

第3章 二维绘图

强大的绘图功能是 MATLAB 的特点之一, MATLAB 提供了一系列的绘图函数, 包括采用不同坐标系, 如直角坐标、对数坐标、极坐标绘制二维图形和三维图形的绘图函数。

二维图形是 MATLAB 图形的基础, 也是应用最广泛的图形类型之一。本章主要介绍 MATLAB 提供的二维图形绘制函数。

【知识要点】

本章主要内容包括 MATLAB 基本二维绘图指令和其他二维图形绘图函数。

【学习目标】

知 识 点	学 习 目 标			
	了 解	理 解	掌 握	运 用
最基本的二维绘图函数			★	★
绘制二维图形的其他函数			★	

在科学计算中, 往往要处理大量的数据。如果把这些数据用图形表现出来, 就能很容易地发现这些数据的内在联系, 大大提高工作效率。MATLAB 正是基于这种考虑, 提供了强大的绘图能力, 可将矩阵中的数值可视化, 如图 3-1 所示。

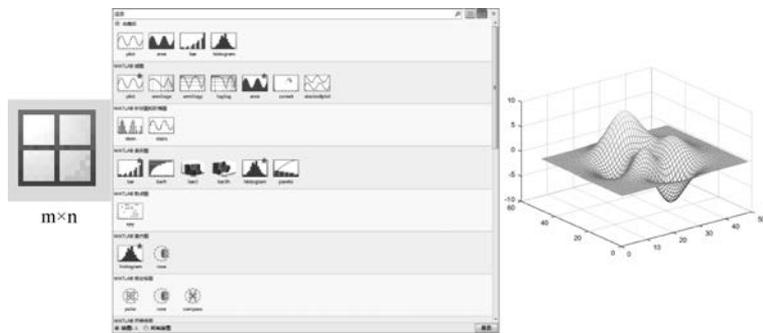


图 3-1 矩阵可视化



视频讲解

3.1 最基本的二维绘图函数

3.1.1 绘制二维曲线的最基本的函数

二维曲线图绘制需要调用 plot 命令。

调用格式: plot(x, y)

说明：以 x 为横坐标， y 为纵坐标，按照坐标 (x_j, y_j) 的有序排列绘制曲线。

【例 3-1】 绘制 0 到 2π 的正弦曲线。在命令行窗口中键入：

```
x = 0:pi/100:2 * pi;           % 构造向量
y = sin(x);                     % 构造对应 y 的坐标
plot(x,y)                       % 绘制以 x 为横坐标,y 为纵坐标的图形
```

绘制的二维图形如图 3-2 所示。

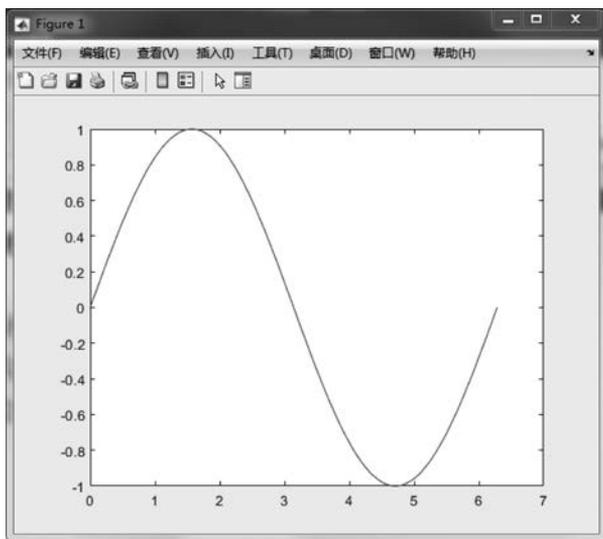


图 3-2 函数 $\text{plot}(x,y)$ 绘制的正弦曲线

3.1.2 绘制图形的类型

可利用 plot 函数绘制多条曲线。

调用格式： $\text{plot}(X_1, Y_1, X_2, Y_2, \dots, X_n, Y_n)$

plot 自动循环地采用颜色板中的各种颜色。

【例 3-2】 绘制 0 到 2π 的正弦曲线和余弦曲线。

在命令行窗口中键入：

```
x = 0:pi/100:2 * pi;           % 构造向量
y1 = sin(x);                   % 构造对应 y1 的坐标
y2 = cos(x);                   % 构造对应 y2 的坐标
plot(x, y1, x, y2)             % 绘制以 x 为横坐标,y1 和 y2 为纵坐标的图形
```

绘制的二维图形如图 3-3 所示。

通常，为了突出图表可视化的效果，常常会对线型、标记符号和颜色进行样式的设置。

调用格式： $\text{plot}(X, Y, \text{'选项'})$

其中：选项用于指定线型、标记和颜色，但线条的类型和颜色可以通过字符串来指定。表 3-1 列出了在这个字符串中允许使用的线条类型和颜色，线条默认 (none) 类型是实线型。

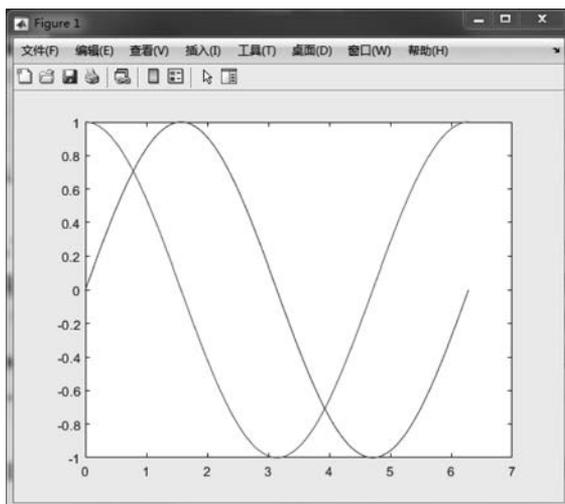
图 3-3 函数 $\text{plot}(x,y)$ 绘制的正弦和余弦曲线

表 3-1 点类型、线类型和颜色

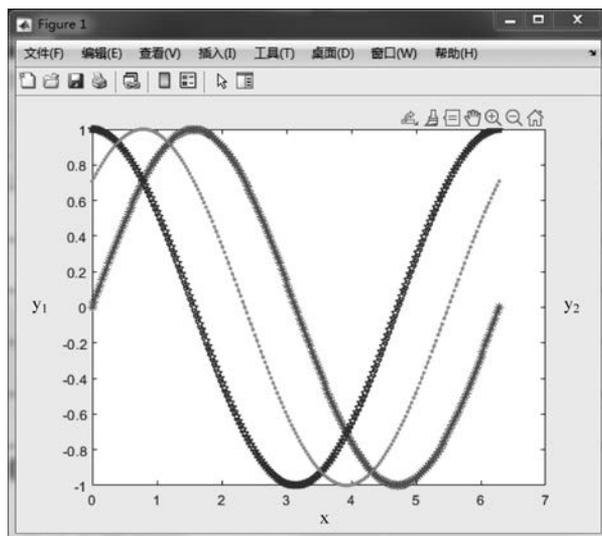
符 号	点 类 型	符 号	线 类 型
.	黑点	-	实线
*	星号	--	虚线
s 或 square	正方形	-.	点画线
d 或 diamond	菱形	:	点线
p 或 pentagram	五角星型	默认 (none)	无线
h 或 hexagram	六角星型		
°	圆圈	符号	颜色
+	加号	g	绿色
×	叉号	m	品红色
<	向左尖三角	b	蓝色
>	向右尖三角	c	灰色
^	向上尖三角	w	白色
v	向下尖三角	r	红色
默认 (none)	无点	k	黑色
		y	黄色

【例 3-3】 绘制函数 $\sin x$ 、 $\cos x$ 和 $\sin\left(x + \frac{\pi}{4}\right)$ 在 $0 \sim 2\pi$ 的曲线。

在命令行窗口中键入：

```
x = 0:pi/100:2 * pi;           % 构造向量
y1 = sin(x);                   % 构造对应 y1 的坐标
y2 = cos(x);                   % 构造对应 y2 的坐标
y3 = sin(x + pi/4);
plot(x, y1, 'm - * ', x, y2, 'b -- p', x, y3, 'g -- .') % 绘制以 x 为横坐标, y1 和 y2 为纵坐标的图形
```

绘制的二维图形如图 3-4 所示。

图 3-4 函数 $\text{plot}(x,y)$ 绘制的正弦(相位 0 和 $\pi/4$)和余弦曲线

3.1.3 图形格式和注释

绘制函数后,还应该给图形进行标注,以增强图形的可读性,如给每个图加上标题、坐标轴标记和曲线说明等。表 3-2 列出了图形标注常用函数及示例。

表 3-2 图形标注常用函数及示例

函 数	示 例
title—添加标题 格式: title('图形名称')	title('两条相交曲线')
xlabel—为坐标轴添加标签 格式: xlabel('x 轴说明')	xlabel('自变量 x') ylabel('函数值 y')
axis—设置坐标轴范围和纵横比 格式: axis([xmin xmax ymin ymax])	axis([0 6 -1 1])
text—向数据点添加文本说明 格式: text(x,y, '图形说明')	text(1.2, 0.8, 'x = 0.989899') text(3.2, 0.2, 'x = 3.0404') text(1.8, 0.4, '1/sinh(x)') text(0.3, 0.2, 'sin(x)')
grid—显示或隐藏坐标区网格线 格式: grid on grid off	grid on
legend—在坐标区上添加图例 格式: legend('图例 1')	legend('cos(x)', '1/cosh(x)', 'Location', 'NorthEast')
hold—图形保持 格式: hold on/off	hold on
line—绘制基本线条 格式: line(x,y)	line([0.989899 0.989899], [-1 1]); line([3.0404 3.0404], [-1 1], 'Color', 'red');

【例 3-4】 绘制 0 到 7 的正弦曲线 $\sin(x)$ 和双曲正弦曲线的倒数 $1/\sinh(x)$, 并为图添加标题、为坐标轴添加标签、添加图例、添加文本说明等。

