

MATLAB 数据可视化的基础方法

本章学习目标

- (1) 掌握二维图形的绘制方法。
- (2) 掌握给图形添加标注的方法。
- (3) 掌握三维图形的绘制方法。
- (4) 熟悉 MATLAB 探查数据特性的其他方法。

数据可视化的基础方法是在二维/三维空间用点、线、面等展示数据的特性。本章首先 介绍 MATLAB 绘制二维图形的基本方法,以及在二维图形上添加标注和调整图形颜色的 方法,然后讲解绘制三维曲线和三维曲面的方法。最后介绍利用 MATLAB 提供的工具探 查数据的基本属性。

5.1 绘制二维图形

二维图形是绘制在平面上的图形,构成二维图形的主要元素是点和线。绘制二维图形, 通常采用笛卡儿坐标系,还可采用对数坐标系、极坐标系。

5.1.1 绘制曲线

MATLAB绘制二维曲线的基本方法是将平面上的数据点用线段连接起来。

1. plot 函数

plot 函数用于绘制基于笛卡儿坐标系的曲线,基本调用格式为

plot(X, Y)

其中,参数 X 和 Y 分别存储要绘制的数据点的横坐标和纵坐标。

1) 输入参数是向量

plot 函数的输入参数 X 和 Y 通常是长度相同的向量,绘制曲线时,用 X、Y 的元素值作为数据点的横、纵坐标,用线段将各数据点连接起来,形成一条曲线。

例 5.1 绘制曲线 $\begin{cases} x = \sin t + \sin 2t \\ y = \cos t - \cos 2t \end{cases}$,其中, $t \in [0, 2\pi]$ 。

先构造向量t,将t代入表达式计算,得到向量x、y,然后使用x、y作为参数,调用 plot

第5章 MATLAB数据可视化的基础方法

函数绘制曲线。程序如下。

```
t=linspace(0, 2*pi, 200);
plot(sin(t)+sin(2*t), cos(t)-cos(2*t));
```

运行程序,将打开一个图形窗口,在其中绘制出二维曲线,如图 5.1(a)所示。

如果单位长度内的数据点太少,绘制的图形不能反映数据的变化特性,例如,将例 5.1 程序的第一条语句改为"t=linspace(0,2 * pi,10);",即只定义 10 个数据点,绘制的图形如 图 5.1(b)所示。



图 5.1 例 5.1 程序绘制的二维曲线

单位长度的数据点越多,曲线越光滑;但数据点过多,又会影响运算速度,浪费存储和计 算资源。在实际应用中,可在调试程序时,观察图形随数据点个数的变化,选取合适的数据 点个数。

2) 输入参数是矩阵

若 plot 函数的输入参数 X、Y 是大小为 m×n 的矩阵,则依次用 X、Y 对应列的元素值 为横、纵坐标绘制曲线,每列元素定义一条曲线数据点的坐标,曲线条数等于矩阵的列数。 例如,在同一坐标区中绘制三条幅值不同的正弦曲线,程序如下。

x = linspace(0, 2 * pi, 100); y = sin(x);	
<pre>plot([x; x; x]', [y; y*2; y*3]')</pre>	%或 plot([x', x', x'],[y', y'*2, y'*3])

输入参数[x; x; x]、[y; y*2; y*3]都是大小为100×3的矩阵。如果不转置,即用 [x; x; x]、[y; y*2; y*3]作为输入参数,将输出100条折线,每条折线上有3个数据点。

如果输入参数 X、Y 一个是向量,一个是矩阵,则矩阵第1 维度或第2 维度的大小应与向量的长度相同。当矩阵的行数与向量长度一致时,绘图时默认用矩阵的每列元素作为数据点的坐标;当矩阵的列数与向量长度一致时,绘图时默认用矩阵的每行元素作为数据点的坐标。例如,在同一坐标区中绘制三条幅值不同的正弦曲线,也可以使用以下程序。

147

x = linspace(0, 2 * pi, 100); y = sin(x);

plot(x, [y; y*2; y*3])%或plot(x, [y; y*2; y*3]') 或plot(x', [y; y*2; y*3])

