

# 第三章 数据的可视化

MATLAB 的数值计算功能是首屈一指的，数值计算结果是大量的离散化数据，人们很难从这些大量离散数据中观察到数据的本质特征。数据的可视化是把离散数据用多种图形方式显示出来，让人们在视觉上直观地看到这些数据的特征。

## 3.1 数据可视化基础

### 3.1.1 图形窗口

MATLAB 自动将图形画在图形窗口上，图形窗口相对于命令窗口是独立的窗口。图形窗口的属性由系统和 MATLAB 共同控制。当 MATLAB 上没有图形窗口时，绘图命令将新建一个图形窗口；当 MATLAB 已经存在一个或多个图形窗口时，MATLAB 一般指定最后一个图形窗口作为当前图形命令的输出窗口。下面介绍和图形窗口有关的 MATLAB 函数。

**1. figure**

用默认的属性创建新的图形窗口，并将新创建的窗口作为当前绘图窗口。

**2. figure('PropertyName',PropertyValue,...)**

用指定的属性创建图形窗口，并将新创建的窗口作为当前绘图窗口。其中'PropertyName'为属性名，PropertyValue 为'PropertyName'属性的值。

**3. figure(h)**

如果整数  $h$  不是某个已经存在的图形窗口的句柄，则创建新的图形窗口，并将  $h$  指定为新窗口的图形句柄；如果  $h$  是已经存在的图形窗口的句柄，则将图形句柄为  $h$  的图形窗口设置为当前图形窗口，并在屏幕的最前端显示。

**4. h = figure(...)**

创建图形窗口，并返回其图形句柄。

**5. subplot(m,n,p)**

将当前图形窗口分为  $m$  行  $n$  列个子窗口，并指定第  $p$  个子窗口为当前的绘图子窗口。子窗口序号  $p$  是按照行优先的次序排列的，这点和矩阵是不同的。如果不存在当前绘图窗口，则先创建一个新的绘图窗口，然后再划分子窗口。

**6. clf**

清除当前绘图窗口内的图形。

**例3.1.1** 运行下面的命令，可以在一个图形窗口中绘制四幅图形，如图3.1.1所示。

```
t=0:pi/20:2*pi; [x,y]=meshgrid(t); %准备绘图数据
subplot(2,2,1) %新建绘图窗口，划分为2×2子窗口，指定第1子窗口为当前绘图窗口
plot(sin(t),cos(t)), axis equal %axis equal表示纵横轴采用同一比例绘制
subplot(2,2,2), z=sin(2*x)+cos(2*y);
plot(t,z), axis([0 2*pi -2 2])
subplot(2,2,3), z=sin(x).^3.*cos(y);
plot(t,z), axis([0 2*pi -1 1])
subplot(2,2,4), z=(sin(x).^3)-(cos(y).^3);
plot(t,z), axis([0 2*pi -1 1]) %axis函数来指定每幅图的坐标轴显示的范围
```

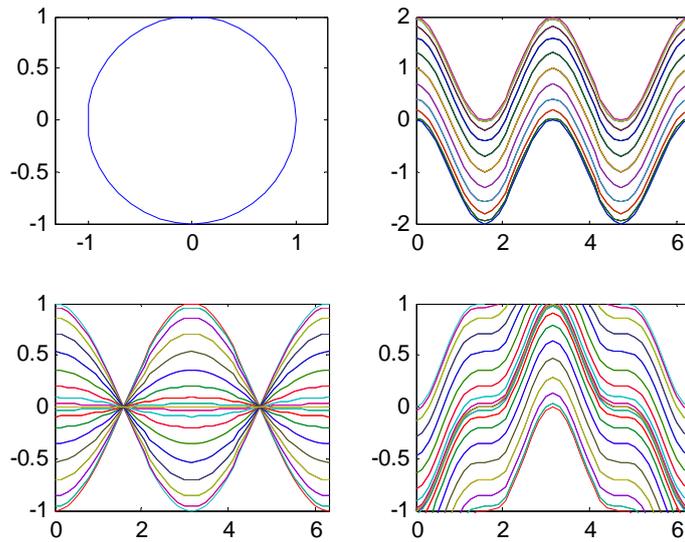


图3.1.1 曲线图形

### 3.1.2 离散数据的准备

MATLAB 是根据离散的数据来绘制图形的，因此调用绘图命令前要为绘图命令准备离散的绘图数据。对于已经存在的离散数据（如数值计算的结果数据）一般不需要再对这些离散数据进行处理，即可调用基本绘图函数。如果要绘制函数的图形，通常需要准备数据。

