

Contents

- [曲线与曲面的绘制](#)
- [matlab可视化](#)
- [散点图](#)
- [多图](#)
- [平面连续图形的绘制](#)
- [同一窗口中绘制不同的曲线](#)
- [修饰](#)
- [微积分课中的一个对无界和无穷大的理解](#)
- [fplot绘制平面曲线](#)
- [polar作极坐标曲线图形](#)
- [ezplot函数作图](#)
- [双y轴坐标](#)
- [空间图形的描绘](#)
- [用plot3描绘空间图形](#)
- [表达式](#)
- [ezplot3函数](#)
- [用mesh函数--三维网格命令](#)
- [带网格线的三维表面图](#)
- [极坐标的方式效果更好](#)
- [三维曲面](#)
- [符号函数的表面图](#)
- [球面](#)
- [柱面](#)
- [等值线](#)
- [图像的进一步处理](#)
- [莫比乌斯带](#)
- [专题研究](#)
- [LaTeX Markup Example](#)

曲线与曲面的绘制

山东理工大学数学学院

```
%shixiangbupt@qq.com  
%2020年4月14日
```

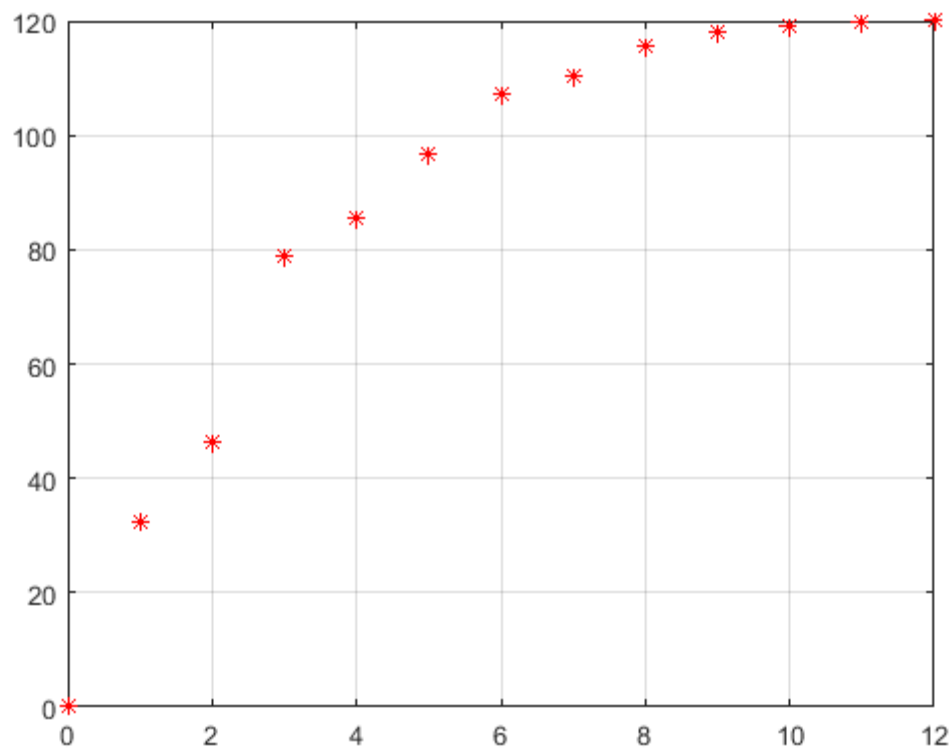
matlab可视化

图形窗口是matlab数据可视化的平台，这个窗口和命令窗口是相互独立的。在matlab命令窗口中输入 plot绘图命令时，系统就会自动建立一个图形窗口。当前窗口通常是最后一个使用的图形窗口。

figure 创建一个图形窗口，figure(n) 创建一个编号为n的窗口。figure('PropertyName',propertyvalue,...) ,指定窗口的各种属性
set(n),get(n) 设置和获取窗口的各种属性

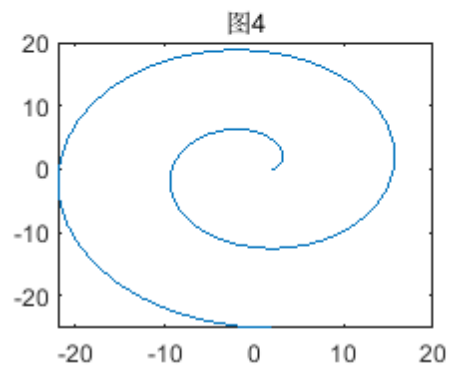
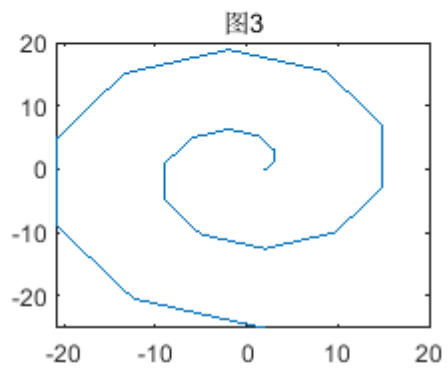
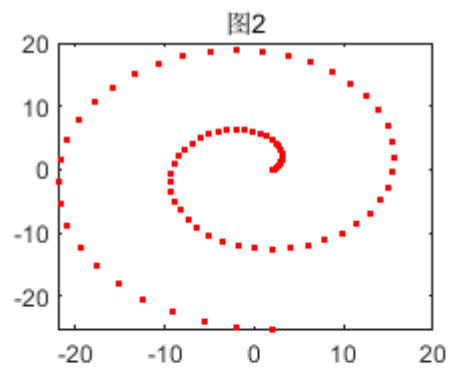
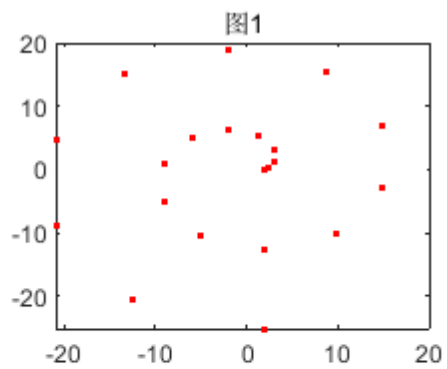
散点图

```
t=0:12;    %输入时间t的数据
T=[0 32.5 46.3 78.8 85.5 96.6 107.3 110.4 115.7 118 119.2 119.8 120]; %输入时间T 的数据
plot(t,T,'r*')    %用红色的' *' 描绘出相应的数据点
grid on          %画出坐标方格
```



多图

```
t1=0:pi/5:4*pi;
t2=0:pi/20:4*pi;
x1=2*(cos(t1)+t1.*sin(t1));
y1=2*(sin(t1)-t1.*cos(t1));
x2=2*(cos(t2)+t2.*sin(t2));
y2=2*(sin(t2)-t2.*cos(t2));
subplot(2,2,1),plot(x1,y1,'r. '),title('图1')
subplot(2,2,2),plot(x2,y2,'r. '),title('图2')
subplot(2,2,3),plot(x1,y1),title('图3')
subplot(2,2,4),plot(x2,y2),title('图4')
```

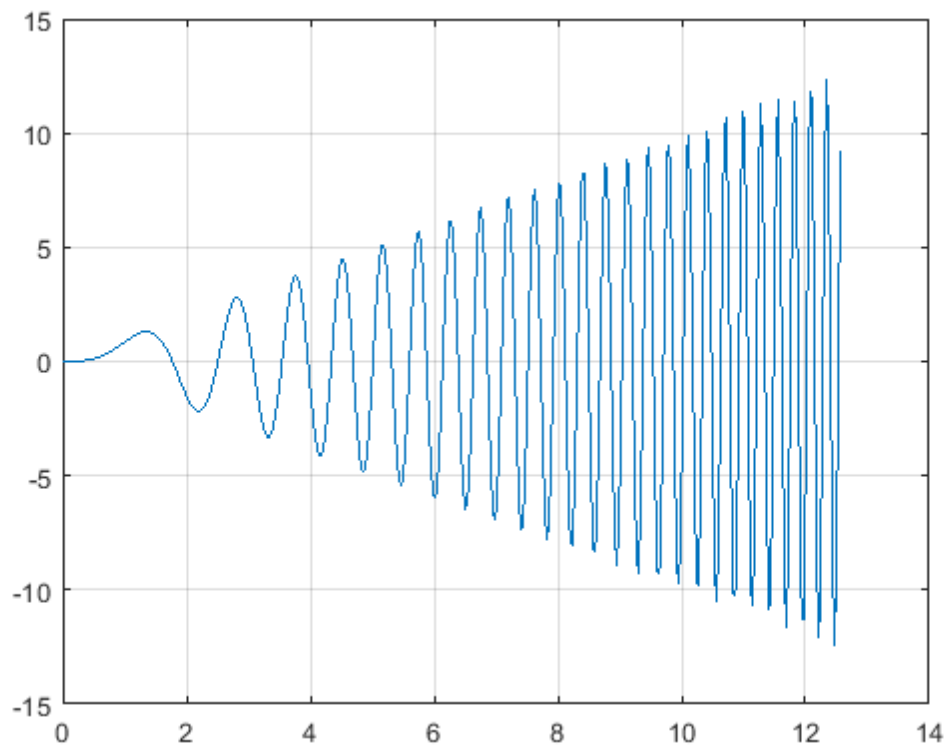


平面连续图形的绘制

画出函数 $y=x\sin(x^2)$ 在 $[0, 4\pi]$ 中的图形。

```
clear,clc,clf
x=0:pi/100:4*pi;
y=x.*sin(x.^2);%注意乘法和幂次的运算都是“点”运算
plot(x,y),grid
```

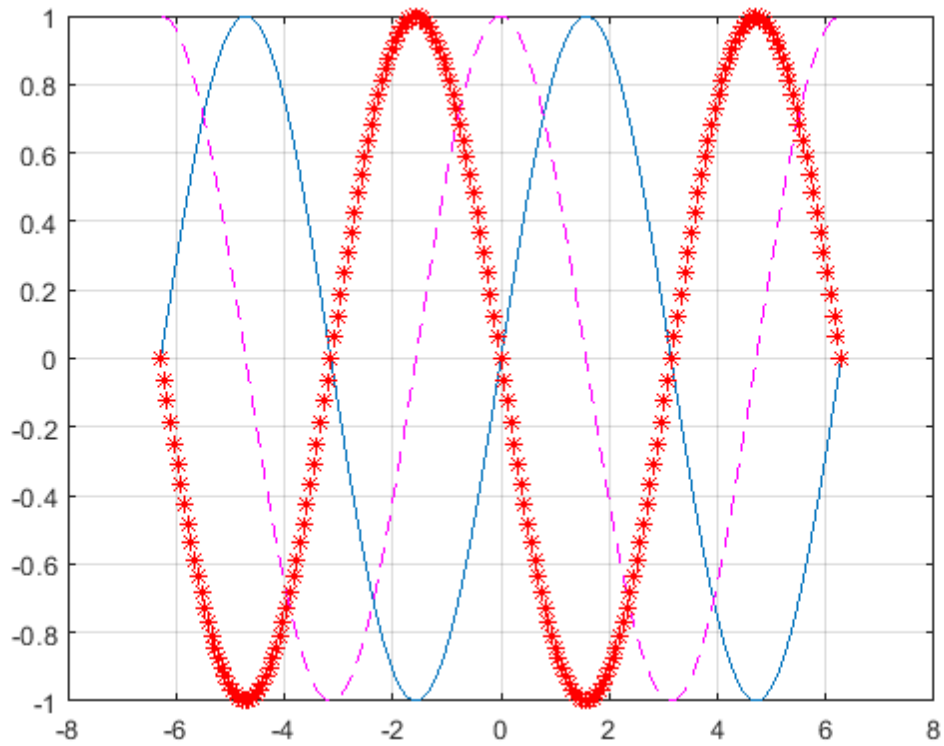
% 图形窗口中工具条的使用，保存文件的格式



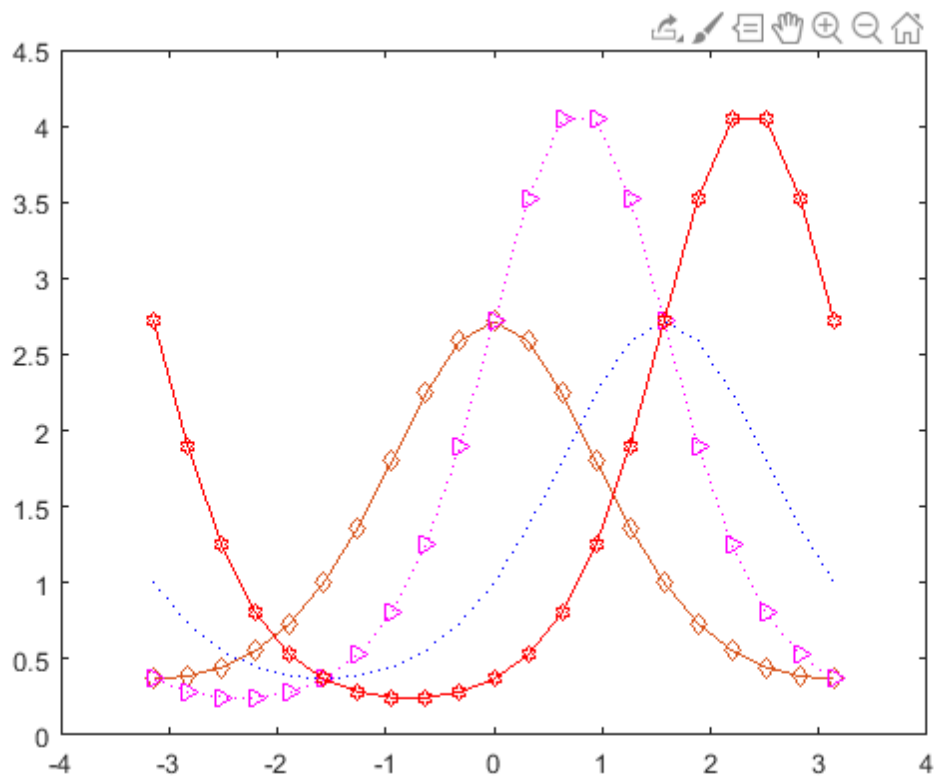
同一窗口中绘制不同的曲线

画出 $y = \cos(x)$, $y = -\sin(x)$, $x \in [-2\pi, 2\pi]$ 的图形

```
clear , clc, clf
x=-2*pi:pi/50:2*pi;
y1=sin(x);y2=cos(x);y3=-sin(x);
plot(x,y1,x,y2,'m--',x,y3,'r*'),grid on
% doc plot% r:red g:绿色, b: 蓝色, m: 紫红色, k: 黑色
```



```
clear
close all
x=-pi:pi/10:pi;
y1=exp(sin(x));
y2=exp(cos(x));
y3=exp(sin(x)+cos(x));
y4=exp(sin(x)-cos(x));
plot(x, y1, 'b:', x, y2, 'd-', x, y3, 'm>:', x, y4, 'rh-')
```

