置标注在现有的图上. 运行命令`gtext('string')`时, 屏幕上出现当前图形, 在图形上出现一个交叉的十字, 该十字随鼠标的移动移动, 当按下鼠标左键时, 该标注`string`放在当前十字交叉的位置.

例 在区间 $[0, 2\pi]$ 画 $\sin(x)$, 并分别标注 “ $\sin(x)$ ” “ $\cos(x)$ ”.

解

```
x=linspace(0, 2*pi, 30);
```

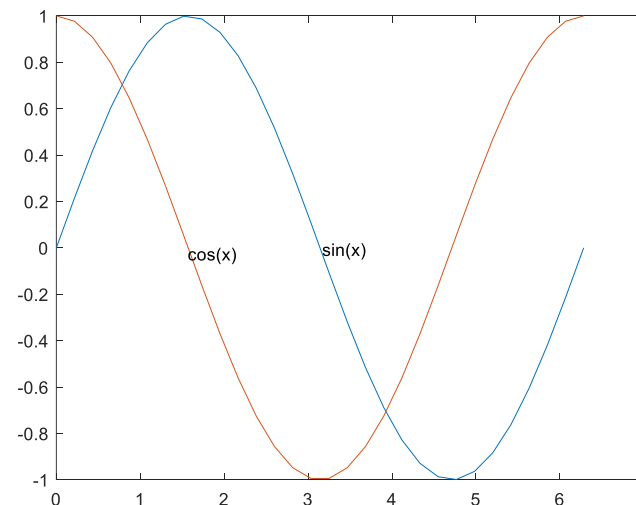
```
y=sin(x);
```

```
z=cos(x);
```

```
plot(x, y, x, z)
```

```
gtext('sin(x)'); gtext('cos(x)')
```

【0,2pi】
之间均分30份



8. MATLAB 作图

`Axis([xmin xmax ymin ymax zmin zmax])` 定制图形坐标

x、y、z的最大、最小值

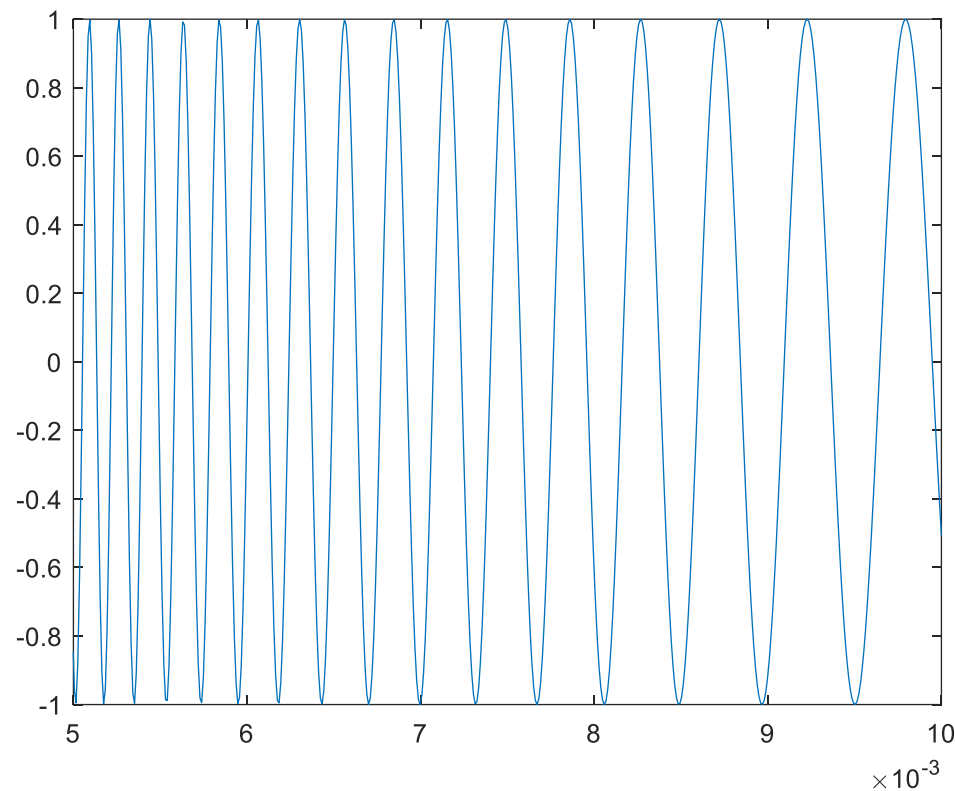
`Axis`
`auto`

将坐标轴返回到自动缺省值

例 在区间 $[0.005, 0.01]$ 显示 $\sin(1/x)$ 的图形.

