

第 3 章

MATLAB 基本绘图功能

3.1 二 维 图 形

任何一个二元标量 (x, y) 都可以用平面上的一点表示；同样，一组向量对 (x, y) 可用一组点表示。在坐标系中表现向量点，就是离散数据的可视化。对函数进行区间量化处理，使之成为向量，这表现的是离散数据。连续可视化场有两种方法：一是使区间更细化，以生成更多点来表现函数的连续变化；二是将相邻两点间线性相连，以近似表现函数的变化。应注意的一点就是，采样点需要足够多，才能够比较真实地反映原函数的连续性变化。下面主要针对 MATLAB 函数讲解可视化的方法。

3.1.1

基本的二维绘图函数

在二维绘图函数中，最重要、最基本的函数是 plot 函数，其他函数基本上都是以它为基础的。表 3-1 列出了 MATLAB 里绘制基本二维曲线图形的函数。

表 3-1 基本二维曲线绘制函数

函 数	说 明
plot	在二维坐标系里绘制线性图形
loglog	在对数坐标系里绘制图形
semilogx	在 x 对数比例坐标系里绘制图形
semilogy	在 y 对数比例坐标系里绘制图形
plotyy	绘制双 y 轴图形

plot 函数的调用格式如下：

```
plot(Y)
plot(X1,Y1,...)
plot(X1,Y1,LineSpec,...)
plot(...,'PropertyName'PropertyValue,...)
h=plot(...)
```

参数说明：

plot(Y)中，若 Y 为实数类型，则绘制以 Y 为下标，以 Y 值为坐标的相连折线；如果 Y 为虚数，则绘制连接(real(Y),imag(Y))的折线。

`plot(x,y)`绘制的是以 (x,y) 为坐标相连的折线。 x 和 y 主要有下面3种情况:

- x 和 y 都为向量时, x 和 y 的长度相同;
- x 和 y 都为数组(非向量)时, x 和 y 的大小必须相同;
- x 和 y 中,一个为向量(行或列向量),一个为数组(非向量)时,行向量或列向量的长度应该和数组的行长度(列长度)相同。

如果 $[x,y]$ 是闭合的数组,则用函数`plot()`可以绘制多边形,多边形的面积可以由函数`polyarea()`求得。

参数`LineStyle`用来设定图形的线形、标记符号、颜色等,3.1.2节将详细说明。

`plot(...,'PropertyName',PropertyValue,...)`可设置由`plot`函数创建`line`图形对象时的属性值。

`h=plot(...)`返回图形对象句柄。

【例 3-1】 `plot` 函数应用实例。

```
x = 0:0.05:5;           % 长度 101 的 x 轴采样点
y = sin(x.^2);
plot(x,y);              % 绘制一条曲线,如图 3-1 所示
y1 = y - 0.25;
y2 = y - 0.5;
plot(x,y,xy1,x,y2);    % 绘制一条曲线,plot 会自动定义不同曲线的颜色,如图 3-2 所示
```

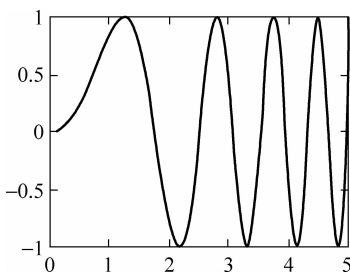


图 3-1 `plot` 函数绘制一条曲线

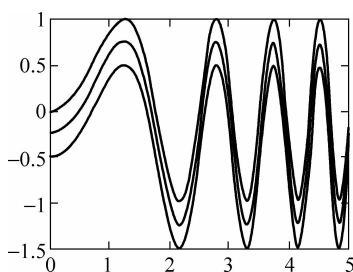


图 3-2 `plot` 函数绘制一组曲线

上面给出了`plot`函数的简单用法说明。表 3-1 中其他函数的具体用法和`plot`函数类似,这里不再叙述。

比较特殊的是双 y 轴绘制函数`plotyy()`。如`plotyy(X1,Y1,X2,Y2)`表示分别以左 Y 轴、右 Y 轴和 X 轴画 $(X1,Y1)$ 和 $(X2,Y2)$ 的图形。