在命令行窗口中键入:

```
x = linspace(0,7,100);
plot(x, sin(x), 'r--', x, 1./sinh(x), 'b-')
xlabel('自变量 x') % 坐标轴标签
ylabel('函数值 y')
title('两条相交曲线')
text(1.2, 0.8, 'x = 0.989899') % 文本说明
text(3.2, 0.2, 'x = 3.0404')
text(1.8, 0.4, '1/sinh(x)')
text(0.3, 0.2, 'sin(x)')
legend('sin(x)', '1/sinh(x)', 'Location', 'NorthEast') % 图例
axis([0 7 -1 1]) % 坐标轴范围
grid on % 显示网格线
line([0.989899 0.989899], [-1 1]); % 绘制线条
line([3.0404 3.0404], [-1 1], 'Color', 'red'); % 绘制线条并设置颜色
```

绘制的二维图形如图 3-5 所示。

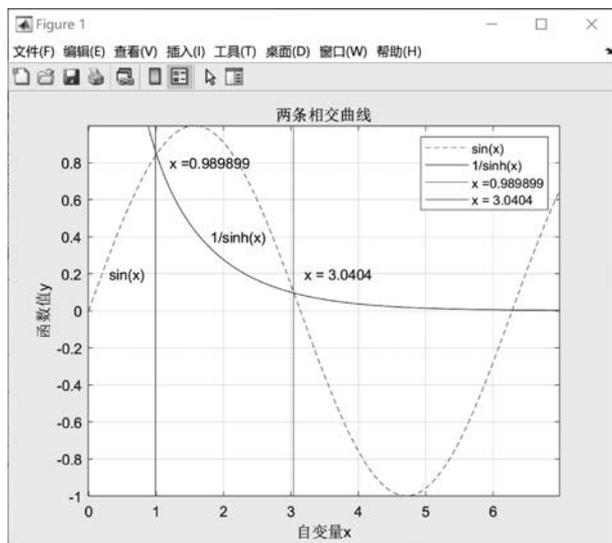


图 3-5 正弦 $\sin(x)$ 和双曲正弦倒数 $1/\sinh(x)$ 曲线

也可以将 plot 与 line 命令合并,代码如下:

```
plot(x, sin(x), 'r--', x, 1./sinh(x), 'b-', [0.989899, 0.989899], [-1, 1], [3.0404, 3.0404], [-1, 1], 'm')
```

绘制的二维图形如图 3-6 所示。

在图 3-7 中,可以显示鼠标所选的图形上某点的坐标值;在图 3-8 中,通过放大镜图标  可实现图形的缩放。

在显示的图像中,单击“编辑”→“图窗属性”,就可以修改图形的颜色、线型、线宽等,如图 3-9 所示。

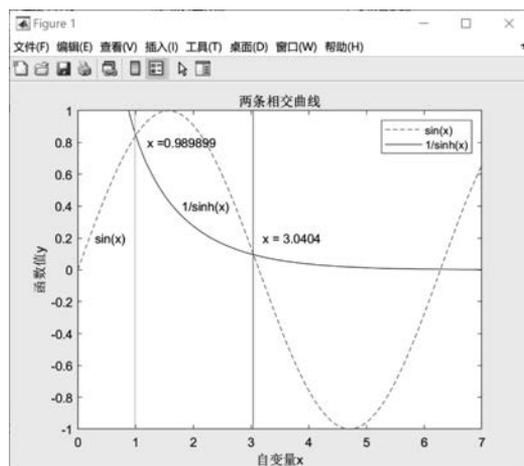


图 3-6 plot 与 line 命令合并绘制的正弦 $\sin(x)$ 和双曲正弦倒数 $1/\sinh(x)$ 曲线

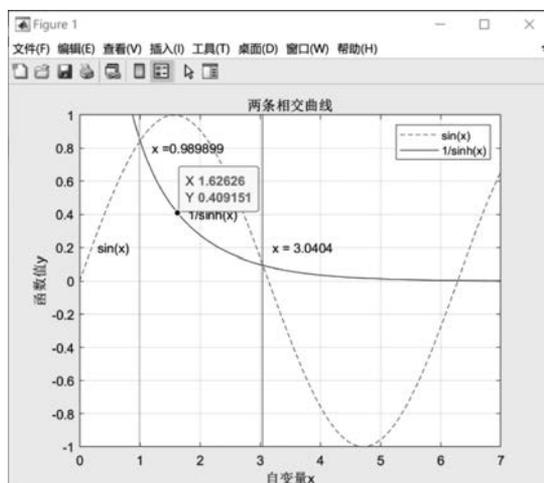


图 3-7 显示鼠标所选的图形上某点的坐标值

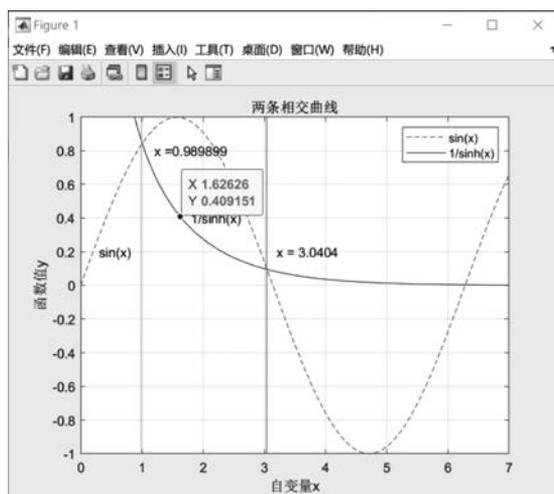


图 3-8 通过放大镜图标实现图形的缩放

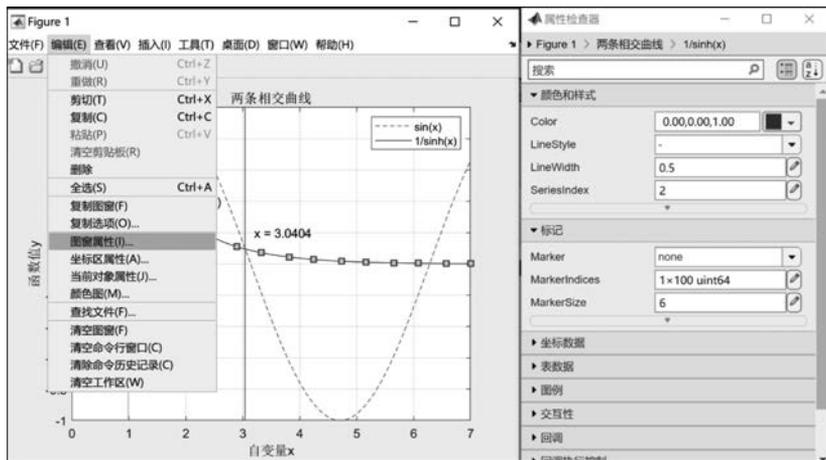


图 3-9 图窗属性的编辑

3.1.4 叠加图绘制

在默认情况下,多个图形的绘制在执行第二个 plot 语句时,将删除第一个 plot。在同一坐标轴中绘制多个图形有多种方法。

一种方法是前面介绍的,调用 $\text{plot}(X_1, Y_1, X_2, Y_2, \dots, X_n, Y_n)$ 绘制多条曲线。

```
x = 0:pi/100:2 * pi;
y1 = sin(x);
y2 = cos(x);
y3 = sin(x + pi/4);
plot(x, y1, 'r -', x, y2, 'g --', x, y3, 'b :')
```

% 构造向量
 % 构造对应 y1 的坐标
 % 构造对应 y2 的坐标
 % 构造对应 y3 的坐标
 % 绘制以 x 为横坐标, y1、y2 为纵坐标的图形

所绘制的曲线如图 3-10 所示。

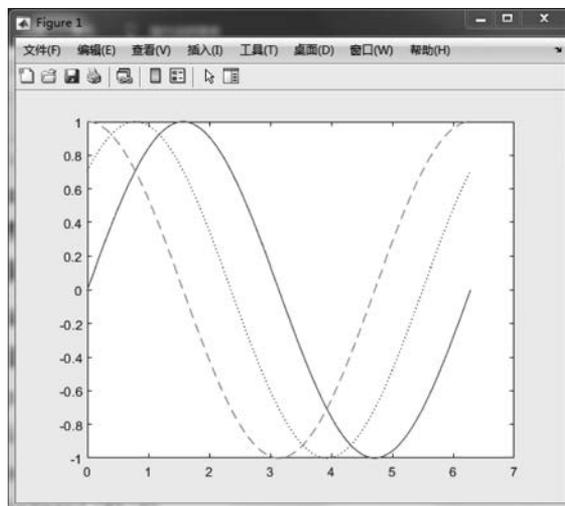


图 3-10 叠加图形的绘制

另一种方法是使用 hold 命令。可以使用 hold on 命令使当前坐标轴及图形保持而不被刷新,使随后绘制的图形叠加到现有图形中。hold off 命令为关闭图形保持功能,不能在当

前坐标轴上再绘制图形。通常每次绘图结束就采用 hold off 将所画曲线清除。

【例 3-5】 在同一坐标轴中绘制 $-\pi$ 到 π 的 $\sin x$ 、 $\cos x$ 、 $\sin(x) + \cos(x)$ 三条曲线，并以不同线型进行区分。

程序如下：

```
x = -pi:pi/30:pi;           % 构造向量
y1 = sin(x);                % 构造对应 y1 的坐标
plot(x, y1, 'r--')          % 绘制以 x 为横坐标, y1 为纵坐标的图形
hold on
y2 = cos(x);                % 构造对应 y2 的坐标
plot(x, y2, 'b*')           % 绘制以 x 为横坐标, y2 为纵坐标的图形
hold on
y3 = sin(x) + cos(x);      % 构造对应 y3 的坐标
plot(x, y3, 'g-.^')        % 绘制以 x 为横坐标, y3 为纵坐标的图形
hold off
```