#### 1. 离散函数

如果函数为离散函数，则只需要在自变量区间内取遍所有离散点的值，从而获得离散的函数值即可。

如：对于函数  $f(n) = \frac{1}{n} + n + n^2$ ，在  $n=1,2,\dots,10$  区间我们可以用下面语句准备数据。

```
n=1:10;           %离散的自变量 n 的值
y=1./n+n+n.^2;   %离散的函数值
```

#### 2. 连续函数

绘制连续函数的曲线，也需要对连续函数的自变量取值区间进行离散采样。因为连续函数的自变量是连续无限可分的，因此采样间隔越近，数据信息就越全面，绘制出的图形就越接近于连续函数。我们可以在连续函数的自变量区间取足够多的离散值，代入连续函数中，从而获取离散的函数值。

如：对于函数  $f(x) = \sin x + \cos x$  在  $[0, 2\pi]$  区间内我们可以按下面语句准备数据。

```
x=linspace(0,2*pi,100); %在[0,2π]之间均匀取 100 个自变量值
y=sin(x)+cos(x);
```

### 3.1.3 数据可视化的基本步骤

数据可视化的一般步骤为：

- ①准备绘图需要的数据；
- ②指定绘图的窗口或者区域；
- ③选择线型、颜色、数据点形状等绘图属性；
- ④调用基本绘图命令；
- ⑤坐标轴控制，包括显示范围、刻度线、比例、网格线；
- ⑥标注控制，包括坐标轴名称、标题、相应文本等；
- ⑦其他更精确的控制，如颜色、视角、剪切和镂空等。

以上数据可视化的 7 个步骤不都是必须的，用户可以根据需要来选取相应的步骤。

## 3.2 二维图形

MATLAB提供了许多绘制二维图形的函数，它们的函数名称不同，但是函数的参数定义和plot函数完全相同，本节将只介绍plot函数。表3.2.1给出了部分二维绘图命令。

表3.2.1 绘制二维图形的命令

函数	轴的刻度方式
loglog	纵横轴均采用对数坐标
semilogx	纵轴采用线性坐标，横轴采用对数坐标
semilogy	纵轴采用对数坐标，横轴采用线性坐标
plotyy	在图的左右两侧分别建立纵坐标轴

### 3.2.1 基本二维图形

plot函数是最基本的二维图形命令，它是以MATLAB的内部函数形式出现的。MATLAB的其他二维函数中的绝大多数是以plot为基础构造的绘图命令。

#### 【调用格式】

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

#### 【说明】

##### (1) plot (y)

- 若y为实向量，则以y的元素为纵坐标，以相应元素下标为横坐标，绘制连线图。
- 若y为实矩阵，则按列绘制每列元素值相对其下标的连线图，图中的曲线数等于矩阵的列数。
- 若y为复矩阵，则分别以每列元素的实部和虚部为横纵坐标绘制多条连线图。

##### (2) ploy (x, y)

- 当y和x为同维向量，则以x为横坐标、y为纵坐标绘制连线图。
- 若x为向量，y为一维或和x同维的矩阵，则绘制多条不同颜色的连线图，曲线的个数等于矩阵的另一维，x作为这些曲线的共同横坐标。
- 若x为矩阵，y为向量，情况与上面相同，只是都是以y为共同的纵坐标。
- 若x和y为同维实数矩阵，则以x、y对应元素为横纵坐标分别绘制曲线，曲线数等于矩阵的列数。
- 若x和y为复数矩阵时，MATLAB将忽略虚数部分。