解

```
x=linspace(0.0001,0.01,1000);  
y=sin(1./x);  
plot(x,y)  
axis([0.005 0.01 -1 1])
```

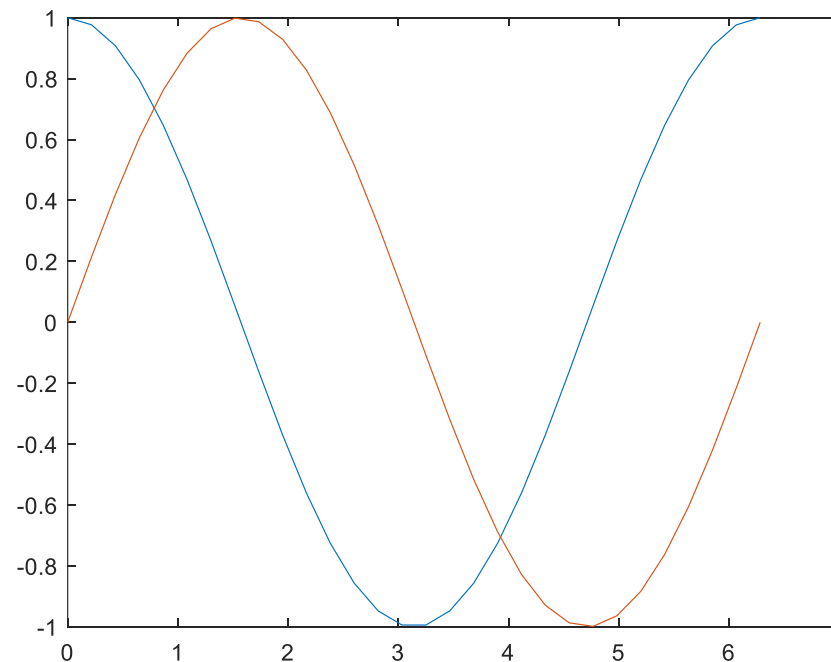


8. MATLAB 作图

(1) `hold on` 保持当前图形, 以便继续画图到当前图上
`hold off` 释放当前图形窗口

例 将 $y=\sin(x)$ 、 $y=\cos(x)$ 分别用点和线画出在同一屏幕上.

解 `x=linspace(0,2*pi,30);`
`y=sin(x);`
`z=cos(x)`
`plot(x,z,:)`
`hold on`
`plot(x,y)`



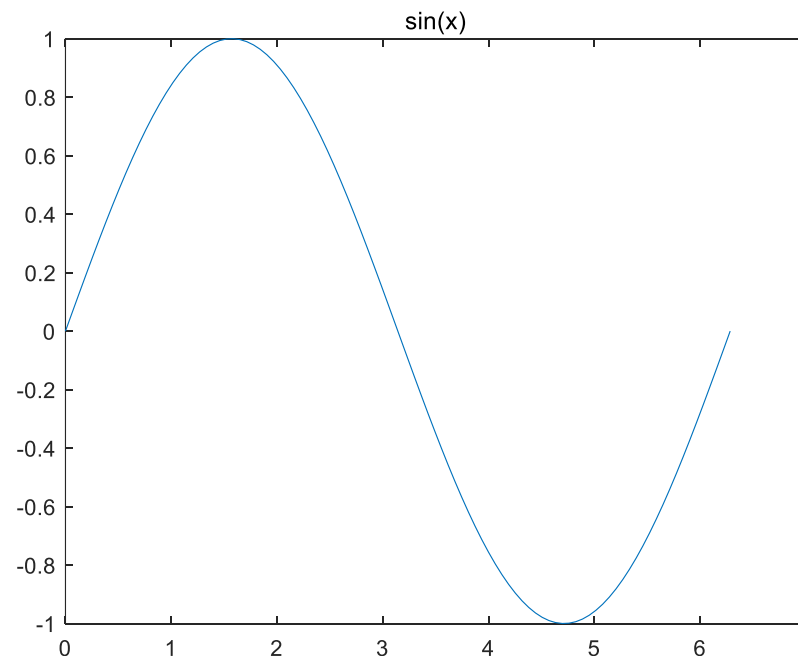
8. MATLAB 作图

(2) `figure(h)` 新建h窗口，激活图形使其可见，并把它置于其它图形之上

例 区间 $[0, 2\pi]$ 新建两个窗口分别画出 $y=\sin(x)$;
 $z=\cos(x)$.