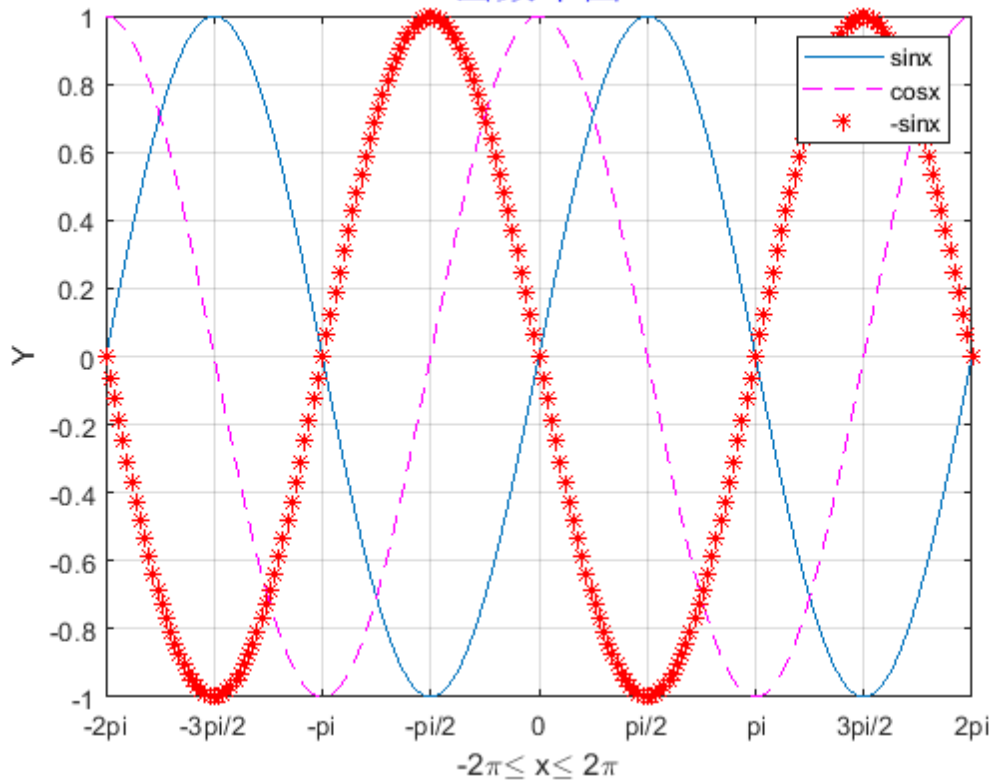


修饰

xlabel 设置坐标轴x ; title : 定义标题 ; text : 增加文本 ; legend : 设置图例 ; axis : 设置坐标轴 ; box : 设置边框

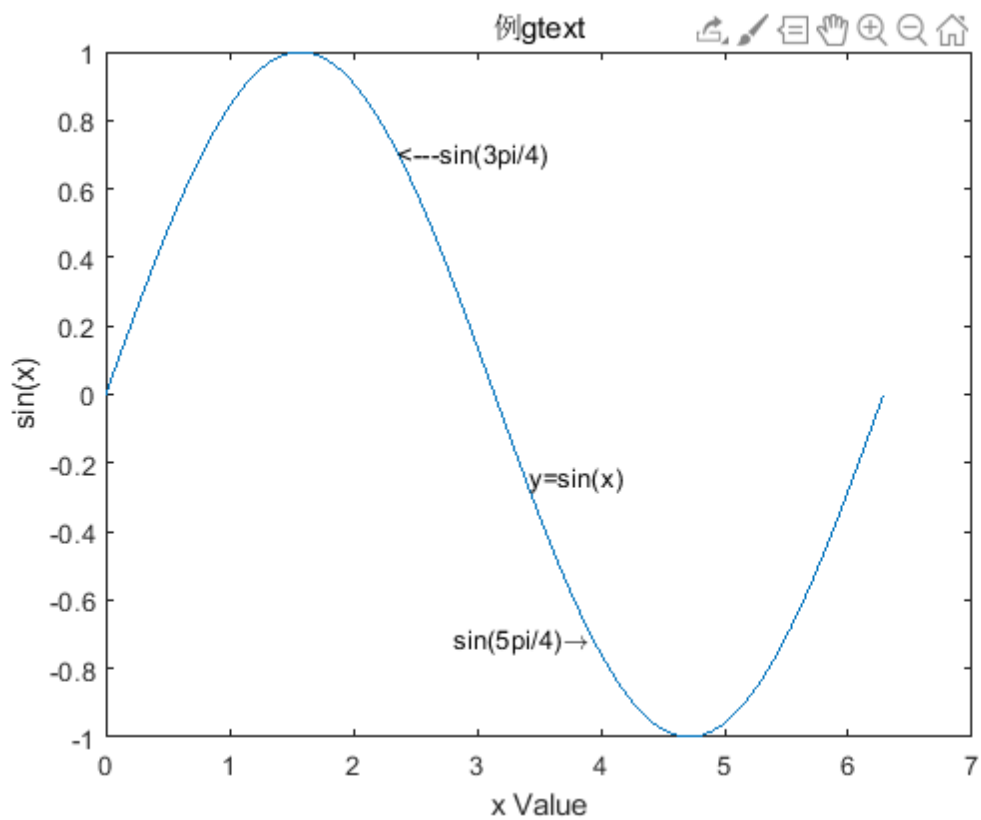
```
clear ,clc,clf
x=-2*pi:pi/50:2*pi;
y1=sin(x);y2=cos(x);y3=-sin(x);
plot(x,y1,x,y2,'m--',x,y3,'r*'),grid on
legend('sinx','cosx','-sinx')
% legend('sinx','cosx','-sinx',4) % 2014a可以
xlabel('-2\pi\leq x\leq 2\pi')
ylabel('Y')
title('函数草图','FontSize',16,'color','b')
axis([-2*pi,2*pi,-1,1])
set(gca,'XTick',-2*pi:pi/2:2*pi) %gca获取当前坐标轴的句柄
set(gca,'XTickLabel',{'-2pi','-3pi/2','-pi','-pi/2','0','pi/2','pi','3pi/2','2pi'})
```

函数草图



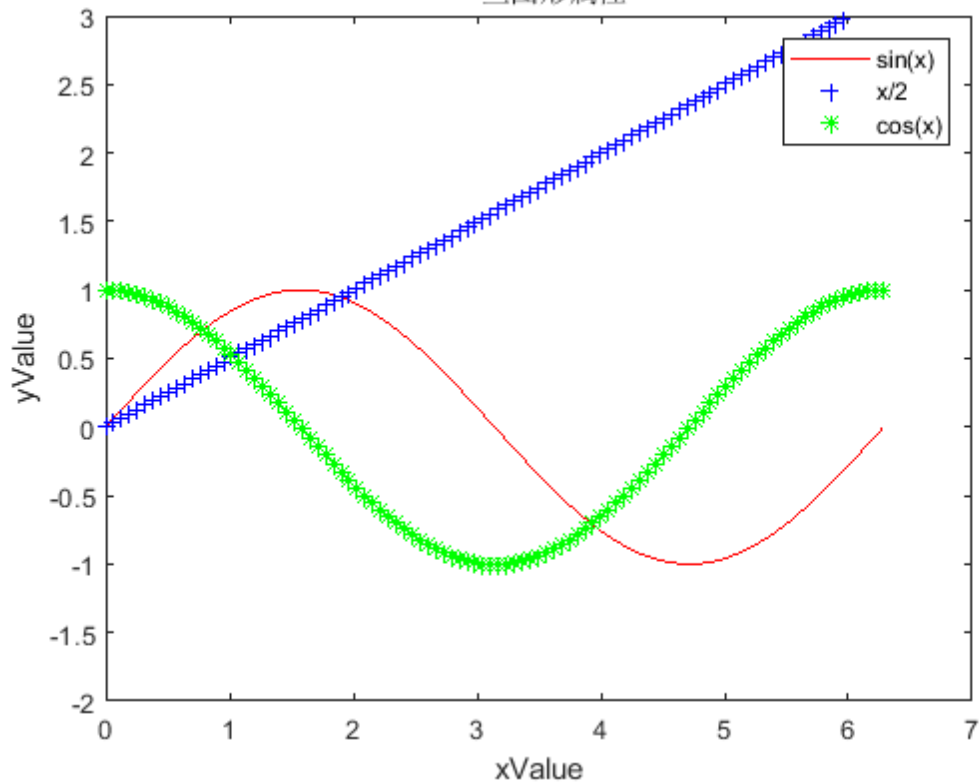
gtext，让鼠标在图形的任意位置进行标注

```
x=0:pi/50:2*pi;
plot(x, sin(x))
title(' 例gtext')
xlabel(' x Value'), ylabel(' sin(x)')
text(3*pi/4, sin(3*pi/4), '<---sin(3pi/4)')
text(5*pi/4, sin(5*pi/4), 'sin(5pi/4)\rightarrow', 'HorizontalAlignment', 'right')
gtext(' y=sin(x)')
```

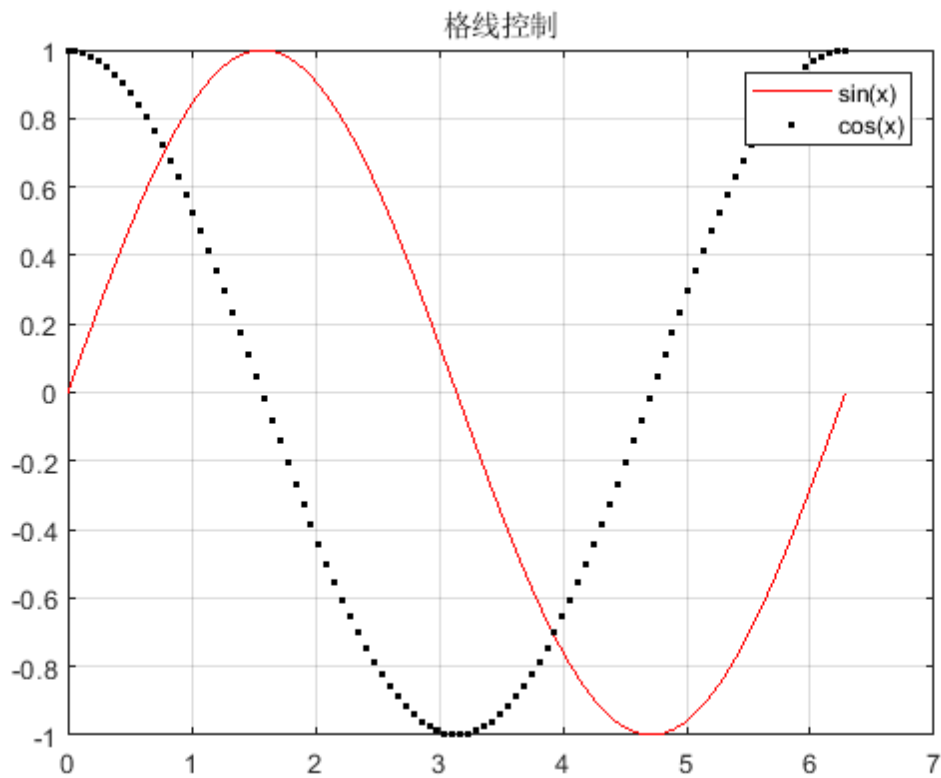


```
close all
x=linspace(0,2*pi,100);
y1=sin(x);
y2=x/2;
y3=cos(x);
plot(x,y1,'-r',x,y2,'+b',x,y3,'*g')
title('一些图形属性')
xlabel('xValue'),ylabel('yValue')
axis([0,7,-2,3])
legend('sin(x)','x/2','cos(x)')
```


一些图形属性



```
clear
close all
x=linspace(0,2*pi,100);
y1=sin(x);
y2=cos(x);
h=plot(x,y1,'-r',x,y2,'.k');
title(' 格线控制')
legend(h,'sin(x)','cos(x)')
grid on
```

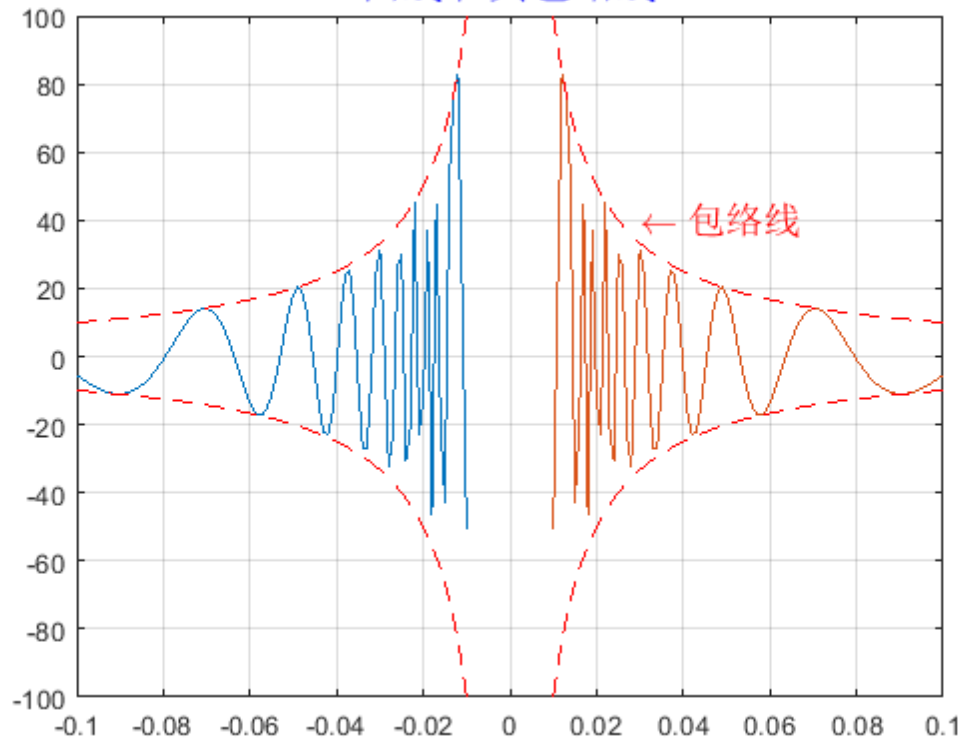


微积分课中的一个对无界和无穷大的理解

函数 $y = \frac{1}{x} \sin \frac{1}{x}$ 在点 $x = 0$ 的任何领域中无界，但不是无穷大。

```
clear ,clc,clf
x1=[-0.1:0.001:-0.01];
x2=0.01:0.001:0.1;
y1=1./x1.*sin(1./x1);
y2=1./x2.*sin(1./x2);
plot(x1,y1),
grid on,hold on % hold on使当前轴及图形保持不变
plot(x2,y2),
y3=1./x1;
plot(x1,y3,'r--',x1,-y3,'r--')
y4=1./x2;
plot(x2,y4,'r--',x2,-y4,'r--')
title('曲线和其包络线','FontSize',16,'color','b')
text(0.03,40,'\leftarrow 包络线','color','r','FontSize',14)
```

曲线和其包络线

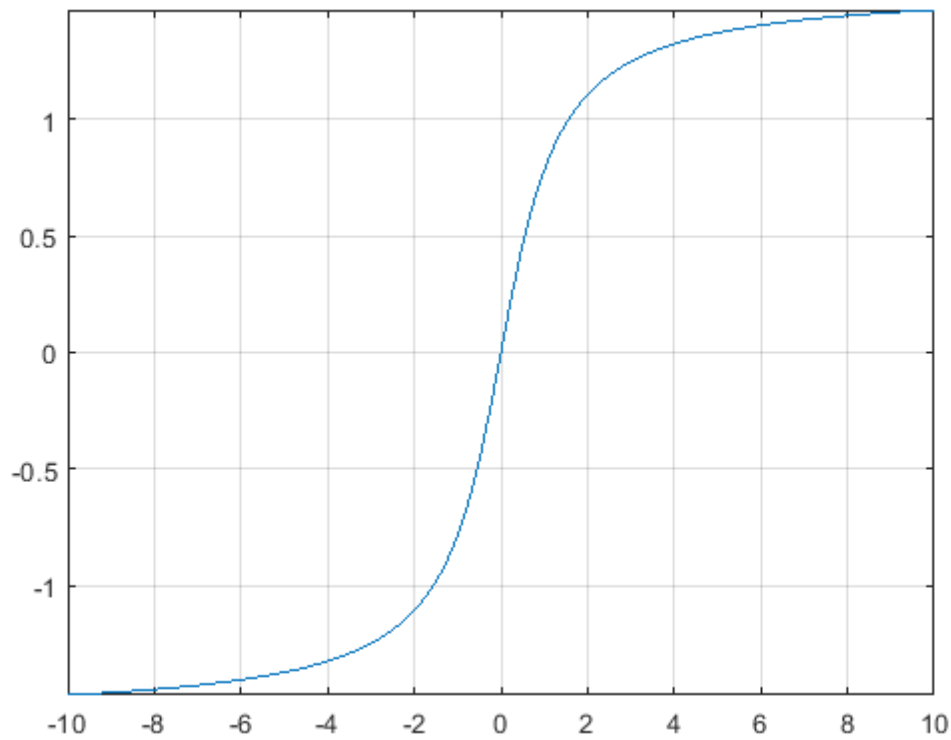


fplot绘制平面曲线

实际中，有时候并不清楚函数的具体情况，因此依据所选取的数据点作的图像可能会忽略真实函数的某些重要特征，内部有自适应算法，做出的图像比用plot命令做出的更光滑。

```
clear, clc, clf
fplot('atan(x)', [-10, 10]), grid
```