【例 3-2】 `plotyy` 函数应用实例(如图 3-3 所示)。

```
x = 0:0.01:20;
y1 = 200 * exp(-0.05 * x) * sin(x);
y2 = 0.8 * exp(-0.5 * x) * sin(10 * x);
[AX,H1,H2] = plotyy(x,y1,x,y2,'plot');           % 绘制数据图形并返回坐标轴和图形句柄
set(get(AX(1),'Ylabel'),'String','Left Y-axis') % 标注左边的 Y 轴,句柄对象中会讲解 set 函
                                                    数和 get 函数的用法
set(get(AX(2),'Ylabel'),'String','Right Y-axis') % 标注右边的 Y 轴
```

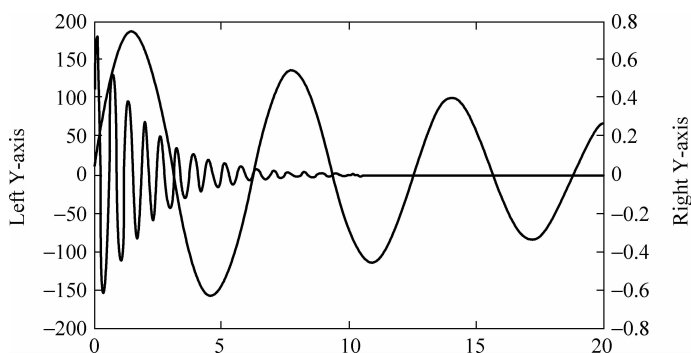


图 3-3 plotyy 函数应用实例

【例 3-3】 多曲线的绘制(如图 3-4 所示)。

```
subplot(211)           % 窗体分割显示轴,选定第一轴
Z = peaks;              % peaks 产生一个 49 × 49 大小的数组,函数为 MATLAB 系统三维绘图典型示例
plot(Z)                % 绘制 49 条曲线,并以不同颜色的曲线显示
subplot(212)
semilogx(1:100, '+' ') % "+" 定义了点型,3.1.3 节将详细说明
hold all               % 在图形上增绘图行,同时新增线型和线形
plot(1:3:300, 1:100, '- - ')
hold off
grid on                % 打开网格线
```

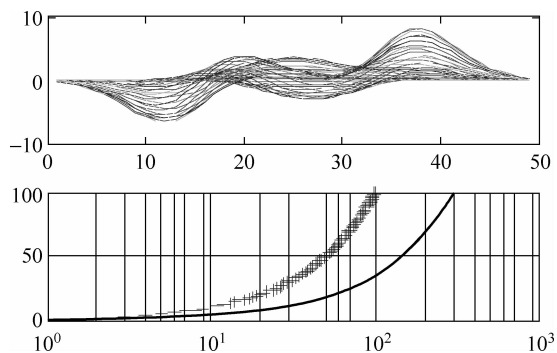


图 3-4 多曲线的绘制

3.1.2

线型、点型、色彩

MATLAB 里针对绘图的命令定义了一些色彩、线型和数据点型(如表 3-2 和表 3-3 所示)。这些参数对应 plot 等函数的 LineSpec 参量的值。

表 3-2 线型、色彩符号

线型符号	含 义	色彩符号	含 义	色彩符号	含 义
—	实线	b	蓝色	m	品红色
--	双划线	g	绿色	y	黄色
:	虚线	r	红色	k	黑色
-. .	点划线	c	青色	w	白色

表 3-3 数据点型符号

符号	含 义	符号	含 义	符号	含 义
+	“+”字符	^	上三角符	s	正方符
o	空心圆	v	下三角符	d	菱形符
*	星号	<	左三角符	h	六星符
.	实心圆	>	右三角符		
x	叉符	p	五星符		

