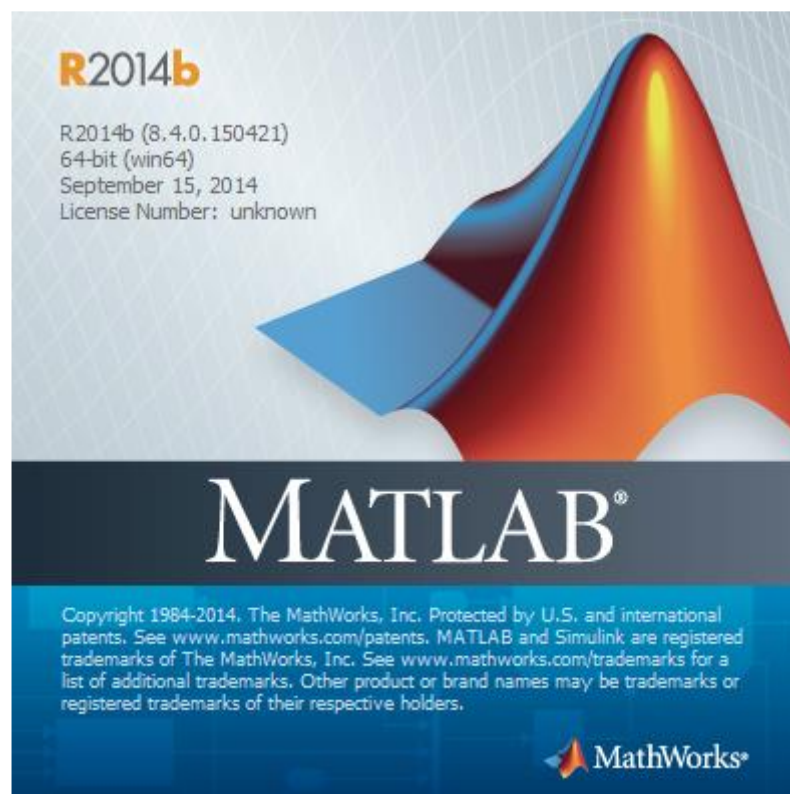


第3章a-20200302更新

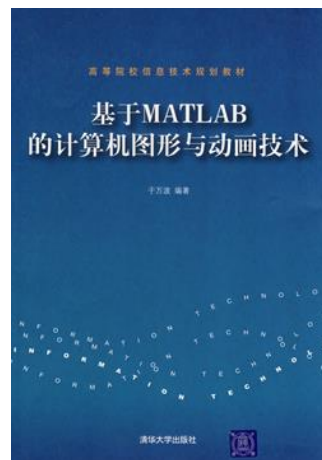
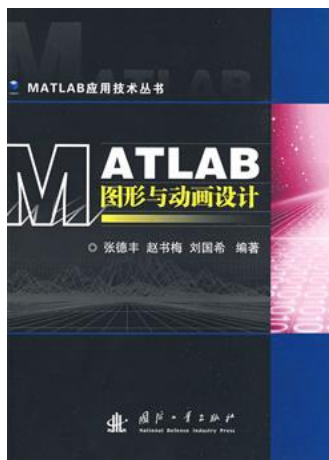
MATLAB软件与应用



第三章 数据可视化

- 二维数据绘图
- 三维数据绘图
- 图形效果的修饰
- 句柄绘图

- MATLAB具有极强大的数据可视化功能，可制作具有出版质量的图形。
- 在前面的课程中，已经使用了数据可视化命令plot。
- 详细介绍MATLAB这一部分的内容可以写一本书。



- 我们重点介绍2D与3D数据可视化基础知识和常用方法，初步了解图形修饰、句柄绘图的基本概念。



二维数据绘图

二维绘图 一些基本概念

- 2D图形：二维坐标系上的数据形成的平面图形；
- 二维坐标可采用直角坐标、对数坐标、极坐标等形式；
- 数据一般以向量或矩阵形式给出，也能以函数表达式形式给出；
- 数据可以是实型或复型；
- 二维绘图除了最常用的曲线图，还有饼状图、等高线图、条形图、矢量图等多种特殊形式
- 可以在一副图片中绘制多条曲线
- 可在一副图片中使用两套纵坐标
- 可在一个图形窗口中绘制使用独立坐标系的多个子图
- 可设置图形的许多细节（曲线颜色、线形、图片标题等）

二维绘图 **plot**

根据输入参数的不同形式，有三类不同用法。

plot(y) plot(x,y) plot(x1,y1,x2,y2)

plot(y)

- **y**为向量($1*N$ 或 $N*1$ 矩阵)时，以元素下标为横坐标、元素值为纵坐标绘出曲线
- **y**为实 $M*N$ 二维矩阵时，按列绘制**每列**元素值相对其下标的曲线，曲线数等于**y**矩阵的列数**N**。
- **y**为复二维矩阵时，按列分别以矩阵的实部和虚部为横、纵坐标绘制多条曲线

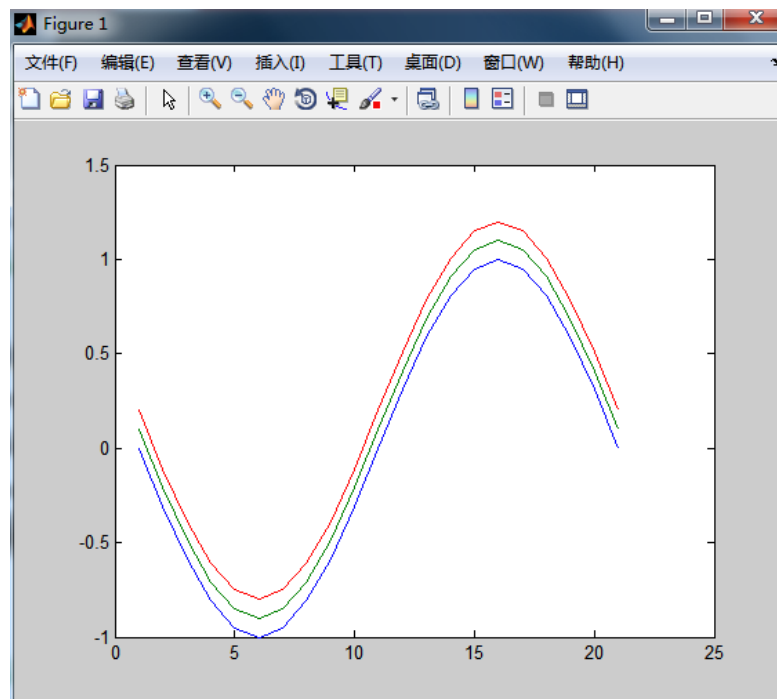
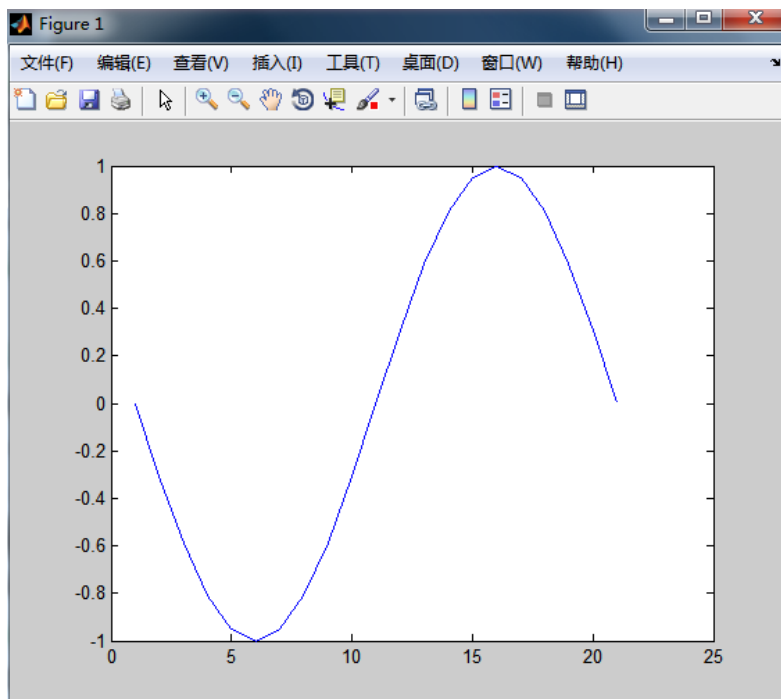
二维绘图 **plot**

【例】 $x = -\pi : \pi/10 : \pi$; $y = \sin(x)$; %21个数据点

$\text{plot}(y)$ % $\text{plot}(y')$ 的结果也相同

$z = [y; y+0.1; y+0.2]$; $\text{plot}(z')$ % z' 是 21×3 矩

三列得到三条线



二维绘图

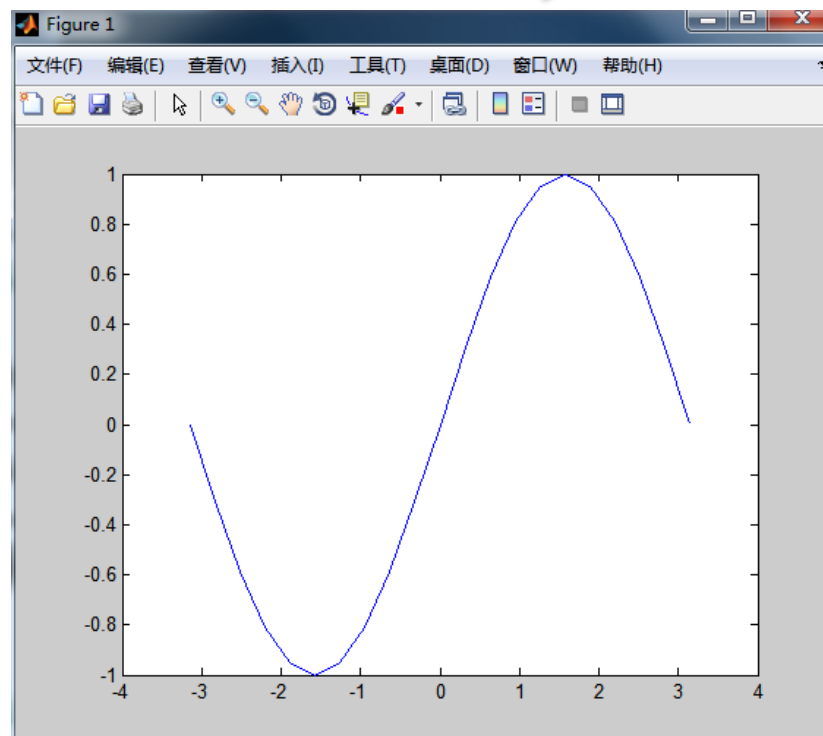
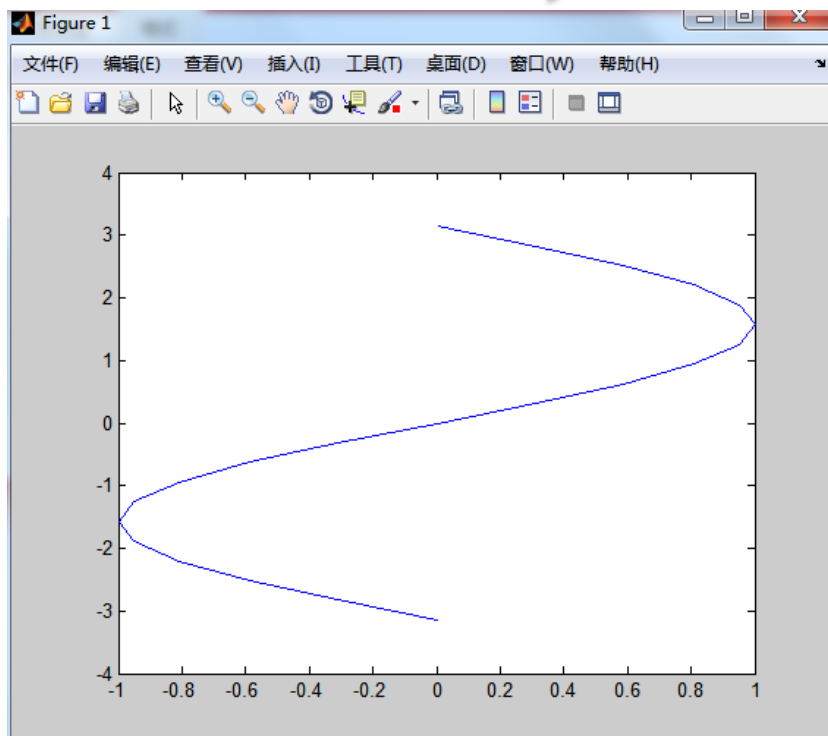
plot

【例】 $x = -\pi:\pi/10:\pi$; $y = \sin(x)$; %21个数据点

%z是1*21复矩阵

$z = y + x*i$; plot(z)

$z = x + y*i$; plot(z)



二维绘图

plot

plot(x, y)

- x、y为同维向量时，绘制以x、y元素为横纵坐标的曲线
- x为含有N个元素的向量（ $1 \times N$ 或 $N \times 1$ ），y为 $M \times N$ 或 $N \times M$ 矩阵时，以x为公共横坐标，绘制M条不同颜色的曲线
- x为二维矩阵，y为向量时，情况与上面类似，只是y仍为纵坐标
- x和y都是 $M \times N$ 矩阵时，绘制N条曲线，每条曲线依次对应x和y的每列数据

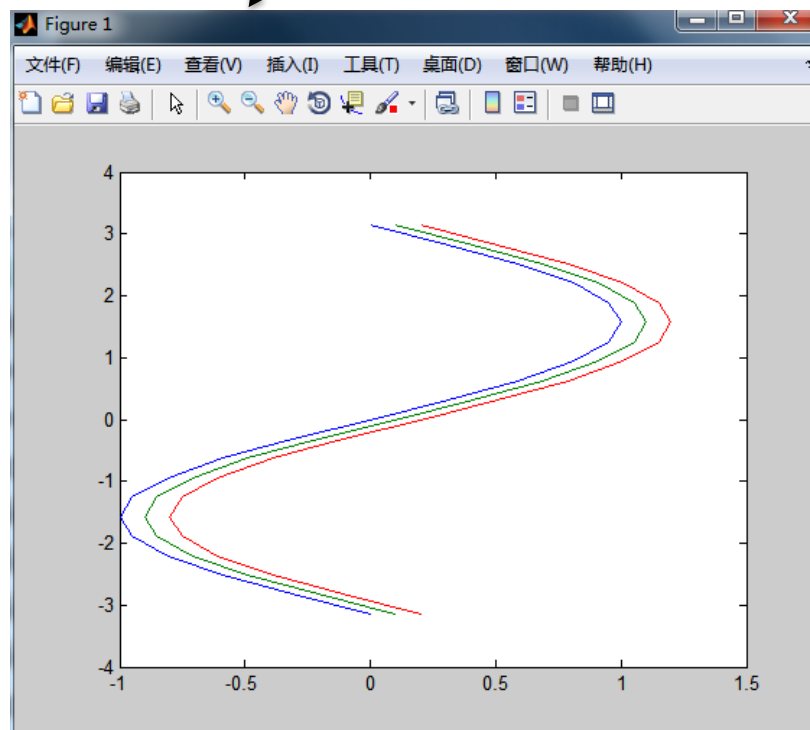
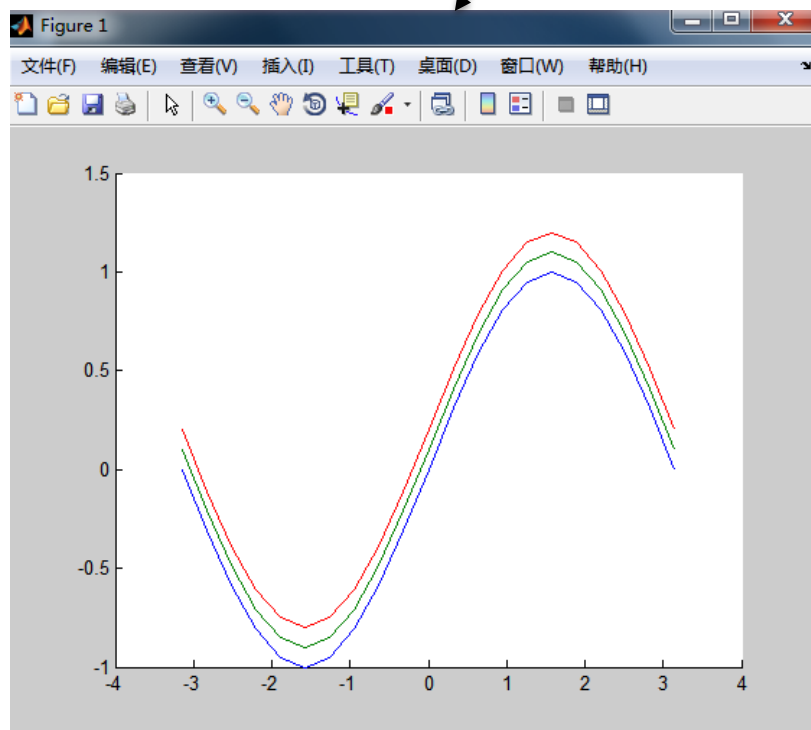
二维绘图 **plot**