在同一坐标轴中绘制的三条曲线如图 3-11 所示。

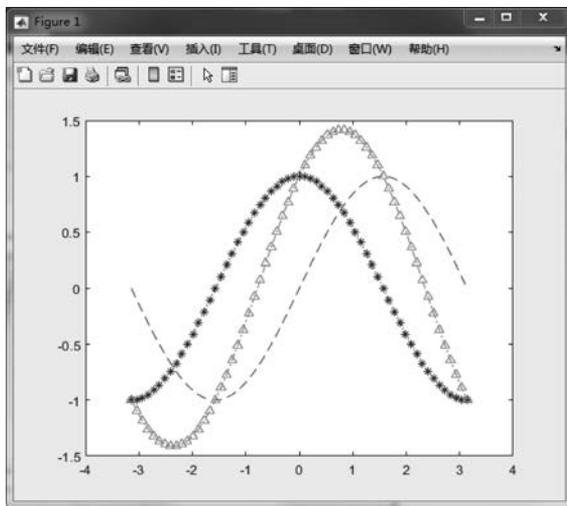


图 3-11 在同一坐标轴中绘制的三条曲线

3.1.5 子图绘制

在图 3-11 中三条曲线挤在同一张图中，某些情况下这样的布局比较好，但在有些情况下则需要将多个曲线分开到不同的子图中绘制。如果想要在一张图中展示多个子图，单纯使用 plot 函数就很难解决了。

如果希望在一个图形窗口中同时绘制多幅相互独立的子图，每个子图也是一个独立的坐标系，需要调用 subplot 命令。

调用格式：subplot(m, n, k) 或 subplot(mnk)

说明：将当前图形窗口分成 $m \times n$ 个绘图区，即共 m 行，每行 n 个，子绘图区的编号按行优先从左到右编号。该函数选定第 k 个子图为当前活动区。在每一个子绘图区允许以不同的坐标系单独绘制图形。subplot 本身并不绘制任何图形，但决定了如何分割图形窗口以及下一幅图将被绘制在哪个子窗口中。

【例 3-6】 将例 3-5 中的三幅图分别绘制在子窗口中。

程序如下：

```
x = -pi:pi/10:pi;
subplot(2,2,1);
plot(x, sin(x), 'r-- ');
subplot(2,2,3);
plot(x, cos(x), 'b: * ');
subplot(2,2,[2 4]);
plot(x, sin(x) + cos(x), 'g- .^');
```

绘制的曲线如图 3-12 所示。

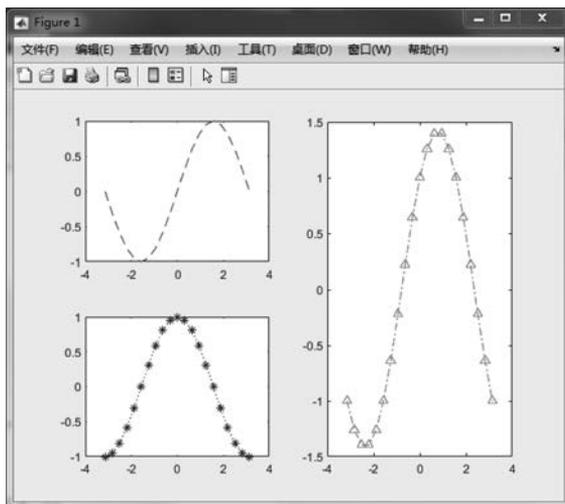


图 3-12 在同一坐标轴中绘制多图例子

subplot(2,2,1)将原始的图像切割为 4 个子图,是 2 行 2 列,并将图绘制在第一个子图区域上; subplot(2,2,[2 4])将图像绘制在第 2 个和第 4 个子图区域上。

3.1.6 复制/粘贴图

在图窗菜单中选择“编辑”→“复制选项”,进入预设项界面,设置剪贴板格式和图窗背景色。选择图元文件并使用图窗颜色,如图 3-13 所示。

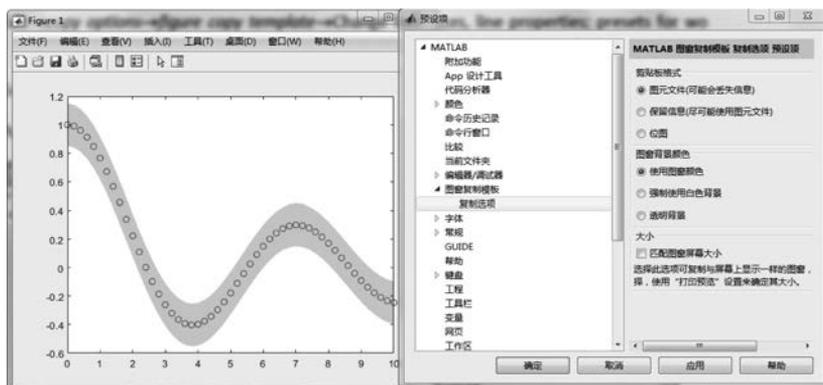


图 3-13 设置剪贴板格式和图窗背景色

选择“编辑”→“复制图窗”，将图窗复制到系统剪贴板，如图 3-14 所示。

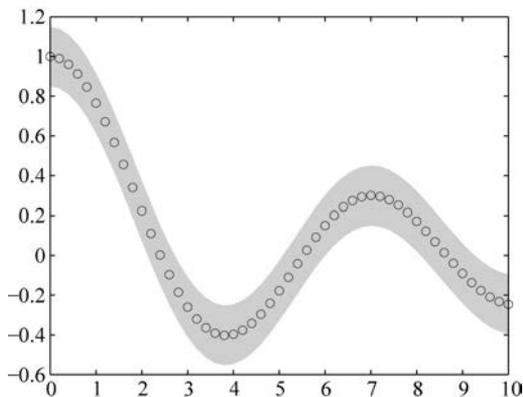


图 3-14 复制/粘贴图

3.1.7 保存图形

可以将图窗保存为特定的文件格式，图形可以保存为多种格式（fig、eps、jpeg、gif、png、bmp 等），常用图形保存格式如表 3-3 所示。

表 3-3 常用图形保存格式

扩展名	文件生成格式
.fig	fig 文件包含了所有信息，包括图窗和内容，可以后期修改
.bmp	未压缩的图像
.eps	高质量可缩放格式(用 latex 编辑时用)，用 PostScript 语言描述的一种 ASCII 图形文件格式，在 PostScript 图形打印机上能打印出高品质的图形图像，最高能表示 32 位图形图像
.pdf	压缩的图像

使用交互式控件保存绘图，单击坐标区工具栏中的导出按钮 ，MATLAB 显示包含文件类型选项的“另存为”对话框，如图 3-15 所示。

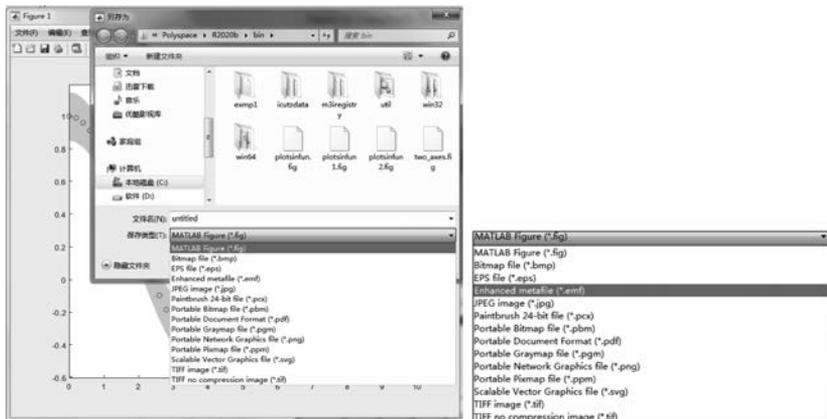


图 3-15 图形保存及格式选择

从 R2016a 开始，保存的图窗大小默认情况下与屏幕上的图窗大小一致。也可以用编程方式将绘图保存为图像或向量图形文件。

调用格式: `saveas(fig, filename)`

表 3-4 为图形保存位图图像格式。

表 3-4 图形保存位图图像格式

选项	位图图像格式	默认文件扩展名
'jpeg'	JPEG 24 位	.jpg
'png'	PNG 24 位	.png
'tiff'	TIFF 24 位(压缩)	.tiff
'bmp'	BMP 24 位	.bmp
'bmpmono'	BMP 黑白	.bmp
'bmp256'	BMP 8 位(256 色)	.bmp

表 3-5 为图形保存矢量图格式。

表 3-5 图形保存矢量图格式

文件格式	矢量图格式	默认文件扩展名
'pdf'	整页可移植文档格式(PDF)颜色	.pdf
'eps'	PostScript(EPS) 3 级黑白	.eps
'eps'	封装的 PostScript(EPS) 3 级彩色	.eps
'meta'	增强型图元文件(仅限 Windows)	.emf
'svg'	SVG 指可伸缩矢量图形(Scalable Vector Graphics)	.svg
'ps'	全页 PostScript (PS) 3 级黑白	.ps
'psc'	全页 PostScript (PS) 3 级彩色	.ps
'ps2'	全页 PostScript (PS) 2 级黑白	.ps
'psc2'	全页 PostScript (PS) 2 级彩色	.ps

【例 3-7】 创建一个条形图并获取当前图窗,然后将该图窗另存为 PNG 文件。

程序如下:

```
x = [2 4 7 2 4 5 2 5 1 4];
bar(x);
saveas(gcf, 'Barchart.png')           % gcf 当前图窗的句柄
```

如果希望清晰度很高的话, `saveas` 就无法处理了,因为分辨率太高,此时就需要函数 `print`。

从 R2020a 开始,可以使用 `exportgraphics` 函数保存下列任一项的内容:坐标区、图窗、可作为图窗子级的图、分块图布局或容器(如面板)。`exportgraphics` 函数支持三种图像格式(PNG、JPEG 和 TIFF)和三种同时支持向量和图像内容的格式(PDF、EPS 和 EMF)。PDF 格式支持嵌入字体。