##### (3) plot (x1,y1,x2,y2,...)

每对x、y必须符合plot (x, y) 中的要求，不同对之间没有影响，命令将对每一对x、y绘制曲线。

【说明】以上的三种格式中，输入参数x、y都可以是表达式，但表达式的结果应符合上述格式要求。

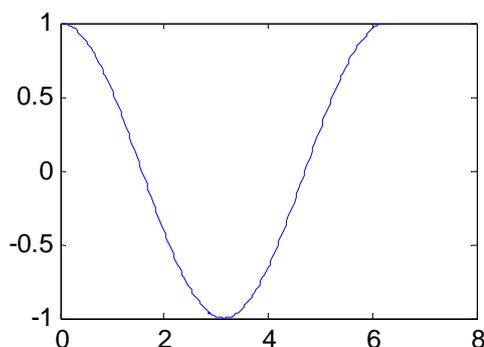


图3.2.1  $y = \cos(t)$  图形

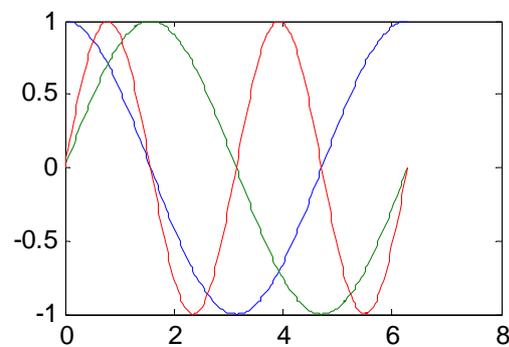


图3.2.2  $y, y_2, y_3$  图形

例3.2.2 运行下面的程序，得到如图3.2.1和图3.2.2所示图形。

```
t=0:pi/100:2*pi;  
y=cos(t); plot(t,y)
```

```
y2=sin(t); y3=2*cos(t).*sin(t);
figure, plot(t, y, t, y2, t, y3)
```

### 3.2.2 绘图属性控制

在调用plot函数时，可以指定线型、颜色和数据点的图标，其调用格式为

**plot (x, y, 'color\_linestyle\_marker')**

【说明】参数color\_linestyle\_marker 为一个字符串，由颜色、线型、数据点的图标组成。其中颜色字符定义如表3.2.2所示，线型字符定义如表3.2.3所示，数据点标记字符定义如3. 2. 4所示。

表3.2.2 颜色字符定义表

字符定义	颜色	字符定义	颜色
y	yellow (黄)	m	magenta(洋红)
c	cyan (青)	r	red (红)
g	green (绿)	b	blue (兰)
w	white (白)	k	black (黑)

表3.2.3 线型字符定义表

字符定义	线型	字符定义	线型
-	实线 (默认)	--	虚线
:	点划线	-.	点连线
none	无线		

表3. 2. 4 数据点标记字符定义表

字符定义	数据点标记	字符定义	颜色
+	加号	^	反勾号
v	勾号	>	大于号
<	小于号	o	小圆圈
*	星号	.	小黑点
x	叉号	pentagram	五角星
square	小正方形	hexagram	六角星
diamond	菱形	none	无标记

例如：plot (x,y,'y:square')，字符串“y:square”表示用黄色点划线和正方形点标记绘制曲线。

例 3.2.3 运行下列语句，观察 plot 的控制串的效果，如图 3.2.3 所示。

```
t=linspace(0,2*pi,360);
x1=4*cos(t); y1=4*sin(t); x2=8*cos(t); y2=4*sin(t); x=-8:8; y=0.5*x;
plot(x1,y1,'r-',x2,y2,'b--',x,y,'b*')
```