【例】 $x = -\pi:\pi/10:\pi$; $y = \sin(x)$; %21个数据点

$z = [y; y + 0.1; y + 0.2]$; %3*21

$\text{plot}(x, z)$;

$\text{plot}(z, x)$;

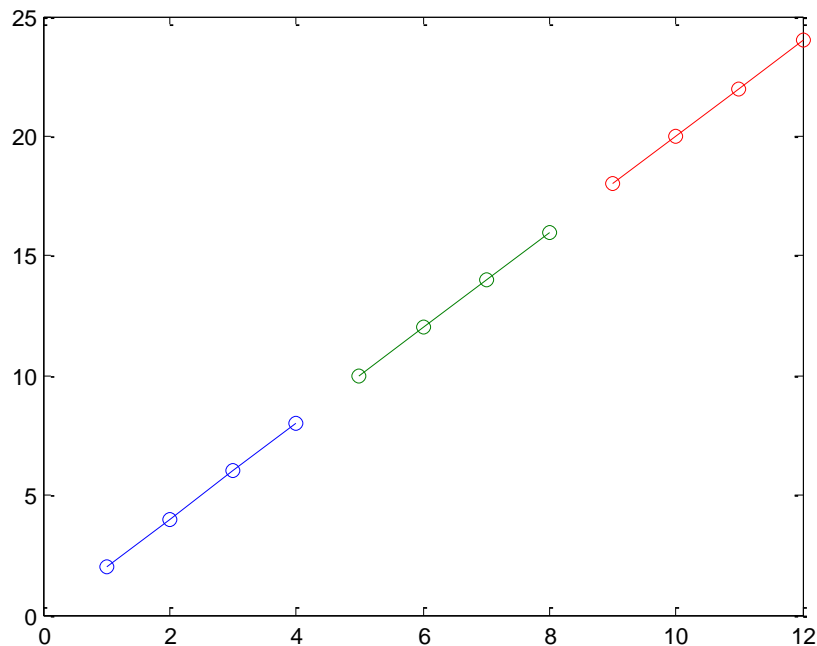


二维绘图

plot

【例】 $x = \begin{bmatrix} 1, & 5, & 9 \\ 2, & 6, & 10 \\ 3, & 7, & 11 \\ 4, & 8, & 12 \end{bmatrix}$
 $y = x^2$; `plot(x,y,'-o')`

x , y 都是 4×3 矩阵，因此以上命令绘制出 3 条曲线，每条曲线的数据来自 x 和 y 的每一列数据，每条曲线包含 4 个数据点



二维绘图

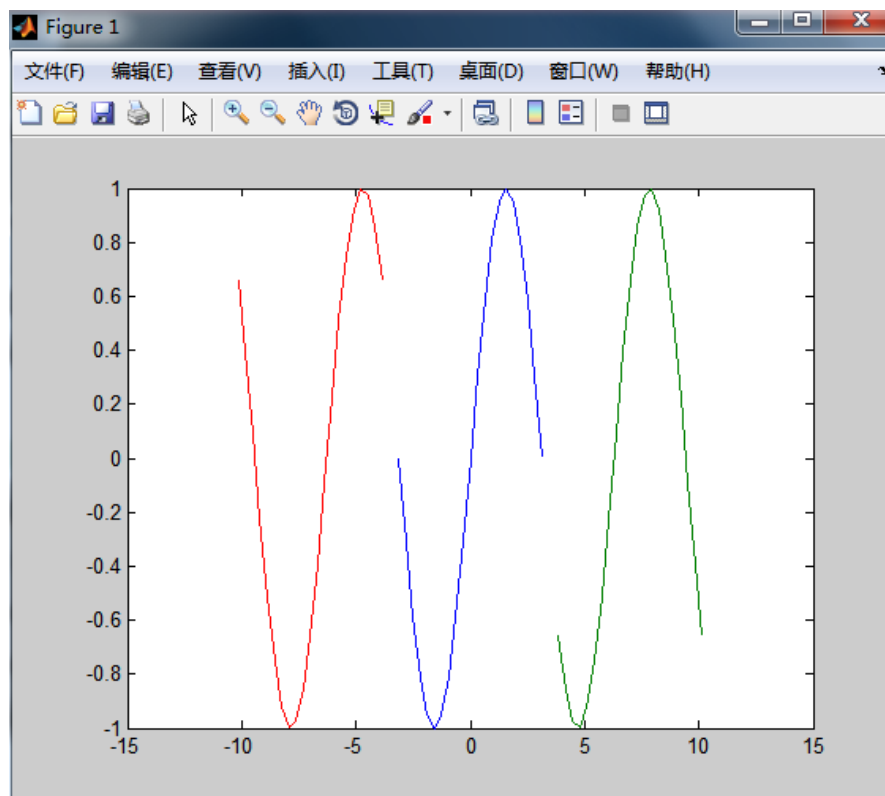
plot

【例】 $x = -\pi:\pi/10:\pi$; %1*21

$x=[x;x+7;x-7];$

$y=\sin(x)$; %x和y 3*21

$\text{plot}(x',y')$ %x'和y' 21*3, $\text{plot}(x,y)$ 是什么结果



二维绘图

plot

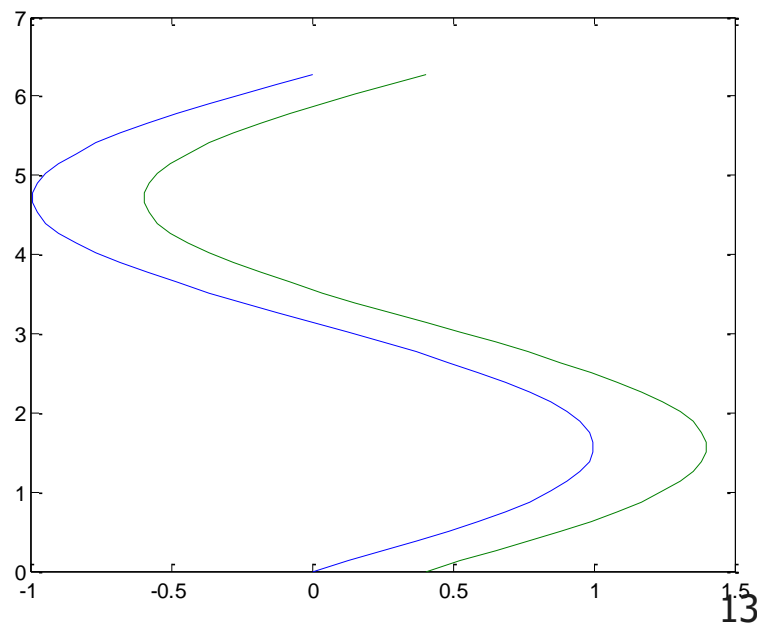
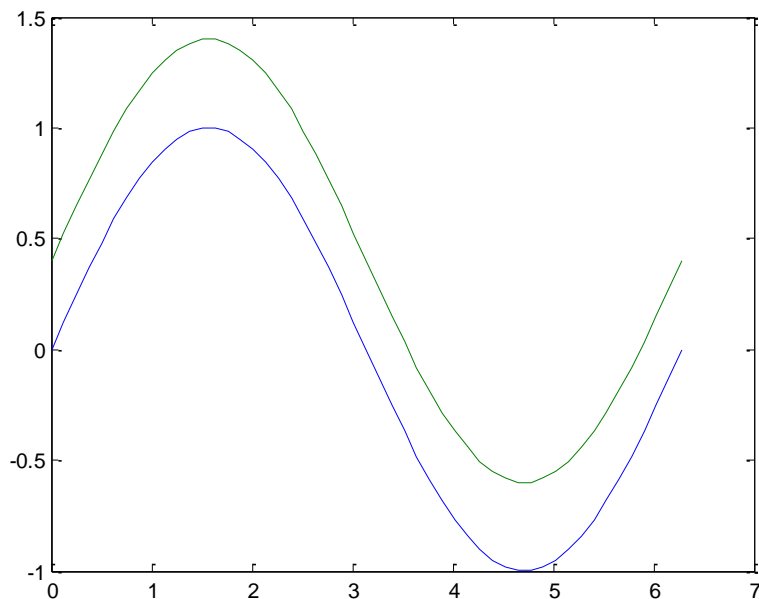
【例】 绘制两条曲线

$$\begin{aligned} y1 &= \sin(x) \\ y2 &= \sin(x) + 0.2, \quad 0 \leq x \leq 2\pi \end{aligned}$$

```
x = 0:pi/50:2*pi;
```

```
y1=sin(x); y2=y1+0.2; y=[y1;y2]
```

```
plot(x,y1,x,y2) ; plot(x,y) ; plot(y,x) ; %注意区别
```



二维绘图

plot 指定曲线样式

曲线样式设定: `plot(x, y, 's')`

曲线线形控制符

符号	-	:	-.	--
含义	实线	虚线	点划线	双划线

曲线颜色控制符

符号	b	g	r	c	m	y	k	w
含义	蓝	绿	红	青	品红	黄	黑	白

二维绘图

plot 指定曲线样式

数据点形控制符

符号	含义	符号	含义
.	实心黑点	d	菱形符 diamond
+	十字符	h	六角星符 hexagram
*	八线符	o	空心圆圈
^	朝上三角符	p	五角星符 pentagram
<	朝左三角符	s	方块符 square
>	朝右三角符	x	叉字符
v	朝下三角符		

- 曲线的线形控制符、颜色控制符、数据点形控制符可以组合使用，也可以单独使用
- 控制符先后次序不影响绘图结果

二维绘图

plot 指定曲线样式

```
x=0 : 0.02*pi : pi;
```

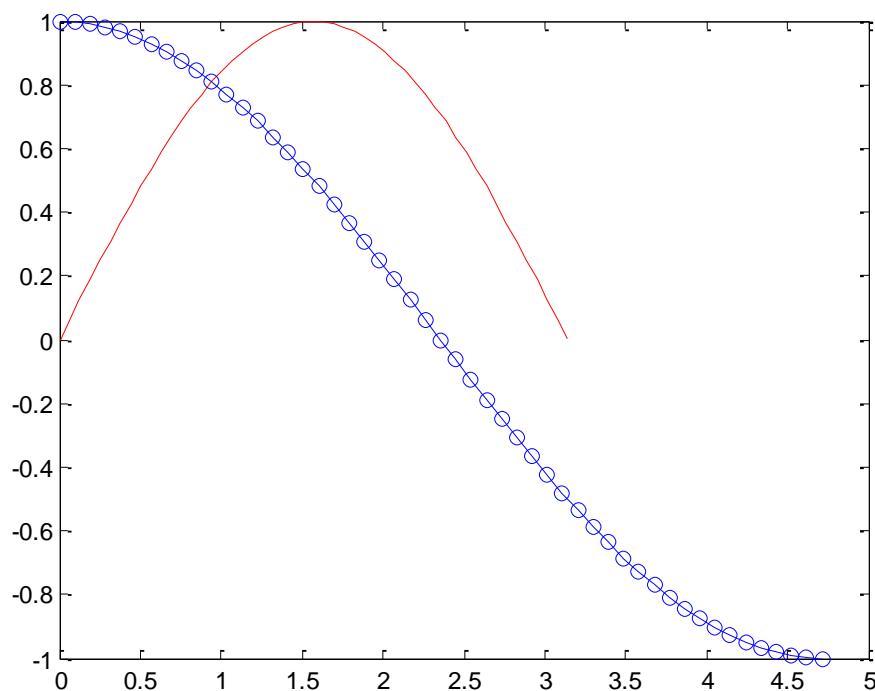
```
x1=x'
```

```
x2=x1*1.5
```

```
y1=sin(x)'
```

```
y2=cos(x)';
```

```
plot(x1,y1, 'r-', x2,y2, 'b-o' )
```