当需要执行以下操作时, `exportgraphics` 函数比 `saveas` 函数更合适:

保存在 App 或 MATLAB Online 中显示的图形;最小化内容周围的空白;用可嵌入的字体保存 PDF 片段;保存图窗中内容的一部分;控制背景颜色,而不必修改图窗的属性。

【例 3-8】 创建一个条形图并获取当前图窗,然后将该图窗另存为 PNG 文件。在本例中,指定每英寸 300 点(DPI)的输出分辨率。

程序如下:

```
bar([1 11 7 8 2 2 9 3 6])
f = gcf;
```

```
% Requires R2020a or later
exportgraphics(f, 'barchart.png', 'Resolution', 300)
```

3.2 线性直角坐标系其他二维图形绘制函数



视频讲解

除了绘制二维曲线的基本函数 `plot` 外,在线性直角坐标系中,其他形式的图形还有火柴杆图、条形图、阶梯图和填充图等。

3.2.1 双纵轴坐标

`plotyy` 函数能把具有不同量纲、不同数量级的两个函数绘制在同一坐标中。

调用格式: `plotyy(x1, y1, x2, y2)`

其中: `x1-y1` 对应一条曲线, `x2-y2` 对应另一条曲线。横坐标的标度相同,纵坐标有两个,左纵坐标用于 `x1-y1` 数据对,右纵坐标用于 `x2-y2` 数据对。

【例 3-9】 在同一坐标中绘制 $y_1 = 200e^{-0.05x} \cdot \sin x$ 和 $y_2 = 0.8e^{-0.5x} \cdot \sin(10x)$ 。

程序如下:

```
%% 使用 plotyy 画两条曲线
clear; clc; close all;
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); % 两条曲线
title('双纵轴坐标曲线'); % 显示标题
```

绘制的曲线如图 3-16 所示。

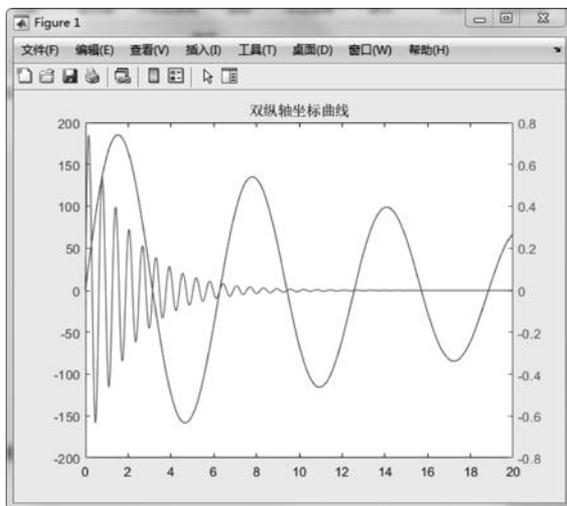


图 3-16 双纵轴坐标曲线图

3.2.2 火柴杆图

`stem` 函数常用于绘制离散数据的图形,画出的图形是火柴杆图或戴着“帽子”的“棒棒糖图”,在数字信号处理中应用较多。

调用格式: `stem(x,y,'选项')`

【例 3-10】 绘制正弦函数 $\sin x$ 的火柴杆图。

程序如下:

```
%% 杆状图
clear; clc; close all;
x = linspace(0, 4 * pi, 40);
y = sin(x);
subplot(1,2,1);
stem(y); % 杆状图
subplot(1,2,2);
stem(y,'fill','r'); % 杆状图
```

绘制的正弦曲线火柴杆图如图 3-17 所示。

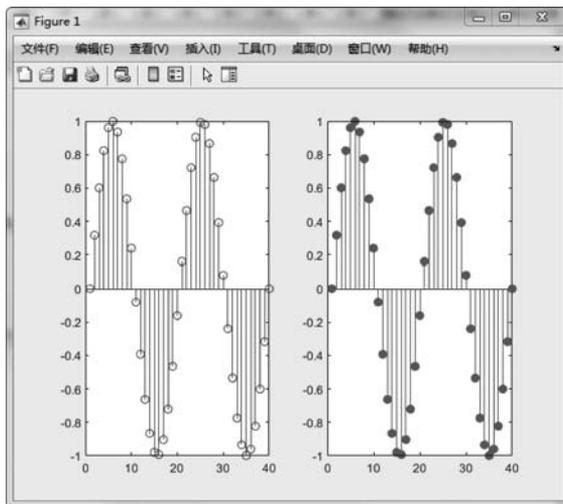


图 3-17 正弦曲线火柴杆图

3.2.3 条形图

`bar` 函数用于绘制二维垂直条形图,用垂直条形显示向量或矩阵中的值。

调用格式: `bar(x,y,'选项')`

其中“选项”默认条形图为堆栈(垂直)的。

【例 3-11】 分别绘制向量 $x=[1\ 2\ 5\ 4\ 8]$ 和 $y=[x;1:5]$ 的一维向量 x 条形图和二维向量 x,y 条形图。

程序如下:

```
clear; clc; close all;
x = [1 2 5 4 8]; % 向量 x
y = [x;1:5]; % 向量 y
subplot(1,2,1);
bar(x); % 一维条形图
title('一维向量 x 条形图');
subplot(1,2,2);
bar(y); % 二维条形图
title('二维向量 x,y 条形图');
```

绘制的条形图如图 3-18 所示。

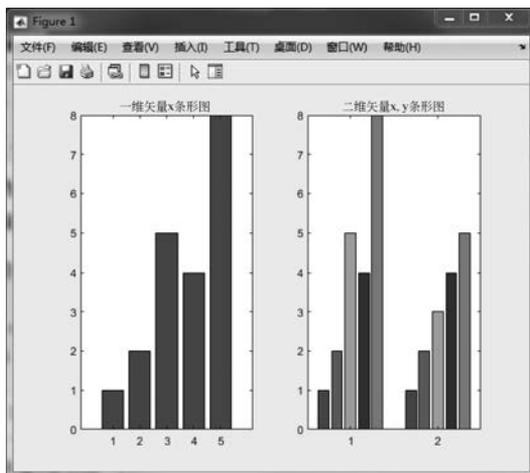


图 3-18 条形图

若将条形图变成水平的,则在 bar 后面加一个 horizontal 的首字母。

调用格式: `barh(y,'选项')`

【例 3-12】 分别绘制向量 $x = [1\ 2\ 5\ 4\ 8]$ 和 $y = [x; 1:5]$ 的堆栈式的条形图和水平式的条形图。

程序如下:

```
x = [1 2 5 4 8];
y = [x; 1:5];
subplot(1,2,1);
bar(y,'stacked');           % 堆栈式的 bar
title('堆栈式的条形图');
subplot(1,2,2);
barh(y);                    % 水平式的 bar
title('水平式的条形图');
```

绘制的条形图如图 3-19 所示。

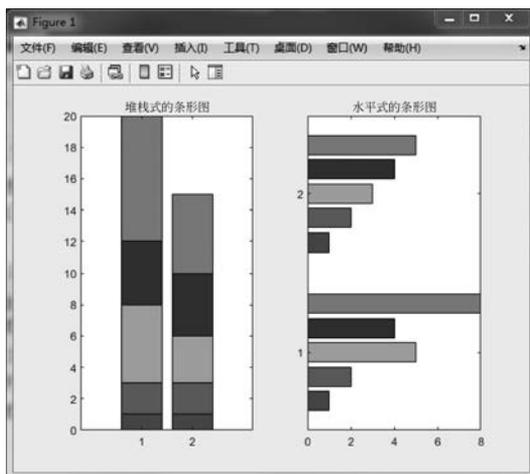


图 3-19 条形图

3.2.4 阶梯图

stairs 函数有助于理解数据阶梯形的变化趋势,主要用于绘制数字信号处理中的采样信号。另外,stairs 函数在图像处理中的直方图均衡化技术中有很大的意义。

调用格式: stairs(x,y,'选项')

【例 3-13】 绘制正弦函数 $\sin x$ 的阶梯图。

程序如下:

```
%% 阶梯图
clear; clc; close all;
x = linspace(0, 4 * pi, 40);
y = sin(x);
stairs(y);                                % 阶梯图
```

绘制的阶梯图如图 3-20 所示。

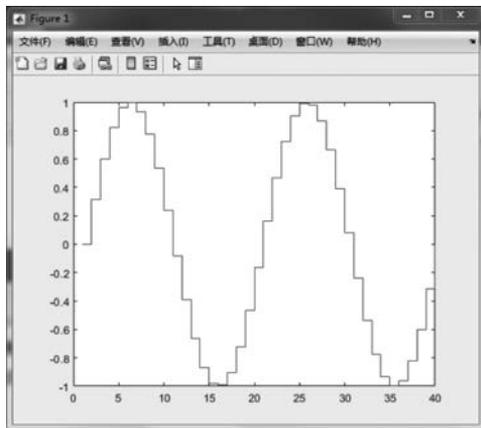


图 3-20 阶梯图

3.2.5 填充图

fill 函数按向量元素下标渐增次序用直线段连接 x,y 对应元素定义的数据点。假如这样连接所得折线不封闭,那么 MATLAB 将自动把该折线的首尾连接起来,构成封闭多边形,然后将多边形内部涂满指定的颜色。

调用格式: fill(x1,y1,'选项 1',x2,y2,'选项 2',...)