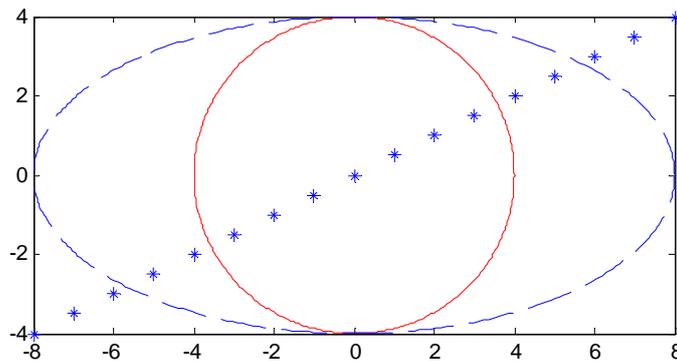


图 3.2.3 plot 的绘图属性

### 3.2.3 绘图窗口的属性控制

绘图窗口的属性可以由 MATLAB 和 Windows 共同控制,通常情况下是用默认绘图属性建立绘图窗口,用户可以通过 MATLAB 提供的命令和函数来修改绘图窗口的属性。

#### 1.坐标控制

可以控制坐标轴的各种属性,包括坐标轴的方向、范围、高宽比等。下面的指令是常用的坐标轴控制指令。

- **axis auto** 使用默认的坐标轴属性
- **axis on** 显示坐标轴背景
- **axis off** 取消坐标轴背景
- **axis([xmin,xmax,ymin,ymax])** 设定 x 和 y 轴的坐标值范围
- **axis equal** 横轴和纵轴的单位刻度相同
- **axis square** 横纵和纵轴的长度相同
- **axis tight** 把数据范围直接设置为坐标值范围
- **axis image** 采用相同的单位刻度,且坐标框紧贴数据范围

#### 2.刻度设置

缺省情况下, MATLAB 自动在坐标范围内生成均匀的刻度,用户也可以使用 set 命令改变刻度值,其使用方法如下:

- **set(gca, 'Xtick', xs, 'Ytick', ys)**

【说明】: xs 和 ys 分别是横轴和纵轴的刻度行向量。

#### 3.栅格控制

栅格是根据显示刻度绘制的虚线分格线,可以控制是否显示栅格。

- **grid on** 显示栅格
- **grid off** 隐藏栅格
- **grid** 切换栅格的显示和隐藏

#### 4.追加控制

绘图窗口绘制图形时,可以指定绘图是否采用追加(层叠)方式。追加方式是指每次的绘图指令都不会擦去图形窗口的原有图形,只是在原图上面层叠绘制新的图形。

- **hold on** 打开追加绘图方式
- **hold off** 关闭追加绘图方式

### 3.2.4 图形标注

很多时候,在图形上需要加上适当的文字说明,即标注,比如图名、坐标轴的名字、坐标点的值等。MATLAB 提供了图形标注的函数,主要的图形标注函数如下:

- **xlabel('string', 'FontSize', size)** 显示横坐标轴的名称
- **ylabel('string', 'FontSize', size)** 显示纵坐标轴的名称
- **title('string', 'FontSize', size)** 显示图形标题
- **text(x, y, 'string')** 在(x,y)坐标处显示字符串 string

其中'FontSize'表示字体尺寸属性, size为使用的字体尺寸,这2个输入变量可以省略,此时使用默认的字体系显示标注信息。

**例 3.2.4** 运行下列程序,观察长轴半径为 3.5,短轴半径为 1.5 的椭圆在不同坐标轴属性下的显示效果,如图 3.2.4 所示。

```
t=linspace(0,2*pi,100);  
x=1.5*sin(t); y=3.5*cos(t);
```

subplot(2,3,1), plot(x,y), axis normal, grid on, title('Normal and Grid')  
 subplot(2,3,2), plot(x,y), axis equal, grid on, title('Equal')  
 subplot(2,3,3), plot(x,y), axis square, grid on, title('Square')  
 subplot(2,3,4), plot(x,y), axis normal, grid off, title('Grid Off')  
 subplot(2,3,5), plot(x,y), axis tight, grid off, title('Tight')  
 subplot(2,3,6), plot(x,y), axis image, grid off, title('Image')