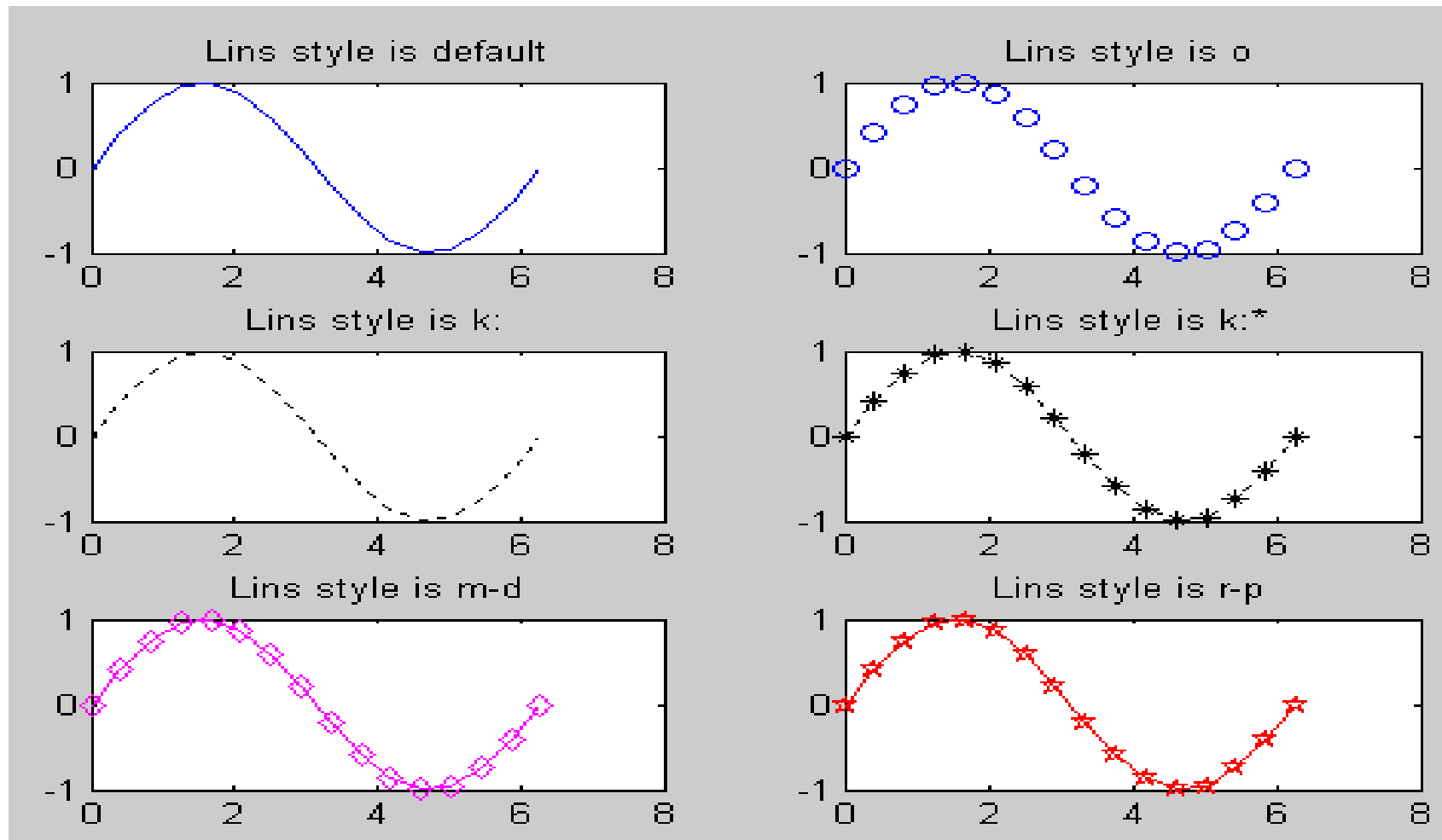


二维绘图

plot

指定曲线样式

【例】各种线形

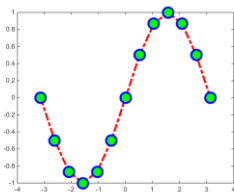


二维绘图

plot 指定曲线样式

更多设置: 曲线宽度,
数据点的大小、边框颜色、内部颜色等

【例】 $x = -\pi:\pi/6:\pi$; $y=\sin(x)$;
`plot(x,y,'ro-.','LineWidth',3,'MarkerEdgeColor','k',...
'MarkerFaceColor','g','MarkerSize',12)`



二维绘图

图形细节控制 标注

图形标注 **title**、**xlabel**、**ylabel**、**zlabel**、**legend**、**text** 等

其中 **title**、**xlabel** 为图形标题和坐标名称，调用格式相似，以xlabel为例：

xlabel('标注', '属性1', 属性值1, '属性2', 属性值2,.....)

属性为文本属性，包括：字体大小、字体类型、字体粗细等。

【例】

```
x=0:0.1*pi:2*pi;
```

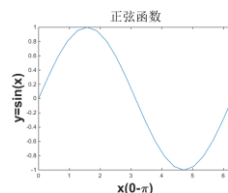
```
y=sin(x), plot(x,y) %先绘图形再设标注
```

```
axis([0,2*pi,-1,1])
```

```
xlabel('x(0-\pi)', 'fontweight', 'bold');
```

```
ylabel('y=sin(x)', 'fontweight', 'bold');
```

```
title('正弦函数', 'fontsize', 22)
```



图例标注 **legend**

`legend('标注1','标注2',..., 'location', 定位代号)`

定位代号包括字符串和数字两类，也可以缺省

字符串代号为'North' 'NorthWest' 'East' 等表示方位的关键词

数字代号为-1~4，详细内容见doc legend

文本标注 **text**

`text(x,y,'标注文本及控制字符串')`

其中x,y给定了标注文本在图中添加的位置

关于字体的设置:

\fontname{arg} \arg \fontsize {arg} string

其中String为要输出的字符串，其前面的均为属性控制

	指令	arg取值	示例
指定字体	\fontname {arg}	Arial 宋体	'\fontname {'宋体'}
指定风格	\arg	Bf (黑体) It 斜体 1 Sl 斜体 2 Rm 正体	'\bf example'
指定大小	\fontsize {arg}	正整数 (缺省 10P)	'\fontsize {16} Example1.'

上下角标的控制

	指令	Arg 取值	举例	
			示例指令	效果
上标	<code>^{\arg}</code>	任何合法字符	<code>'\exp^{(-t)}\sin(t)'</code>	$e^{-t} \sin(t)$
下标	<code>_{\arg}</code>	任何合法字符	<code>'U_{\alpha}'</code>	U_{α}

希腊字母与特殊字符

Character Sequence	Symbol	Character Sequence	Symbol	Character Sequence	Symbol
<code>\alpha</code>	α	<code>\upsilon</code>	υ	<code>\sim</code>	\sim
<code>\beta</code>	β	<code>\phi</code>	ϕ	<code>\leq</code>	\leq
<code>\gamma</code>	γ	<code>\chi</code>	χ	<code>\infty</code>	∞
<code>\delta</code>	δ	<code>\psi</code>	ψ	<code>\clubsuit</code>	\clubsuit
<code>\epsilon</code>	ϵ	<code>\omega</code>	ω	<code>\diamondsuit</code>	\diamondsuit
<code>\zeta</code>	ζ	<code>\Gamma</code>	Γ	<code>\heartsuit</code>	\heartsuit

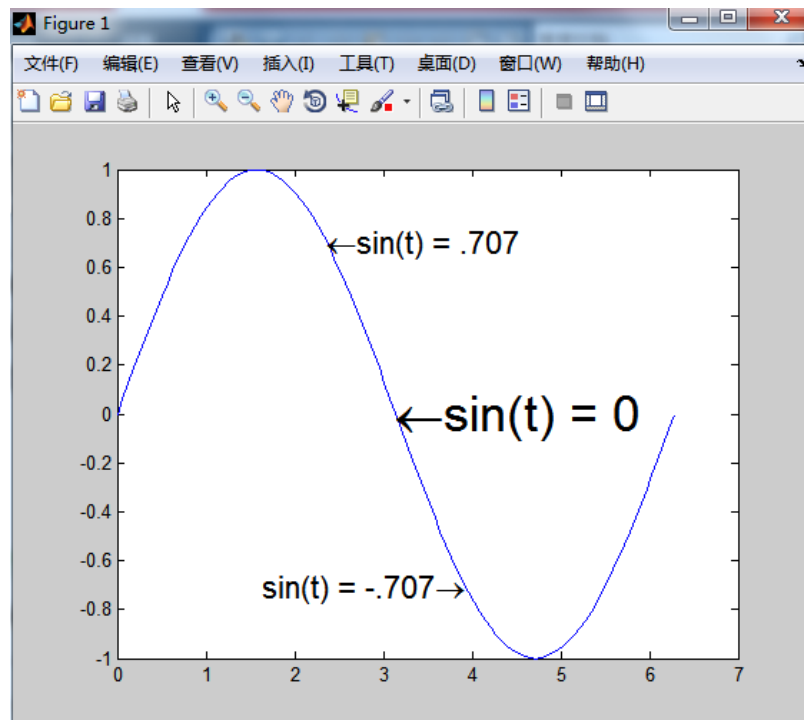
二维绘图

图形细节控制 标注

【例】

```
t=(0:100)/100*2*pi; y=sin(t); plot(t, y);  
text(3*pi/4, sin(3*pi/4), '\fontsize{16}\leftarrowsin(t) = .707 ');  
text(pi, sin(pi), '\fontsize{26}\leftarrowsin(t) = 0 ');  
text(5*pi/4, sin(5*pi/4), '\fontsize{16}sin(t) = -.707\rightarrow',...  
    'HorizontalAlignment','right')
```

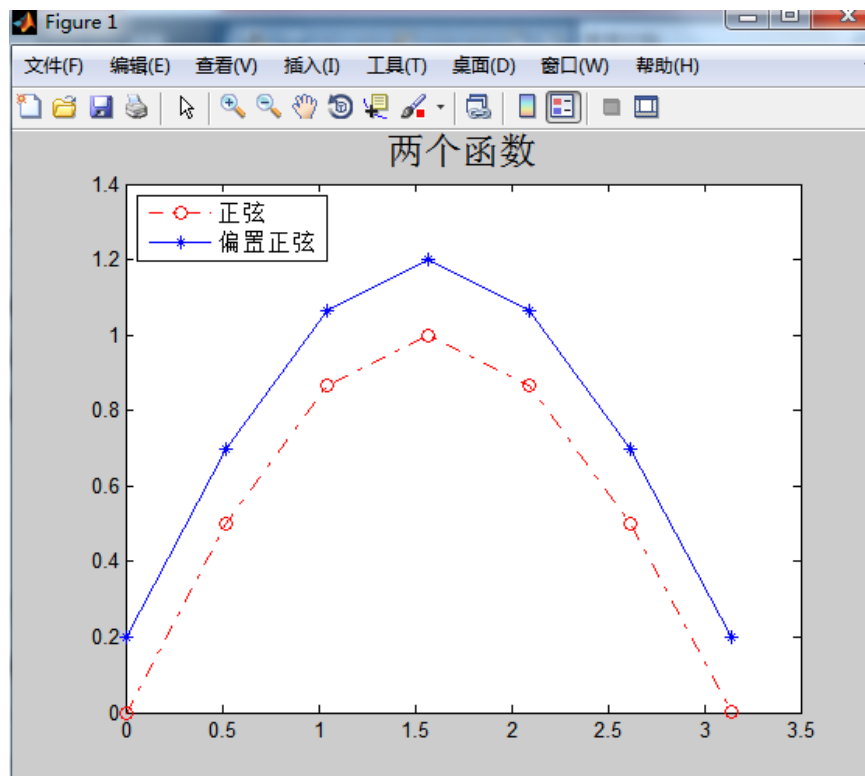
‘HorizontalAlignment’, ‘right’
设置图形标识为水平右对齐



二维绘图

图形细节控制 标注

【例】 $x = 0:\pi/6:\pi;$ $y1=\sin(x);$ $y2=y1+0.2;$
plot(x,y1,'ro-.',x,y2,'b*-')
title('两个函数','fontsize',18)
legend('正弦','偏置正弦','location','NorthWest')



二维绘图

图形细节控制

网格

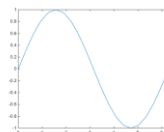
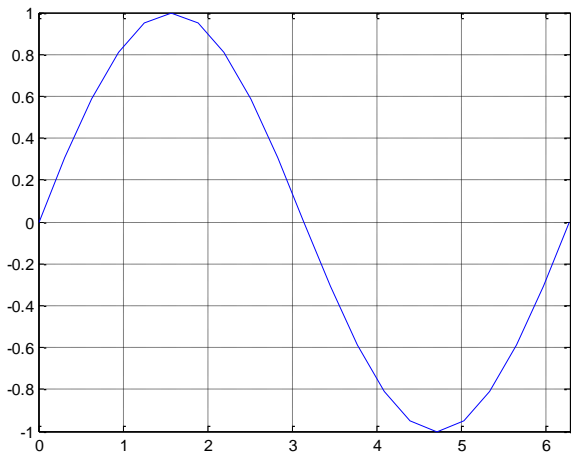
平面坐标的网格函数 **grid**

grid on 表示在图形中绘制坐标网格, **grid off** 表示取消坐标网格。
单独的**grid**函数将实现**grid on**与**grid off**状态间的转换。

【例】 $x=0:0.1*\pi:2*\pi$; $y=\sin(x)$; **plot(x,y)**

axis([0,2*pi,-1,1])

grid on; **grid** %或**grid off**,之后图形变成



二维绘图

图形细节控制

背景

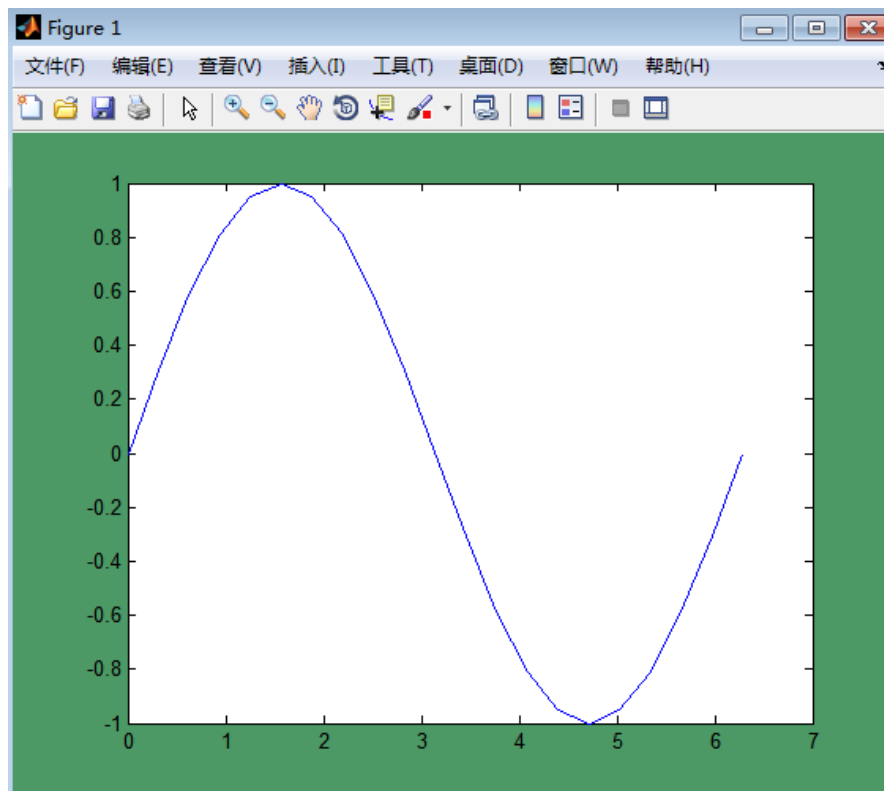
通过**set** 或**figure**函数可设置图片背景颜色

【例】 $x=0:0.1*\pi:2*\pi$; $y=\sin(x)$;

$back1=[0.3 \ 0.6 \ 0.4]$

figure('color',back1) %也可用**set(gcf,...)**来设置背景颜色

plot(x,y)



坐标轴范围控制函数——axis

- 基础调用形式：**axis(V)**。其中V为一个用于存储坐标轴的坐标范围的矩阵。
- 对于二维图形：**V=[xmin,xmax,ymin,ymax]**
- 对于三维图形：**V=[xmin,xmax,ymin,ymax,zmin,zmax]**

部分常用的坐标控制函数（更多内容见**doc axis**）

坐标轴控制方式、取向、范围		坐标轴的高宽比	
axis auto	使用缺省设置	axis equal	纵、横坐标等长刻度
axis manual	是当前坐标范围不变	axis image	纵、横坐标等长刻度,坐标框紧贴数据范围
axis off	不显示坐标轴	axis square	产生方形坐标系
axis on	显示坐标轴	axis normal	缺省矩形坐标系
axis ij	坐标原点在左上方		
axis xy	坐标原点在左下方		
axis (v) v=[x1, x2, y1, y2]	设定坐标范围		