【例 3-4】 曲线色彩与线型、数据点型示例(如图 3-5 所示)。

```
t=0:pi/20:2*pi;
y=sin(t);
y1=sin(t-0.25);
y2=sin(t-0.5);
y3=sin(t-0.75);
hold on
plot(t,y)           % 使用默认曲线色彩和线型,没有点型
plot(t,y1,':k')     % 定义曲线色彩为黑色,线型为虚线,没有定义点型
plot(t,y2,'om')     % 定义曲线色彩为品红色,点型为空心圆,没有绘制出点相连折线(线形),这
                    % 是表现离散数据可视化的方法
plot(t,y3,'- .gp')  % 定义曲线色彩为绿色,点型为五星符,相连的折线为点划线
hold off
```

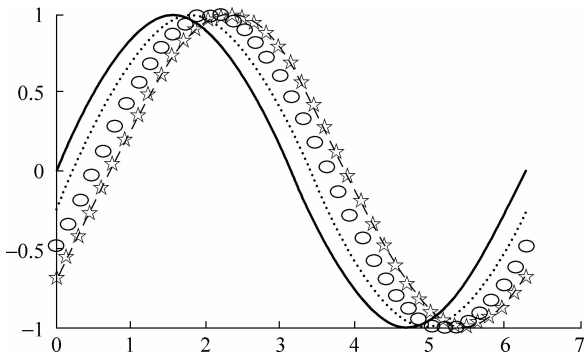


图 3-5 曲线色彩与线型、数据点型示例

在 MATLAB 中,也可以对上述符号的显示样式进行控制,如线宽、点型大小等。这主要通过设置属性来定义,对于 plot 函数,即用 plot(..., 'PropertyName', PropertyValue...) 设置样式。表 3-4 中列出了这些属性。

表 3-4 符号样式控制属性

参 数	含 义
LineWidth	线宽
MarkerEdgeColor	点型边色(空心圆、正方符、菱形符、五星符、六星符及 4 种三角符)
MarkerFaceColor	点型填充色
MarkerSize	点型大小

【例 3-5】 符号样式控制示例(如图 3-6 所示)。

```
x = -pi:pi/10:pi;
y = tan(sin(x)) - sin(tan(x));
plot(x,y,'--rs','LineWidth',2,...
      'MarkerEdgeColor','k',...
      'MarkerFaceColor','g',...
      'MarkerSize',10)
% 设置双划线、红色与正方符,设置线宽为 2,正方符边黑色,
  正方符填充绿色,正方符大小为 10
```

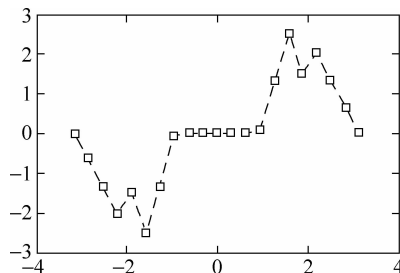


图 3-6 符号样式控制示例

3.1.3

窗口控制

(1) figure 函数用于创建窗口图形对象,其调用格式如下:

```
figure
figure('PropertyName', PropertyValue,...)
figure(h)
h = figure(...)
```

参数说明:

figure 创建一个默认的参数新窗口对象。

figure('PropertyName', PropertyValue,...) 创建自定义参数的新窗口对象。figure(h) 首先寻找句柄 h 的对象窗口是否存在,如果存在,就把它作为当前窗口显示出;如果不存在,但 h 是个整数,就创建一个句柄为 h 的新对象窗口,不是整数的话,就会出错。

(2) subplot 函数用于创建和控制多坐标轴,其调用格式如下:

```
subplot(m,n,p)
subplot(m,n,p,'replace')
subplot(h)
subplot('Position',[left bottom width height])
h = subplot(...)
```