MATLAB 数据可视化(微课视频版)

3) 输入参数只有一个

若输入参数 Y 是实型向量,则以该向量元素的索引为横坐标、元素值为纵坐标绘制出 一条曲线;若输入参数 Y 的元素是复数,则分别以元素实部为横坐标、虚部为纵坐标绘制一 条曲线。例如,绘制 0°~360°的正弦曲线,可以使用以下命令。

```
x = 0:1:360;
plot(sind(x)) %或 plot(x+1i*sind(x))
```

若输入参数 Y 是矩阵, 且 Y 的所有元素是实数,则以每列元素的行下标为横坐标、以每 列元素的值为纵坐标绘制多条曲线,曲线条数等于输入参数 Y 的列数; 若 Y 是复数矩阵,则 按列分别以元素实部、虚部为横、纵坐标绘制多条曲线。例如,绘制三个同心圆,程序如下。

```
t = linspace(0,2*pi,100);
x = cos(t)+1i*sin(t);
plot([x.', 2*x.', 3*x.'])
```

圆是一条闭合曲线,曲线起点与终点坐标相同。

4) 多对输入参数

当 plot 函数有多对输入参数,且都为向量时,即

plot(x1, y1, x2, y2, ..., xn, yn)

其中,x1和 y1、x2和 y2、…、xn和 yn 组成 n 组向量对,分别以每一组向量对的元素为横、纵 坐标绘制出一条曲线。采用这种格式时,同组的向量长度必须一致,不同向量对的长度可以 不同。例如,在同一坐标区中绘制曲线 $x = \cos t + t \sin t$, $t \in [0, 3\pi]$ 和 $y = \sin t + t \cos t$, $t \in [0, 2\pi]$,程序如下。

t1 = linspace(0,3*pi, 90); x = cos(t1)+t1.*sin(t1); t2 = linspace(0,2*pi, 50); y = sin(t2)-t2.*cos(t2); plot(t1,x, t2,y);

2. fplot 函数

使用 plot 函数绘图时,先要取得横、纵坐标,然后再绘制曲线,横坐标往往采取等间距。 在实际应用中,表达两个变量关系的表达式随着自变量的变化趋势未知,或者在不同区段频 率特性差别大,此时使用 plot 函数绘制图形,如果数据点的间距设置不合理,则无法完整反 1

映表达式的原有特性。例如,绘制曲线 $\sin \frac{1}{x}$, $x \in [0, 0.1]$, 程序如下。

148

```
x = eps:0.005:0.1;
plot(x, sin(1./x))
```

因为x为0时,运算 $\frac{1}{x}$ 无数学意义,因此x的最小值设为 eps(eps 表示极小值 2⁻⁵²)。

第5章 MATLAB数据可视化的基础方法■

图 5.2(a)是步长为 0.005 时绘制的图形,图 5.2(b)是步长为 0.001 时绘制的图形。图 5.2(b)显示,在 0~0.1 范围有多个振荡周期,因变量的值变化大,而 0.04 以后变化较平缓。当自变量的间距设置为 0.005 时,绘制的曲线(图 5.2(a))没有反映出 0.01~0.04 区段的变化规律。



MATLAB的 fplot 函数可以很好地解决此类问题。fplot 函数根据输入参数表达式的 变化特性自适应地调整自变量元素的间距。因变量的值变化缓慢的区段,自变量元素间距 较大;因变量的值变化剧烈的区段,自变量元素间距较小。

1) 基本调用格式

fplot 函数的输入参数是表达式,表达式描述数据点纵坐标与横坐标的关系。fplot 函数的调用格式如下。

fplot(fun, lims)

其中,输入参数 fun 是定义数据点纵坐标的表达式,通常采用函数句柄的形式。lims 为 fun 的自变量的取值范围,用二元行向量[xmin,xmax]描述,省略时,lims 默认为[-5,5]。例 如,绘制图形 sin $\frac{1}{x}$,x \in [0, 0.1],可以使用以下命令。

>> fplot(@(x) sin(1./x),[0,0.1])