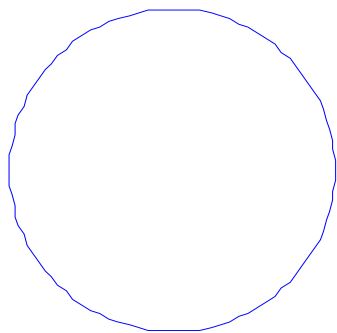
【例】 绘制椭圆，长轴为3.25，短轴为1.15

```
t=0:2*pi/99:2*pi;  
x=1.15*cos(t); y=3.25*sin(t); % y为长轴，x为短轴  
subplot(2,2,1); plot(x, y);  
axis off  
title('axis off');  
subplot(2,2,2); plot(x,y);  
axis image;  
title('axis image');  
subplot(2,2,3); plot(x,y);  
axis equal;  
title('axis Equal');  
subplot(2,2,4); plot(x,y);  
axis square;  
title('axis Square');
```

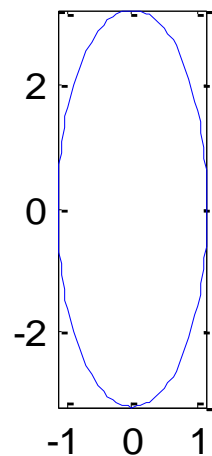
二维绘图

图形细节控制 axis

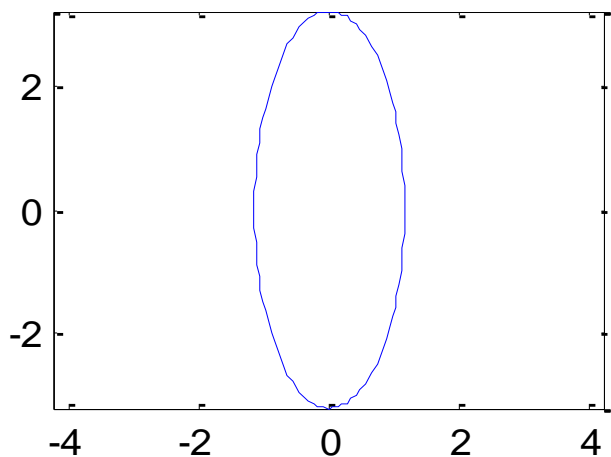
axis off



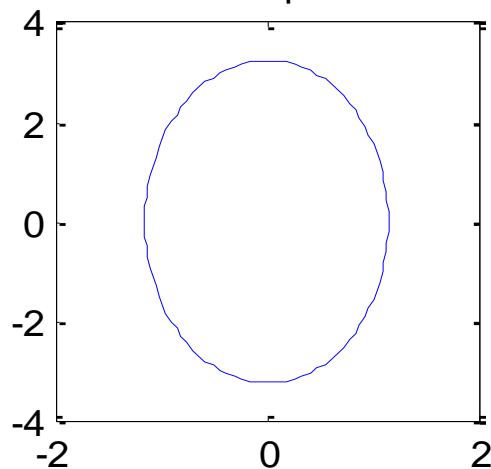
axis image



axis Equal



axis Square



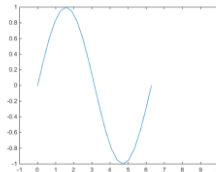
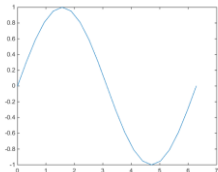
二维绘图

图形细节控制 xlim

用**xlim**, **ylim**单独设置坐标显示范围

【例】 $x=0:0.1*\pi:2*\pi$; $y=\sin(x)$;

plot(x,y); xlim([-1,10]);



坐标轴放缩函数——**zoom** **zoom+**‘控制字符串’

对图形的放缩不会影响图形的原始尺寸，也不会影响图形的横纵坐标比例。

控制字符串	说明
空	在zoom on与zoom off间切换
(factor)	以factor作为放缩因子进行坐标轴放缩
on (off)	允许（禁止）对图形进行放缩
out	恢复所进行的一切放缩
xon (yon)	只允许对x (y) 坐标轴进行放缩
reset	清除放缩点

二维绘图

图形细节控制

边框/刻度

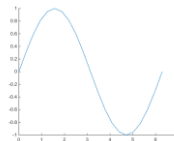
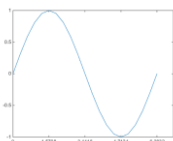
■ 坐标边框设置 **box**

- **box on** 加边框线(默认)
- **box off** 不加边框线

■ 刻度设置 **set**

- 指令及格式: **set (gca, 'xtick', xs, 'ytick', ys)**
- xs、ys可以使任何合法的实数向量, 用于分别设置x、y轴的刻度。

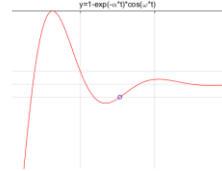
【例】 $x=0:0.1*\pi:2*\pi$; $y=\sin(x)$; **plot(x,y); box off;**
set(gca,'xtick',0: $\pi/2$: $2*\pi$, 'ytick', [-1,-0.5,0,0.5,1])



二维绘图 绘图实例

【例】 绘制 $y=1-\exp(-0.3*t).*\cos(0.7*t)$

```
t=6*pi*(0:100)/100;  
y=1-exp(-0.3*t).*cos(0.7*t);  
tt=t(find(abs(y-1)>0.05));  
ts=max(tt);  
plot(t,y,'r-');  
grid on;  
axis([0,6*pi,0.6,max(y)]);  
title('y=1-exp(-\alpha*t)*cos(\omega*t)');  
hold on;  
plot(ts,0.95,'bo');  
hold off;  
set(gca,'xtick',[2*pi,4*pi,6*pi],'ytick',[0.95,1,1.05,max(y)]);  
grid on;
```



二维绘图

多次绘图叠加

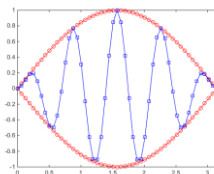
- 多次调用**plot**命令可在一幅图上绘制多条曲线，需要**hold**指令的配合。
- **hold on** 保持当前坐标轴和图形，并可以叠加下一次绘制。
- **hold off** 取消当前坐标轴和图形保持，这种状态下，调用**plot**绘制完全新的图形，不保留以前的坐标格式、曲线。
- 在不使用**hold**指令时，系统默认不叠加

二维绘图

多次绘图叠加

【例】叠绘波形 $y = \sin(t) \sin(9t)$ 及其包络线。

```
t=(0:pi/50:pi)';    %长度为101的时间采样列向量  
y1=sin(t)*[1,-1];  %包络线函数值，是（101x2）的矩阵  
y2=sin(t).*sin(9*t);    %长度为101的调制波列向量  
plot(t,y1,'r-o')  
hold on  
plot(t,y2,'b-s')  
axis([0,pi,-1,1])  
hold off
```



二维绘图

双纵坐标 **plotyy**

■ **plotyy(x1, y1, x2, y2)**

x1-y1曲线y轴在左， **x2-y2**曲线y轴在右。

【例】

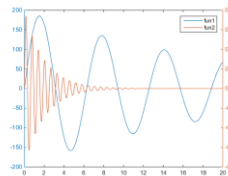
```
x=0:0.01:20;
```

```
y1=200*exp(-0.05*x).*sin(x);
```

```
y2=0.8*exp(-0.5*x).*sin(10*x);
```

```
plotyy(x,y1,x,y2);
```

```
legend('fun1','fun2');
```



二维绘图

多子图 subplot

可在同一图形窗口布置几幅独立的子图(一窗多图)

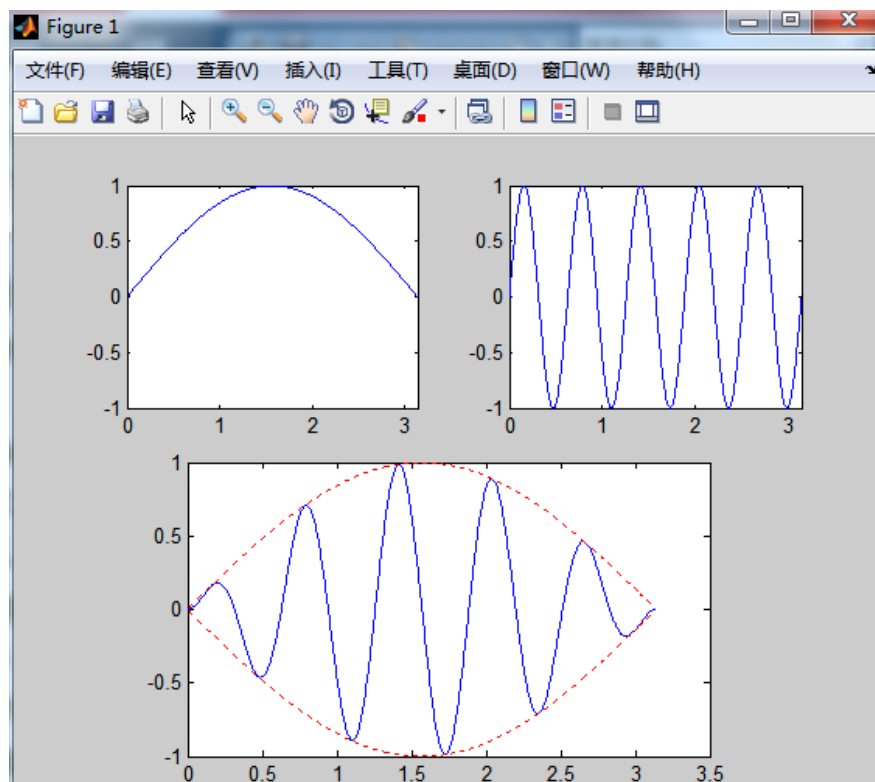
- `subplot(m, n, k)` 使 $m*n$ 幅子图中第 k 个子图成为当前图
- `subplot('postion', [left, bottom, width, height])` 在指定位置开辟子图，并成为当前图
- `subplot(m, n, k)`的含义：图形窗口包含 $(m*n)$ 个子图， k 为要指定的当前子图的编号。**编号原则**：左上方为第1子图，然后向右向下依次排序。该指令按缺省值分割子图区域。
- `subplot('postion', [left, bottom, width, height])`用于手工指定子图位置，指定位置的四元组采用归一化的标称单位，即认为整个图形窗口绘图区域的高、宽的取值范围都是 $[0, 1]$ ，而左下角为 $(0,0)$ 坐标。
- 产生的子图彼此独立。所有的绘图指令均可以在子图中使用

二维绘图

多子图 subplot

```
【例】 t=(pi*(0:1000)/1000)';  
y1=sin(t); y2=sin(10*t); y12=sin(t).*sin(10*t);  
subplot(2,2,1);    plot(t,y1);    axis([0,pi,-1,1])  
subplot(2,2,2);    plot(t,y2);    axis([0,pi,-1,1])  
subplot('position',[0.2,0.05,0.6,0.45]);  
plot(t,y12,'b-',t,[y1,-y1],'r:');
```

子图的线形等细节
可以分别设置



二维绘图 特殊坐标系

- 特殊坐标系的二维图形函数

semilogx: x坐标为对数坐标的二维图

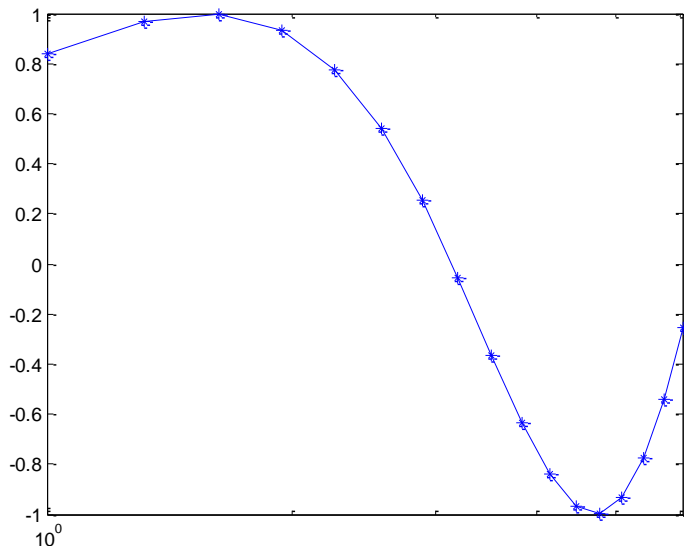
semilogy: y坐标为对数坐标的二维图

loglog: 双对数坐标二维图

polar: 极坐标二维图（弧度）

$x=1:0.1*\pi:2*\pi;$ $y=\sin(x);$

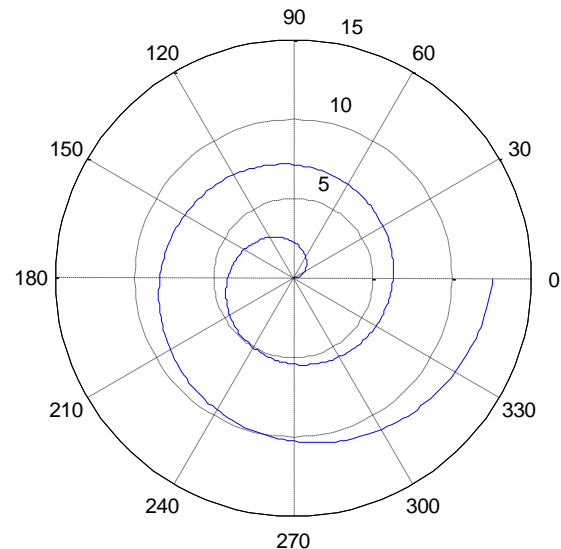
semilogx(x,y,'-*')



$x=0:0.01*\pi:4*\pi;$

$y=\sin(x/2)+x;$

polar(x,y,'-')



二维绘图 特殊图形

■ 二维特殊图形

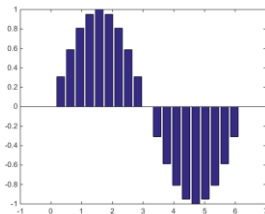
函数名	说明	函数名	说明
area	填充绘图	feather	羽状图
bar	条形图	fill	多边形填充
barh	水平条形图	fplot	函数图
comet	彗星图	hist	直方图
errorbar	误差带图	pie	饼状图
ezplot	简单绘制函数图	contour	等高线图
stairs	阶梯图	stem	离散杆图
quiver	矢量图		

二维绘图 特殊图形

【例】绘制条形图

```
x=0:0.1*pi:2*pi; y=sin(x);
```

```
bar(x,y)
```

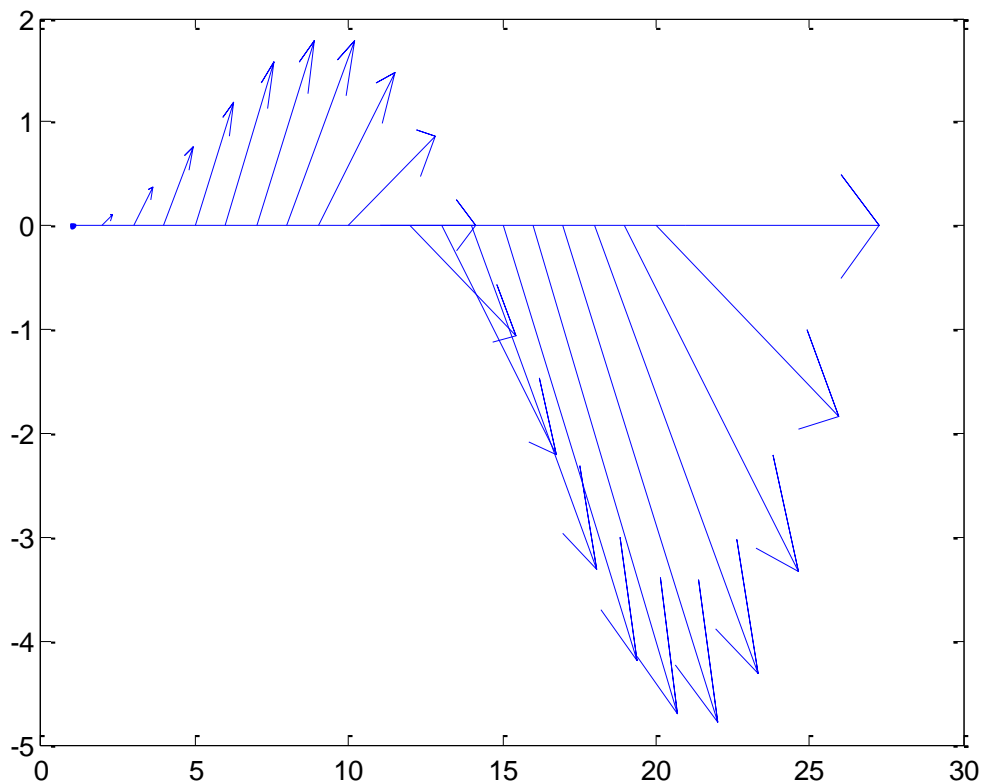


二维绘图 特殊图形

【例】绘制羽状图

$x=0:0.1*\pi:2*\pi$, $y=\sin(x).*x$;

feather(x,y)