解

```
x=linspace(0,2*pi,100);  
y=sin(x);z=cos(x);  
plot(x,y);  
title('sin(x)');  
pause  
figure(2);  
plot(x,z);  
title('cos(x)');
```



8. MATLAB 作图

```
h=subplot(mrows,ncols,thisplot)
```

划分整个作图区域为**mrows*ncols**块（逐行对块访问）并激活第**thisplot**块，其后的作图语句将图形画在该块上.

```
subplot(mrows,ncols,thisplot)
```

激活已划分为**mrows*ncols**块的屏幕中的第**thisplot**块，其后的作图语句将图形画在该块上.

```
subplot(1,1,1)
```

命令**Subplot(1,1,1)** 返回非分割状态.

8. MATLAB 作图

例 将屏幕分割为四块，并分别画出 $y=\sin(x)$ ， $z=\cos(x)$ ， $a=\sin(x) \times \cos(x)$ ， $b=\sin(x) / \cos(x)$ 。

解`x=linspace(0,2*pi,100);`

`y=sin(x); z=cos(x);`

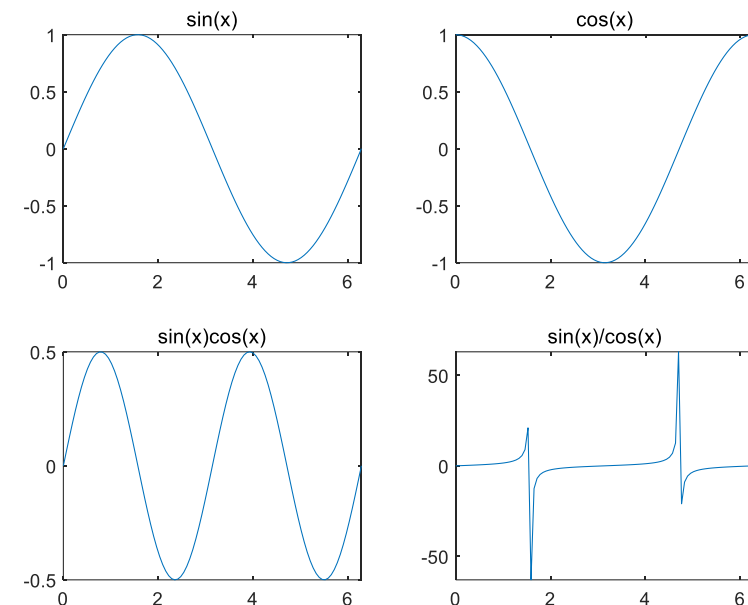
`a=sin(x).*cos(x);b=sin(x)./(cos(x)+eps)`