参数说明:

subplot(m,n,p) 表示把当前窗口对象分成 $m \times n$ 块矩形区域,并在第 p 块区创建一个新的坐标轴,将这个新的坐标轴设定为当前坐标轴。参数 replace 表示当前坐标轴存在,删除它并创建一个新的坐标轴。

subplot('Position',[left bottom width height]) 是在规格化的窗口对象(范围为 0.0~

1.0)里创建一个位置为[left bottom width height]的坐标轴。

`h = subplot(...)` 返回创建新坐标轴的句柄。

【例 3-6】 窗口分割示例(如图 3-7 所示)

```
figure % 新打开一个窗口,并且将其默认为当前窗口
subplot(2,3,1:2) % 把窗体分割成 2×3 大小的窗口,p = 1 : 2,(subplot(m,n,p))为一个向量的情况
t = 0:.01:1;
y1 = sin(t * 2 * pi) .* cos(t * pi);
plot(t,y1);
subplot(233) % 等价于 subplot(2,3,3)
y2 = log(t + 1);
plot(t,y2);
subplot(2,3,[4 5 6])
alpha = 5.5;
y3 = exp(-alpha * t) .* sin(.5 * t);
plot(t,y3);
```

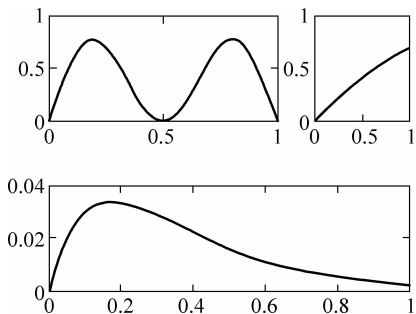


图 3-7 窗口分割示例

3.1.4

坐标轴控制命令

在缺省情况下,MATLAB 图形坐标轴的状态为:轴的比例控制为“自动(auto)”、坐标轴为“显示(on)”、采用的坐标比例为“直角(xy)”。在这种状态下,用户不需要对图形坐标进行任何干预,坐标刻度范围将根据绘图函数中矢量和矩阵元素值的范围自动确定,这种缺省状态给绘图带来很大方便。

图形是多种多样的,统一的坐标模式不可能总是最有效地表现出所绘图形的特征。因此,MATLAB 设计了控制坐标状态的 `axis` 函数。`axis` 函数的调用格式为:

- `axis([xmin,xmax,ymin,ymax])`。指定二维图形 x 轴和 y 轴的刻度范围。
- `axis auto`。设置坐标轴为自动刻度(默认值)。
- `axis manual`(或 `axis(axis)`)。保持刻度范围不随数据的大小而变化。
- `axis tight`。以数据的大小为坐标轴的范围。
- `axis ij`。设置坐标轴系统的原点在左上角,i 坐标为纵坐标,自上而下为数据增大,j 坐标为横坐标,由左至右为数据增大。
- `axis xy`。使坐标轴回到直角坐标系(默认值)。
- `axis equal`。使各坐标轴的刻度增量相同。
- `axis square`。使各坐标轴的长度相同(正方形或立方体),但刻度增量未必相同。
- `axis normal`。自动调节轴与数据的外表比例,使其他设置失效。
- `axis off`。使坐标轴消隐。
- `axis on`。绘制坐标轴(默认值)。

1. 坐标轴的范围

二维图形的坐标轴范围在默认状态下是根据数据的大小自动设置的,如果需要改变坐标范围,可以通过调用函数 `axis([xmin,xmax,ymin,ymax])` 来实现。其中,由 4 个参数组

成的矢量分别表示 x 轴的最小值和最大值,以及 y 轴的最小值和最大值。

【例 3-7】 对比坐标轴范围对正切函数图形表现的影响,生成数据点,分别按默认坐标轴范围和指定坐标轴范围绘制曲线(如图 3-8 所示)。

```
x = 0:.01:pi/2;
plot(x,tan(x),'-ro')      % 默认坐标轴范围
axis([0,pi/2,0,5])       % 指定坐标轴范围
```