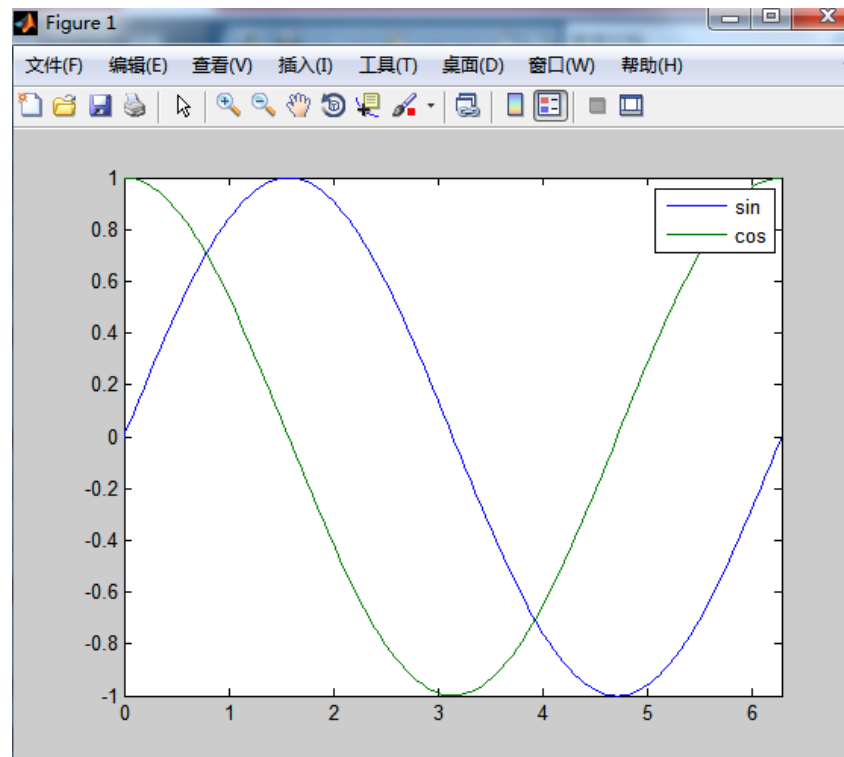


二维绘图 特殊图形

【例】绘制函数图形（只需给出坐标表达式和区间，不需预先计算出坐标值）

`lim=[0,2*pi,-1,1];`

`fplot('sin(x),cos(x)',lim); legend('sin','cos');`



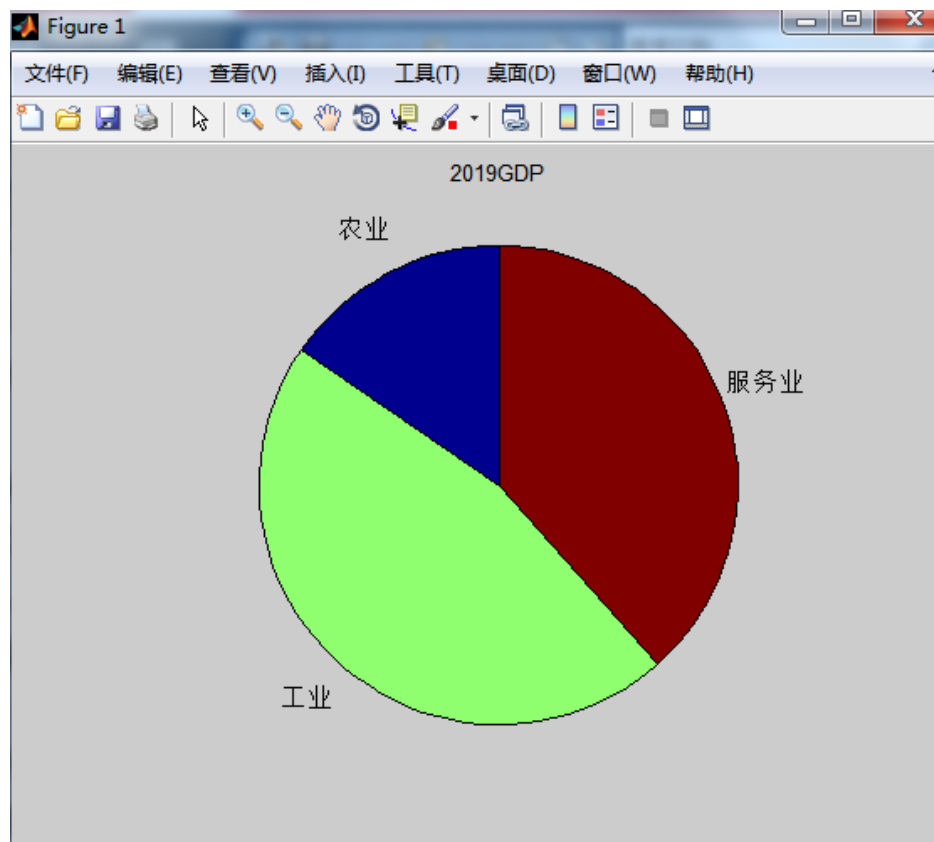
二维绘图 特殊图形

【例】绘制饼状图

`x=[1,3,2.5];`

`pie(x,{'农业', '工业', '服务业'})`

`title('2019GDP')`



二维绘图 特殊图形

【例】 **stem**和**stairs**绘制离散杆图和阶梯图

```
t=2*pi*(0:20)/20;
```

```
y=cos(t).*exp(-0.4*t);
```

```
stem(t,y,'g'); hold on; %绿色离散杆图
```

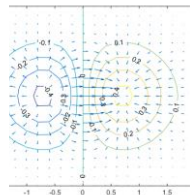
```
stairs(t,y,'r'); hold off; %红色阶梯图
```

45

二维绘图 特殊图形

【例】 用 **contour** 和 **quiver** 绘制二维等高线图和矢量图

```
[x,y]=meshgrid(-2:0.2:2);  
z=x.*exp(-x.^2-y.^2);  
[dx,dy]=gradient(z,0.2,0.2);  
contour(x,y,z,'ShowText','on');  
hold on  
quiver(x,y,dx,dy);
```





三维数据绘图

三维绘图 三维曲线plot3

plot3用来绘制三维曲线，基本调用格式同plot类似

plot3(x,y,z)

- x,y,z为同维向量时，绘出以x,y,z为坐标点的曲线
- x,y,z为同维矩阵时，例如m*n矩阵，则将其中n个列向量取出依次绘制n条曲线

plot3(x1, y1, z1, 's', x2, y2, z2, 's' ...)

- 可以依次输入多组曲线坐标，并用`s`指定曲线的颜色、线型等，同plot相似

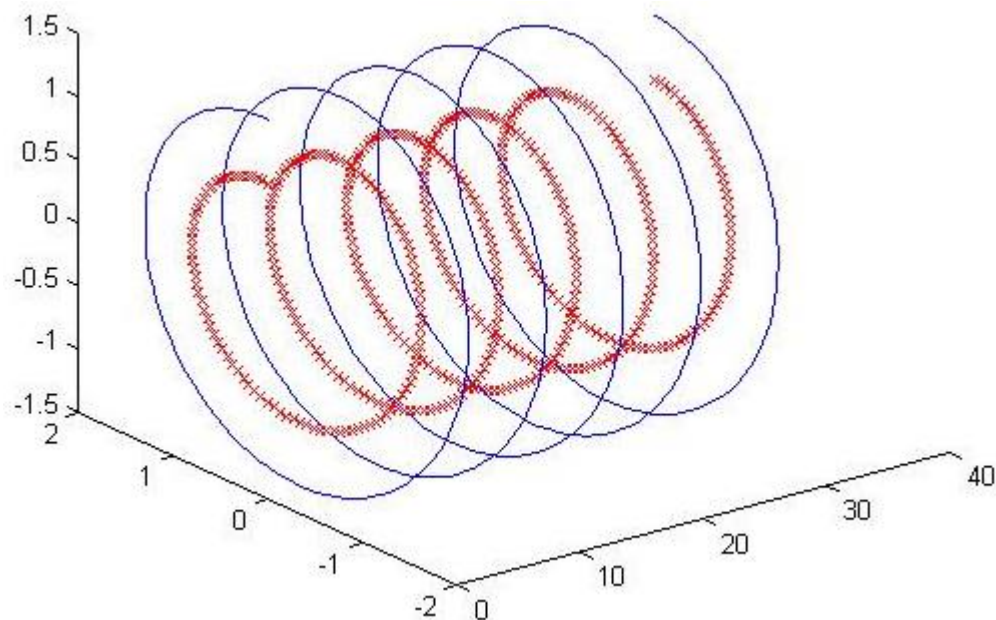
plot3(x, y, z, 'propertyname', propertyvalue)

- 也可以用 属性名+属性值 的方式定义曲线显示方式的相关属性

三维绘图 三维曲线plot3

【例】

```
x1=0:pi/50:10*pi;  
y1=sin(x1); z1=cos(x1);  
x2=x1; y2=y1*1.5;  
z2=z1*1.5;  
plot3(x1,y1,z1,'rx',x2,y2,z2,'b');
```



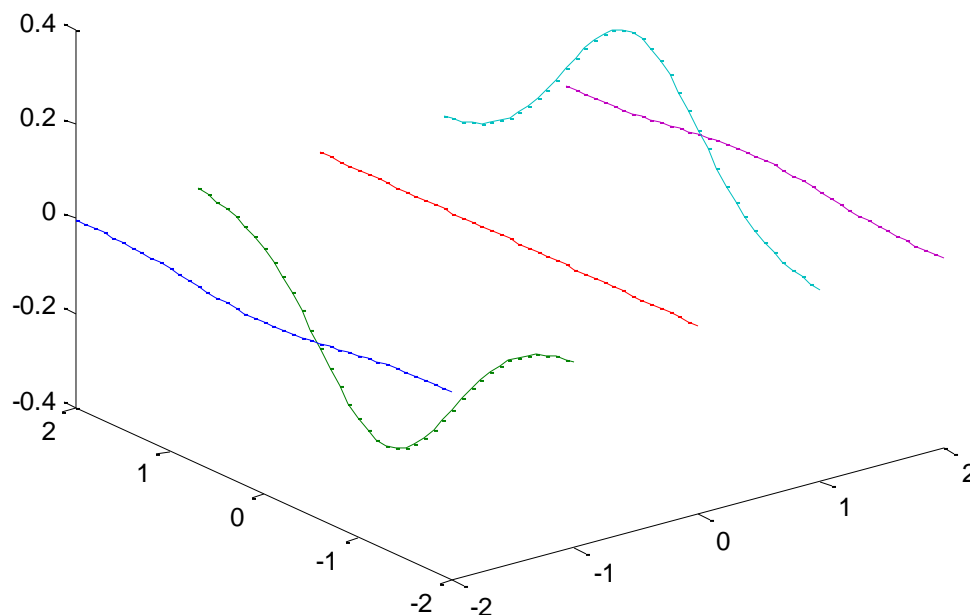
三维绘图 三维曲线plot3

【例】

```
[x,y]=meshgrid(-2:1:2,-2:0.1:2); %5*41的经纬线
```

```
z=x.*exp(-x.^2-y.^2);
```

```
plot3(x,y,z) %5条线，每条对应z的一列，z是41x5矩阵
```



三维绘图 三维曲面的绘图

- **mesh(x,y,z)** 绘制三维面的网格图，还有**meshc**等变种。如果缺省x，y则相当于用z的下标作为x，y。
- **surf(x,y,z)** 绘制三维着色面图，还有**surfc**等变种。
- **[X,Y]=meshgrid(x,y)** 划分平面网格。x，y为给定的向量，矩阵X，Y是网格划分后的数据矩阵。常配合**mesh**等使用。

三维绘图 三维曲面的绘图

绘制函数 $z=f(x,y)$ 代表的三维曲面，**要做以下准备工作：**

- 确定自变量的取值范围和取值间隔。

`x=x1:dx:x2;`

`y=y1:dy:y2;`

- 构成 x - y 平面上的自变量采样“格点”矩阵，形成所谓“网格”。可利用指令`meshgrid`产生网格。

`[xa, ya]=meshgrid(x,y);`

- 计算函数在自变量采样“格点”上的函数值

三维绘图 三维曲面的绘图

【例】绘制函数 $z=x^2+y^2$ 的曲面

$x=-4:4$; $y=x$;

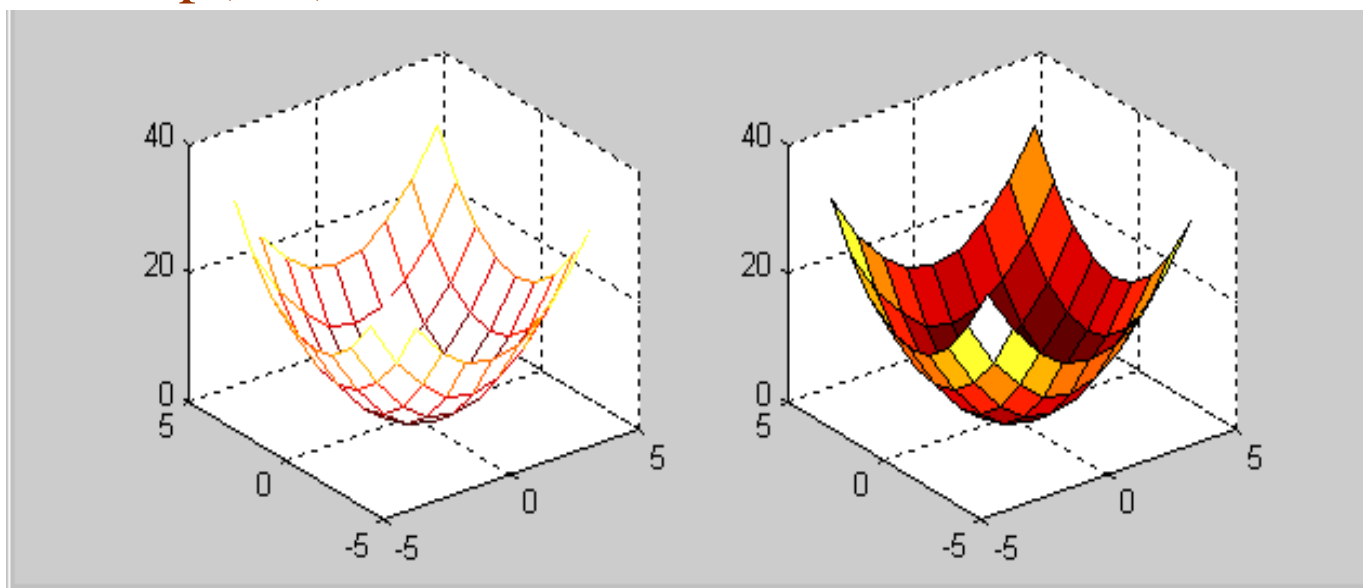
$[x,y]=\text{meshgrid}(x,y)$; %生成 x-y 坐标“格点”矩阵