`subplot(2,2,1);plot(x,y),title('\sin(x)')`

`subplot(2,2,2);plot(x,z),title('\cos(x)')`

`subplot(2,2,3);plot(x,a),title('\sin(x)\cos(x)')`

`subplot(2,2,4);plot(x,b),title('\sin(x)/\cos(x)')`



8. MATLAB 作图

zoom on 为当前图形打开缩放模式

单击鼠标左键，则在当前图形窗口中，以鼠标点中的点为中心的图形放大2倍；单击鼠标右键，则缩小2倍。

zoom off 关闭缩放模式

例 缩放 $y=\sin(x)$ 的图形。

解

```
x=linspace(0,2*pi,30);  
y=sin(x);  
plot(x,y)  
zoom on
```



8. MATLAB 作图-改变视角

(1) `view(a,b)`

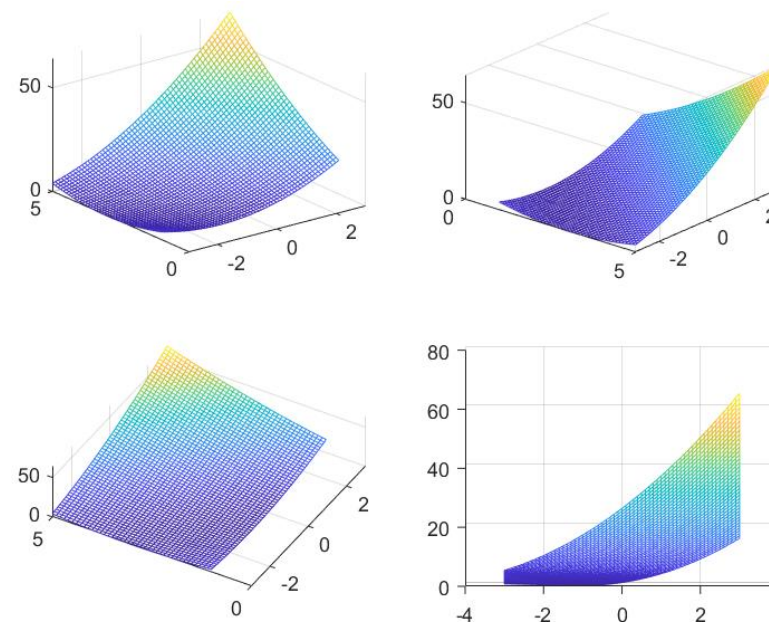
命令`view(a,b)`改变视角到 (a,b) , a 是方位角, b 为仰角.
缺省视角为 $(-37.5, 30)$.

(2) `view([x, y, z])`

`view`用空间向量表示的, 三个量只关心它们的比例, 与数值的大小无关, x 轴`view([1, 0, 0])`, y 轴`view([0, 1, 0])`, z 轴`view([0, 0, 1])`.

例 画出曲面 $z = (x+y)^2$ 在不同视角的网格图.

```
解 x=-3:0.1:3;    y=1:0.1:5;  
[X,Y]=meshgrid(x,y);  
Z=(X+Y).^2;  
subplot(2,2,1),  mesh(X,Y,Z)  
subplot(2,2,2),  mesh(X,Y,Z), view(50,-34)  
subplot(2,2,3),  mesh(X,Y,Z), view(-60,70)  
subplot(2,2,4),  mesh(X,Y,Z), view(0,1,1)
```



8. MATLAB 作图-动画

`Moviein()`, `getframe`, `movie()`

函数`Moviein()`产生一个帧矩阵来存放动画中的帧；函数`getframe`对当前的图像进行快照；函数`movie()`按顺序回放各帧。

例 将曲面`peaks`做成动画。

解

```
[x,y,z]=peaks(30);  
surf(x,y,z)  
axis([-3 3 -3 3 -10 10])  
m=moviein(15);  
for i=1:15  
    view(-37.5+24*(i-1),30)  
    m(:,i)=getframe;  
end  
movie(m)
```

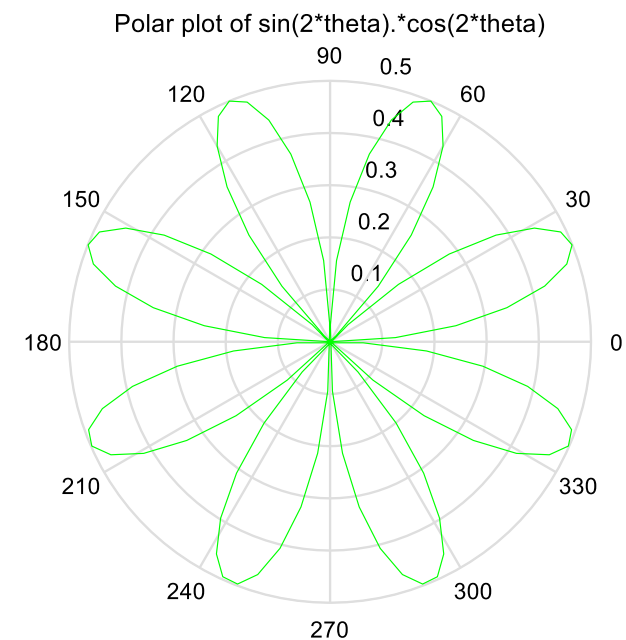

8. MATLAB 作图

极坐标图: `polar (theta,rho,s)`

用角度`theta`（弧度表示）和极半径`rho`作极坐标图，用`s`指定线型.