用 `axis` 函数设定坐标轴范围时要注意,任何一个坐标轴的最小值都必须小于其最大值。

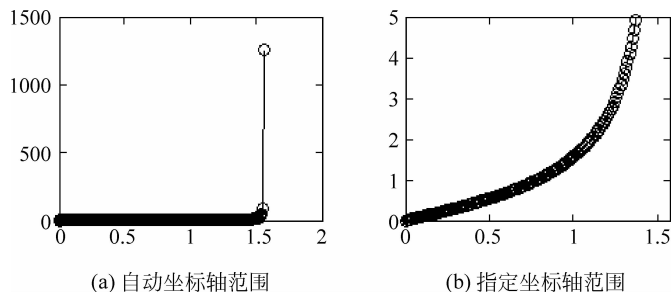


图 3-8 坐标轴范围对图形表现的影响

2. 显示比例对绘图结果的影响

为了能够充分地利用屏幕的显示空间,MATLAB 在默认的情况下,按照与屏幕相同的纵横比绘制图形。但为了满足不同情况的需要,MATLAB 还提供了几种不同的显示方法。

【例 3-8】 几种不同的显示方法比较。生成数据点,绘制曲线,并使用不同的显示比例(如图 3-9 所示)。

```
t = 0:pi/20:2 * pi;
plot(sin(t),2 * cos(t))      % 默认值
grid on
axis square                  % 正方形图
axis equal                   % X、Y 刻度相等
axis tight                   % 紧缩形式
```

(1) 本例绘制了一个椭圆,其长轴(y 轴)与短轴(x 轴)之比为 $2:1$ 。

(2) 图 3-9(a)所示为默认状态,没有设置任何坐标轴参数,坐标轴自动按照数据的大小确定刻度,图形的形状则尽可能按照窗口的比例。这种方式的优点是能够得到最大的图形显示。

(3) 图 3-9(b)中调用函数 `axis square`,将两个坐标轴定义为相等的长度(但刻度的增量不一定相等),即绘制正方形坐标下的图形。这种显示方式通常在两个轴具有相同比例(如画圆)时使用。

(4) 图 3-9(c)调用函数 `axis equal`,使得两个坐标轴具有相同的刻度,因此绘制的图形是真实的比例情况。

(5) 图 3-9(d)调用函数 `axis tight`,得到了最紧凑的显示形式,即将坐标轴收缩到图形的尺寸,不留空余空间。

本例中,图(a)和图(d)恰好相同。

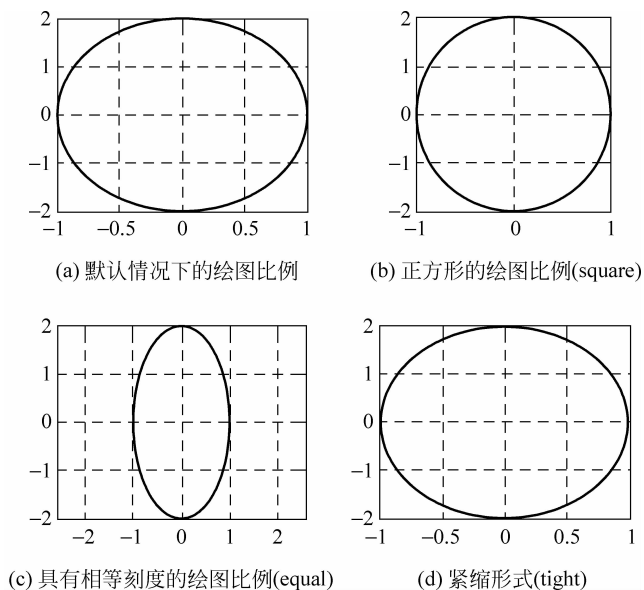


图 3-9 几种不同的显示方式

3.1.5

图形标注

一个好的图形必须有适当的标注, MATLAB 提供了一系列方便的图形标注函数, 包括:

- `title`。图形标题。
- `xlabel`。x 轴标识。
- `ylabel`。y 轴标识。
- `zlabel`。z 轴标识。
- `text`。任意位置加注文本。
- `gtext`。鼠标定位加注文本。
- `legend`。标注图例。