$z=x.^2+y.^2$; %计算格点上的函数值

$\text{subplot}(1,2,1), \text{mesh}(x,y,z)$; %三维网格图

$\text{subplot}(1,2,2), \text{surf}(x,y,z)$; %三维曲面图

$\text{colormap}(\text{hot})$;



三维绘图

三维曲面的绘图

【例】

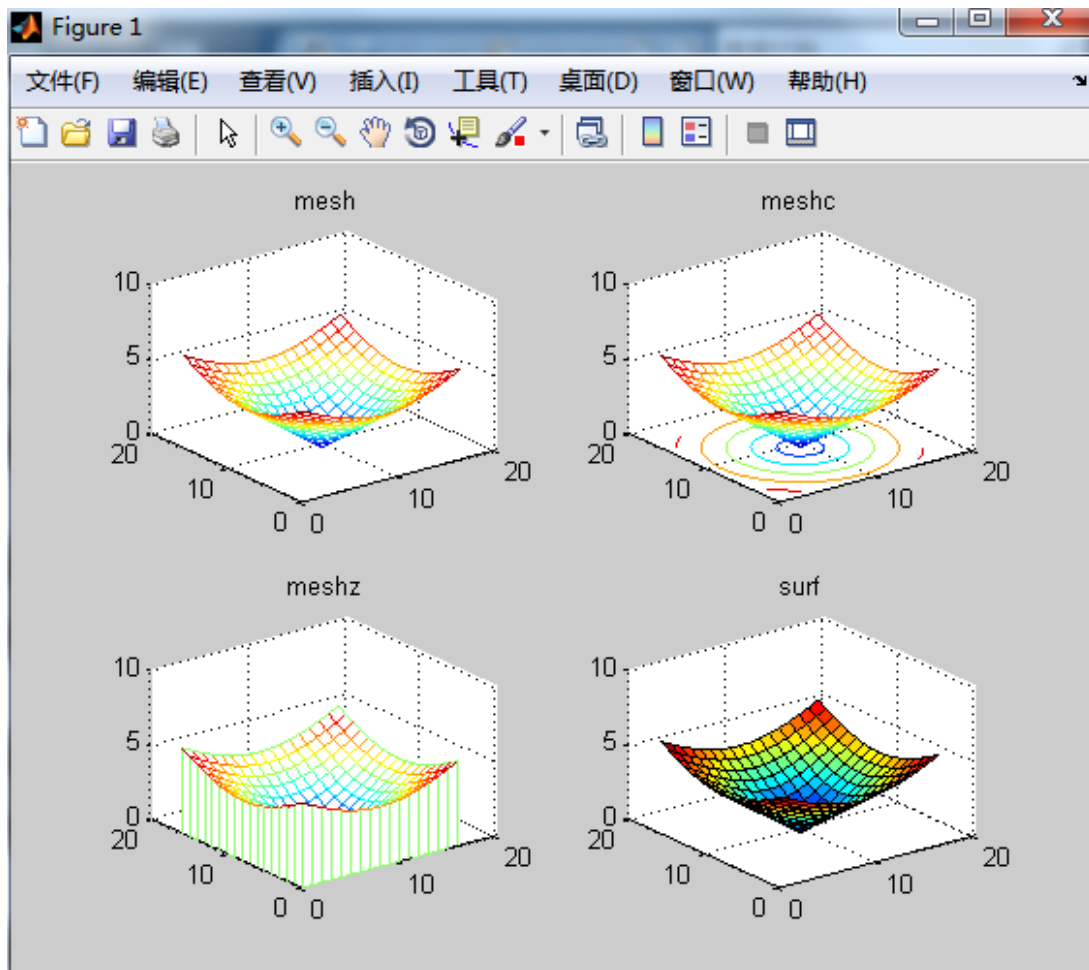
```
[X,Y]=meshgrid([-4:0.5:4]);  
z=sqrt(X.^2+Y.^2);
```

```
subplot(2,2,1)  
mesh(z);title('mesh');
```

```
subplot(2,2,2)  
meshc(z);title('meshc');
```

```
subplot(2,2,3)  
meshz(z);title('meshz');
```

```
subplot(2,2,4)  
surf(z);title('surf');
```

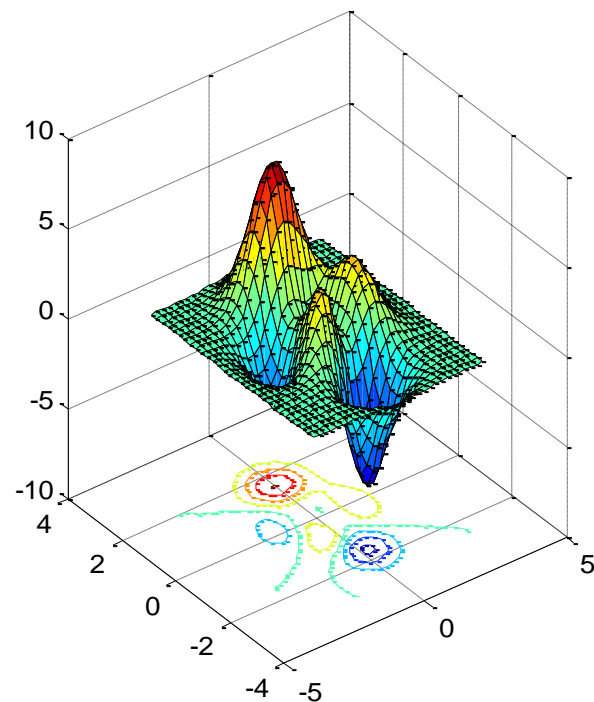
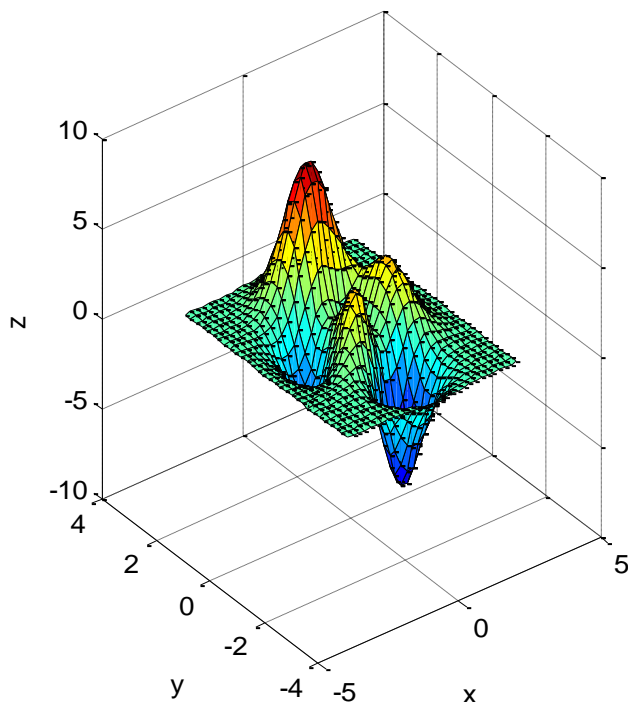


三维绘图

三维曲面的绘图

【例】

```
[x,y,z]=peaks(30);  
subplot(1,2,1);  
title('surf exam');  
surf(x,y,z);  
xlabel('x');ylabel('y');zlabel('z');  
subplot(1,2,2);  
surfc(x,y,z);
```

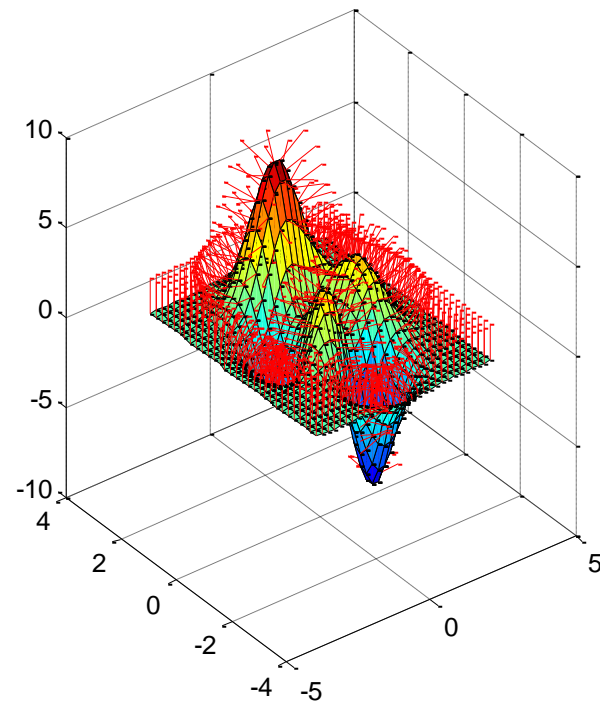
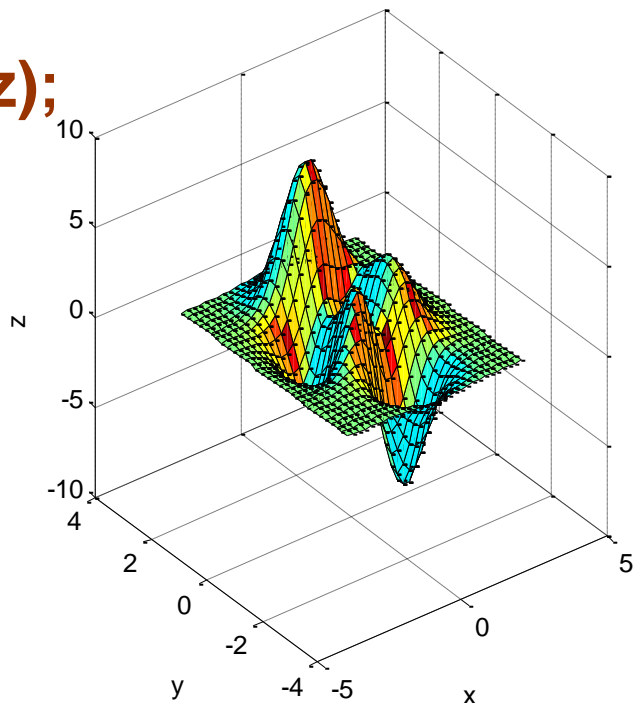


三维绘图

三维曲面的绘图

【例】

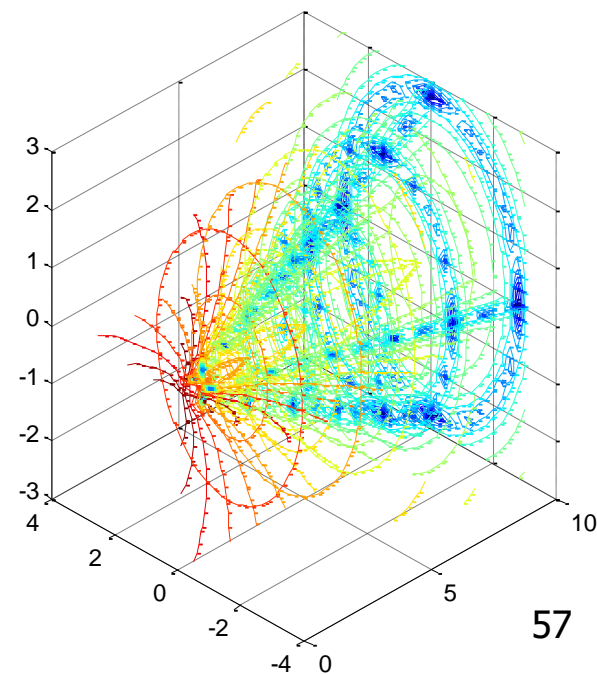
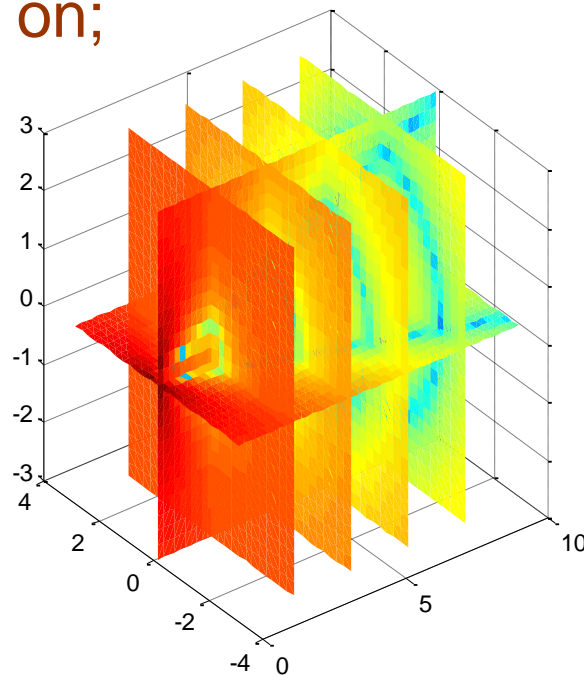
```
[x,y,z]=peaks(30);  
subplot(1,2,1);  
title('surf exam');  
surf(x,y,z);  
xlabel('x');ylabel('y');zlabel('z');  
subplot(1,2,2);  
surfnorm(x,y,z);
```



三维绘图 准四维绘图 slice, contourslice

```
[x,y,z,v]=flow;  sx=2:2:10  
slice(x,y,z,v,sx,0,0); shading interp;
```

```
subplot(1,2,2);  
vmin=min(min(min(v))); vmax=max(max(max(v)));  
sv=linspace(vmin+1,vmax-1,20);  
contourslice(x,y,z,v,sx,0,0,sv);  
view([-45,30]); grid on;
```



三维绘图

特殊的三维绘图函数

函数名	说明	函数名	说明
bar3	三维条形图	quiver3	三维矢量场
comet3	三维彗星轨迹图	trisurf	三角形表面图
ezgraph3	函数控制绘制三维图	trimesh	三角形网格图
pie3	三维饼状图	waterfall	瀑布图
stem3	三维离散数据图	scatter3	三维散射图
cylinder	柱面图	contour3	三维等高图
sphere	球面图	cplxmap	复数变量绘图