警告：在以后的版本中，fplot 将不接受字符向量或字符串输入。请改用 `fplot(@(x)atan(x))`。



```
close all
% f_compare=@(x) sin(1./x);
x=linspace(0.01,0.02,50);
y=f_compare(x);
subplot(2,1,1),plot(x,y)
subplot(2,1,2),fplot('f_compare',[0.01,0.02])

[X,Y]=fplot('f_compare',[0.01,0.02]);
[n,m]=size(X)
% plot中作图的修饰也可以稍微简单一点。
%
% hh = fplot(ff);
% hh.LineWidth = 2; %这里要注意字母大小写。
% hh.LineStyle = '--';
```

警告：在以后的版本中，fplot 将不接受字符向量或字符串输入。请改用 fplot(@f_compare)。

警告：在以后的版本中，fplot 将不接受字符向量或字符串输入。请改用 fplot(@f_compare)。

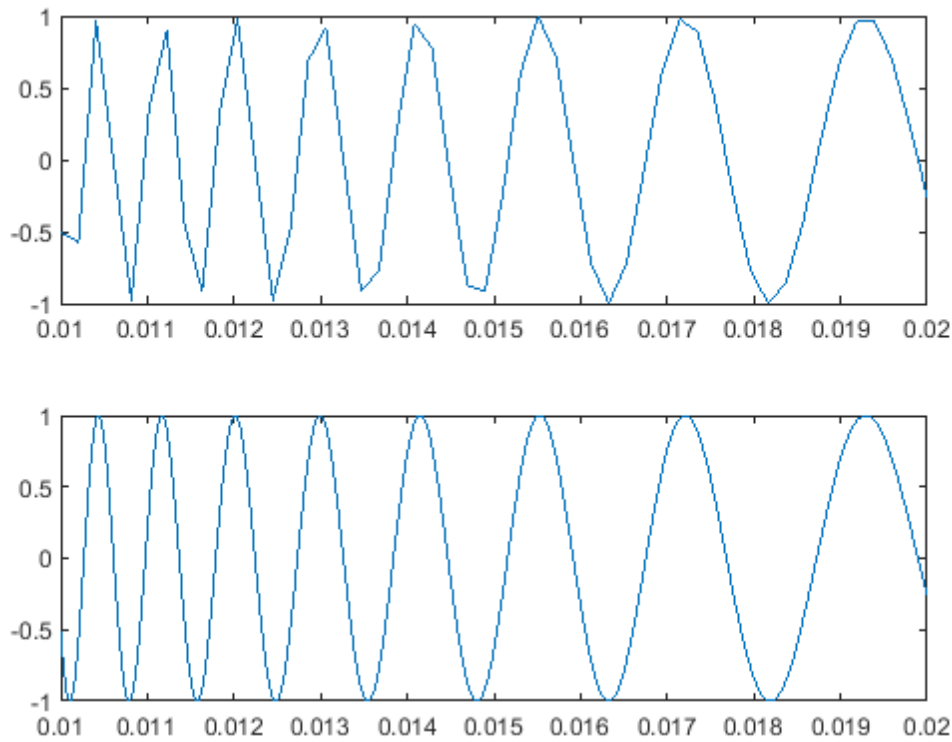
警告：以后的版本中将会删除 fplot 带有两个输出参数的情况。请改用 XData 和 YData 属性。

n =

1

m =

21852



```
clear
close all
% f1=@(x) sin(x)/(x^2-x+0.5)+cos(x)/(x^2+2*x-0.5);
% f2=@(x) log(sin(x)^2+2*sin(x)+8);
% f3=@(x,a,b) exp(a*sin(x)-b*cos(x));
subplot(2,2,1), fplot('f1',[0,1])% 在[0,1]范围
subplot(2,2,2), fplot('f2',[-2*pi,2*pi],1e-6)%相对误差e=1*10^-6
% subplot(2,2,3), fplot('f3',[-4*pi,4*pi],[],[],[],4,2)%默认的值用空矩阵,%高本版有问题
% subplot(2,2,4), fplot(' [sin(x),x,tan(x)]',[0,pi/2,0,2]) %x,y的范围
```

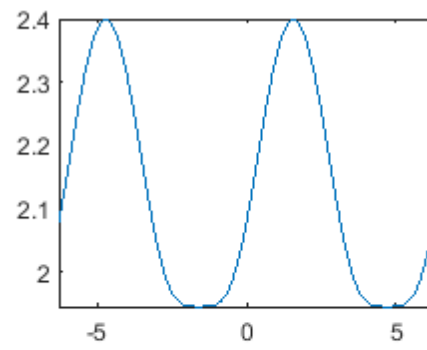
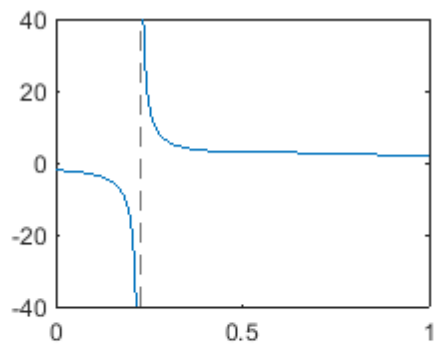
警告：在以后的版本中，fplot 将不接受字符向量或字符串输入。请改用 fplot(@f1)。

警告：函数处理数组输入时行为异常。要改善性能，请将您的函数正确向量化，以返回大小和形状与输入参数相同的输出。

警告：fplot 的容差参数输入已删除。

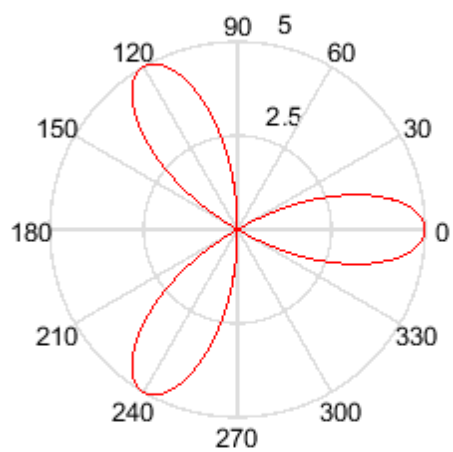
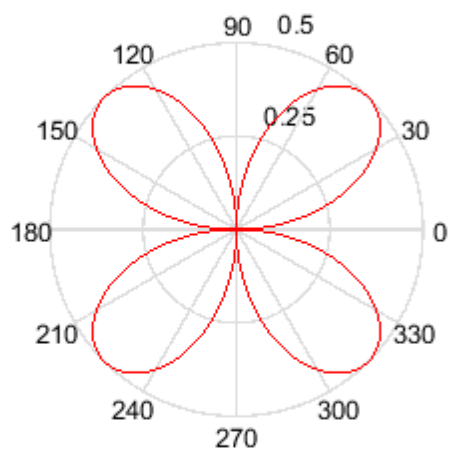
警告：在以后的版本中，fplot 将不接受字符向量或字符串输入。请改用 fplot(@f2)。

警告：函数处理数组输入时行为异常。要改善性能，请将您的函数正确向量化，以返回大小和形状与输入参数相同的输出。

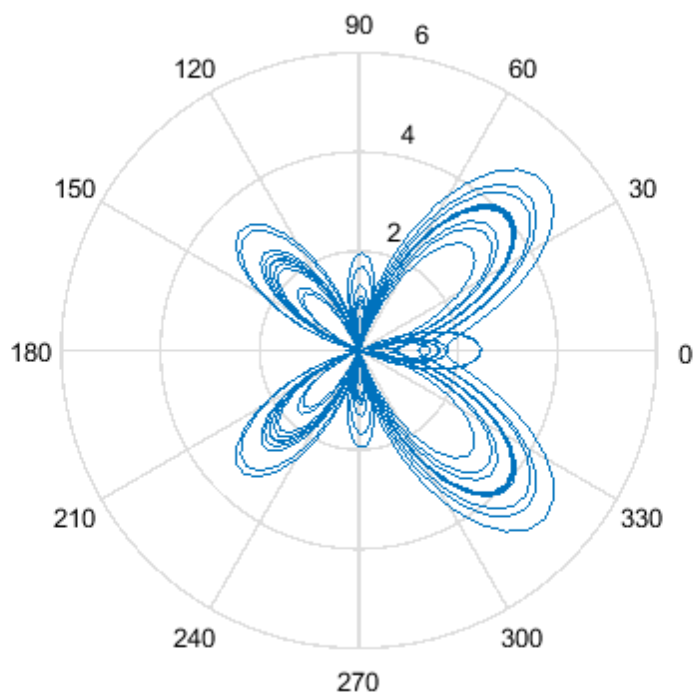


polar作极坐标曲线图形

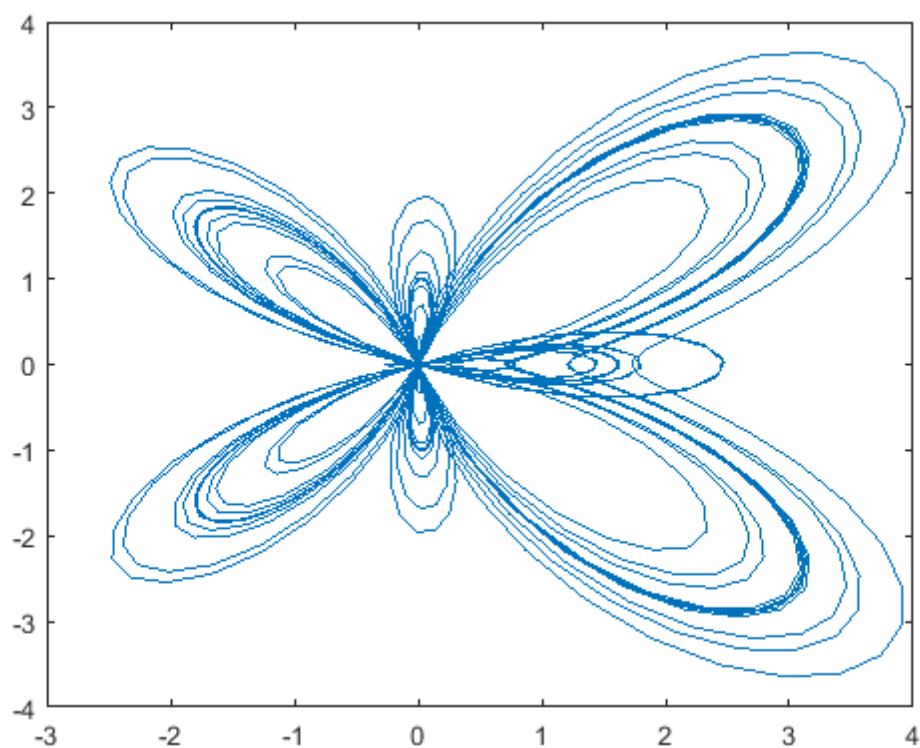
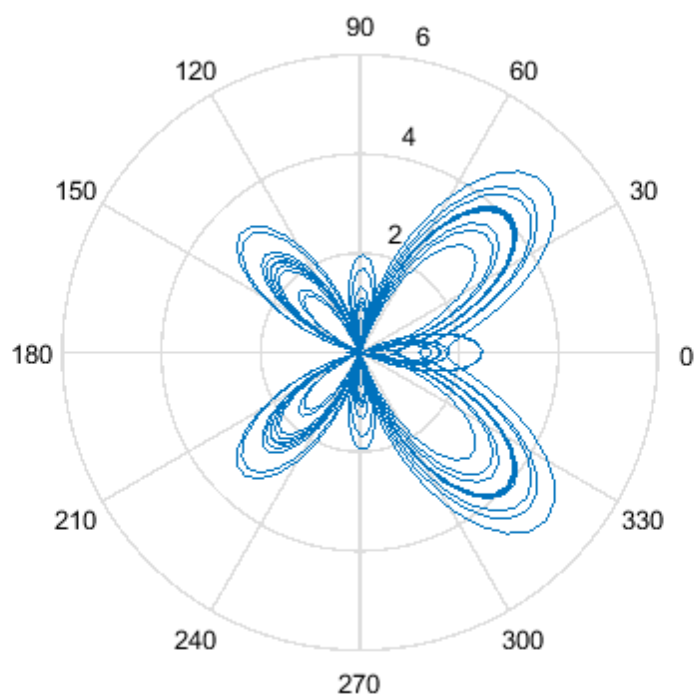
```
clear, clc, clf
theta=0:.01:2*pi;
rho1=sin(2*theta)/2;%四叶玫瑰线方程
subplot(1,2,1);
polar(theta, rho1, 'r')
rho2=5*cos(3*theta);
subplot(1,2,2);
polar(theta, rho2, 'r')
```



```
close all
t=linspace(0,24*pi,1000);
r=exp(cos(t))-2*cos(4.*t)+(sin(t./12)).^5;
polar(t,r)
```



```
[x,y]=pol2cart(t,r); %直角坐标系下图形
figure
plot(x,y)
```



ezplot函数作图

用ezplot作出 $y = x^2$, 心形线, 隐函数 $e^x + \sin(xy) = 0$ 和隐函数 $x^3 + y^3 - 3xy = 0$ 的图形。将一个符号函数图形化