例 $r = \sin 2\theta \times \cos 2\theta$ 的极坐标图形.

```
解: theta=linspace(0,2*pi),  
rho=sin(2*theta).*cos(2*theta);  
polar(theta,rho,'g')  
title('Polar plot of ...  
sin(2*theta).*cos(2*theta)');
```



8. MATLAB 作图

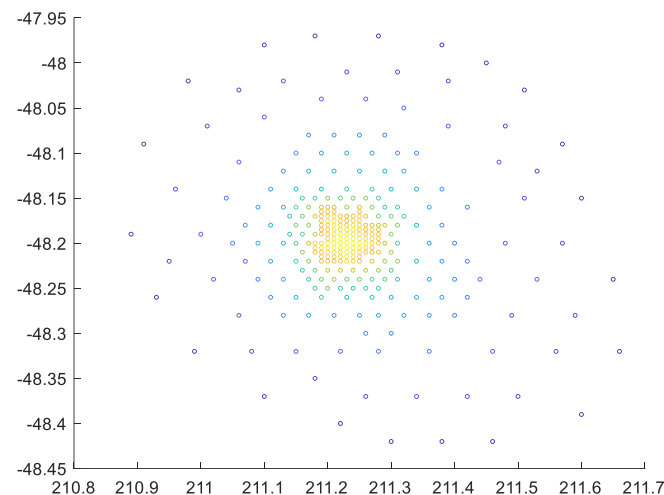
散点图: **scatter** (**x,y,s,c**)

在向量x和y的指定位置显示彩色圈. x和y必须大小相同.

例 绘制seamount散点图

解 输入命令:

```
load seamount
scatter(x,y,5,z)
```

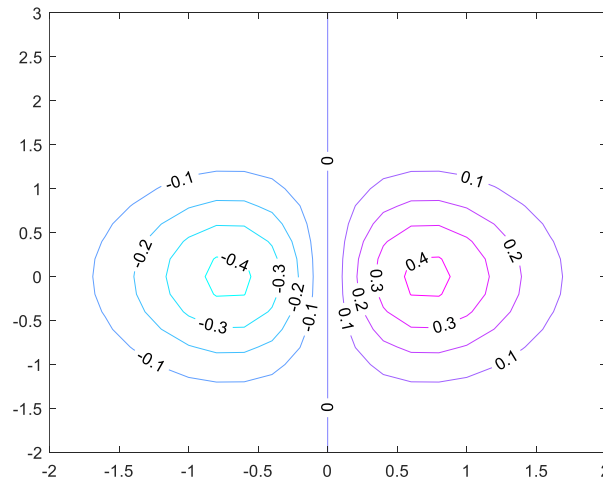


平面等值线图: **contour** (**x,y,z,n**) 绘制n个等值线的二维等值线图

例 在范围 $-2 < x < 2, -2 < y < 3$ 内绘 $z = xe^{-x^2-y^2}$ 的等值线图.

解 输入命令:

```
[X,Y]=meshgrid(-2:.2:2,-2:.2:3);
Z=X.*exp(-X.^2-Y.^2);
[C,h]=contour(X,Y,Z);
clabel(C,h)
colormap cool
```



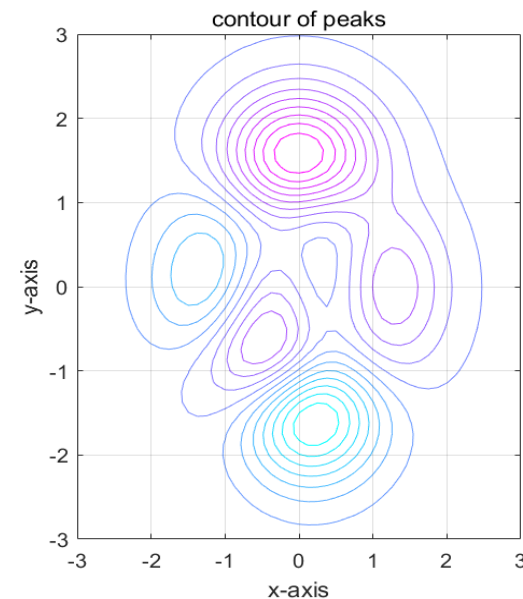
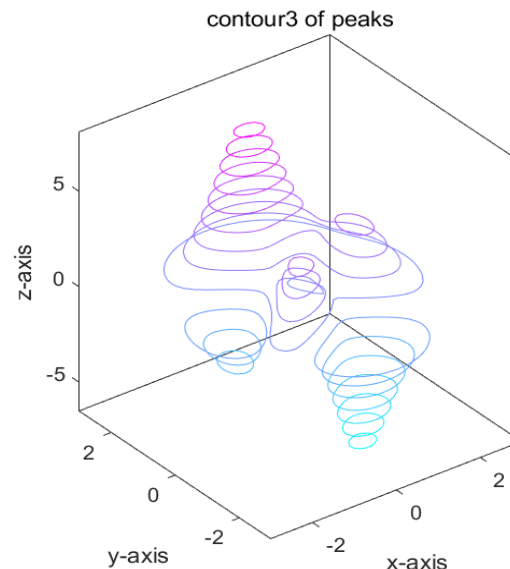
8. MATLAB 作图