执行命令,绘制的图形如图 5.3 所示。图形显示,在 0~0.01 区间,函数对应的图形剧烈 振荡变化,这是上面调用 plot 函数绘制的图形完全没有表现出来的情形。对于某些数学上 无意义的数据点,fplot 函数绘制图形时,会根据前后数据点的变化趋势拟合一个符合实际 情况的数据点,例如,此例中 *x* 为 0 的点。

2) 其他调用格式

如果数据点的横、纵坐标都是与某个变量相关的表达式,则 fplot 函数的调用格式如下。

fplot(funx, funy, lims)

其中,输入参数 funx、funy 是仅有一个自变量的表达式,通常采用函数句柄的形式。lims 指定这个变量的取值范围,用二元向量[tmin,tmax]表示,默认为[-5,5]。例如,例 5.1 中 x、

MATLAB 数据可视化(微课视频版)





>> fplot(@(t) sin(t) + sin(2 * t), @(t) cos(t) - cos(2 * t), [0,2 * pi])

plot 函数的输入参数是数据点的坐标值,fplot 函数的输入参数是描述数据点坐标的表达式。使用 plot 函数和 fplot 函数求解同一个问题,方法有区别。

例 5.2 已知 $f(x) = \begin{cases} 1 - e^{-1(1-x)^2}, & x < 1 \\ \ln x, & x \ge 1 \end{cases}, x \in [0,5]. \end{cases}$

分别调用 plot 函数、fplot 函数绘制曲线。

(1) 调用 plot 函数绘制图形,先计算出数据点的横、纵坐标。定义一个匿名函数 fun, 存储题目中的表达式。将表示横坐标的变量 *x* 作为参数代入函数 fun,计算纵坐标。由于 运算对象是变量 *x* 的各个元素,因此,表达式中要使用数组运算符(.*和./)。程序如下。

```
fun = @(x) (1-exp(-(1-x).^2)).* (x<1)+log(eps+x).* (x>=1);
x = 0:0.1:5;
plot(x, fun(x))
```

说明: log0 无数学意义,为了避开计算 log0,程序中将 x 加上 eps。运行以上程序,绘制的图形如图 5.4 所示。



第5章 MATLAB数据可视化的基础方法

(2) 调用 fplot 函数绘制图形,不需要计算出数据点的横、纵坐标,而需要指定绘图区间。程序如下。

```
fun = @(x) (1 - exp(-(1-x).^2)). * (x<1) + log(x). * (x >= 1);
fplot(fun, [0,5])
```

或者使用符号函数形式,程序如下。

```
syms x;
fplot(1-exp(-(1-x)^2), [0,1-eps])
hold on
fplot(log(x), [1,5])
```

使用符号函数形式时,分两次绘制图形,两段曲线各使用一种颜色。调用 fplot 函数绘制第二个图形时,系统会先清除前面的图形,通过 hold on 命令保持前一个图形,再继续绘制第二个图形。

3. fimplicit 函数

如果给定了定义曲线的显式表达式,可以根据表达式计算出所有数据点坐标,调用 plot 函数绘制图形;或者用函数句柄作为参数,调用 fplot 函数绘制图形。但如果曲线用隐函数 形式定义,如 $x^3+y^3-5xy+\frac{1}{5}=0$,y 没有直接表示为自变量x 的表达式,则不适合用 plot 函数和 fplot 函数绘制图形。MATLAB 提供了 fimplicit 函数绘制隐函数图形,其调用格式 如下。

fimplicit(f, [xmin, xmax, ymin, ymax])

其中,输入参数f存储表达式,可以用匿名函数、函数句柄、符号函数表示;第2个输入参数中,xmin和 xmax 指定绘图区水平方向的最小值和最大值,ymin和 ymax 指定竖直方向的最小值和最大值。当 ymin和 ymax 省略时,默认水平方向和竖直方向的绘图区间均为 [xmin, xmax]。当省略第2个输入参数时,水平和竖直方向的绘图区间默认为[-5,5]。

例 5.3 绘制曲线 $x^3 + y^3 - 5xy + \frac{1}{5} = 0$,其中, $x \in [-5,5]$, $y \in [-5,5]$ 。 使用匿名函数形式定义 fimplicit 函数的输入参数,命令如下。