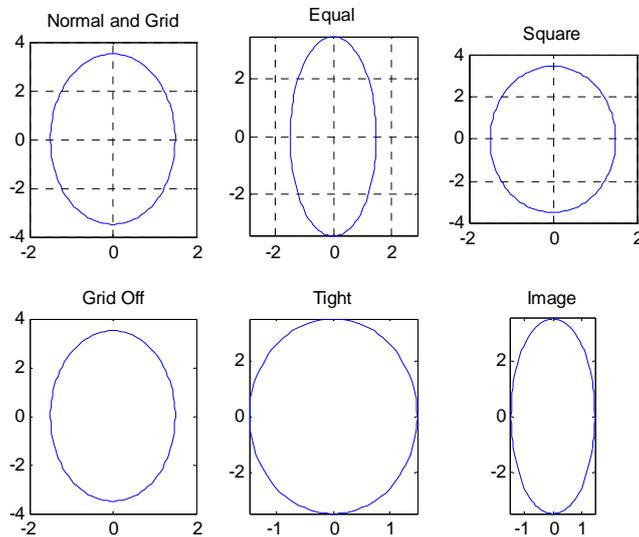


图3.2.4 椭圆在不同坐标轴属性下的显示效果

例3.2.5 运行下面的程序，并得到如图3.2.5所示图形。

```
x=-pi:pi/50:pi;
plot(x,cos(x));
xlabel('x=[-\pi \pi]', 'FontSize', 16)
ylabel('cos(x)', 'FontSize', 16)
title('余弦函数')
text(3*pi/4, cos(3*pi/4), '\leftarrow cos(x)=-.707')
text(-pi/2, cos(pi/2), '\leftarrow cos(x)=0')
text(pi/4, cos(pi/4), 'cos(x)=0.707\rightarrow', 'HorizontalAlignment', 'right')
```

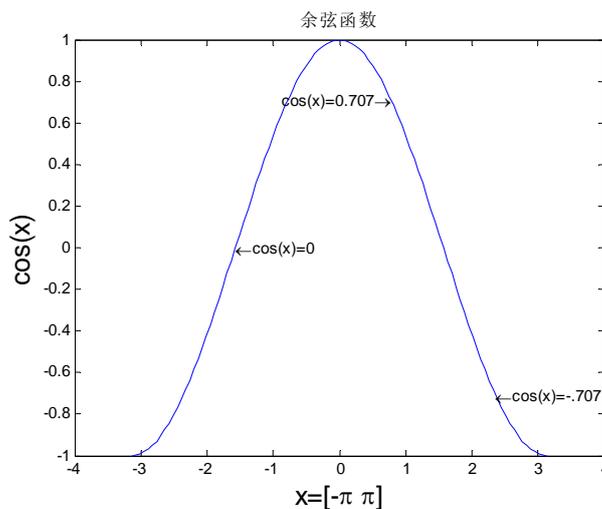


图3.2.5 绘图与标注

【说明】程序中使用了MATLAB约定的转义字符串指令。转义字符串指令用“\”引导，其后加上相应的转义命令名字。如“\pi”表示字符 $\pi$ ，“\rightarrow”表示右箭头图形，“\alpha”表示希腊字母 $\alpha$ 。

## 3.2.5 图形交互命令

MATLAB 提供了一些和鼠标操作相关的图形命令，用户可以通过这些命令和图形窗口进行交互。

### 1. ginput

- **[x,y]=ginput(n)**  
用鼠标从二维图形上拾取 n 个数据点的坐标(x,y)
- **[x,y]=ginput**  
用鼠标从二维图形上拾取多个数据点的坐标(x,y)，直到输入回车按键。
- **[x,y, button]=ginput(...)**  
除了返回数据点的坐标外，button 中还返回鼠标或者按键信息。1 表示鼠标左键，2 表示鼠标中间按键，3 表示鼠标右键，ASCII 码表示按键的键值。

### 2. gtext

格式: **gtext(arg)**

【说明】: 用鼠标把字符串或者字符串细胞数组 arg 放置到图形上。

### 3. zoom

格式:

- **zoom off**            关闭当前图形的变焦功能
- **zoom on**            打开当前图形的变焦功能
- **zoom out**           是图形返回初始尺寸
- **zoom(fact)**        设置变焦因子（每次变焦的倍数），缺省的变焦因子为 2

【说明】如果图形在变焦状态下，用户可以通过鼠标来放大或者缩小图形，包括用鼠标左键放大、用鼠标右键缩小、用鼠标选定显示范围等操作。

## 3.3 三维图形

### 3.3.1 基本三维图形

plot3 为基本的三维绘图函数，其使用方法和 plot 函数基本相同。

【调用格式】

`plot3(X,Y,Z, 's')`

【说明】

1. 当 X, Y, Z 为同向量组时，绘制以 X、Y、Z 的元素为 x、y、z 坐标的三维曲线。
2. 当 X, Y, Z 为同维数组时，分别以其对应的列向量元素为 x、y、z 坐标绘制多条三维曲线，曲线的条数等于矩阵的列数。
3. 字符串 s 表示颜色、线型和点形状控制，其定义和 plot 函数相同。
4. 可以使用多组参数绘图，每组参数之间没有约束关系，如 `plot3(x1,y1,z1,x2,y2,z2)`。
5. plot3 主要用来绘制参数方程决定的三维曲线。

例 3.3.1 设方程组为 
$$\begin{cases} x(t) = \sin(t) \\ y(t) = \cos(t) \\ z(t) = t \end{cases}, \quad 0 < t < 6\pi$$
，试绘制由变量 x, y, z 确定的三维曲线。

解：运行下列程序，程序绘制的图形如图 3.3.1 所示。

```
t=0:pi/100:6*pi;  
x=sin(t); y=cos(t); z=t;
```

```
plot3(x,y,z)
axis square; grid on
```

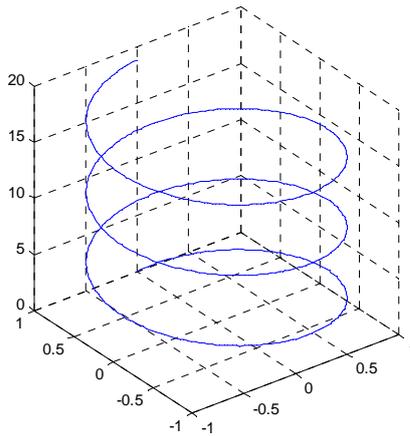


图 3.3.1 参数方程组的三维曲线

### 3.3.2 三维网线和曲面

三维网线和曲面的绘制比三维曲线要复杂。这主要表现在数据的准备和三维图形的色彩、光照、视点和消隐控制。

#### 1. 三维网线图

三维网线是指绘图点之间用曲线连接起来。MATLAB 中绘制网线图的函数是 `mesh`。

- `mesh(Z)` 用矩阵  $Z$  的列、行下标作为  $x$  轴和  $y$  轴变量，画网线图
- `mesh(X,Y,Z)` 分别用  $X, Y, Z$  作为  $x, y, z$  轴的坐标进行绘图
- `mesh(X,Y,Z,C)` 用数据  $X, Y, Z$  和颜色属性矩阵  $C$  绘图

#### 2. 曲面图

曲面图是指绘图点之间用曲面连接起来。MATLAB 中绘制曲面图的函数是 `surf`。

【调用格式】

`surf(Z)`

`surf(X,Y,Z,C)`

【说明】其调用格式和 `mesh` 函数完全相同。

例 3.3.2 绘制由函数  $z = \sqrt{x^2 + y^2}$  在自变量取  $x \in [-8, 8], y \in [-8, 8]$  范围内的网线图和曲面图。