空间等值线图: `contour3(x,y,z,n)`

其中n表示等值线数.

例 山峰的三维和二维等值线图.

```
解 [x,y,z]=peaks;  
subplot(1,2,1)  
contour3(x,y,z,16,'s')  
grid,    xlabel('x-axis'),ylabel('y-axis')  
zlabel('z-axis')  
title('contour3 of peaks');  
subplot(1,2,2)  
contour(x,y,z,16,'s')  
grid,    xlabel('x-axis'), ylabel('y-axis')  
title('contour of peaks');
```



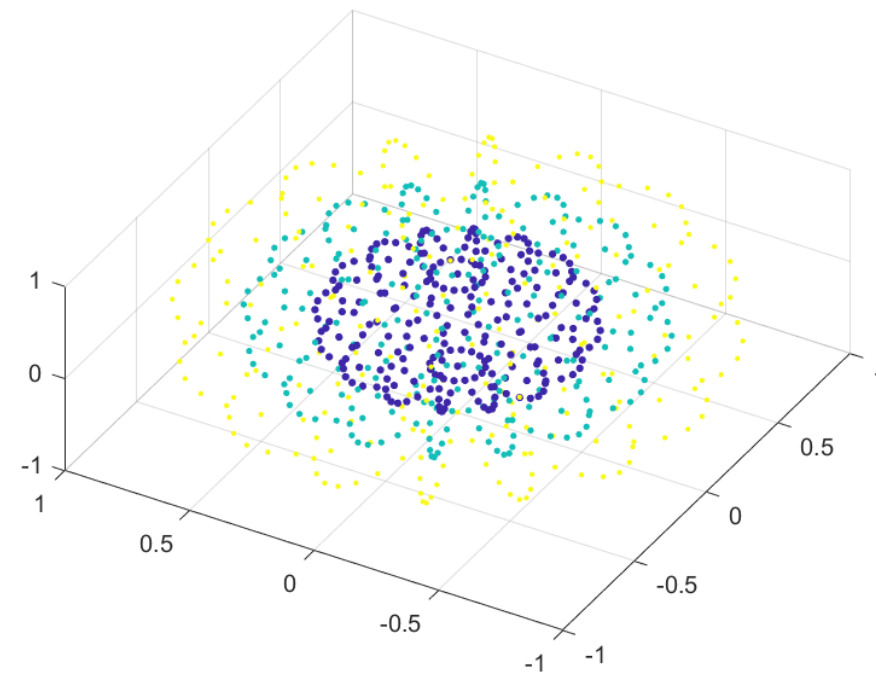
8. MATLAB 作图

三维散点图 `scatter3 (X,Y,Z,S,C)` 在向量 x , y 和 z 指定的位置上显示彩色圆圈. 向量 x , y 和 z 的大小必须相同.

例 绘制三维散点图.