补充： 图像文件的处理

- **imread**

读取图像文件（ BMP, GIF , PNG, JPEG, and TIFF）

- **imshow**

显示图像

- **imwrite**

保存图像

例： 读取图像文件

```
img1=imread('disney.jpg');    % 读入图片
```

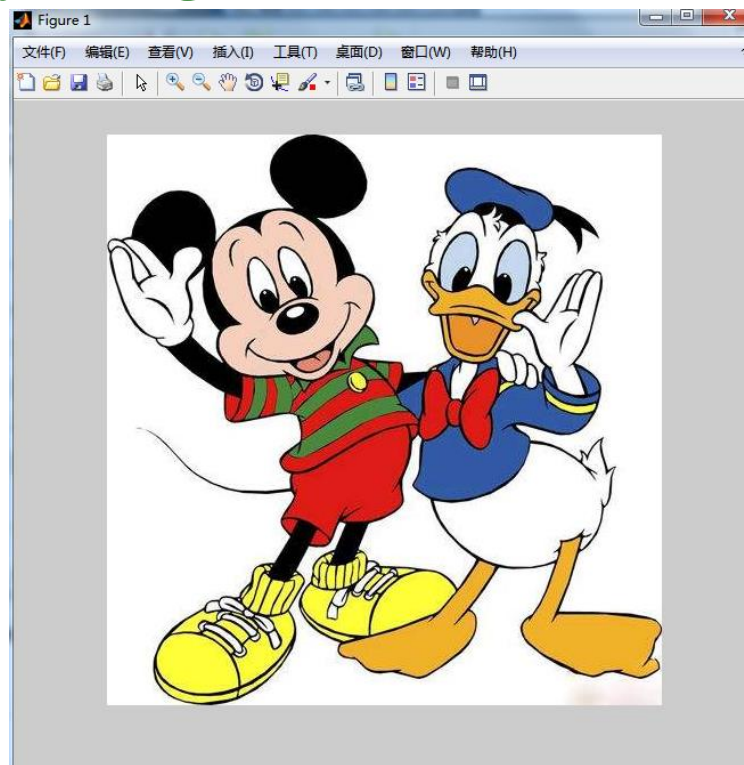
```
whos img1
```

补充： 图像文件的处理

Name	Size	Bytes	Class	Attributes
img1	500x487x3	730500	uint8	

显示图像：

```
imshow(img1); % Display image
```



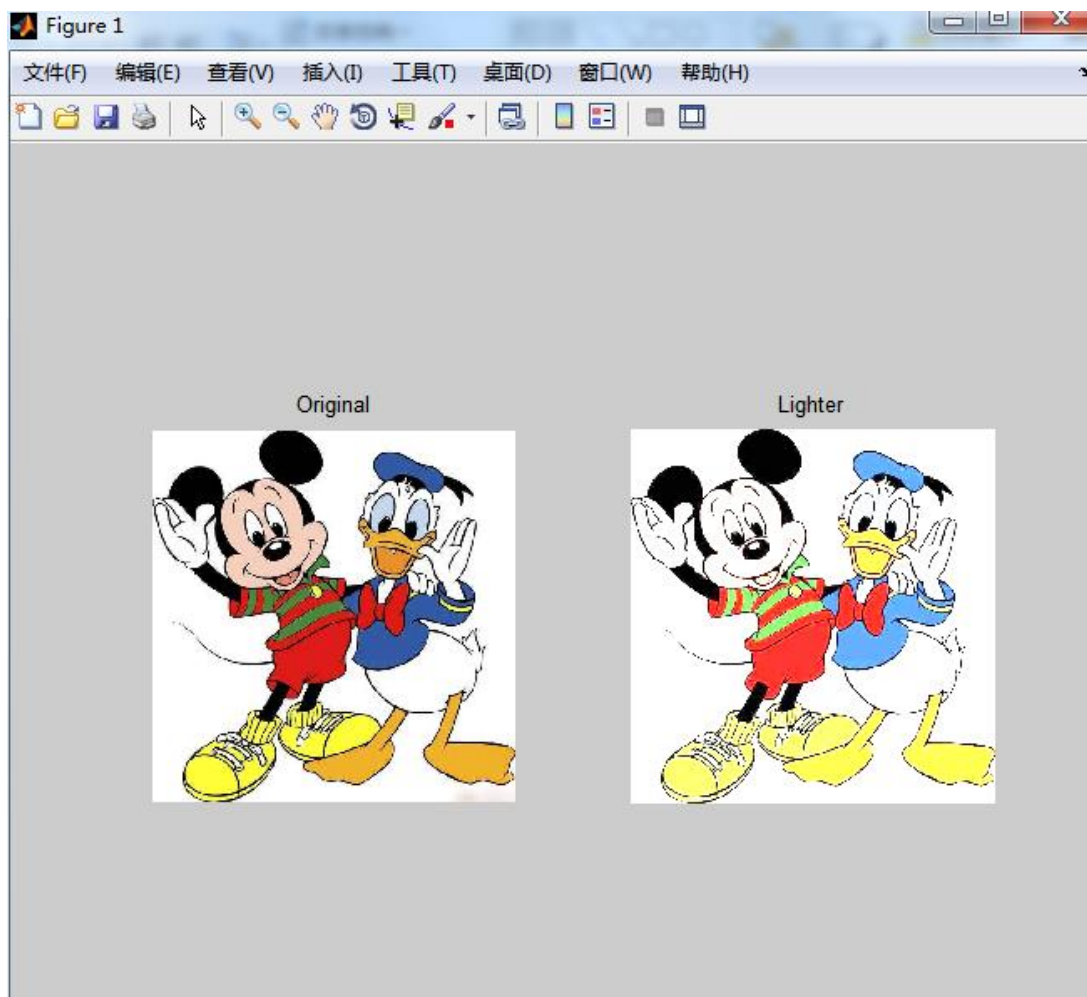
补充： 图像文件的处理

■ 简单图像处理

```
lighter = 2 * img1;  
subplot(1,2,1);  
imshow(img1); % 显示原始图片  
title('Original');  
subplot(1,2,2);  
imshow(lighter); % 显示增亮后的图片  
title('Lighter');
```

补充： 图像文件的处理

图像处理前后的比较



补充： 图像文件的处理

- 保存图像

imwrite(lighter, 'mysaved.jpg')

- 查看保存结果

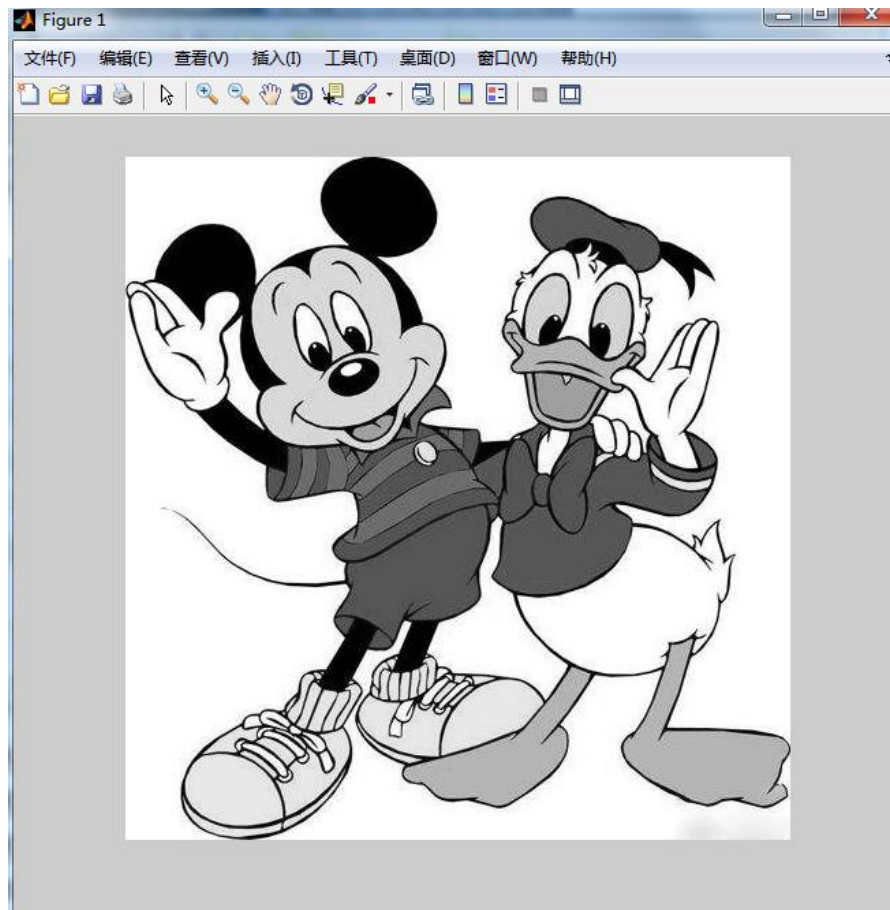
dir mysaved.*

mysaved.jpg

补充： 图像文件的处理

彩色图像转换为灰度图像

```
img2 = rgb2gray(img1);  
imshow(img2)
```



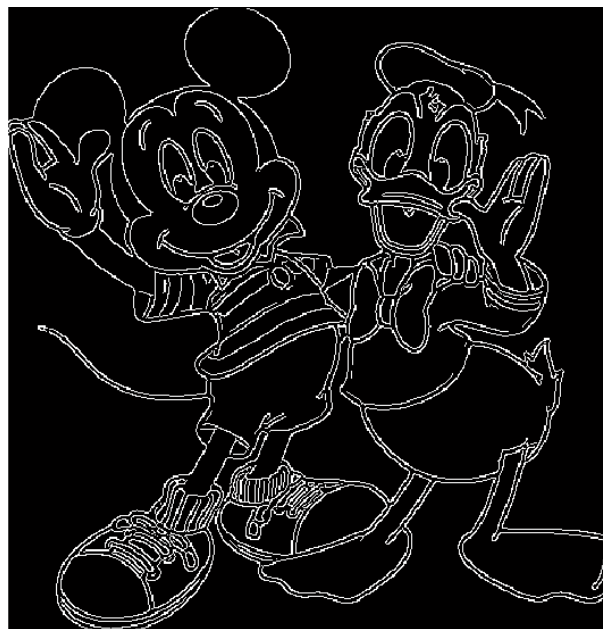
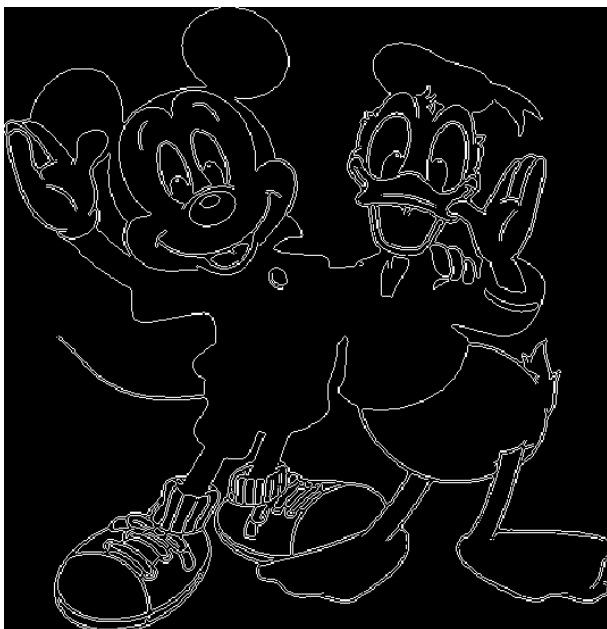
补充： 图像文件的处理

图像特征提取

```
imag_edge1 = edge(img2, 'sobel');  
subplot(121),imshow(imag_edge1)  
imag_edge2 = edge(img2, 'canny');  
subplot(122),imshow(imag_edge2)
```

%sobel边缘提取算法

%canny边缘提取算法



附录：二维特殊图形更多示例 直方图

■ 人员统计的直方图

	1990年	1995年	2000年
第一产业	90.7	70.6	73.9 (万人)
第二产业	281.6	271	214.6
第三产业	254.8	323.7	326.5

执行以下语句：

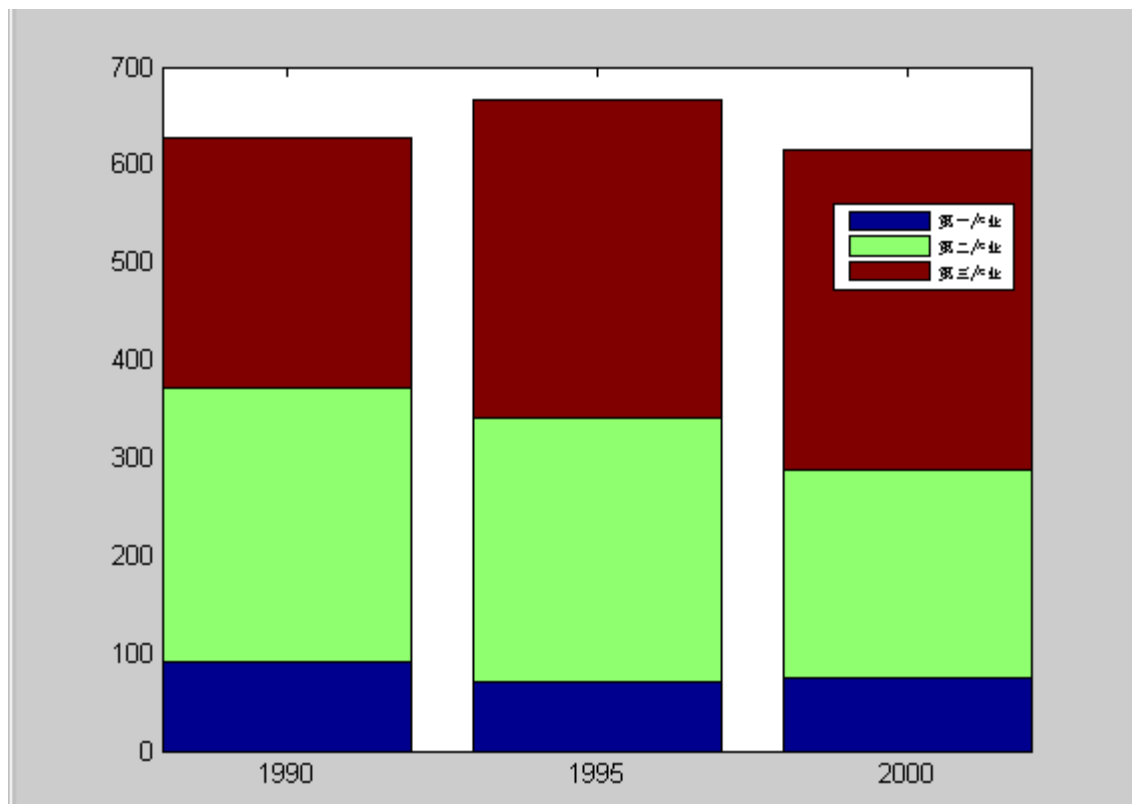
```
year=[1990 1995 2000];people=[90.7 281.6 254.8; 70.6 271  
323.7; 73.9 214.6 326.5];
```

```
bar(year, people, 'stack');
```

```
legend('\fontsize{6}第一产业', '\fontsize{6}第二产业',  
'\fontsize{6}第三产业');
```