【例 3-14】 绘制 $y=2e^{-0.5x}$ 的填充图。

程序如下:

```
x = 0:0.35:7;
y = 2 * exp(-0.5 * x);
fill(x,y,'r');
title('填充图');
axis([0,7,0,2]);
xlabel('自变量 x');
ylabel('函数值 y');
```

绘制的填充图如图 3-21 所示。

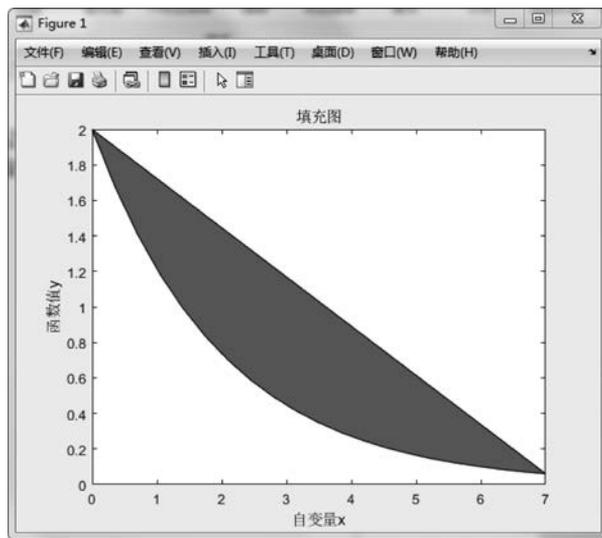


图 3-21 填充图

【例 3-15】 绘制一个红底白字的八边形 STOP 标识。

程序如下：

```
t = (1:2:15)' * pi/8;          % 画八边形的八个角
y = sin(t);
x = cos(t);
fill(x,y,'r');                % 填充
axis square off;
text(0,0,'STOP','color','w','fontsize',80,'fontweight','bold','horizontalalignment','center');
```

绘制的红底白字八边形 STOP 标识如图 3-22 所示。

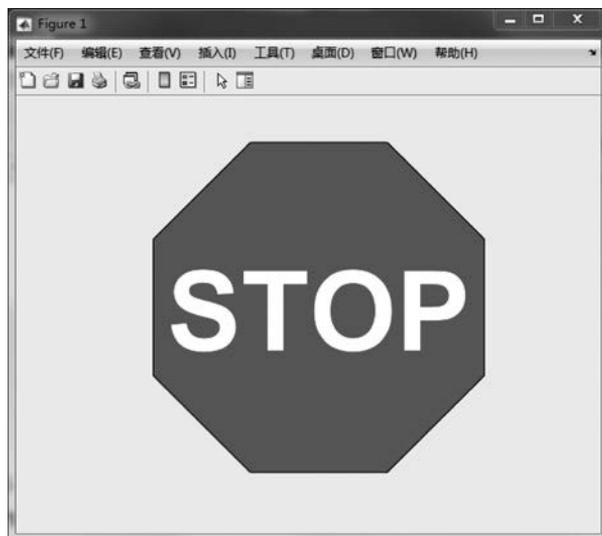


图 3-22 红底白字八边形 STOP 标识



彩色图片



视频讲解

3.3 特殊坐标系二维图形绘制函数

在使用基本的绘图函数时,坐标轴刻度为线性刻度。当自变量的少许变化引起因变量极大变化时,即当实际的数据呈指数变化时,使用对数坐标系可使曲线最大变化范围伸长,图形轮廓更加清楚,起到压缩坐标、扩大视野的作用。

在平面直角坐标系中表示两点间的关系只能使用三角函数来表示,而在极坐标系中用夹角和距离则很容易表示,甚至对于某些曲线来说,只有极坐标方程能够表示。极坐标系的应用领域十分广泛,包括数学、物理、工程、航海以及机器人等领域。

3.3.1 极坐标绘图

polar 函数用来绘制极坐标图。

调用格式: polar(theta, rho, '选项')

其中: theta 为极坐标极角, rho 为极坐标矢径,“选项”的内容与 plot 函数相似。

【例 3-16】 绘制 $\rho=2\sin(4\theta) \cdot \cos(2\theta)$ 的极坐标图。

程序如下:

```
theta = 0:0.01:2 * pi;
rho = 2 * sin(4 * theta) . * cos(2 * theta);
polar(theta, rho, 'r');
```

绘制的极坐标图如图 3-23 所示。

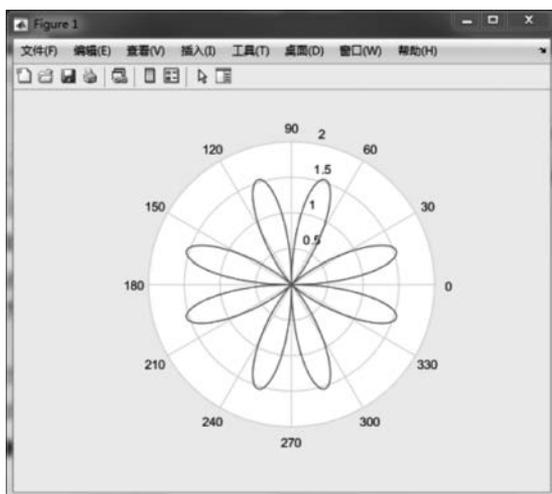


图 3-23 极坐标图

3.3.2 半对数和双对数坐标系绘图

MATLAB 提供了绘制半对数和双对数坐标曲线的函数,半对数 semilogx 表示 x 轴以对数尺度绘图,半对数 semilogy 表示 y 轴以对数尺度绘图,loglog 表示 x 轴和 y 轴都以对数尺度绘图。

调用格式：`semilogx(x1,y1,'选项 1',x2,y2,'选项 2',...)`
`semilogy(x1,y1,'选项 1',x2,y2,'选项 2',...)`
`loglog(x1,y1,'选项 1',x2,y2,'选项 2',...)`

【例 3-17】 绘制 $y=5x^2$ 的自然对数和坐标(半对数和双对数坐标)曲线图。
 程序如下：

```
clear ;clc; close all;
x = 0:0.1:100;
y = 5 * x. * x;
subplot(2,2,1);
plot(x,log(y));
title('自然对数曲线');
grid on;
subplot(2,2,2);
semilogx(x,y);
title('半对数曲线(x轴刻度)');
grid on;
subplot(2,2,3);
semilogy(x,y);
title('半对数曲线(y轴刻度)');
grid on;
subplot(2,2,4);
loglog(x,y);
title('双对数曲线');
grid on;
```

绘制的对数坐标曲线如图 3-24 所示。

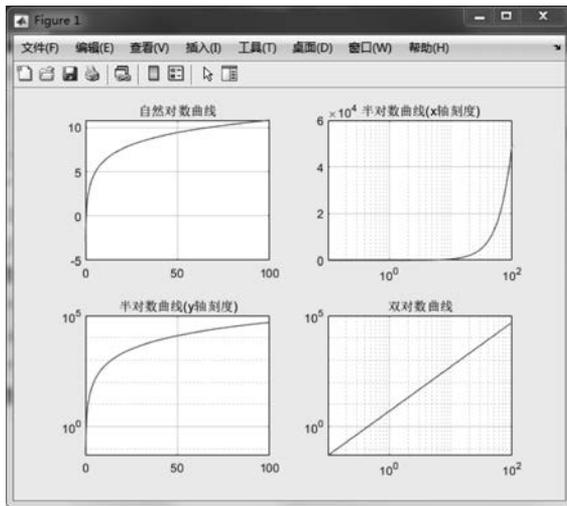


图 3-24 对数坐标曲线

3.4 其他形式二维特殊图形绘制函数

在 MATLAB 中,除了可以通过最基本的二维绘图函数 `plot`、直角坐标系其他二维图形绘制函数、常见的特殊二维图形函数等绘制图形外,还可以通过一些特殊函数绘饼图、直方图、散点图等特殊图形。

3.4.1 饼图

pie 函数用于绘制饼图。

调用格式: pie(x)

【例 3-18】 某次考试优秀、良好、中等、及格、不及格的人数分别为: 7,17,23,19,5,试用饼图进行成绩统计分析。

程序如下:

```
pie([7,17,23,19,5]);
title('饼图');
legend('优秀','良好','中等','及格','不及格');
```

绘制的饼图如图 3-25 所示。

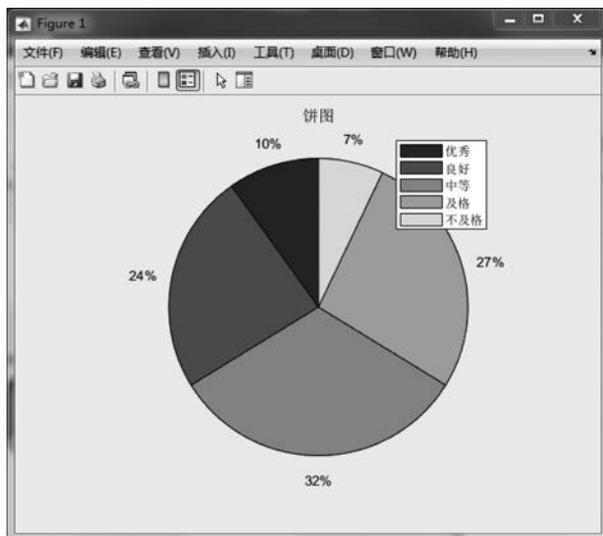


图 3-25 饼图

【例 3-19】 某统计数据所占百分比分为 10%、15%、20%、30%。试绘制该统计数据的饼图。绘制饼图并将第四个饼图提取出来。试绘制该统计数据的三维饼图,并且将最后一个提取出来。

程序如下:

```
%% 饼图
clear; clc; close all;
a = [10 15 20 30]; % 数据的占比
subplot(1,3,1); % 画出饼图,并且自动计算出百分比
pie(a);
subplot(1,3,2); % 将第四个饼图提取出来
pie(a, [0,0,0,1]);
subplot(1,3,3); % 画三维饼图,并且最后一个提取出来
pie3(a, [0,0,0,1]);
```