解 输入命令:

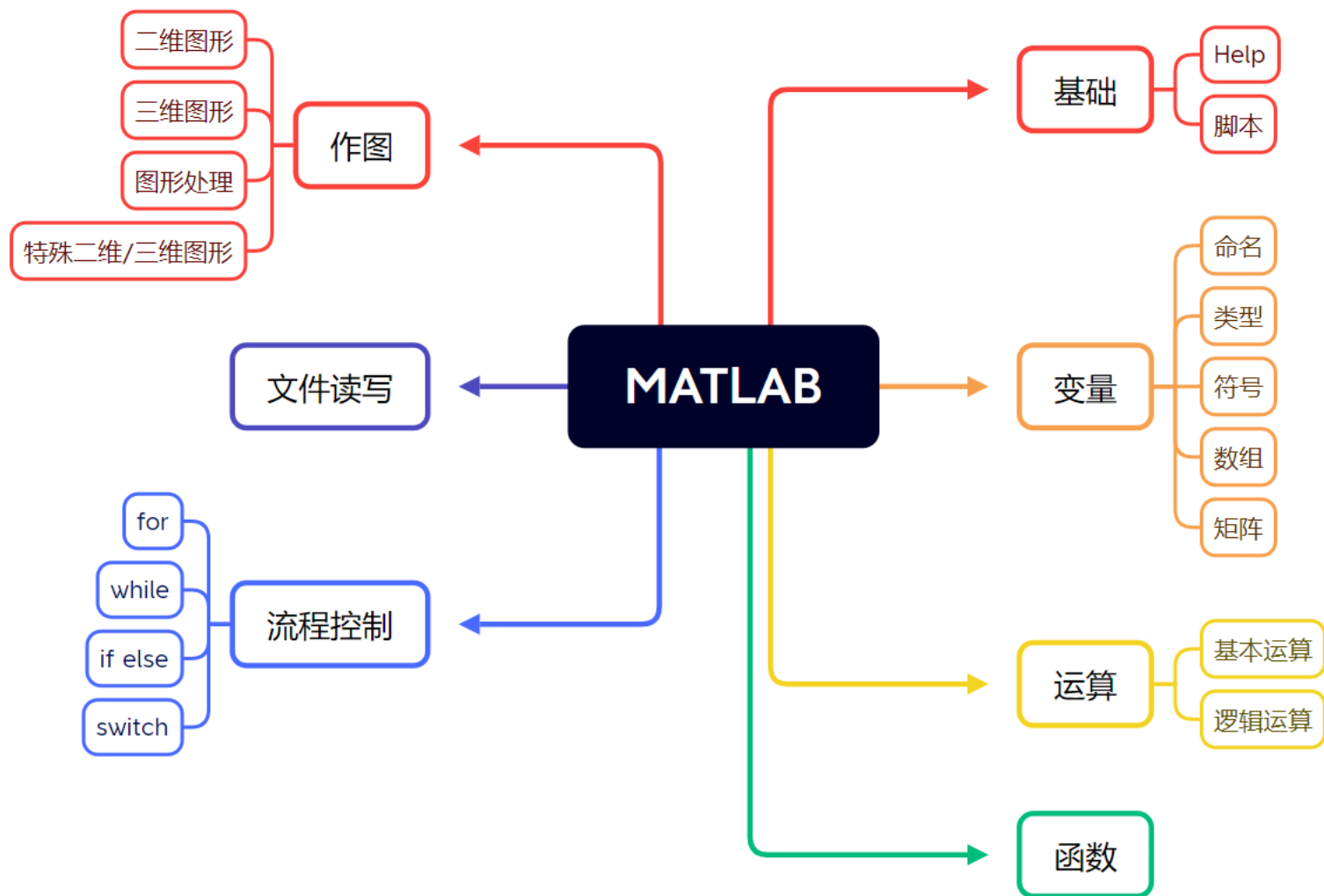
```
[x,y,z]=sphere(16);  
X=[x(:)*.5 x(:)*.75 x(:)];  
Y=[y(:)*.5 y(:)*.75 y(:)];  
Z=[z(:)*.5 z(:)*.75 z(:)];  
S= repmat([1 .75 .5]*10,prod(size(x)),1);  
C= repmat([1 2 3],prod(size(x)),1);  
scatter3(X(:),Y(:),Z(:),S(:),C(:),'filled'),  
view(-60,60)
```



9. Try some interesting...

- fifteen
- knot
- spy
- logo
- life
- lorenz

总结





Q&A?

下节课内容 实验五：素数

翟晓雅

Email: xiaoyazhai@ustc.edu.cn

Homepage: <https://xiaoyazhai.github.io/>

Lab: <http://gcl.ustc.edu.cn/>