附录： 二维特殊图形更多示例 直方图

累计式直方图（垂直型）

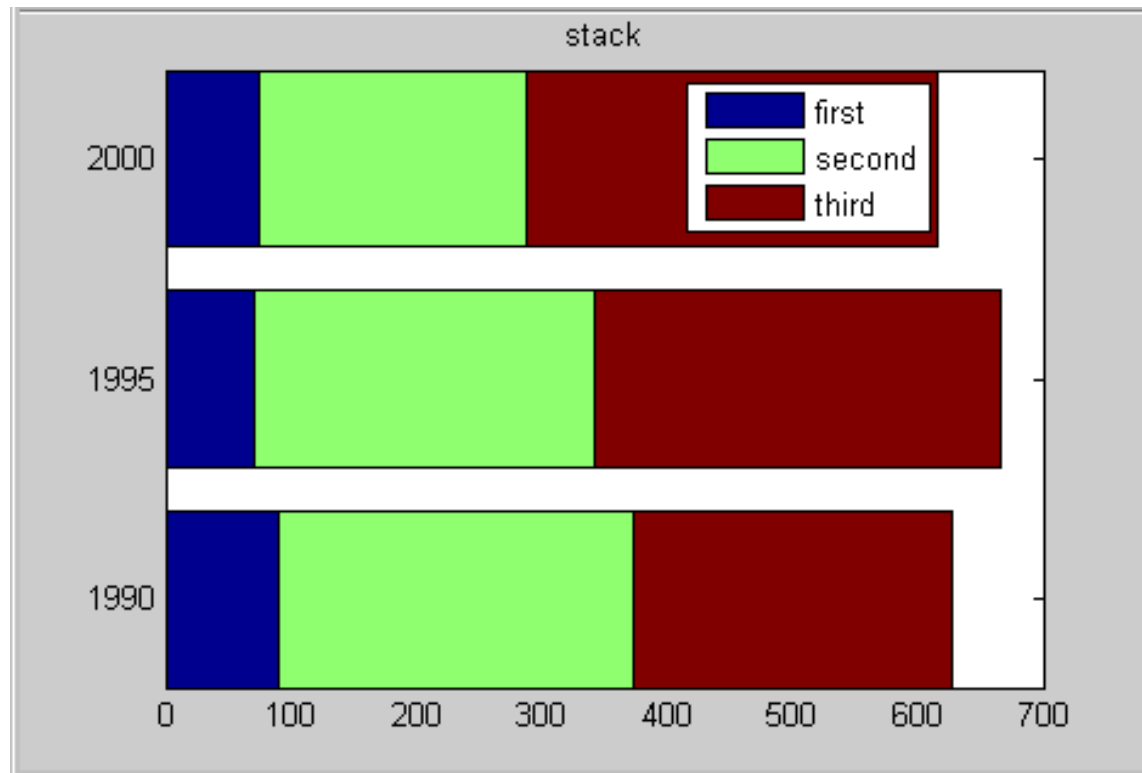


附录： 二维特殊图形更多示例 直方图

```
barh(year, people, 'stack');
```

```
legend('\fontsize{6} first', '\fontsize{6}second', '\fontsize{6}third');
```

累计式直方图（水平型）

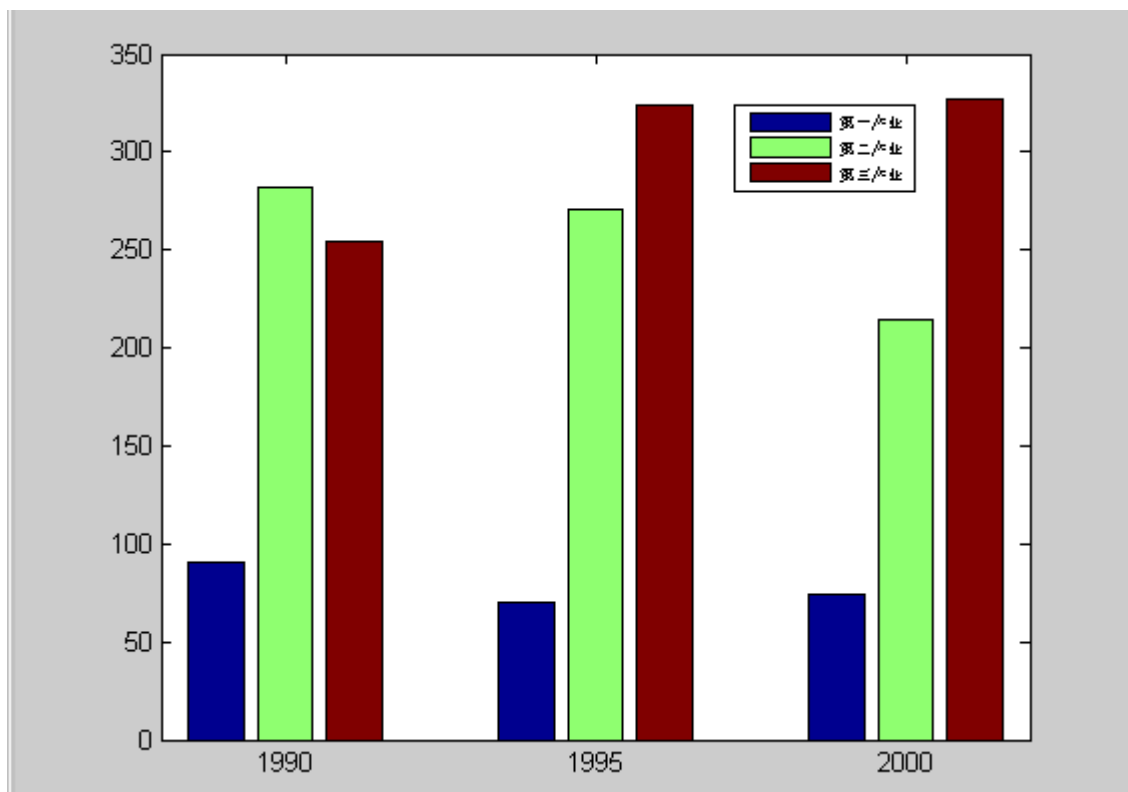


附录： 二维特殊图形更多示例 直方图

```
bar(year, people, 'group');
```

```
legend('\fontsize{6}第一产业' , '\fontsize{6}第二产业' , '\fontsize{6}第三产业');
```

分组式直方图（垂直型）

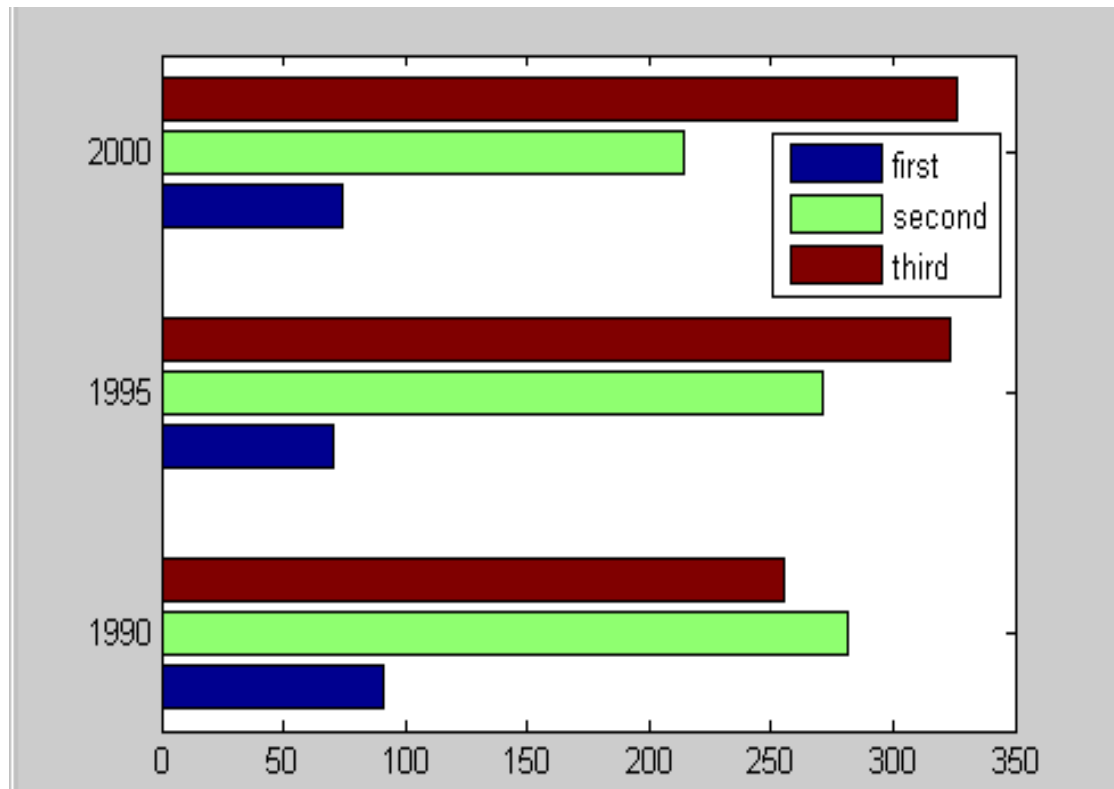


附录： 二维特殊图形更多示例 直方图

`barh(year, people, 'group');`

`legend('\fontsize{6}first', '\fontsize{6}second', '\fontsize{6}third');`

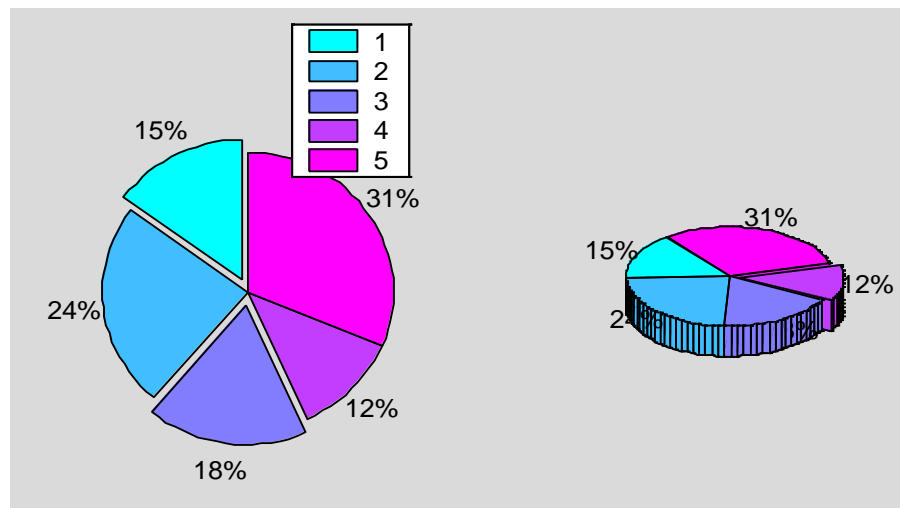
分组式直方图（水平型）



附录： 二维特殊图形更多示例 饼状图

- 饼状图 用来表示各元素占总和的百分数

```
a=[1,1.6,1.2,0.8,2.1];  
subplot(1,2,1),pie(a,[1 0 1 0 0]),  
legend({'1','2','3','4','5'})  
subplot(1,2,2), b=int8(a==min(a))  
pie3(a,b)  
colormap(cool)
```



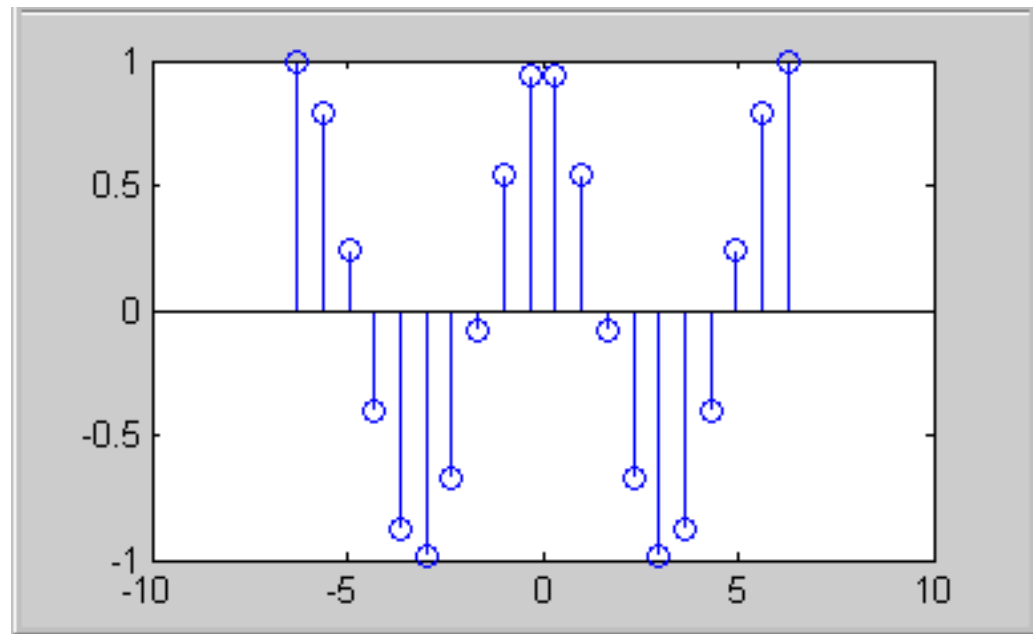
a中各元素求和得到总量，按照各元素占总量的百分比画出对应扇形构成整个圆饼，b中元素为1则将对对应扇形外推以示醒目。

附录： 二维特殊图形更多示例 离散杆图

余弦波的采样信号图

```
t = linspace(-2*pi,2*pi,20);
```

```
h = stem(t,cos(t));
```



附录： 二维特殊图形更多示例

条形图、填充图、阶梯图、离散杆图

```
x = 0:0.35:7;
```

```
y = 2*exp(-0.5*x);
```

```
subplot(221);bar(x,y,'g');
```

```
title('bar(x,y,\'g\')');axis([0,7,0,2]);
```

```
subplot(222);fill(x,y,'r');
```

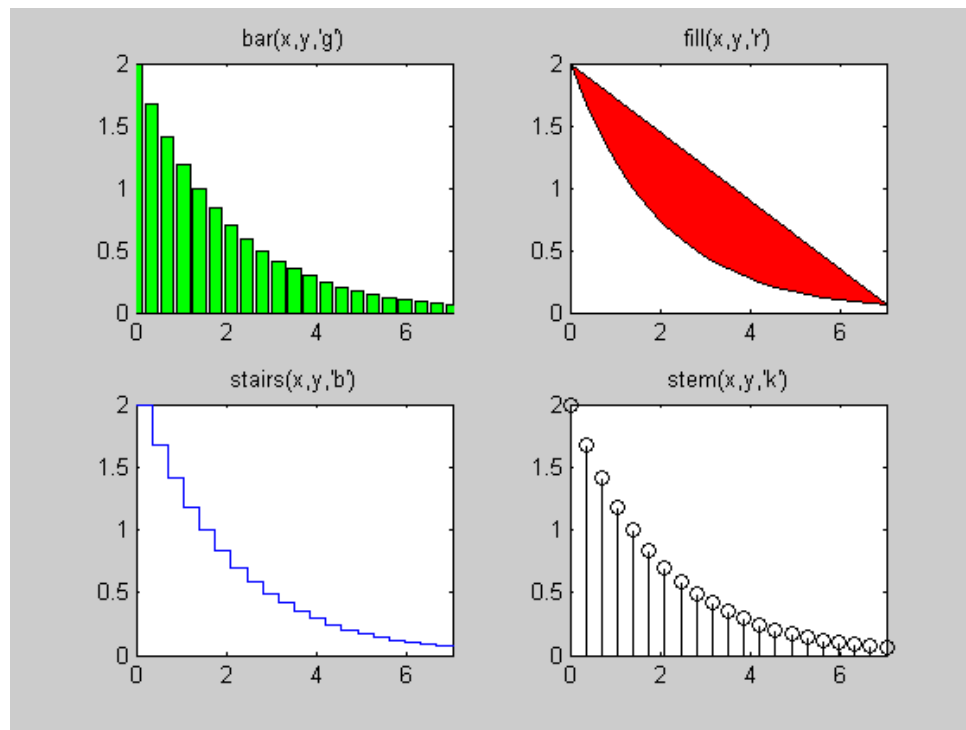
```
title('fill(x,y,\'r\')');axis([0,7,0,2]);
```

```
subplot(223);stairs(x,y,'b');
```

```
title('stairs(x,y,\'b\')');axis([0,7,0,2]);
```

```
subplot(224);stem(x,y,'k');
```

```
title('stem(x,y,\'k\')');axis([0,7,0,2]);
```



本节结束
谢谢