绘制的饼图如图 3-26 所示。

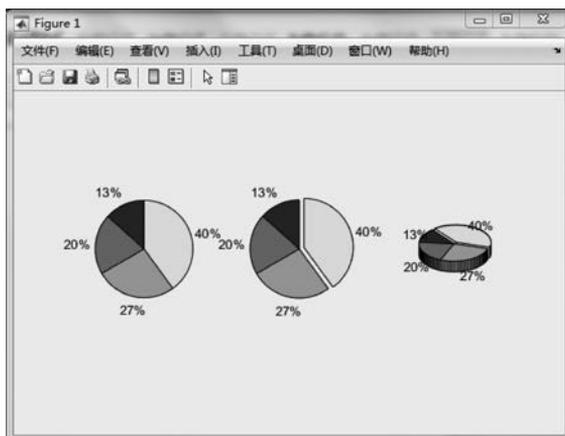


图 3-26 统计数据的饼图

3.4.2 直方图

在旧版本中 `hist` 函数用于绘制二维条形直方图,可以显示出数据的分布情况,但由于该函数适用于某些常规用途,总体能力有限,故新版本中用 `histogram` 函数替换了旧的 `hist` 函数。

调用格式: `histogram(x)`

【例 3-20】 绘制 1000 个随机数的直方图。

程序如下:

```

y = randn(1,1000);           % 由 randn 函数产生 1000 个随机数
subplot(2,1,1);
histogram(y,10);            % 包含 10 个长矩形
title('长矩形数 Bins = 10');
subplot(2,1,2);
histogram(y,50);           % 包含 50 个长矩形
title('长矩形数 Bins = 50');

```

绘制的随机数直方图如图 3-27 所示。

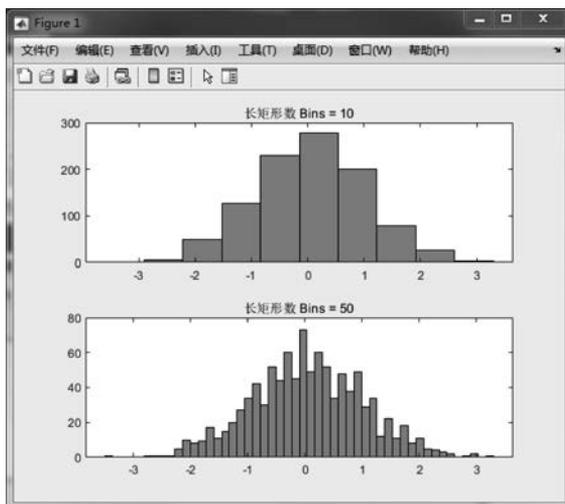


图 3-27 随机数直方图

3.4.3 填充区二维图

area 函数用来绘制填充区二维图。

调用格式: `area(x, Y)`

说明: 绘制 Y 中的值对 x 坐标的图。然后, 该函数根据 Y 的形状填充曲线之间的区域: ①如果 Y 是向量, 则该图包含一条曲线, `area` 填充该曲线和水平轴之间的区域。②如果 Y 是矩阵, 则该图对 Y 中的每列都包含一条曲线, `area` 填充这些曲线之间的区域并堆叠它们, 从而显示在每个 x 坐标处每个行元素在总高度中的相对量。

【例 3-21】 绘制向量 $x=[10\ 11\ 12]$, 矩阵 $Y=[21.6\ 25.4; 70.8\ 66.1; 58.0\ 43.6]$ 的填充区二维图。假设 x 为一个包含三个汽车经销商 ID 的向量, Y 表示每个车型售出的汽车数量。

程序如下:

```
x = [10 11 12];
Y = [21.6 25.4; 70.8 66.1; 58.0 43.6];
area(x, Y)
xlabel('汽车经销商 ID')
ylabel('汽车销售量')
legend({'模型 A', '模型 B'})
```

绘制的每个车型售出的汽车数量填充区二维图如图 3-28 所示。

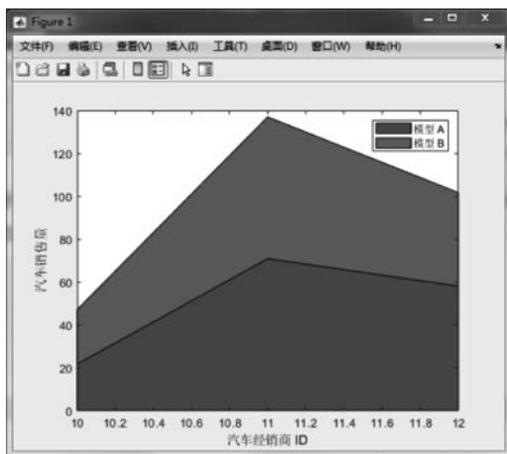


图 3-28 每个车型售出的汽车数量填充区二维图

3.4.4 散点图

scatter 函数用于绘制散点图。

调用格式: `scatter(x, y, s, c)`

说明: 以 x 的值为横坐标, 以 y 的值为纵坐标, 绘制散点。参数 s 设置散点的大小, 参数 c 设置散点的颜色。

【例 3-22】 绘制余弦加均匀分布随机数的散点图。

程序如下:

```
x = linspace(0,3 * pi,200);
y = cos(x) + rand(1,200);           % 余弦加均匀分布随机数
sc = 25;
c = linspace(1,10,length(x));
scatter(x,y,sc,c,'filled')
title('余弦加均匀分布随机数的散点图');
```

绘制的余弦加均匀分布随机数的散点图如图 3-29 所示。

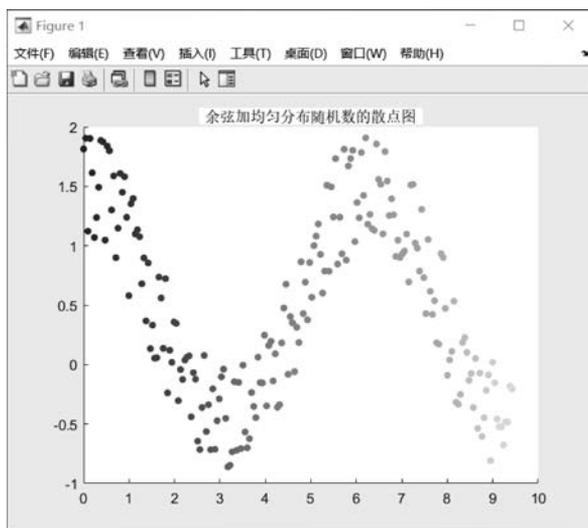


图 3-29 余弦加均匀分布随机数的散点图

3.4.5 散点图矩阵

plotmatrix 函数用来绘制散点图矩阵。

调用格式: plotmatrix(x)

说明: 该函数相当于 plotmatrix(x,x),当参数 x 为 $p \times n$ 的矩阵时,绘制出的是 $n \times n$ 个散点图。该图的对角块画出的是矩阵 x 的每列数据的频数直方图。

【例 3-23】 产生正态分布随机数,并绘制散点图矩阵。

程序如下:

```
X = randn(50,3);                   % 正态分布随机数,产生 50x3 矩阵
plotmatrix(X);
title('正态分布随机数散点图矩阵');
```

绘制的正态分布随机数散点图矩阵如图 3-30 所示。

3.4.6 箱形图或盒图

boxplot 函数用来绘制箱形图,即用箱形图可视化汇总统计量。

调用格式: boxplot(x)

如果 x 是向量,boxplot 绘制一个箱子。如果 x 是矩阵,boxplot 为 x 的每列绘制一个箱子。

在每个箱子上,中心标记表示中位数,箱子的底边和顶边分别表示第 25 个和第 75 个百分位数。虚线会延伸到不是离群值的最远端数据点,离群值会以“+”符号单独绘制。

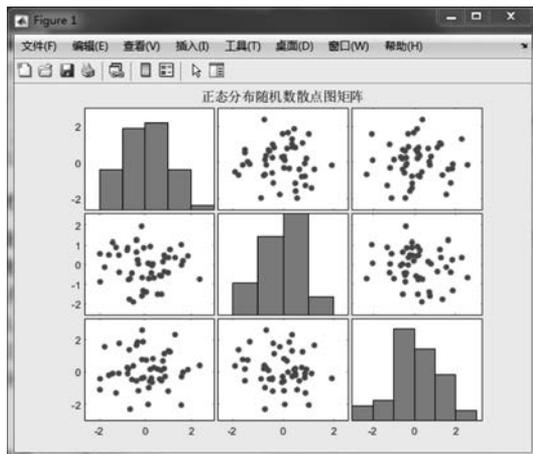


图 3-30 正态分布随机数散点图矩阵

【例 3-24】 已知一组测量数据,其矩阵形式为:

$$x = \begin{bmatrix} 0.7582 & 0.9809 & 0.9089 & 0.9481 \\ 0.9529 & 0.9365 & 0.8307 & 0.8270 \\ 0.9254 & 0.7601 & 0.9708 & 0.8859 \\ 0.8475 & 0.9449 & 0.9100 & 0.9198 \\ 0.8599 & 0.9539 & 0.7721 & 0.7754 \end{bmatrix}$$

绘制该矩阵的箱形图。

程序如下:

```
x = [0.7582 0.9809 0.9089 0.9841
      0.9529 0.9365 0.8307 0.8270
      0.9254 0.7601 0.9708 0.8859
      0.8475 0.9449 0.9100 0.9198
      0.8599 0.9539 0.7721 0.7754];
boxplot(x)
title('测量数据的箱形图')
```

绘制的测量数据的箱形图如图 3-31 所示。

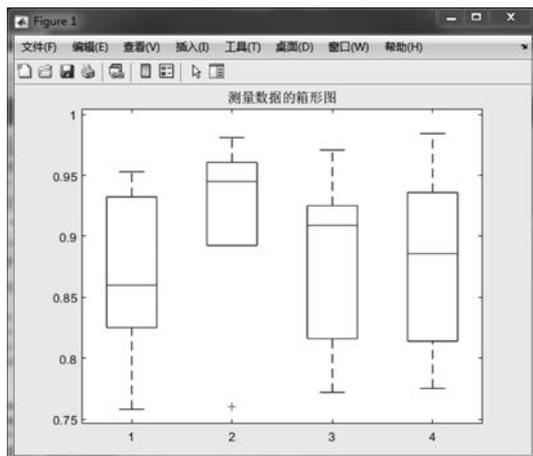


图 3-31 测量数据的箱形图

3.4.7 误差条

errorbar 函数可以绘制误差条图,它是统计学中常用的图形,涉及数据的“平均值”和“标准差”。

调用函数: errorbar(x,y,err)