图形标注使用的文字可以是字母和数字, MATLAB 也可以使用汉字, 还可以按照规定的方法表示希腊字母、数学符号和变形体, 例如 `\pi` 表示 π , `\leq` 表示 $\leq$, `\it` 表示斜体字等。本节只介绍函数输入方法, 其他方法请参见其他参考资料。

1. 加注坐标轴标识和图形标题

`xlabel`、`ylabel` 和 `zlabel` 函数用于在当前坐标轴上加注 x 轴、y 轴和 z 轴的标识, `title` 函数用于在当前图形窗口上加标题。这 4 个函数最简单的调用形式是给定一个字符串格式的

输入参数,它们的位置自动设定,如“xlabel('时间(t)')”表示在 x 轴上标注标识“时间(t)”,“title('电压信号')”表示将“电压信号”标注为图形的标题。

【例 3-9】 在图形中加注坐标轴标识和标题。生成数据点,绘制曲线,并在图形中加注坐标轴标识和标题(如图 3-10 所示)。

```
t = 0:pi/100:2 * pi;
y = sin(t);
plot(t,y)
axis([0 2 * pi - 1 1])
xlabel('0\leq t \leq \pi', 'FontSize', 11)
ylabel('sin(t)', 'FontSize', 11)
title('正弦函数图形', 'FontName', '隶书', 'FontSize', 11)
```

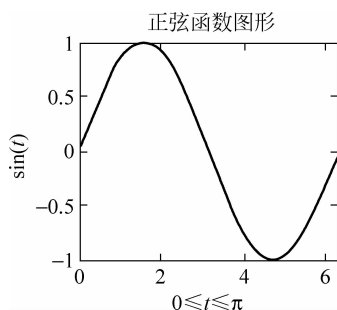


图 3-10 加注坐标轴标识和标题

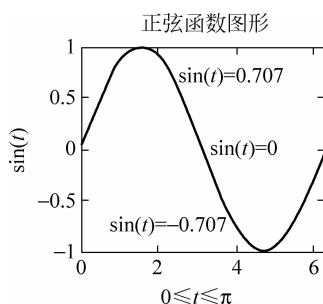


图 3-11 在图形中加注文本

在 xlabel 语句中, \pi 表示希腊字母 π , \leq 表示 $\leq$, \it 表示后面的字为斜体字, \rm 表示后面的字恢复为正体字。FontSize 表示字体的大小, FontName 表示字体的类型。

2. 在图形中加注文本

text 函数用于在图形窗口的任何位置加注文本,该函数最简单的调用形式为:

```
text(x,y,'字符串')
```

x 和 y 指定加注文本的位置。text 函数也可以通过指定属性来增加文本形式的变化。

【例 3-10】 在图形中的任意位置加入文本。在例 3-10 绘制的图形中的任意位置加入文本(如图 3-11 所示)。

```
text(pi, sin(pi), '\leftarrowsin(t) = 0', 'FontSize', 11)
text(3 * pi/4, sin(3 * pi/4), '\leftarrowsin(t) = 0.707', 'FontSize', 11)
text(5 * pi/4, sin(5 * pi/4), 'sin(t) = -0.707\rightarrow', ...
    'FontSize', 11, 'HorizontalAlignment', 'right')
```

语句中的 \leftarrow 表示加一个左箭头, \rightarrow 表示加一个右箭头, HorizontalAlignment 表示水平右对齐。

MATLAB 还提供了一个函数 gtext, 用于使用鼠标在指定位置加注文本, 其调用格式为:

```
gtext('字符串')
```

当调用这个函数时,在图形窗口中出现一个随鼠标移动的大“十”字交叉线。移动鼠标,将“十”字交叉点移动到适当位置后,单击鼠标左键, `gtext` 参数中的字符串就标注在该位置上。