>> fimplicit(@(x,y)x.*x.*x+y.*y.*y-5*x.*y+1/5)

执行命令,输出的图形如图 5.5 所示。

也可以使用符号函数形式定义 fimplicit 函数的输入参数,命令如下。

```
>> syms x y
>> fimplicit(x*x*x+y*y*y-5*x*y+1/5)
```

MATLAB 数据可视化(微课视频版)



4. 曲线样式

在一个坐标区绘制多条曲线时,为了加强对比效果,可设置不同线型和数据点标记区分曲线。

1) 线型

线型用字符串或字符向量描述,表 5.1 列出了 MATLAB 线条的线型可取值,未指定时,默认线型为实线。下面以 plot 函数为例,说明线条线型的设置方法。

通常在调用 plot 函数、fplot 函数、fimplicit 函数绘制曲线时,用参数形式指定线型,例如:

```
x = linspace(0,2*pi,100);
plot(x,sin(x),':', x,sin(2*x),'--', x,sin(3*x),'-.')
```

线型取值	描述	表示的线条	
"_"	实线		
""	虚线		
": "	点线		
"_ "	点画线		
"none"	无线条		

表 5.1	MATLAB	线条的线型
-------	--------	-------

也可以在绘制图形后,通过图形对象的 LineStyle 属性设置线型。例如:

```
phs = plot(x, [sin(x); sin(2 * x); sin(3 * x)]);
phs(1).LineStyle=":";
phs(2).LineStyle="--";
phs(3).LineStyle="-.";
```

152

除了设置线型,还可以通过 LineWidth 属性设置线条宽度,其值为正,以磅(1磅=1/72

英寸,1英寸=2.54 厘米)为单位,省略时,默认为0.5。例如,设置第2条曲线的线宽为2,命 令如下。

phs(2).LineWidth = 2;

2) 数据点标记

数据点标记用字符串或字符向量描述,表 5.2 列出了 MATLAB 数据点的标记可取值, 未指定时,不显示数据点标记。

标识符	标 记 样 式	标识符	标 记 样 式
'+'	加号	's'	正方形
'o'	空心圆	'd'	菱形
'*'	星号	1/1	上三角形
::	点	'v'	下三角形
'x'	叉号	'>'	右三角形
<u>'</u> '	水平线条	'<'	左三角形
י י	垂直线条	'p'	五角形
		'h'	六角形

表 5.2 MATLAB 数据点的标记

通常在调用 plot 函数、fplot 函数、fimplicit 函数绘制曲线时,用参数形式指定数据点标记。例如,用五角形标记数据点,命令如下。

ph1 = plot([71, 69, 64, 67.5, 64], 'p');

若仅指定数据点标记,则只呈现数据点,不显示线条;若同时指定数据点标记和线型,才 会既显示数据点标记,又显示曲线。也可以在绘制图形后,通过图形的 Marker 属性添加数 据点的标记。例如:

```
ph = fplot(@(x) sin(x),[0,2*pi]);
ph.Marker = "p";
```

除了设置数据点的标记样式,还可以通过 MarkerSize 属性设置标记大小,其值以磅为 单位,省略时,默认为6。例如:

ph.MarkerSize = 8;

5.1.2 管理坐标区

数据可视化的第一步是指定一个绘图区,然后相对这个区域内的原点绘制图形。在 MATLAB中,坐标区视为图形的容器,绘制图形都是在某个坐标区内。MATLAB绘制图 形的函数大多会在绘图时自动创建坐标区对象,然后在这个坐标区内绘制图形。

■ MATLAB 数 据 可 视 化 (微 课 视 频 版)

坐标系是能够使数据在指定维度空间内找到映射关系的定位系统,在数据可视化中,常用的平面坐标系包括笛卡儿坐标系、对数坐标系和极坐标系。笛卡儿坐标系和对数坐标系中的数据点用与原点的相对距离表示,笛卡儿坐标系坐标轴上的刻度是线性渐变的,对数坐标系坐标轴上的刻度按对数规律变化。极坐标系中的数据点用极径和极角表示。