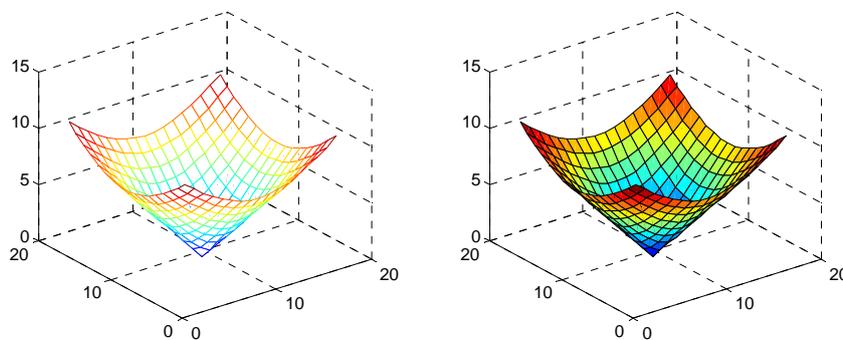


图 3.3.2 网线图和曲面图

解：运行下面的程序，得到网线图如图 3.3.2 左图所示，曲面图如图 3.3.2 右图所示。

```
x=-8:8; y=x;
[X,Y]=meshgrid(x,y); %生成离散采样格点
```

```
Z=sqrt(X.^2+Y.^2);
subplot(1,2,1); mesh(Z); grid on;
subplot(1,2,2); surf(Z); grid on
```

### 3.4 常用函数的绘图

绘制一个连续函数的曲线，要为绘图函数准备数据，这会使函数曲线的绘制工作变得比较麻烦。为了让用户更轻松的实现函数的可视化，MATLAB 提供了一些简洁的绘图指令，这些指令的特点是：无需准备数据就直接画出字符串函数或者符号函数的图形。这个系列的函数的名字都用“ez”开头。下面简单介绍常用的两个函数，其他函数读者可以参阅 MATLAB 的超文本帮助。

#### 3.4.1 一元函数的绘图

【调用格式】

```
ezplot(F,[xmin,ymin])
```

【说明】

1. F 是一元函数的字符表达式、符号函数、函数 M 文件或者内联函数，且只能有一个自变量。
2. [xmin, xmax]是自变量的取值区间。如果不指定自变量取值范围，MATLAB 会根据函数自动选取自变量的值。

例 3.4.1 绘制  $y(t) = 1 - 1.5e^{-0.4t} \sin(t + 0.6)$  在  $0 < t < 20$  区间的函数曲线。

解：运行以下语句，曲线如图 3.4.1 所示。

```
syms t; %定义为 t 符号
y=1-1.5*exp(-0.4*t)*sin(t+0.6);
ezplot(y,[0,20]);
axis tight; grid on;
```

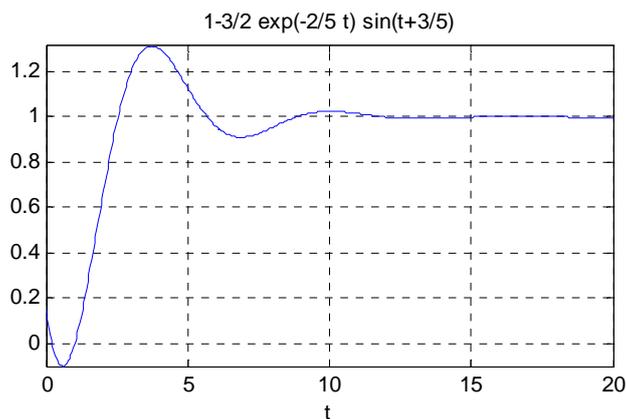


图 3.4.1 一元函数曲线

#### 3.4.2 二元函数的绘图

【调用格式】

```
ezsurf(F, domain, ngrid) 在指定矩形区域上，用指定格点数画二元函数曲面
ezsurf(F, domain, 'circ') 在极坐标中绘制二元函数曲面
```

### 【说明】

1.  $F$  是字符表达式、符号函数、函数 M 文件或者内联函数； $F$  是二元函数，只能包含 2 个自由变量；
2. 若  $\text{domain}=[a, b]$ ，则自变量取值范围为  $a \leq x \leq b$ ,  $a \leq y \leq b$ ；
3. 若  $\text{domain}=[a, b, c, d]$ 时，自变量取值范围是： $a \leq x \leq b$ ,  $c \leq y \leq d$ 。
4.  $\text{ngrid}$  是绘图的格点数。格点数越多，图形越细腻，默认的  $\text{ngrid}$  值是 60。