说明: 绘制 y 对 x 的图,并在每个数据点处绘制一个垂直误差条,总误差条长度是误差 err 值的两倍。

【例 3-25】 绘制正弦函数 $\sin x$ 在 0 到 2π 范围内带垂直误差条的线图、带水平误差条的线图和带垂直和水平误差条的线图。误差值已给定。

程序如下:

```
x = 0:pi/10:2 * pi;
y = sin(x);
err = 0.3 * ones(size(y));           % 给定误差值
subplot(311)
errorbar(x,y,err)                   % 创建带垂直误差条的线图
title('带垂直误差条的线图');
subplot(312)
errorbar(x,y,err,'horizontal')     % 创建带水平误差条的线图
title('带水平误差条的线图');
subplot(313)
errorbar(x,y,err,'both','-s','MarkerSize',10,...
'MarkerEdgeColor','red','MarkerFaceColor','red') % 创建带垂直和水平误差条的线图
title('带垂直和水平误差条的线图');
```

绘制的误差条线图如图 3-32 所示。

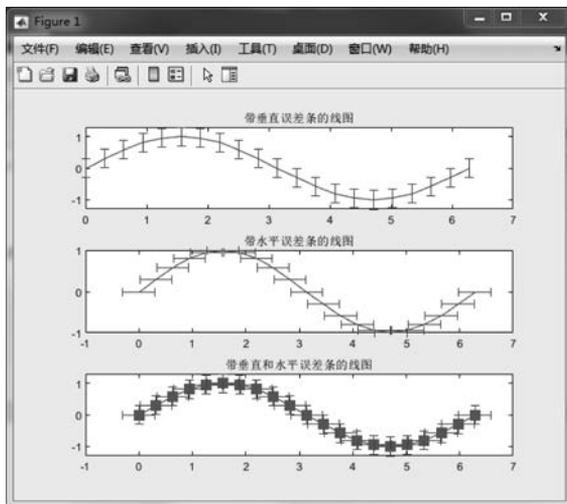


图 3-32 误差条线图

3.4.8 罗盘图

compass 函数用来绘制一个从原点出发、由 (x,y) 组成的向量箭头图形,也称罗盘图。

调用格式: compass(x,y)

【例 3-26】 绘制向量 $x=[1 \ -3 \ 5 \ -6 \ 8 \ 9]$, $y=[5 \ 7 \ -9 \ 12 \ 15 \ -9]$ 的罗盘图。
程序如下:

```
%% 绘制罗盘图
clear ;clc; close all;
x=[1 -3 5 -6 8 9];
y=[5 7 -9 12 15 -9];
figure;
compass(x,y,'r');
```

绘制的罗盘图如图 3-33 所示。

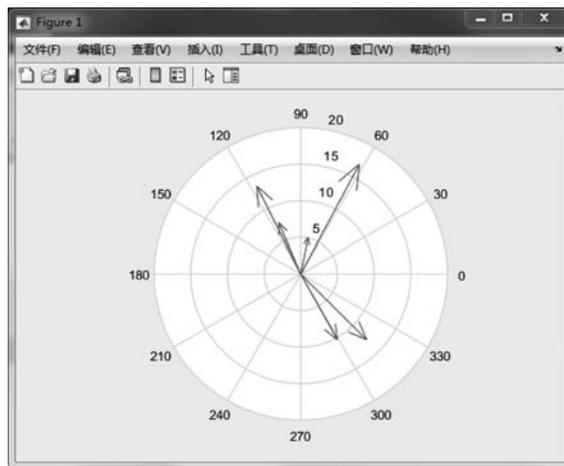


图 3-33 罗盘图

【例 3-27】 绘制复数 $3+2i$, $5.5-i$ 和 $-1.5+5i$ 的相量图。
程序如下:

```
compass([3+2i, 5.5-i, -1.5+5i]);
title('相量图');
```

绘制的复数相量图如图 3-34 所示。

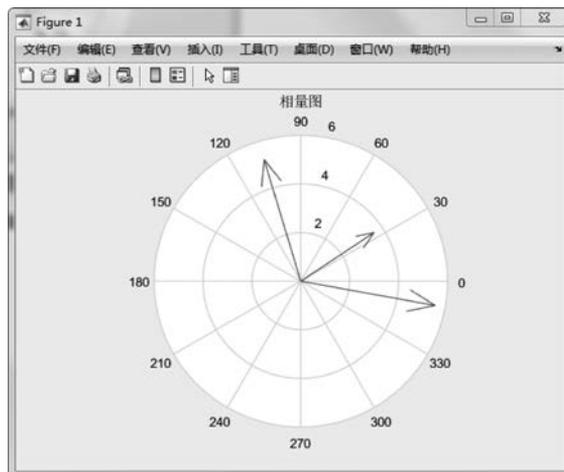


图 3-34 复数相量图

3.4.9 羽毛图

feather 函数用来绘制羽毛图(速度向量图),创建以 x 轴为起点的箭头。

调用格式: feather(x,y)

【例 3-28】 绘制向量 $x=[1\ 3\ 5\ 6\ 8\ 9]$ 和 $y=[5\ 7\ -9\ 3\ -5\ 2]$ 的羽毛图。

程序如下:

```
% % 绘制羽毛图
clear;clc;
close all;
x = [1 3 5 6 8 9];
y = [5 7 -9 3 -5 2];
figure;
feather(x,y);
```

绘制的向量羽毛图如图 3-35 所示。

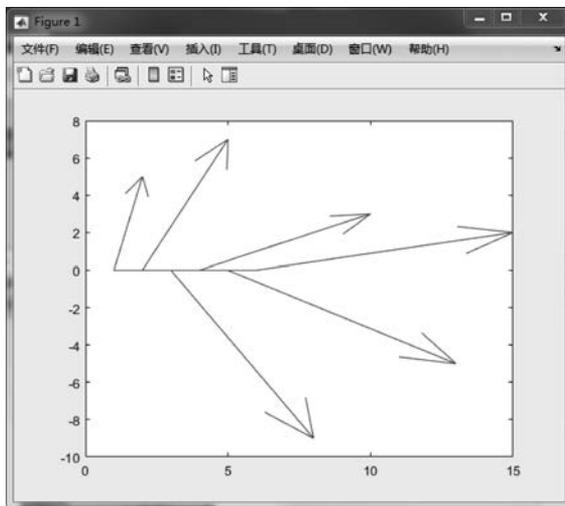


图 3-35 向量羽毛图

3.4.10 箭头图或向量场图

quiver 函数用来绘制箭头图或向量场图。

调用格式: quiver(x,y,u,v)

说明: quiver(x,y,u,v)函数可在坐标(x,y)处绘制向量场图,其中(u,v)为速度分量。

quiver(u,v)函数用来绘制向量场图。

【例 3-29】 绘制速度分量(u,v)的向量场图,其中 $u=\sin x$, $v=\cos x$ 。

程序如下:

```
clear; clc;
close all;
[X,Y] = meshgrid(-pi:pi/8:pi, -pi:pi/8:pi); % meshgrid 创建 x 和 y 形成的二维网格
U = sin(Y);
V = cos(X);
```

```
quiver(X,Y,U,V,'r')
title('向量场图');
```

绘制的速度分量的向量场图如图 3-36 所示。

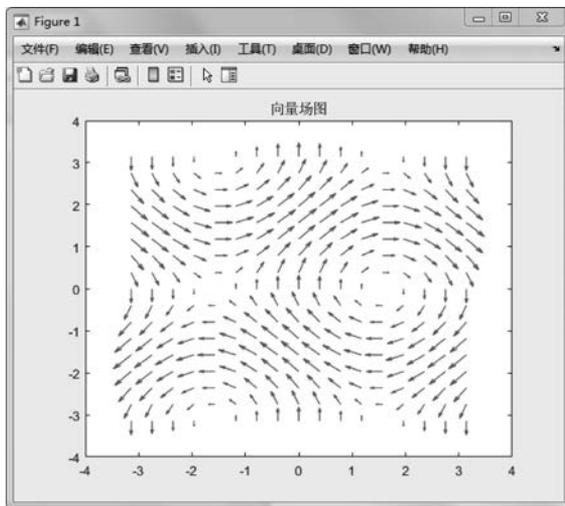


图 3-36 速度分量的向量场图

3.4.11 彗星图

函数 comet 用来绘制彗星图,可产生质点动画。

调用格式: comet(x,y)

说明: 该函数绘制由向量 x 和 y 确定的路线的彗星图; comet(x,y,p)函数设置彗星体的长度为 $p \times \text{length}(y)$,参数 p 的默认值为 0.1。

【例 3-30】 绘制 $y = \sin x$ 在 0 到 2π 范围内的彗星图。

程序如下:

```
% % 绘制彗星图,动态图
clear ;clc; close all;
x = 0:pi/50:2 * pi;
y = sin(x);
comet(x,y); % 画动态图
```

绘制的彗星图(动态)如图 3-37 所示。

3.4.12 伪彩图

pcolor 函数可以绘制伪彩图。

调用格式: pcolor(X,Y,C)

说明: 采用参数 X 确定横坐标,参数 Y 确定纵坐标,绘制矩阵 C 的伪彩图。pcolor(C) 为绘制矩阵 C 的伪彩图。

【例 3-31】 若横坐标 $X = [1 \ 2 \ 3; 1 \ 2 \ 3; 1 \ 2 \ 3]$,纵坐标 $Y = X'$,绘制矩阵 $C = [3 \ 4 \ 5; 1 \ 2 \ 5; 5 \ 5 \ 5]$ 的伪彩图。