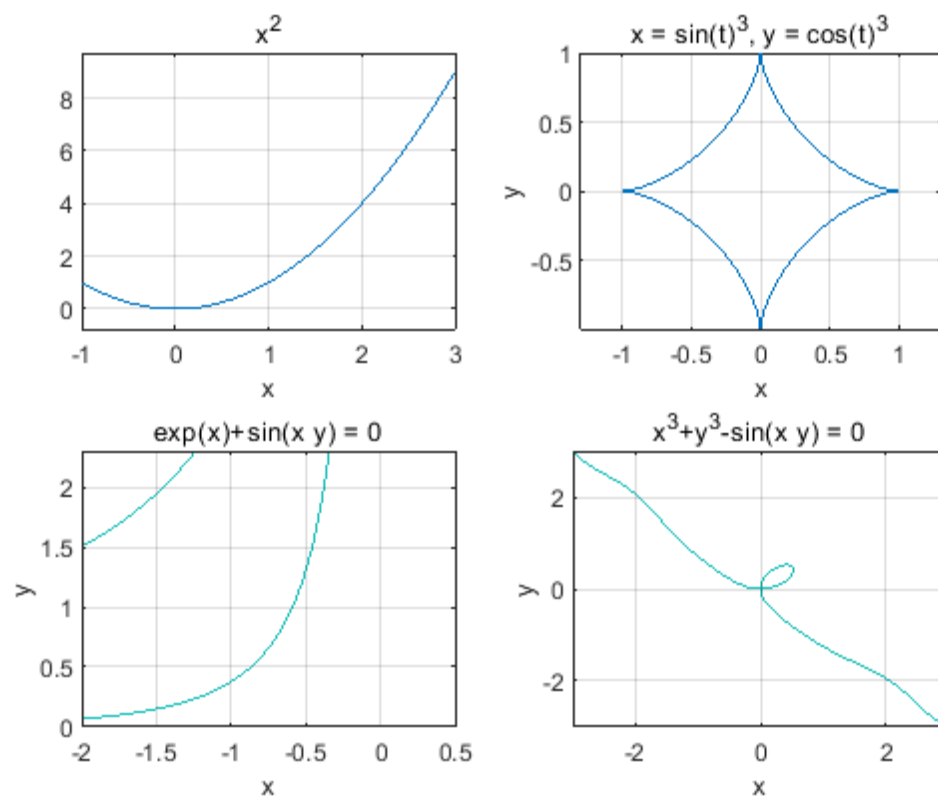
```
clear, clc, clf
subplot(2, 2, 1),
ezplot('x^2', [-1, 3]), grid
```



```

subplot(2,2,2),
ezplot('sin(t)^3','cos(t)^3',[0,2*pi]),grid
subplot(2,2,3),
ezplot('exp(x)+sin(x*y)',[-2,0.5,0,2.3]),grid
subplot(2,2,4),
ezplot('x^3+y^3-sin(x*y)',[-3,3]),grid

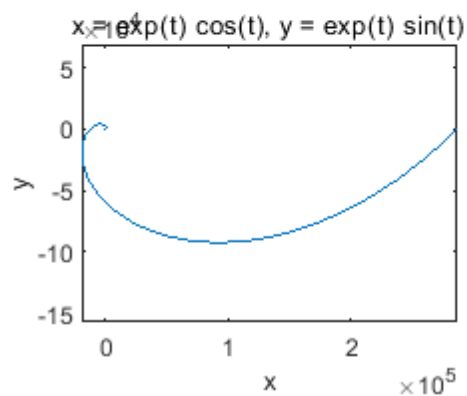
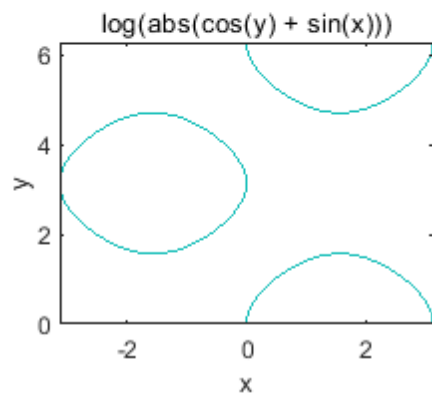
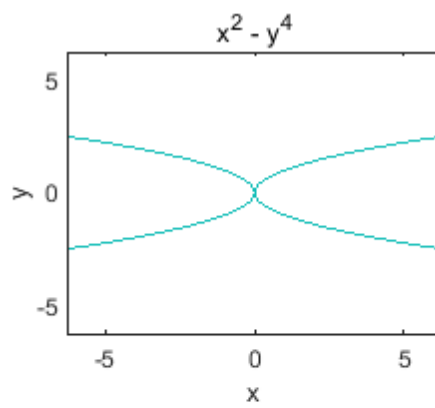
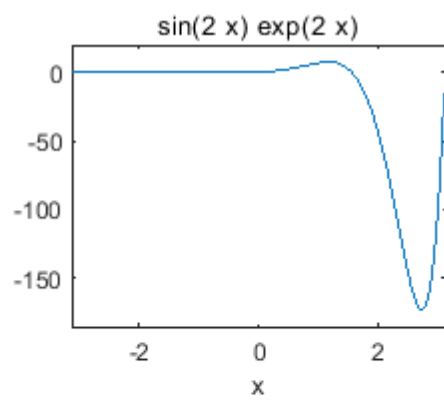
```



```

clear
syms x y t
f1=exp(2*x)*sin(2*x); %符号函数
f2=x^2-y^4; %隐函数
f3=log(abs(sin(x)+cos(y))); %隐函数
X=exp(t)*cos(t);Y=exp(t)*sin(t); %参数方程
subplot(2,2,1),ezplot(f1,[-pi,pi])
subplot(2,2,2),ezplot(f2)
subplot(2,2,3),ezplot(f3,[-pi,pi,0,2*pi])
subplot(2,2,4),ezplot(X,Y,[-4*pi,4*pi])

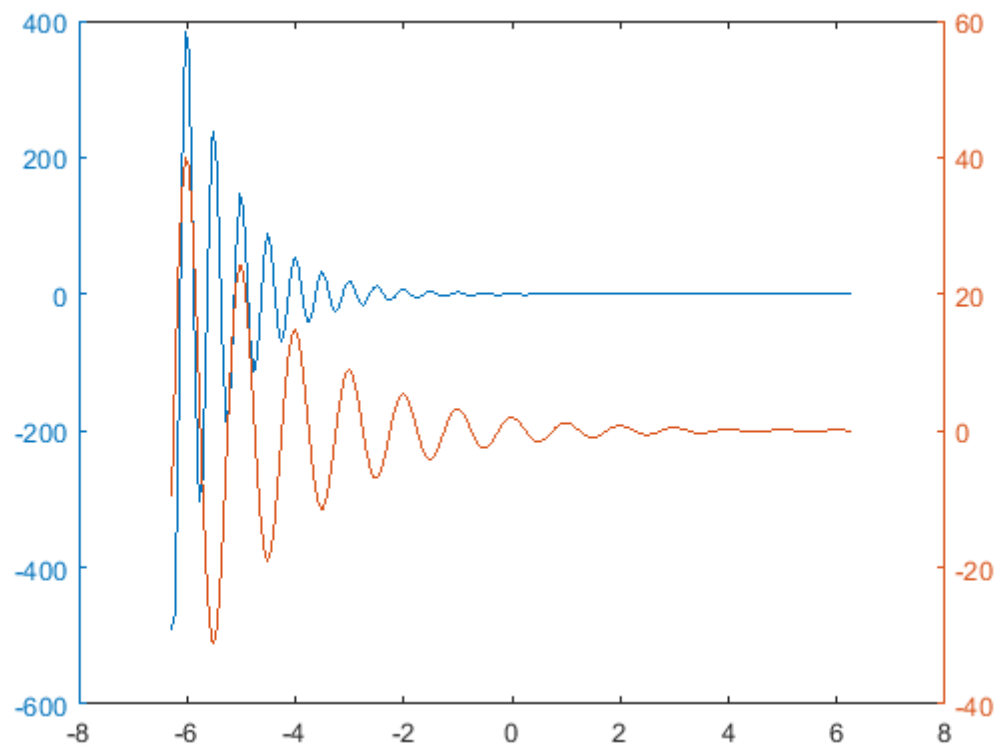
```



双y轴坐标

比较两个函数图像

```
close all
x=linspace(-2*pi,2*pi,200);
y1=exp(-x).*cos(4*pi*x);
y2=2*exp(-0.5*x).*cos(2*pi*x);
plotyy(x,y1,x,y2,'plot')
```



空间图形的描绘

用plot3描绘空间图形

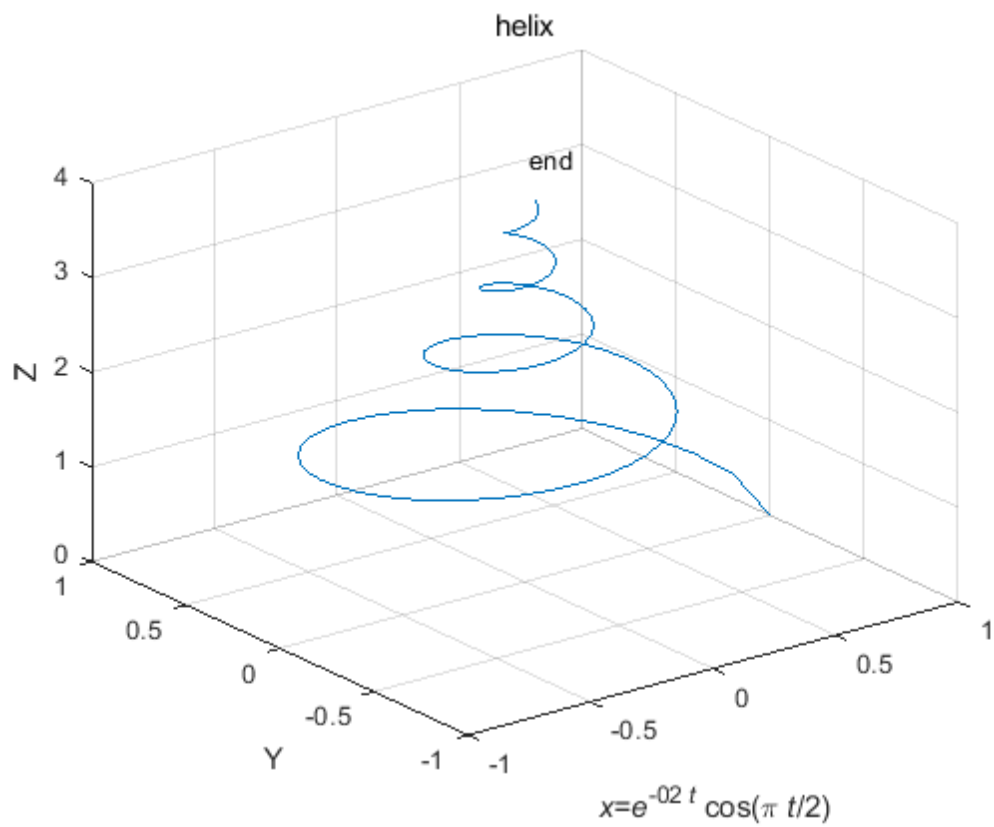
格式plot3(x,y,z,options) 作出由参数方程

表达式

$$\begin{cases} x = e^{-0.2t} \cos \frac{\pi t}{2}, \\ z = \sqrt{t}, \\ y = e^{-0.2t} \sin \frac{\pi t}{2} \end{cases} \quad 0 \leq t \leq 20$$

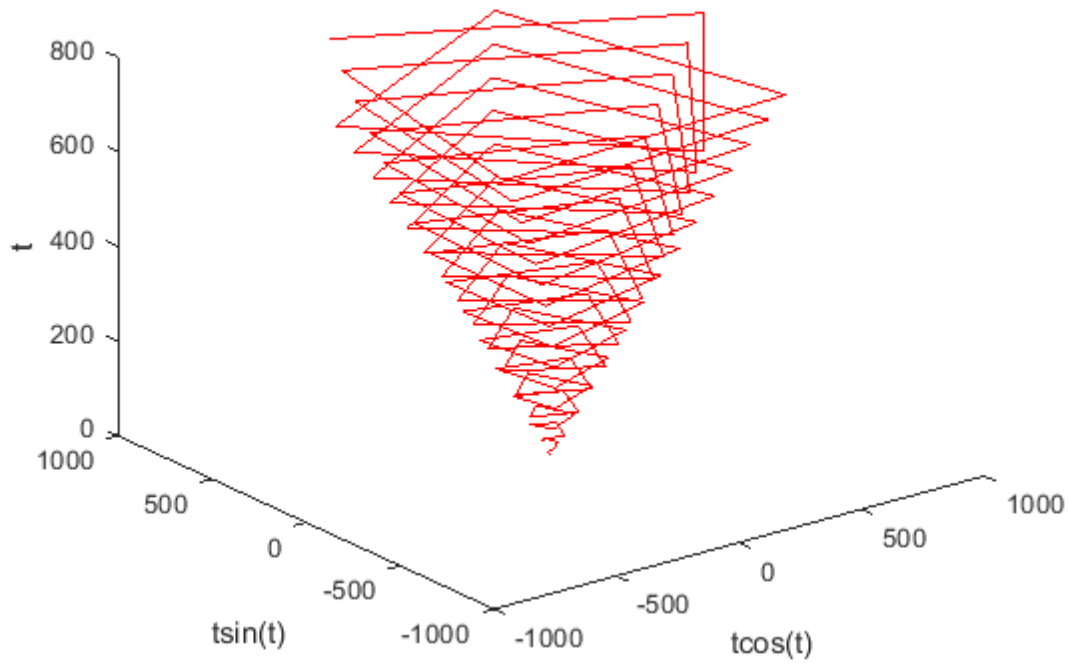
所确定的函数对应的曲线图形。

```
clear,clc,clf
t=0:0.1:20;r=exp(-0.2*t);th=0.5*pi*t;
x=r.*cos(th);y=r.*sin(th);z=sqrt(t);
plot3(x,y,z)
title('helix');text(x(end),y(end),z(end),'end')
xlabel('\it x=e^{-0.2\it t}\rm cos(\pi\it t\rm/2)')
ylabel('Y');
zlabel('Z');
axis([-1,1,-1,1,0,4])
grid on
```



```
close all
clear all
t=linspace( 0.2*pi,800);
x=t.*cos(t);
y=t.*sin(t);
z=t;
plot3(x,y,z,'r')
title('圆锥螺旋线')
xlabel('tcos(t)'),ylabel('tsin(t)'),zlabel('t')
```

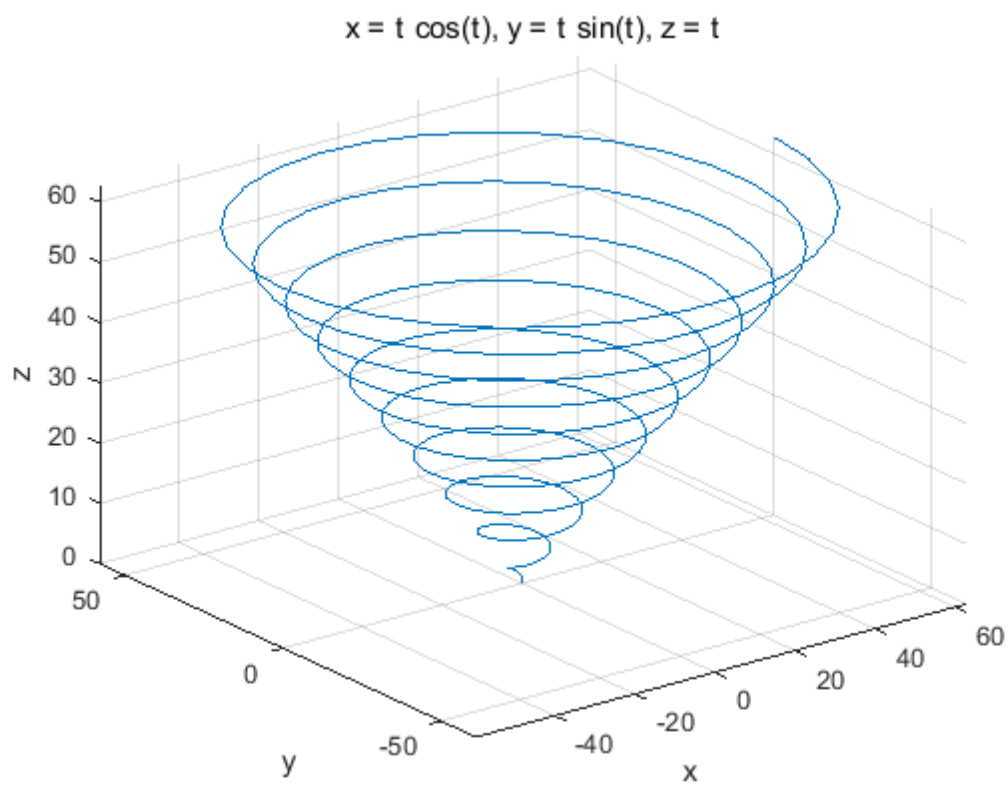
圆锥螺线



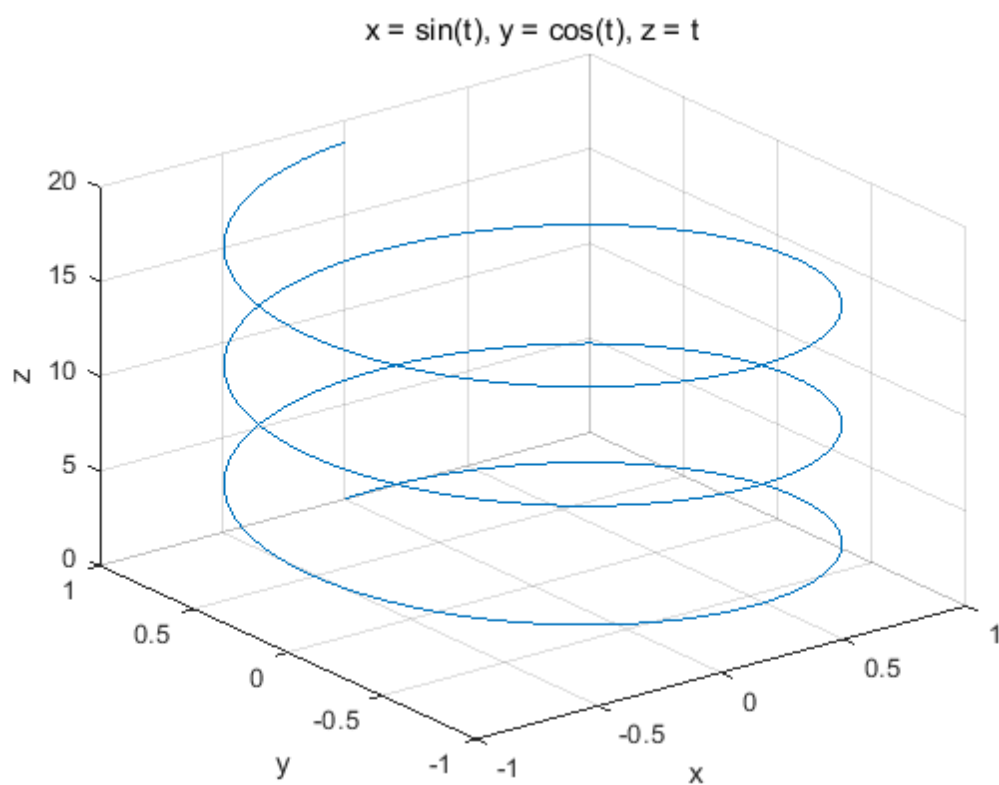
ezplot3函数

```
clear all
syms t
x=t*cos(t);
y=t*sin(t);
z=t;
ezplot3(x,y,z,[0,20*pi]) % 画出的图有问题，不够精细

ezplot3('t*cos(t)','t*sin(t)','t',[0,20*pi]) % 应该与上面一样的图
```



```
ezplot3('sin(t)', 'cos(t)', 't', [0, 6*pi])
```