3. 指定 TeX 字符

MATLAB 中的文本对象支持 TeX 字符。使用 TeX 字符可以在文本中使用各种特殊的字木和符号。

【例 3-11】 在图形中使用指定的 TeX 字符。生成数据,绘制图形,并在图形标注中使用指定的 TeX 字符(如图 3-12 所示)。

```
t = 0:pi/100:2 * pi;
alpha = -0.8; beta = 15;
y = sin(beta * t) . * exp(alpha * t);
plot(t,y)
title('\itAe^{\-alpha\itt}sin\betaa{\tt}\alpha<<\betaa')
xlabel('时间\musec. '), ylabel('赋值')
title 的字符串为“Aeαt sin (βt), α<<β”。
```

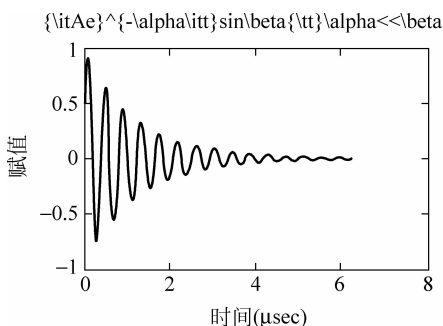


图 3-12 在图形标注中使用指定的 TeX 字符

4. legend 加图例框

调用格式如下:

<code>legend('string1','string2',...)</code>	用指定的文字 <code>string</code> 在当前坐标轴中对所给数据的每一部分显示一个图例;
<code>legend(h,'string1','string2',...)</code>	指定的句柄对象 <code>h</code> 添加图例;
<code>legend(string_matrix,...)</code>	用字符数组来表示图例(每行对应);
<code>legend(h = string_matrix,...)</code>	指定的句柄对象 <code>h</code> 添加图例;
<code>legend(axes_handle,...)</code>	指定的句柄对象 <code>axes_handle</code> ;
<code>legend('off')</code>	删除图例;
<code>legend('hide')</code>	隐藏图例;
<code>legend('show')</code>	显示图例;
<code>legend('boxoff')</code>	移出图例边框;
<code>legend('boxon')</code>	显示图例边框;
<code>legend(...,pos)</code>	<code>pos</code> 的参数如表 3-5 所示。

表 3-5 pos 的参数值

pos 取值	图例位置	pos 取值	图例位置
-1	坐标轴之外的右边	2	坐标轴的左上角
0	坐标轴之内,有可能遮挡部分图形	3	坐标轴的左下角
1	坐标轴的右上角(默认位置)	4	坐标轴的右下角

【例 3-12】 在图形中加上图例(如图 3-13 所示)。

```
x = -pi:pi/20:pi;
plot(x,cos(x),'-ro',x,sin(x),'-b')
h = legend('cos','sin',2);
```

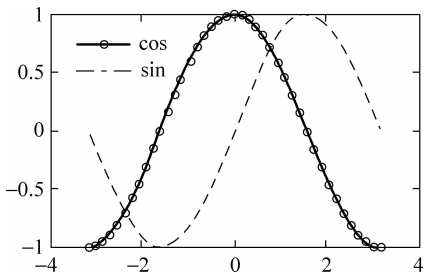


图 3-13 图形加图例示例

3.2 特殊二维图形绘制

在 MATLAB 中只有 plot 函数和 plot3 函数等是远远不够的,和其他编程语言一样, MATLAB 还提供了一些特殊的绘图函数。这些绘图函数具有其他编程语言所不具备的特点,即简易、功能强大、函数全,并且容易掌握。下面将具体说明这些函数。

3.2.1

条形图和面域图

条形图和面域图主要用于在一定时域内比较不同的数据结果,并且显示这些数据的总和。条形图用于离散数据显示,而面域图显示连续数据的情况。条形图在维数上有两种图形:二维条形图和三维条形图;在方位上也有两种图形,即垂直条形图和水平条形图。所以,条形图共有 4 个函数,如表 3-6 所示。

表 3-6 条形图函数

	二维	三维
垂直	bar	bar3
水平	barh	bar3h

另外,条形图的表现模式有两种:累积式和分组式,分别用'stacked'和'grouped'表示。面域图表示函数 area()的用法与条形图一样。