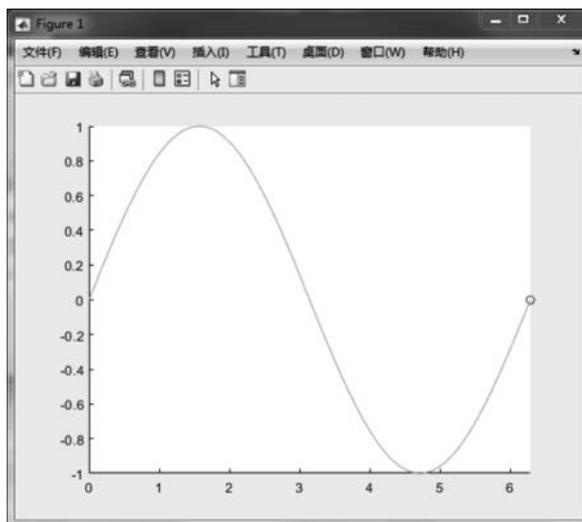


图 3-37 彗星图(动态)

程序如下:

```
X = [1 2 3; 1 2 3; 1 2 3];
Y = X';
C = [3 4 5; 1 2 5; 5 5 5];
pcolor(X,Y,C) % 采用参数 X 确定横坐标, 参数 Y 确定纵坐标, 绘制伪彩图
title('矩阵 C 的伪彩图');
```

绘制的由参数 X 确定横坐标, 参数 Y 确定纵坐标, 矩阵 C 的伪彩图如图 3-38 所示。

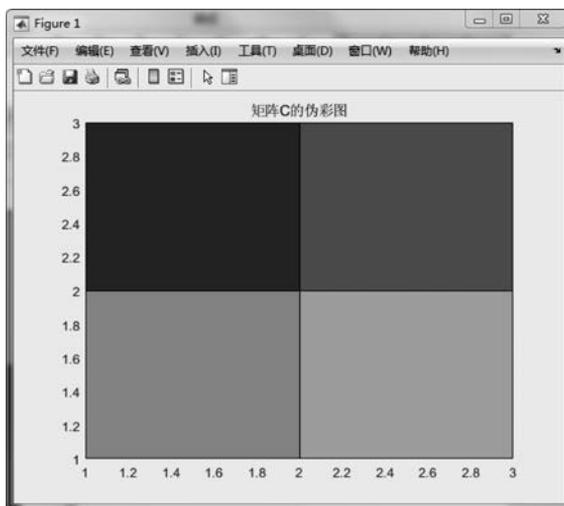


图 3-38 矩阵 C 的伪彩图



彩色图片

3.4.13 图形对象句柄

一个图形由很多对象组成, 包括图形对象 (figure object)、线条对象 (line object) 和坐标轴对象 (axes object), 如图 3-39 所示。

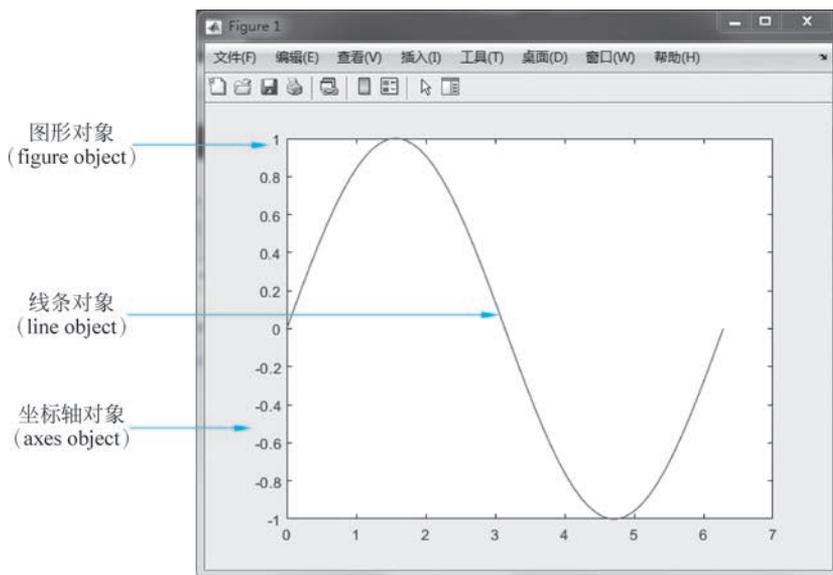
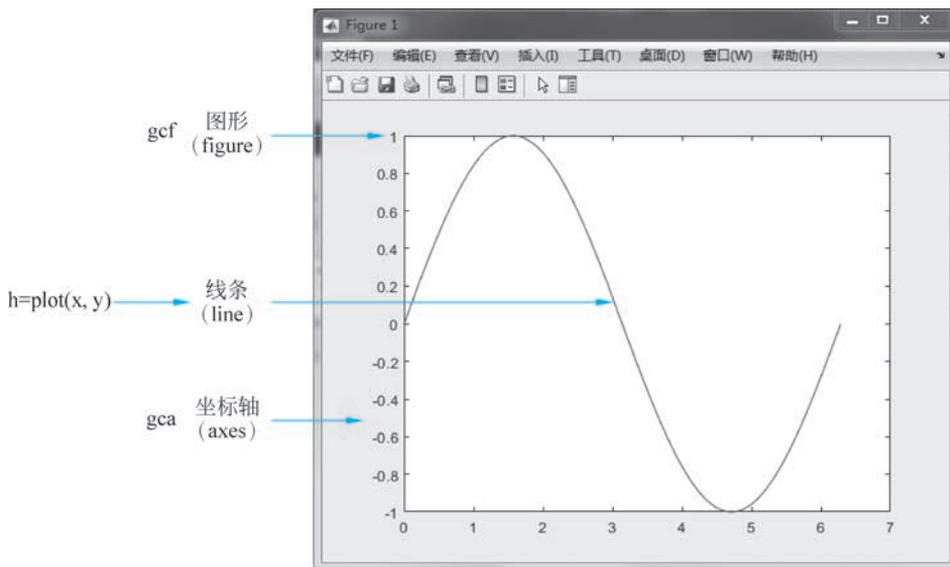


图 3-39 图形对象示意图

图 3-40 中 $h=\text{plot}(x,y)$ 表示返回由图形线条对象组成的列向量。在创建特定的图形线条后,可以使用 h 修改其属性。

h 可以是一个或多个图形线条对象,以标量或向量的形式返回。

图 3-40 $h=\text{plot}(x,y)$ 表示返回由图形线条对象组成的列向量

在 MATLAB 中,通过 gca 函数获得当前图形中坐标轴的句柄,常用函数如表 3-6 所示。

表 3-6 常用函数

函 数	用 途
gca	返回当前图窗中的当前坐标轴
gcf	返回当前图窗的句柄
allchild	查找指定对象的所有子级

续表

函 数	用 途
ancestor	图形对象的前代
delete	删除对象
findall	查找所有图形对象

利用 get 函数获取图形句柄的属性值。

调用格式: get(h, '属性')

该函数获取句柄为“属性”(PropertyName)的图形对象的属性值。

利用 set 函数设置图形句柄的属性值。

调用格式: set(h, '属性', 属性值)

该函数将句柄为“属性”的图形对象的属性值设置为“属性值”(PropertyValue)。

通过坐标轴句柄,可以利用函数 get 获取坐标轴的属性值,也可以通过函数 set 对坐标轴的属性值进行设置。

【例 3-32】 获取例 3-9 的双纵轴坐标曲线的坐标轴句柄和两条线条的句柄,并对坐标轴和线条线型属性进行设置。

程序如下:

```

%% 使用 plotyy 画两条曲线,修改曲线的属性
clear; clc; close all;
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);           % 获取 axes 句柄和两条线条的句柄
set(get(AX(1), 'Ylabel'), 'String', '左边 Y-轴'); % 设置线段 1 的 label
set(get(AX(2), 'Ylabel'), 'String', '右边 Y-轴'); % 设置线段 2 的 label
title('双纵轴坐标曲线');                       % 显示标题
set(H1, 'LineStyle', '--');                     % 设置两条线的风格
set(H2, 'LineStyle', ':');

```

获取例 3-9 的双纵轴坐标曲线的坐标轴句柄和两条线条的句柄如图 3-41 所示。

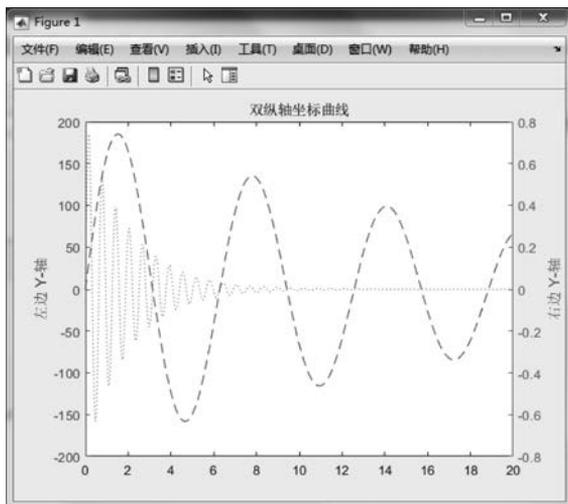


图 3-41 获取例 3-9 的双纵轴坐标曲线的坐标轴句柄和两条线条的句柄

本章小结

本章重点介绍 MATLAB 二维图形的绘制方法,包括最基本而且应用最为广泛的绘制二维图形的函数 plot、叠加图绘制、子图绘制等。此外,还介绍了其他二维图形绘制函数,包括火柴杆图、极坐标绘图、对数和半对数坐标系绘图、双纵轴坐标等。

【思政元素融入】

所谓一图胜千言,图形化的信息表示更加形象直观,使抽象数据的数量比较关系或变化趋势变得一目了然。掌握 MATLAB 最基本而且应用最广泛的二维绘图方法,可以将复杂的数据转变为直观的甚至动态可视化的图像,加深对事物本质的领悟和理解,通过抽象思维与形象思维的结合,提高学习的兴趣和积极性,有利于逻辑思维、辩证思维和创新思维能力的培养。