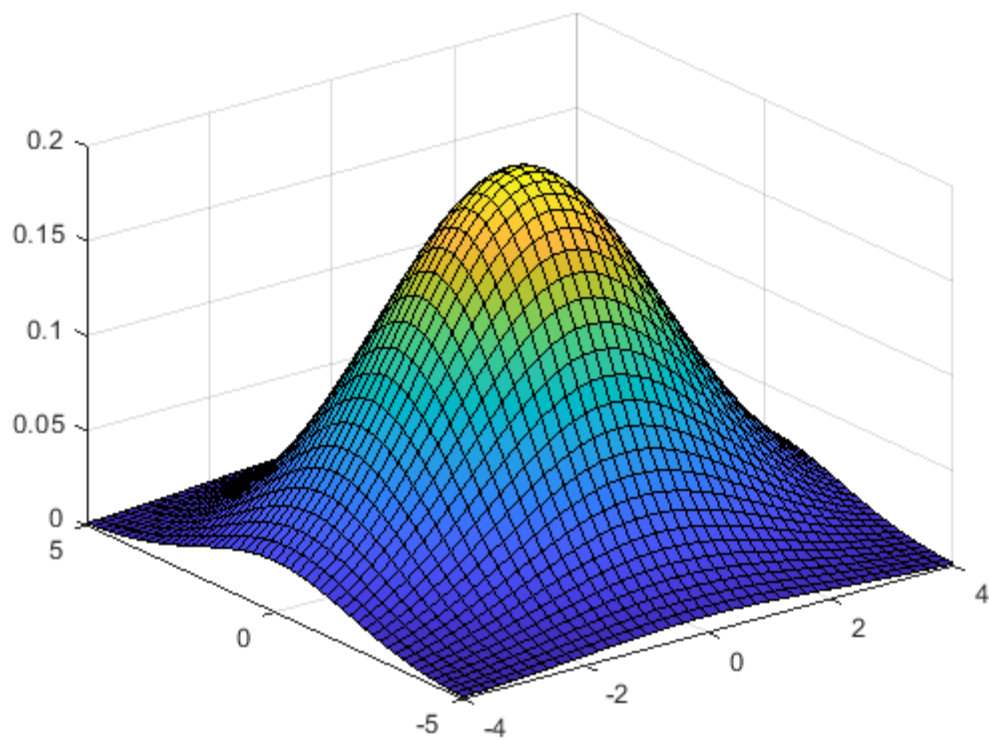
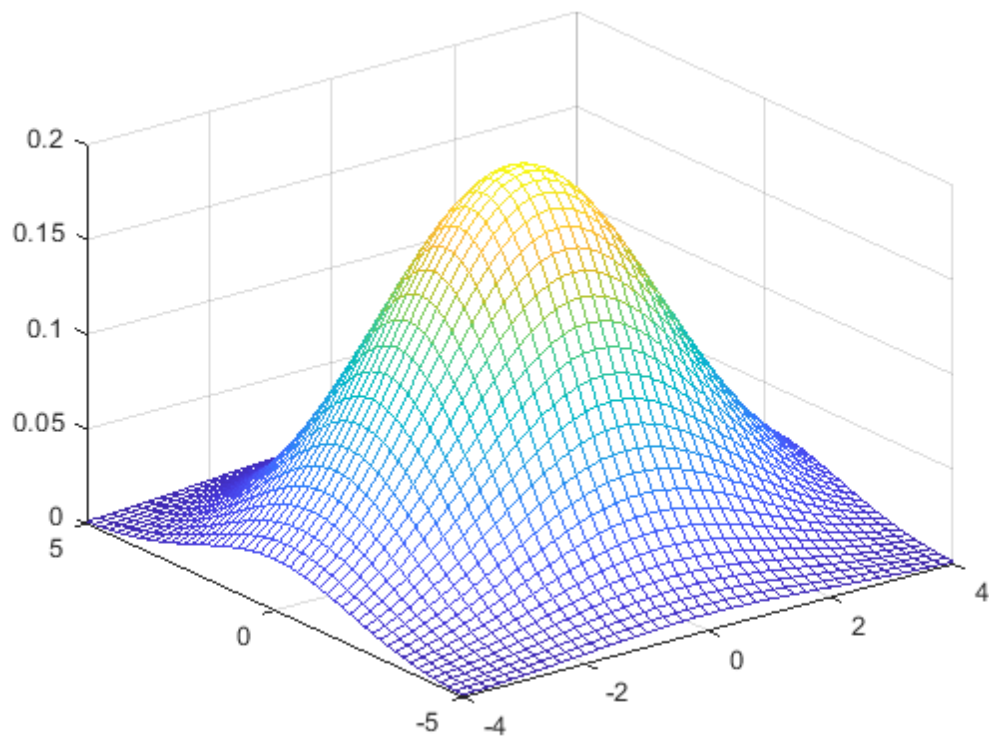


用mesh函数--三维网格命令

mesh(x,y,z,options) $z = \frac{1}{2\sqrt{2\pi}} e^{-\frac{x^2+y^2}{8}}$ \$ 在区间[-4,4],[-5,5]上的图形

```
clear ,clc
x=1:4;y=5:10;
[X,Y]=meshgrid(x,y);%产生网格点
disp(X),disp(Y)%交叉点形成平面坐标
clear ,clc,clf
x=-4:.2:4;
y=-5:.2:5;
[X,Y]=meshgrid(x,y);
Z=exp(-(X.^2+Y.^2)/8)/(2*sqrt(2*pi)); %二维正态分布的密度函数图像
mesh(X,Y,Z)
figure(2)
surf(X,Y,Z)
% surf 以网格线和补片填充色彩作图
% meshc 网格作图方式，带等高线
% surfc 以网格线和补片填充色彩作图，带等高线
```

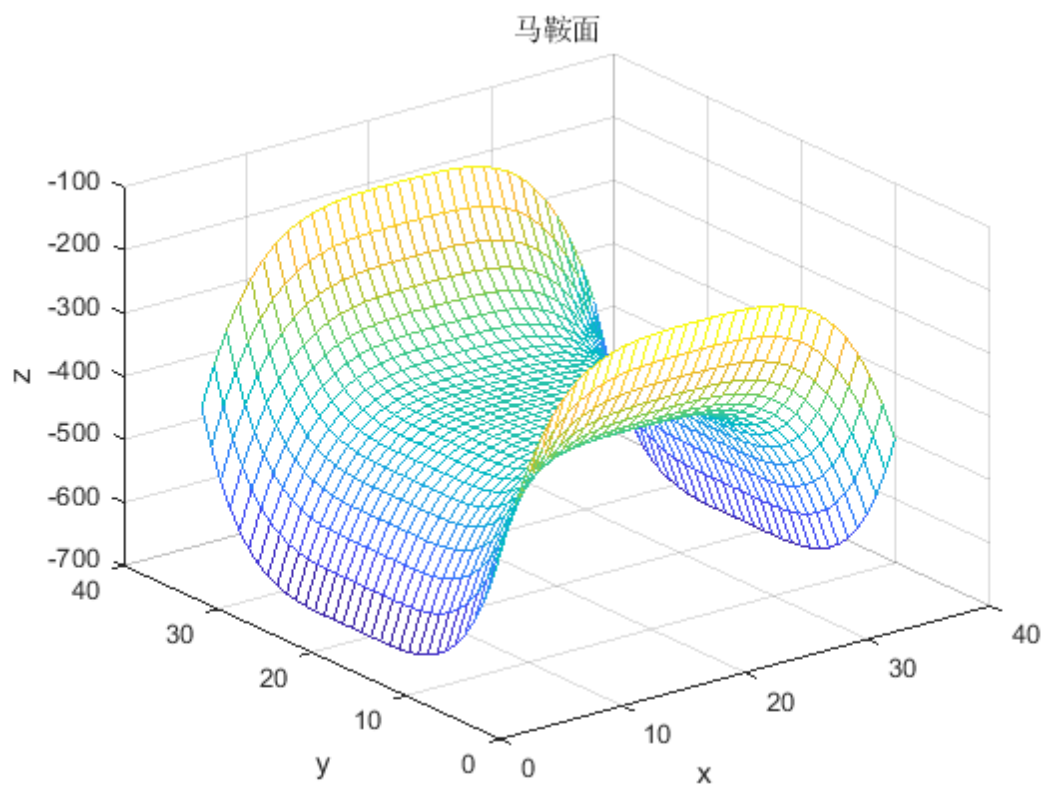
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9
10	10	10	10



```

close all
x=-4:0.25:4;
y=x;
[X,Y]=meshgrid(x,y);
Z=-X.^4+Y.^4-X.^2-Y.^2-2*X*Y;
mesh(Z)
title('马鞍面')
xlabel('x'),ylabel('y'),zlabel('z')

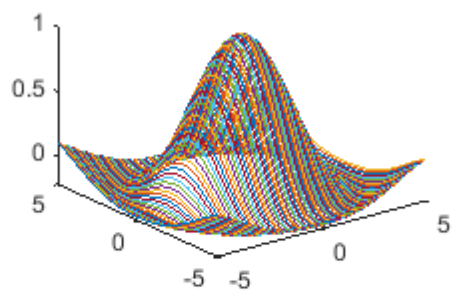
```

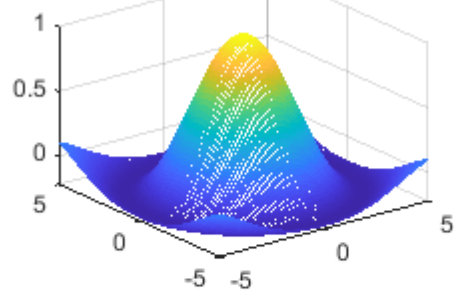
比较

```
close all
x=-5:0.1:5;
[X,Y]=meshgrid(x);
Z=sin(sqrt(X.^2+Y.^2))./sqrt(X.^2+Y.^2);
subplot(2,2,1)
plot3(X,Y,Z)
title('plot3作图')
subplot(2,2,2)
mesh(X,Y,Z)
title('mesh作图')
subplot(2,2,3)
meshc(X,Y,Z)
title('meshc作图')
subplot(2,2,4)
meshz(X,Y,Z)
title('meshz作图')
```

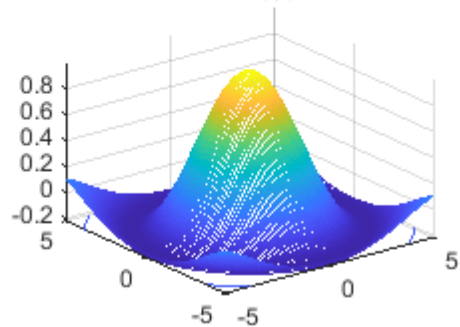
plot3作图



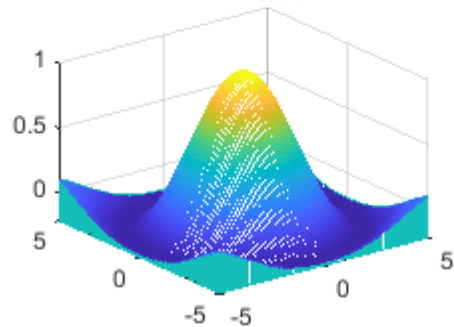
mesh作图



meshc作图



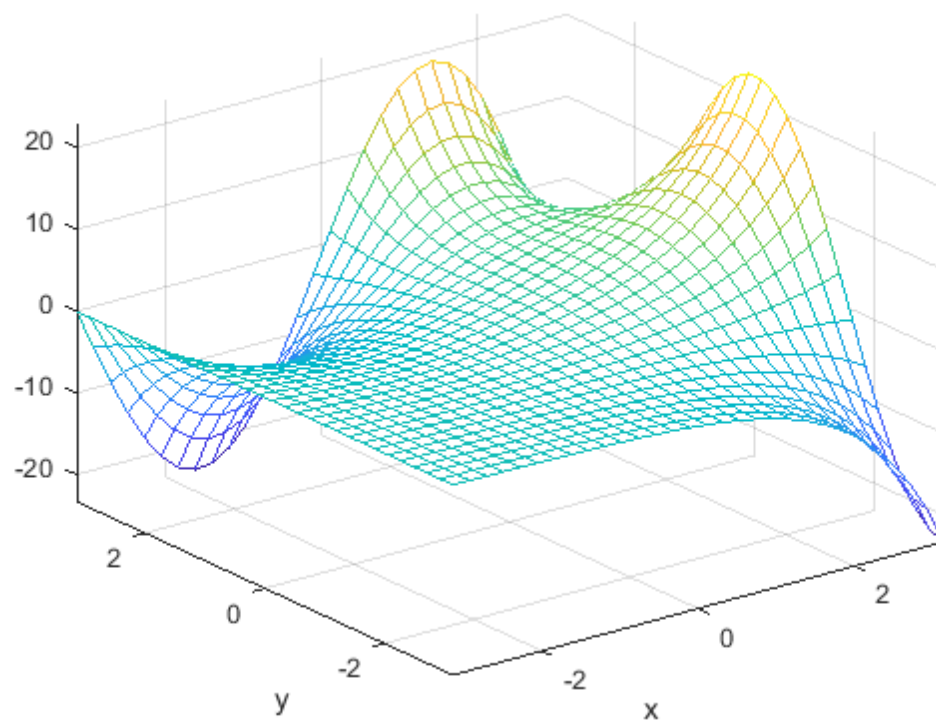
meshz作图



带网格线的三维表面图

```
close all
syms x y
f=sin(x)*exp(y)+cos(y)*exp(x);
ezmesh(f, [-pi, pi], 30)
title('带网格线的三维表面图')
```

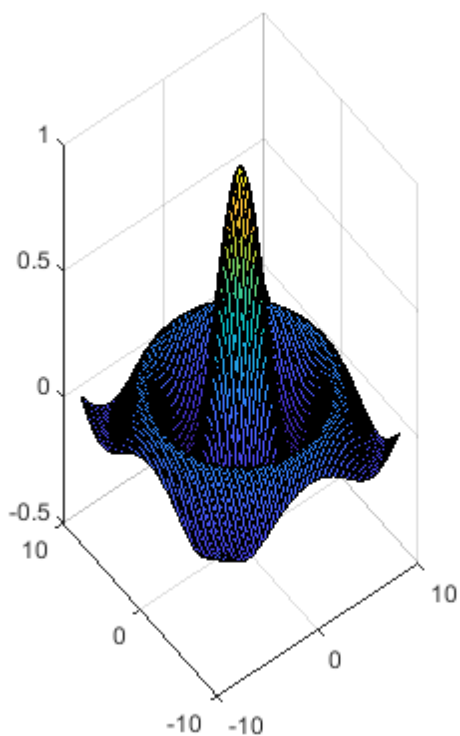
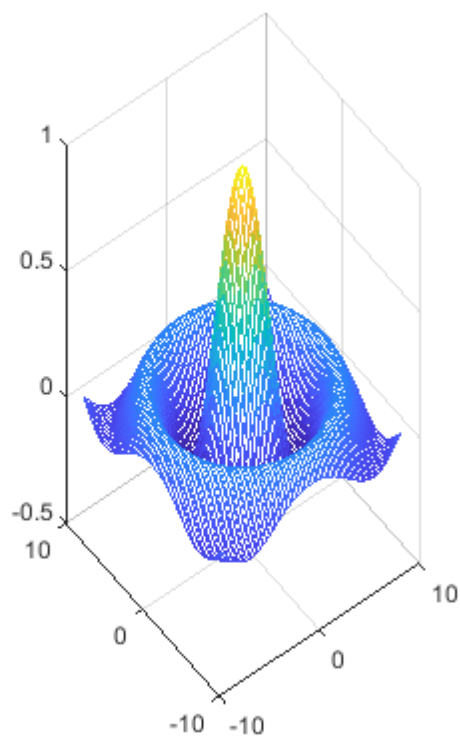
带网格线的三维表面图