1. 设置坐标区刻度

绘制图形时,MATLAB根据绘制数据的值域自动创建坐标区,并确定合适的坐标刻度,使得曲线尽可能完整、清晰地显示出来。

1) axis 函数

若绘图时需要自己定义坐标区的坐标范围,可以调用 axis 函数来实现。axis 函数的基本调用格式如下。

axis([xmin, xmax, ymin, ymax, zmin, zmax, cmin, cmax])

系统按照给出的三个维度的最小值(xmin, ymin, zmin)和最大值(xmax, ymax, zmax)设置坐标区范围,并按指定的颜色最小值 cmin 和最大值 cmax 建立数据与颜色的映射关系。绘制二维图形时通常只给出横、纵坐标范围。例如,在绘制了二维曲线后,指定当前坐标区的坐标范围,命令如下。

axis([-6, 5, -5.5, 4])

axis 函数的其他用法如下。

(1) axis style.

style 设置坐标区坐标呈现模式,可取值包括 tight、padded、equal、square、fill 和 vis3d 等。tight 样式指定坐标范围与数据值域相同,坐标区外轮廓与图形呈现区域无间距; padded 样式指定坐标区外轮廓临近图形呈现区域,两者之间存在间距; equal 样式指定坐标 区的每个轴使用相同的单位长度; square 样式指定坐标区为正方形; fill 样式启用"伸展填 充"(默认值),每个维度轴线的长度等于坐标区的 Position 属性所定义的矩形边长; vis3d 样 式锁定纵横比。

例如,绘制一个边长为1的正方形,可以使用以下程序。

x = [0, 1, 1, 0, 0];	y = [0, 0, 1, 1, 0];	
plot(x,y)		
axis padded		%使曲线与坐标区轮廓不重合
axis square;		%使图形呈现为正方形

运行以上程序,将绘制出如图 5.6(a)所示图形。

(2) axis mode.

mode 用于设置坐标范围的调整模式。manual 模式指定将所有坐标区范围锁定在当前 值;auto 模式根据数据值域自动确定坐标区范围;'auto x'模式根据表示水平位置的数据值 域自动确定 x 轴范围;'auto y'模式根据表示竖直位置的数据值域自动确定 y 轴范围;'auto z' 模式根据表示高度的数据值域自动确定 z 轴范围;'auto xy'、'auto xz'、'auto yz'分别表示自动

第5章 MATLAB数据可视化的基础方法



确定 x-y 轴、x-z 轴、y-z 轴的范围。

(3) axis visibility.

visibility 用于设置坐标轴的可见性。on 指定显示坐标轴, off 指定不显示坐标轴。

2) 坐标区对象的属性

建立坐标区后,可以通过坐标区对象的属性调整坐标区的外观。

(1) XLim, YLim, ZLim 属性。

XLim、YLim、ZLim 属性用于获取和设置坐标区的坐标范围,其值是一个二元向量,分别对应坐标的最小值和最大值。例如,绘制上述正方形后,修改坐标区横轴的坐标范围,命令如下。

>> ax = gca; >> ax.XLim = [-0.5 1.5];

第1行命令通过预定义变量 gca 获取当前坐标区对象的句柄,存于变量 ax。执行以上 命令,显示如图 5.6(b)所示图形,横轴的坐标范围变为-0.5~1.5。

(2) XTick, YTick, ZTick 属性。

坐标区坐标轴的刻度值默认是线性递增的。XTick、YTick、ZTick 属性用于获取和设置坐标区坐标轴的刻度。例如:

>> ax.XTick = [-0.5 0.3 0.6 1.0 1.2];

执行命令,横轴的刻度变为-0.5、0.3、0.6、1.0、1.2。

(3) XTickLabel, YTickLabel, ZTickLabel 属性。

坐标区坐标轴的刻度标签默认采用刻度值。XTickLabel、YTickLabel、ZTickLabel 属性用于获取和设置坐标轴的刻度标签。例如:

155

>> ax.XTickLabel = {"周一","周二","周三","周四","周五"};