例 3.4.2 在极坐标系中绘制  $z = xy$  的图形。

解：执行下列语句，结果如图 3.4.2 所示

```
ezsurf('x*y', 'circ'); shading flat; view([-10,30])
```

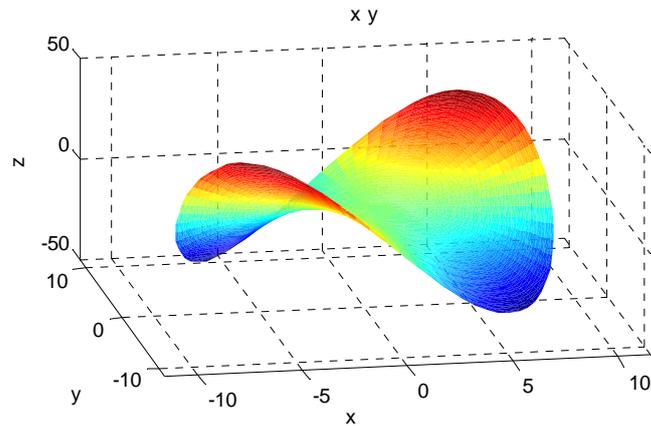


图 3.4.2 二元函数确定的曲面

## 3.5 数据可视化的其他方面

MATLAB 的数据可视化功能非常强大，其提供的函数多种多样。本节简单地介绍除了前面各节介绍的功能外，其他方面的数据可视化操作。

### 3.5.1 特殊图形

MATLAB 提供了一些常用的特殊图形的绘制函数，下面仅仅列出相关函数名字，读者使用这些函数时请参考帮助信息。

直方图	bar	bar3	barh	bar3h
面积图	area			
统计频数直方图	hist	rose		
填色图	fill	fill3		
火柴杆图	stem	stem3		
矢量图	compass	feather	quiver	quiver3
等高线图	clabel	contour	contour3	countourf
彩带图	ribbon			

### 3.5.2 三维图形的精细控制

为了能获得更好的视觉效果，MATLAB 提供了对三维图形更为精确的控制函数。下面列出和这些精细控制有关的函数或者命令。

1. 透视与消隐：处理被遮盖在下面的那部分图形

- hidden
- 2. 视点控制：从不同的视角看到的图形效果
  - view
- 3. 旋转：图形旋转一定角度后的效果
  - rotate
- 4. 色彩控制：用不同的颜色组显示图形
  - colordef whitebg colormap shading
- 5. 照明和材质处理：有光源照射、反射时图形的效果
  - light lighting material surfl
- 6. 透明处理：采用透明化技术揭示复杂图形的内部结构
  - alpha alim alphamap

### 3.5.3 图像

图像数据文件主要保存的是图像上各个像素的颜色值，根据图像颜色数的不同，有 8 位色（256 色）图像、16 位色（65535）图像等。MATLAB 的数值数组的数据类型是 double 双精度型，每个元素占用 64 位，用数值数组来保存颜色值显然是浪费内存的。MATLAB 中定义了 unit8 和 unit16 数据类型用以保存颜色值，由 unit8 或者 unit16 构成的矩阵用来保存图像各个像素的颜色值，即保存图像数据（8 位图像、16 位图像）。

和图像显示和读取有关的函数包括：

image colormap imfinfo imread imwrite

### 小结

MATLAB 提供了强大的图形处理能力，提供给用户两类图形指令。一类是“高级绘图命令”，这些高级绘图命令易于理解和使用，可以绘制一般科技绘图软件所能绘制的几乎所有图形。另外一类是“句柄图形”，句柄图形属于底层的绘图控制，属于 MATLAB 的面向对象的图形系统，用句柄图形可以实现对图形显示的更精密的控制。

本章对常用的高级绘图命令的格式和使用方法做详细的介绍，另外还简单介绍了复杂的可视化操作涉及到的函数和命令。