极坐标的方式效果更好

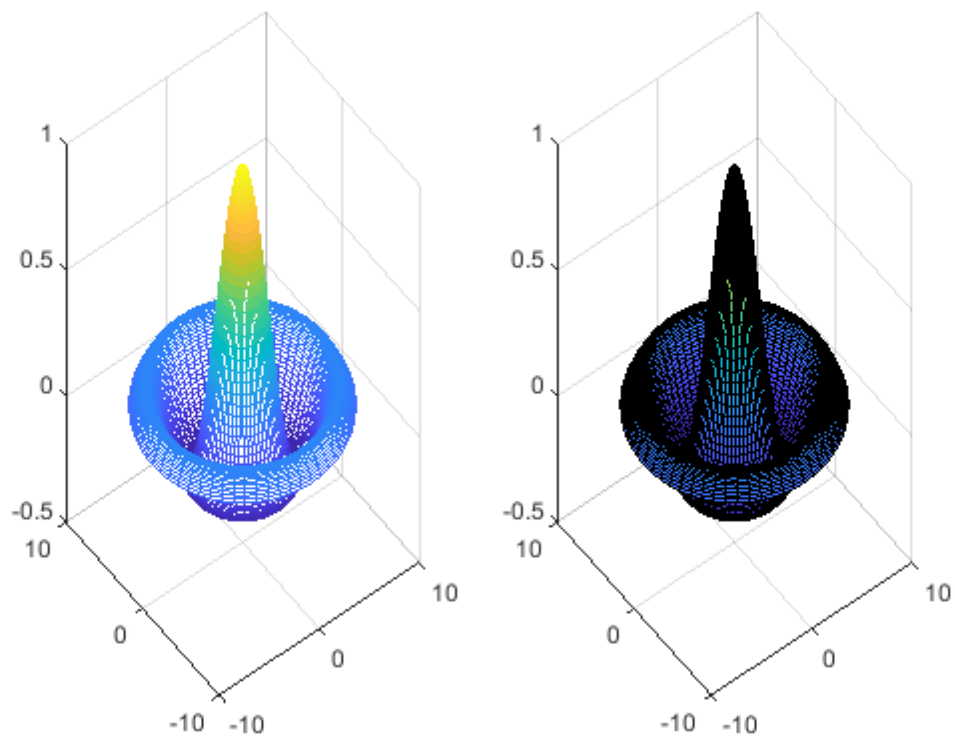
曲面 $z = \frac{\sin \sqrt{x^2+y^2}}{\sqrt{x^2+y^2}}$ 在区域 $[-9,9]*[-9,9]$ 上的图形。

```
clear, clc, clf
x=-9:.3:9;
[y, x]=meshgrid(x);
r=sqrt(x.^2+y.^2)+eps;%避免分母为零
z=sin(r)./r;
subplot(1, 2, 1), mesh(x, y, z)
subplot(1, 2, 2), surf(x, y, z)
```



%效果不好，因为曲面为选择曲面，可考虑用极坐标方式。

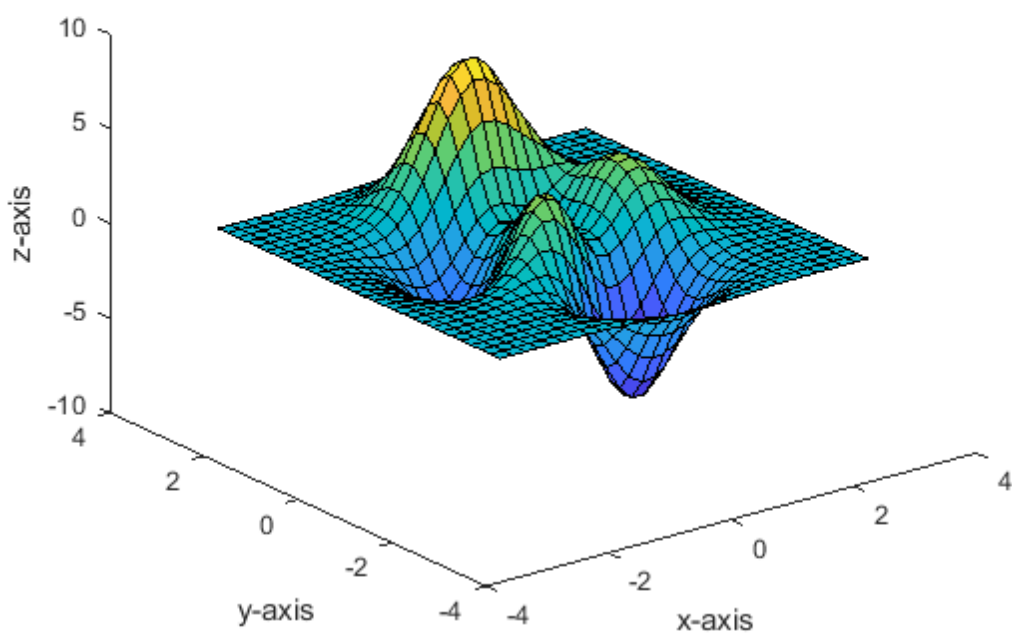
```
clear, clc
theta=0:pi/30:2*pi;
rho=0:.1:9;
[T, R]=meshgrid(theta, rho);
x=R.*cos(T); y=R.*sin(T);
r=sqrt(x.^2+y.^2)+eps;
z=sin(r)./r;
subplot(1, 2, 1), mesh(x, y, z)
subplot(1, 2, 2), surf(x, y, z)
```



三维曲面

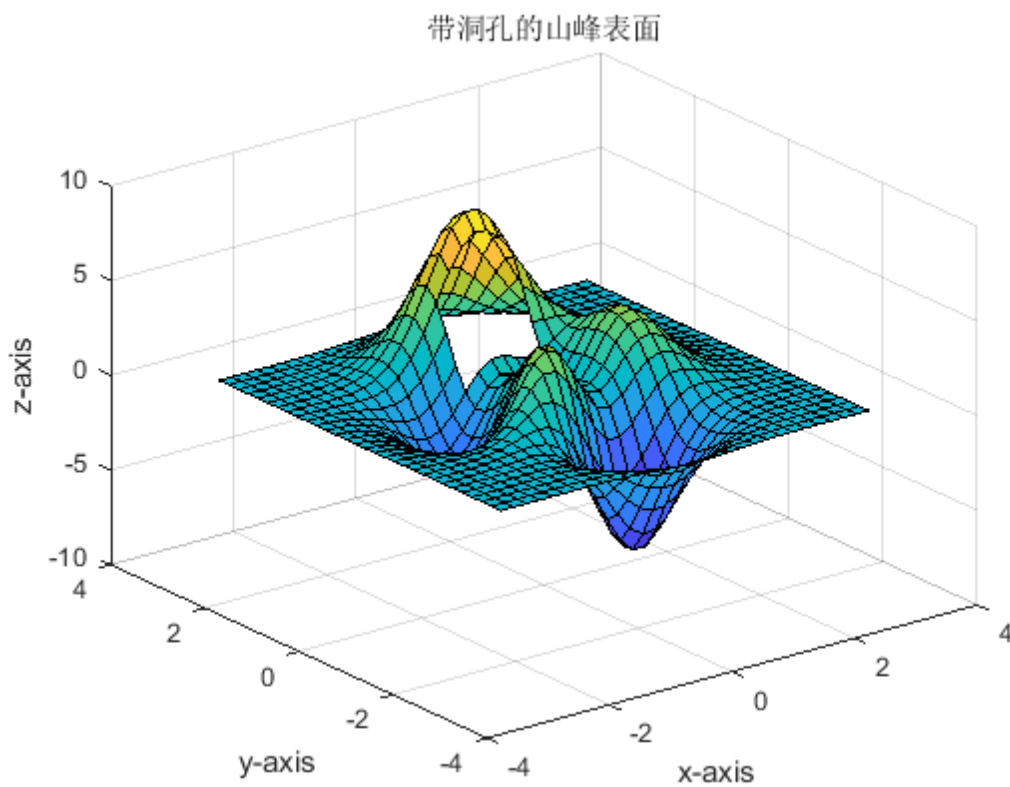
```
close all
[X,Y,Z]=peaks(30);
surf(X,Y,Z)
title(' 山峰表面')
xlabel('x-axis'),ylabel('y-axis '),zlabel('z-axis')
grid
```

山峰表面



带空洞的三维曲面

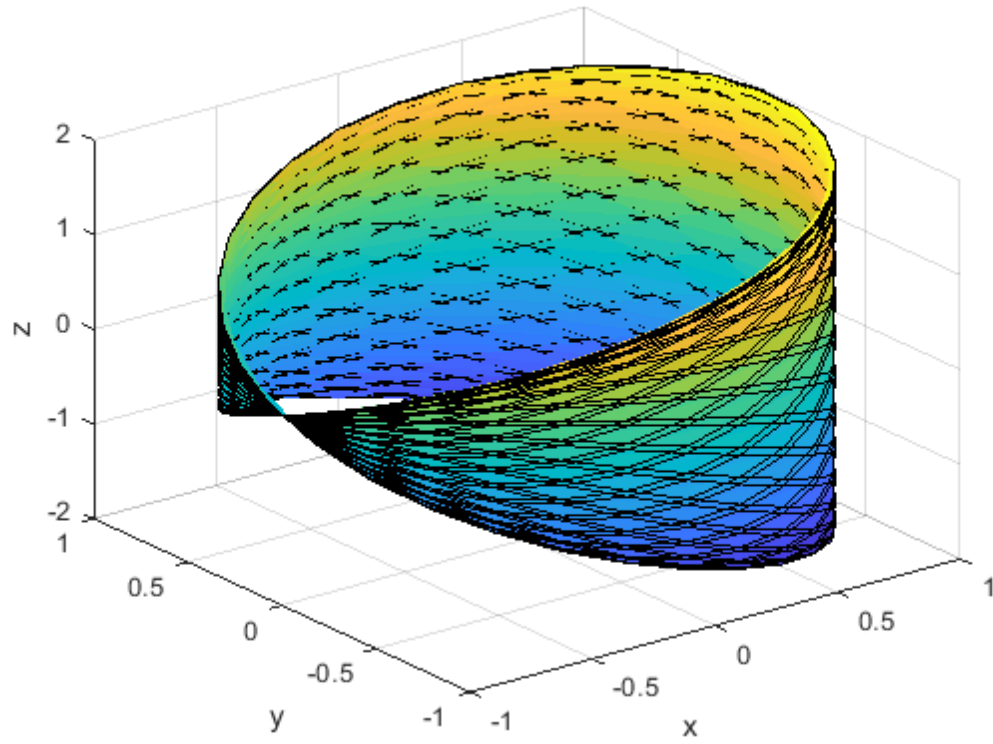
```
close all
[X,Y,Z]=peaks(30);
x=X(1,:);
y=Y(:,1);
i=find(y>0.8 & y<1.2);
j=find(x>-.6 & x<.5);
Z(i,j)=nan*Z(i,j);
surf(X,Y,Z);
title('带洞孔的山峰表面');
xlabel('x-axis'),ylabel('y-axis '),zlabel('z-axis')
```



符号函数的表面图

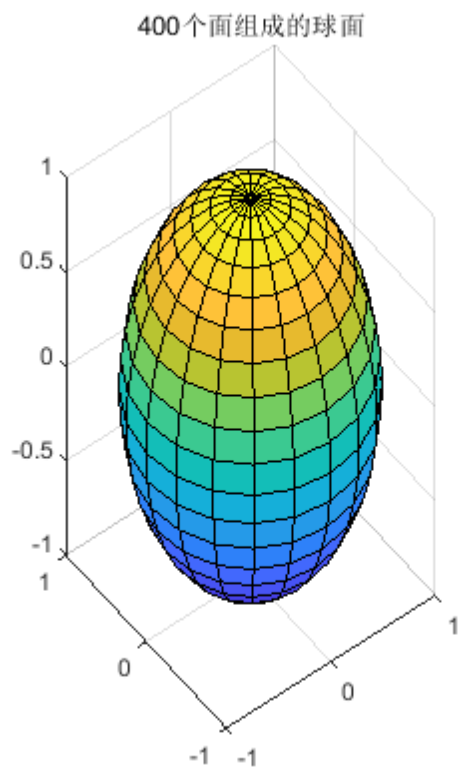
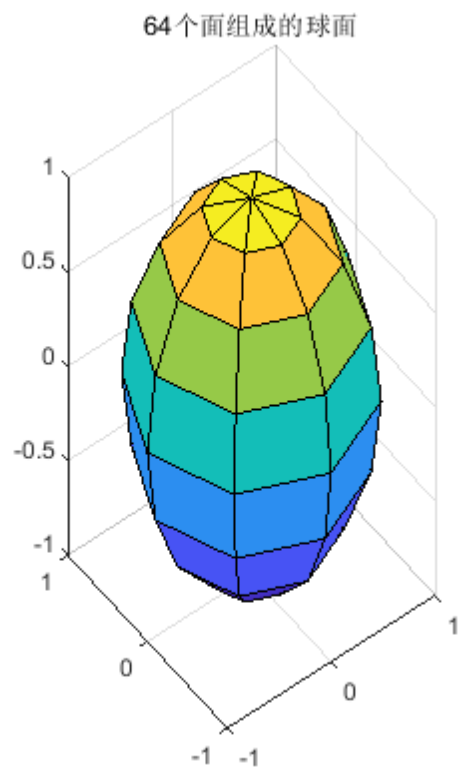
```
close all
syms s t
x=sin(s+t);
y=cos(s+t);
z=sin(s)+cos(t);
ezsurf(x,y,z,[-pi,pi],30)
title('符号函数曲面图')
```

符号函数曲面图



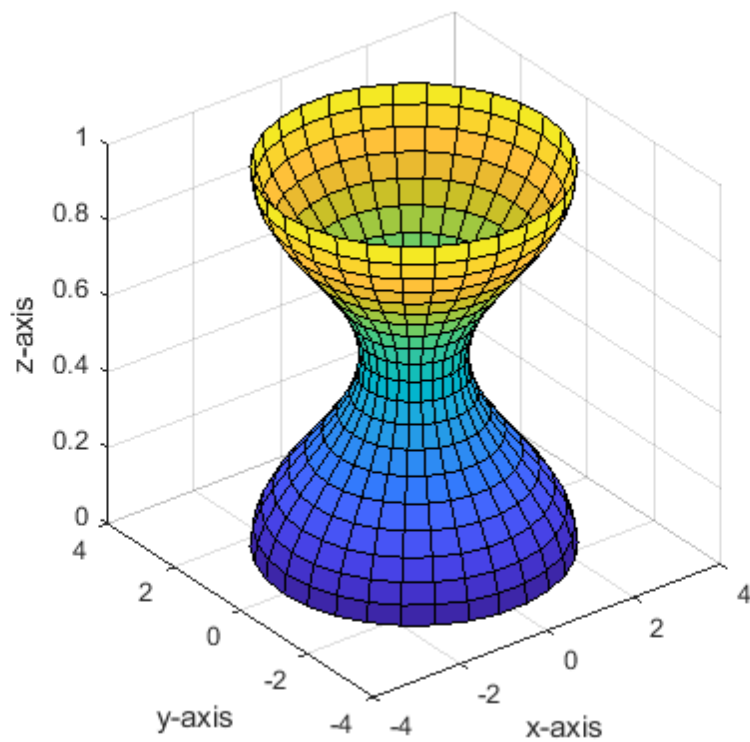
球面

```
close all
[X1,Y1,Z1]=sphere(8);
[X2,Y2,Z2]=sphere(20);
subplot(1,2,1)
surf(X1,Y1,Z1)
title(' 64个面组成的球面')
subplot(1,2,2)
surf(X2,Y2,Z2)
title(' 400个面组成的球面')
```



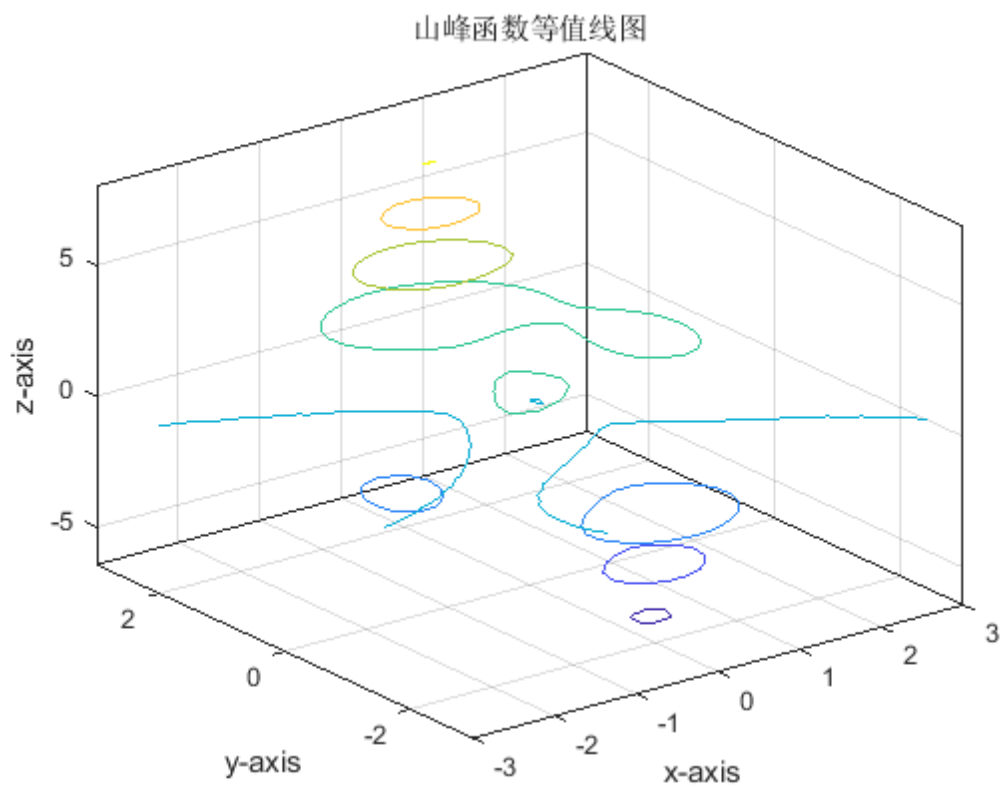
柱面

```
close all
t=0:pi/10:2*pi;
[X,Y,Z]=cylinder(2+cos(t),30);
surf(X,Y,Z)
axis square
xlabel('x-axis'),ylabel('y-axis'),zlabel('z-axis')
% 需要matlab高版本
```

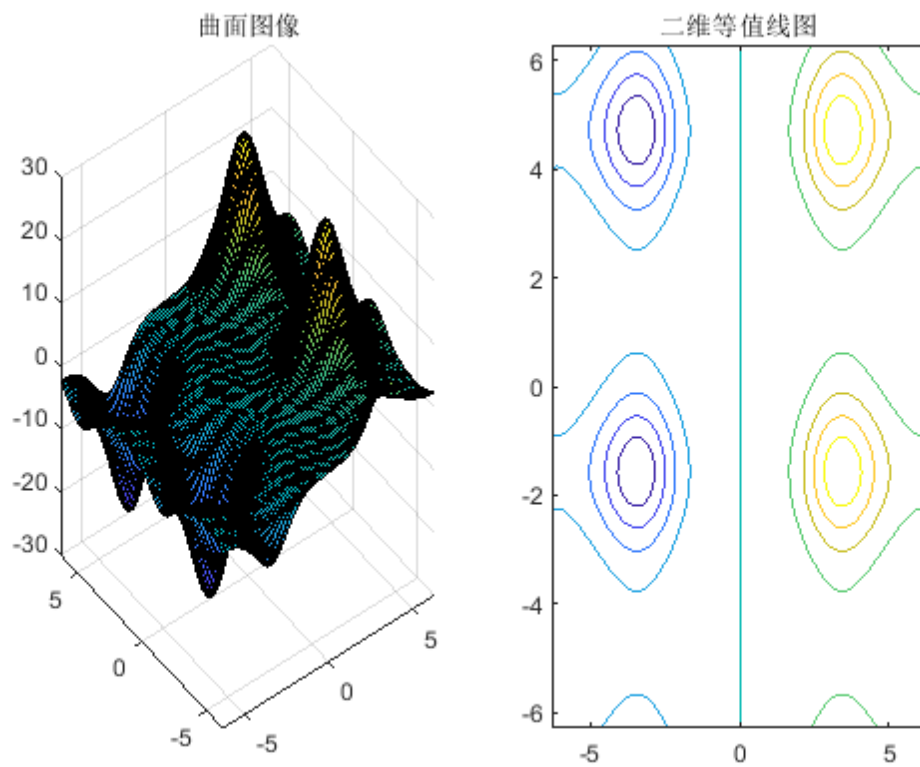


等值线

```
close all
[x, y, z]=peaks(30);
contour3(x, y, z);
title(' 山峰函数等值线图');
xlabel('x-axis'),ylabel('y-axis '),zlabel('z-axis')
```

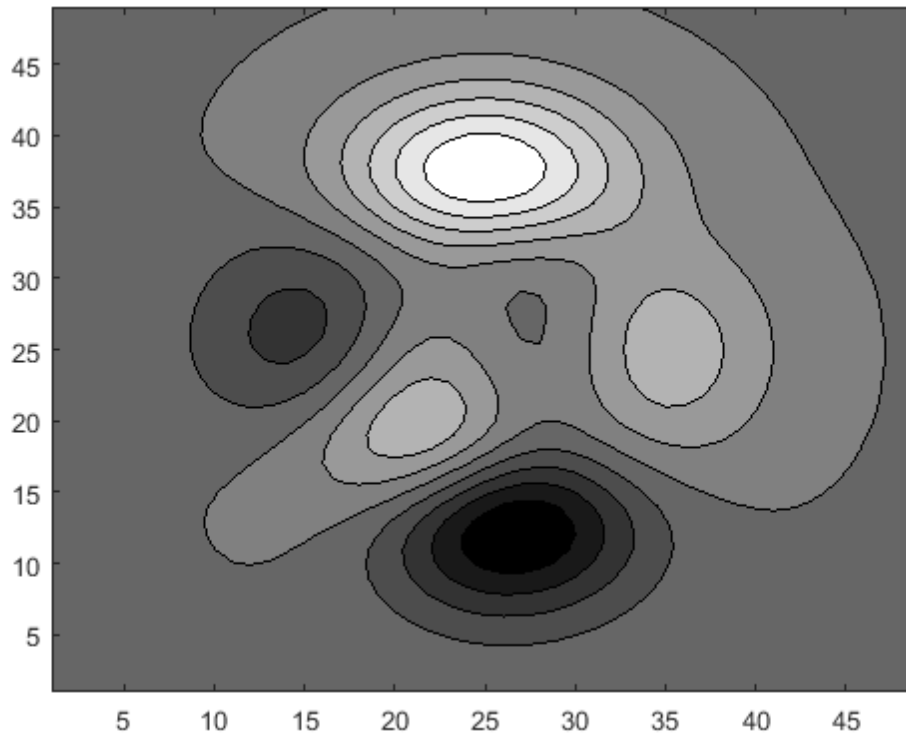



```
close all
x=linspace(-2*pi,2*pi,100);
y=x;
[X,Y]=meshgrid(x,y);
Z=X.*exp(-cos(X)-sin(Y));
subplot(1,2,1);
surf(X,Y,Z);
title(' 曲面图像');
subplot(1,2,2);
contour(X,Y,Z);
title(' 二维等值线图')
```



```
close all
Z=peaks;
[C,h]=contourf(Z,10);
colormap gray;
title(' 二维等值线图及颜色填充')
```

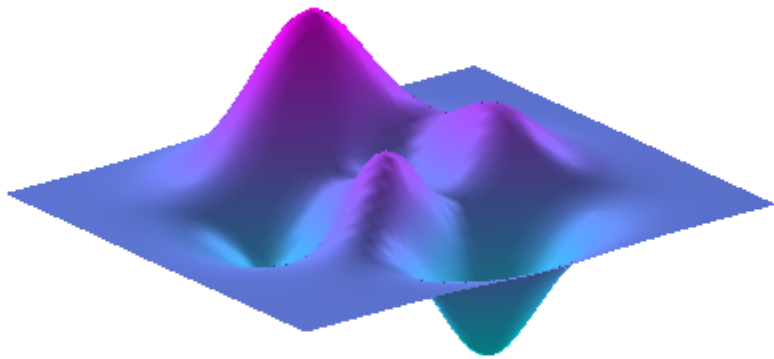
二维等值线图及颜色填充



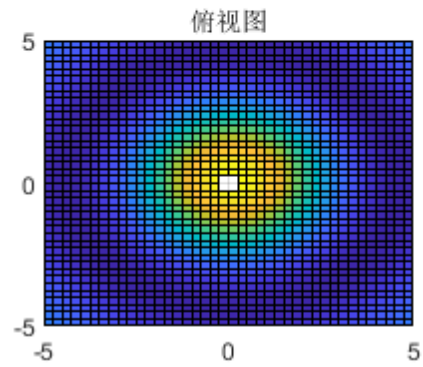
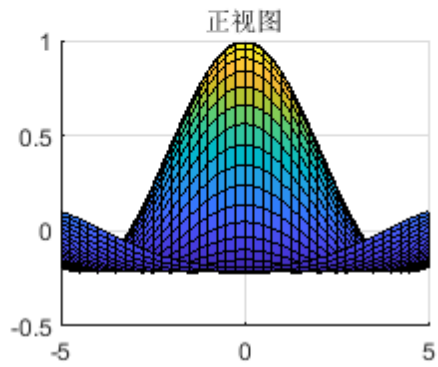
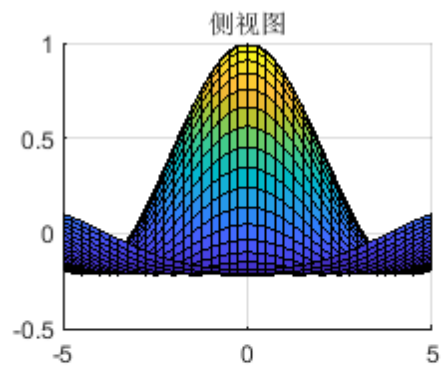
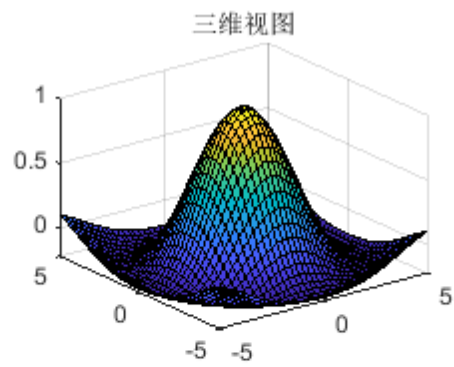
图像的进一步处理

设置观察点，`view(alpha, beta)`, `alpha`为方位角，表达视点与y轴负向的夹角，`beta`变大视点与坐标原点的连线与xoy平面的夹角，默认`alpha=-37.5`，`beta=30` 光源设置：`light('position',[n1,n2,n3])` 着色处理：`colormap('设置')`

```
clear, clc, clf
h=surf(peaks);
set(h, 'FaceLighting', 'phong', 'FaceColor', 'interp', 'AmbientStrength', 0.5)
light('Position', [1 0 0], 'Style', 'infinite');
lighting phong
axis off
set(h, 'facecolor', [1, 0.8, 1])
colormap('cool')
shading interp
camlight(100, 150)
```

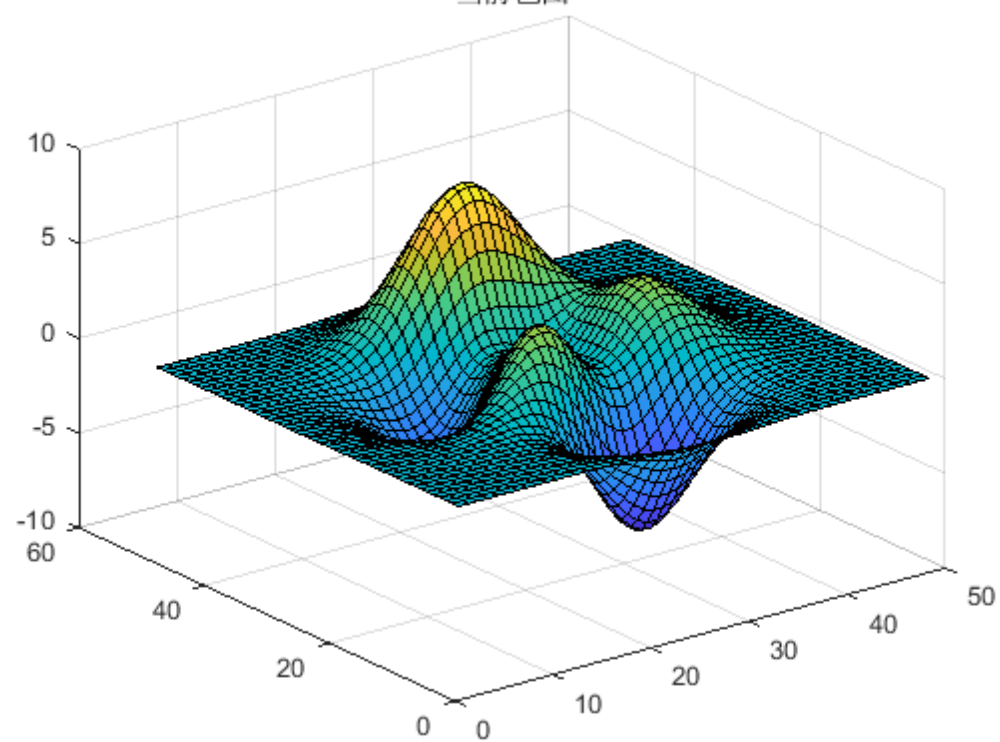


```
close all
[X,Y]=meshgrid(-5:0.25:5);
Z=sin(sqrt(X.^2+Y.^2))./sqrt(X.^2+Y.^2);
subplot(2,2,1)
surf(X,Y,Z),title('三维视图')
subplot(2,2,2)
surf(X,Y,Z),view(90,0)
title('侧视图')
subplot(2,2,3)
surf(X,Y,Z),view(0,0)
title('正视图')
subplot(2,2,4)
surf(X,Y,Z),view(0,90)
title('俯视图')
```

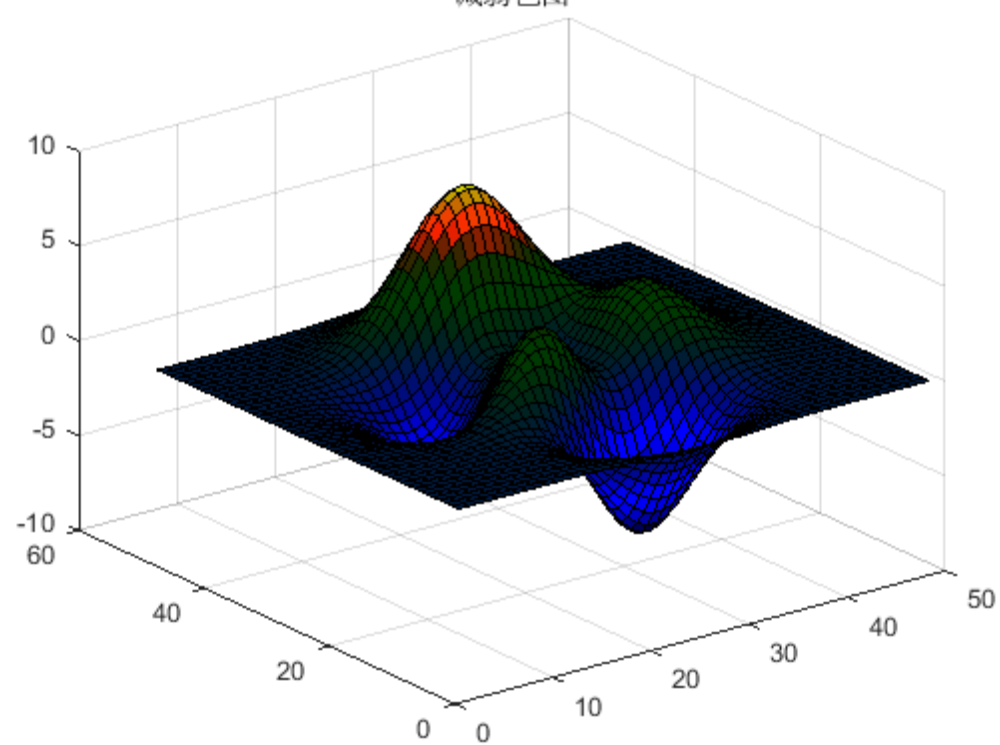


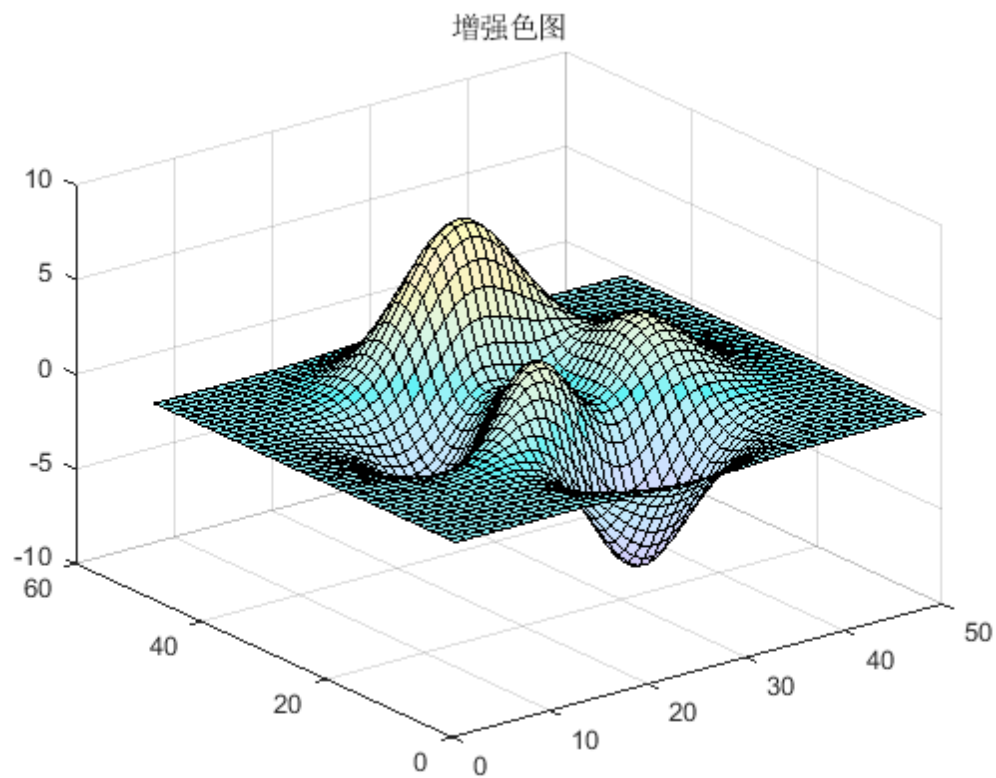
```
h1=figure;
surf(peaks);
title('当前色图')
h2=figure;
surf(peaks),brighten(-0.85)
title('减弱色图')
h3=figure;
surf(peaks),brighten(0.85)
title('增强色图')
```

当前色图



减弱色图



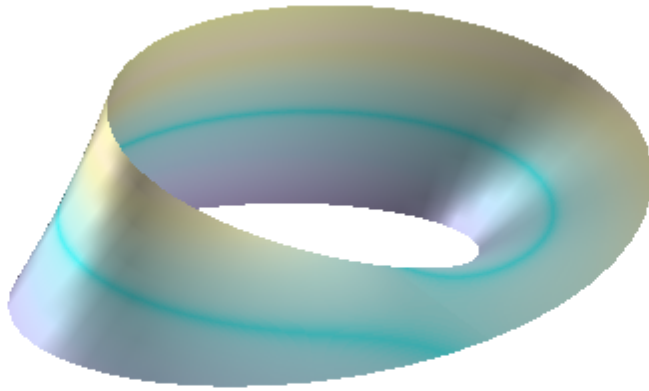


莫比乌斯带

曲面方程 $x = r(t, v) \cos t, y = r(t, v) \sin(t), z = bv \sin \frac{t}{2}$, 其中 $r(t, v) = a + bv \cos \frac{t}{2}$, a, b 为常数, $t \in [0, 2\pi], v \in [-1, 1]$.

```
clear ,clc
ezsurf(' (0.5+0.2*v*cos(t/2))*cos(t)', ' (0.5+0.2*v*cos(t/2))*sin(t)', ' 0.2*v*sin(t/2)', [0, 2*pi, -1, 1])
light('position', [2, 1, 2])
lighting phong;%照明设置
shading interp;axis off%不显示坐标轴
camlight(-220, -170)% 设置光照位置
axis equal
view(60, 25)
```

$$x = (0.5 + 0.2 \cos(t/2)) \cos(t), y = (0.5 + 0.2 \cos(t/2)) \sin(t), z = 0.2 \sin(t/2)$$

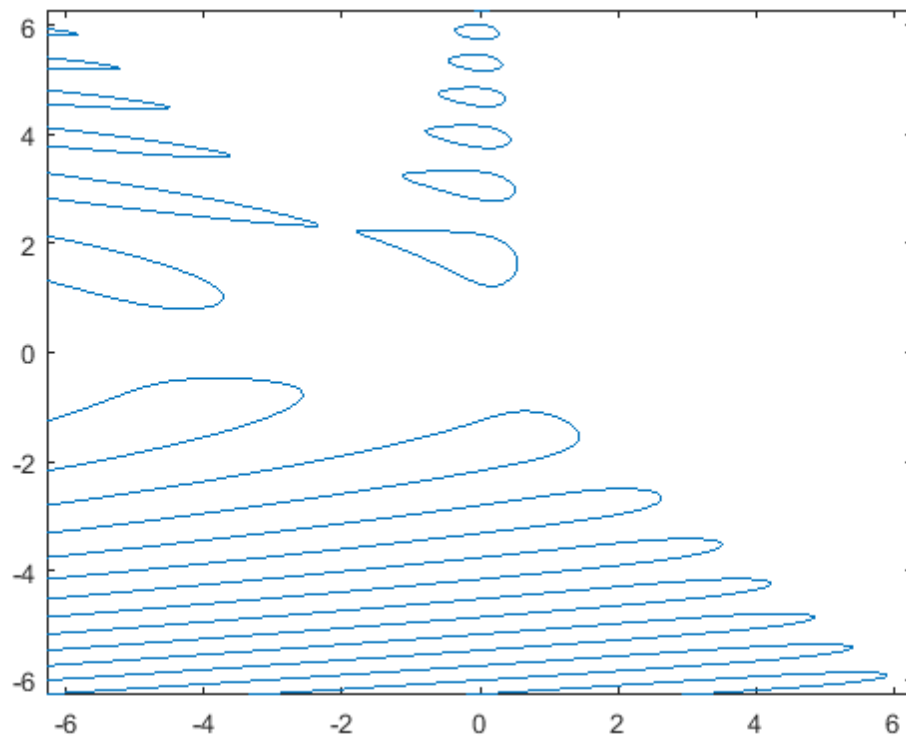


专题研究

图解法解方程 绘制隐函数 $y^2 \cos(x + y^2) + x^2 e^{x+y} = 0$ 在 $(-2\pi, 2\pi)$ 上图像。

```
syms x y;  
p=2*pi;  
f=y^2*cos(x+y^2)+x^2*exp(x+y);  
fimplicit(f, [-p, p])
```

% 需要在matlab高版本下测试，但因为默认网格密度小，曲线不光滑。
fimplicit(f, [-p, p], 'Meshdensity', 500) %更光滑

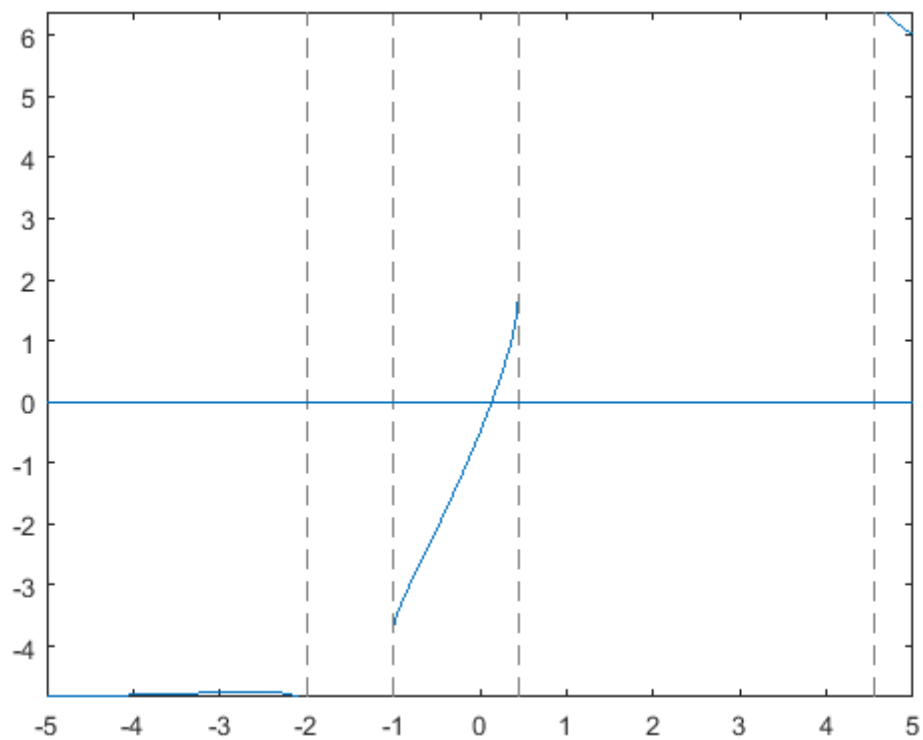


```
syms x
f=sqrt(2*x^2+3)+sqrt(x^2+3*x+2)-sqrt(2*x^2-3*x+5)-sqrt(x^2-5*x+2);
% f=matlabFunction(f);
f= str2func(['@(x)',vectorize(f)]) %将符号表达式转化为函数句柄的方法
fplot(f, [-5, 5]), line([-5 5], [0 0])
% fplot(f), line([-5 5], [0 0]) %绘制曲线，方程与横轴只有一个交点
% x=0.13809878
```

f =

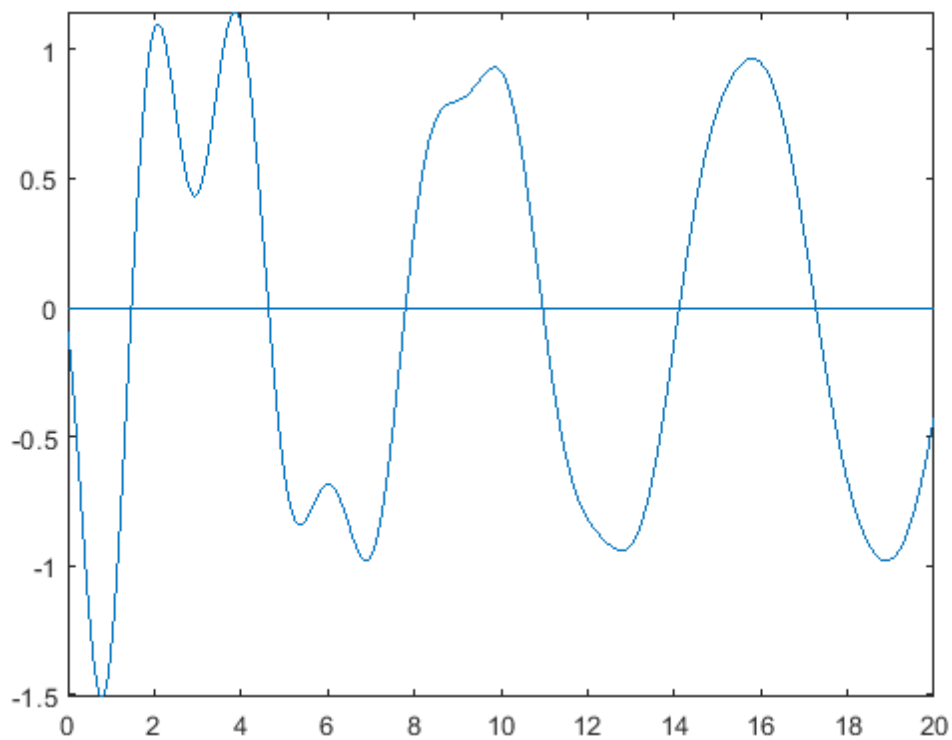
包含以下值的 function_handle:

```
@(x) (3.*x+x.^2+2).^(1./2)-(x.^2-5.*x+2).^(1./2)-(2.*x.^2-3.*x+5).^(1./2)+(2.*x.^2+3).^(1./2)
```

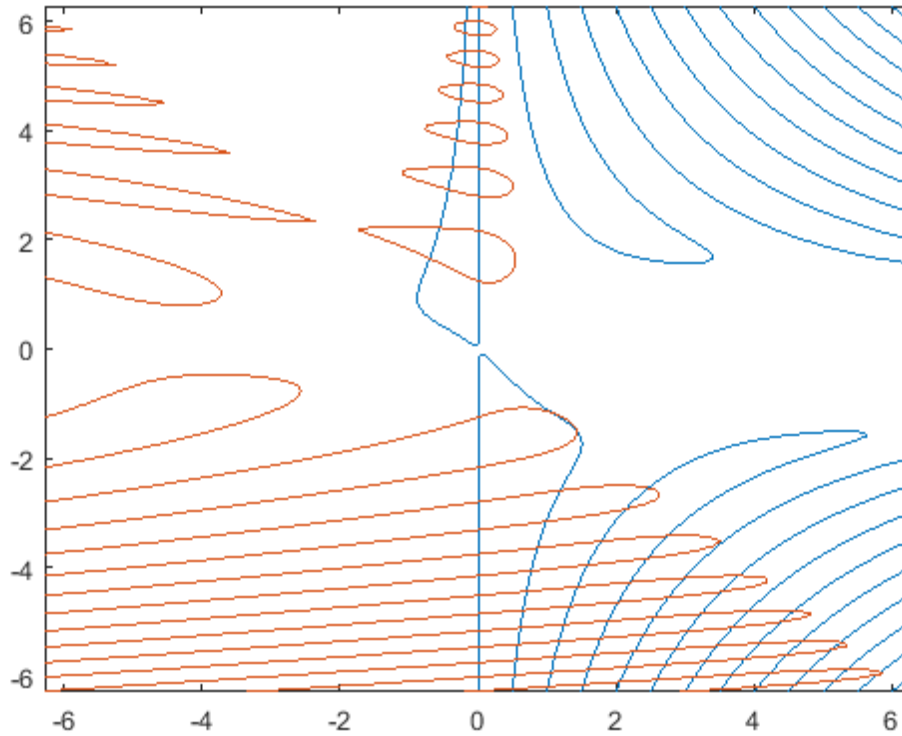
超越方程无解析解

```
f=@(x) exp(-0.2*x).*sin(3*x+2)-cos(x);
fplot(f, [0, 20]), line([0, 20], [0, 0])
% x=10.9601
```



二元方程

```
syms x y
f1=x^2*exp(-x*y^2/2)+exp(-x/2)*sin(x*y);
f2=y^2*cos(x+y^2)+x^2*exp(x+y);
fimplicit([f1 f2],[-2*pi,2*pi])
%需要高本版
% 局部放大，不过交点(x, y)太多，一个一个去处理
```



http://cn.mathworks.com/help/matlab/matlab_prog/marking-up-matlab-comments-for-publishing.html

LaTeX Markup Example

This is a table:

Published with MATLAB® R2019b