第 7 讲　拟合模型

司守奎
烟台市，海军航空大学
Email：sishoukui@163.com

插值：求过已知有限个数据点的近似函数。

拟合：已知有限个数据点，求近似函数，不要求过已知数据点，只要求在某种意义下它在这些点上的总偏差最小。

插值和拟合都是要根据一组数据构造一个函数作为近似，由于近似的要求不同，二者的数学方法上是完全不同的。而面对一个实际问题，究竟应该用插值还是拟合，有时容易确定，有时则并不明显。

7.1　线性最小二乘法的数学原理

以一元函数的拟合为例，已知平面上的 n 个点 (x_i, y_i)， $i=1,2,\cdots,n$， x_i 互不相同，寻求一个函数（曲线） $y=f(x)$，使得

$$f(x)=a_1 r_1(x)+a_2 r_2(x)+\cdots+a_m r_m(x), \tag{1}$$

其中 $r_k(x)$ 是事先选定的一组线性无关的函数，a_k 是待定系数 $(k=1,2,\cdots,m,m<n)$；所谓的线性是指函数 y 关于待定参数是线性的。拟合的准则是使 y_i（$i=1,2,\cdots,n$）与 $f(x_i)$ 的距离 δ_i 的平方和

$$J(a_1,\cdots,a_m)=\sum_{i=1}^n \delta_i^2=\sum_{i=1}^n [f(x_i)-y_i]^2 \tag{2}$$

最小，称为最小二乘准则。

7.1.1　系数 a_k 的确定

为求 $a_1,\cdots,a_m$ 使 J 达到最小，只需利用极值的必要条件 $\dfrac{\partial J}{\partial a_k}=0\ (k=1,\cdots,m)$，得到关于 $a_1,\cdots,a_m$ 的线性方程组

$$\sum_{i=1}^n r_j(x_i)[\sum_{k=1}^n a_k r_k(x_i)-y_i]=0,\quad (j=1,\cdots,m),$$

即

$$\sum_{k=1}^n a_k[\sum_{k=1}^n r_j(x_i) r_k(x_i)]=\sum_{i=1}^n r_j(x_i) y_i,\quad (j=1,\cdots,m). \tag{3}$$

记

$$R=\begin{bmatrix} r_1(x_1) & \cdots & r_m(x_1) \\ \vdots & \vdots & \vdots \\ r_1(x_n) & \cdots & r_m(x_n) \end{bmatrix}_{n\times m},$$

$$A=[a_1,\cdots,a_m]^T,\quad Y=[y_1,\cdots,y_n]^T.$$

方程组（3）可表为

$$R^T R A=R^T Y. \tag{4}$$

当 $\{r_1(x),\cdots,r_m(x)\}$ 线性无关时， R 列满秩， $R^T R$ 可逆，于是方程组（4）有唯一解

$$A=(R^T R)^{-1} R^T Y.$$

7.1.2　函数 $r_k(x)$ 的选取

面对一组数据 $(x_i, y_i), i=1,2,\cdots,n$，用线性最小二乘法作曲线拟合时，首要的、也是关键的一步是恰当地选取 $r_1(x),\cdots,r_m(x)$。如果通过机理分析，能够知道 y 与 x 之间应该有什么样的函数关系，则 $r_1(x),\cdots,r_m(x)$ 容易确定。若无法知道 y 与 x 之间的关系，通常可以将数据 (x_i, y_i)（$i=1,2,\cdots,n$）作图，直观地判断应该用什么样的曲线去作拟合。人们常用的曲线有

（i）直线　$y=a_1 x+a_2$；

（ii）多项式　$y=a_1 x^m+\cdots+a_m x+a_{m+1}$（一般 $m=2,3$，不宜太高）；

（iii）双曲线（一支） $y=\frac{a_1}{x}+a_2$；

（iv）指数曲线 $y=a_1e^{a_2x}$.

对于指数曲线，拟合前需作变量代换，化为对 a_1,a_2 的线性函数。

已知一组数据，用什么样的曲线拟合最好，可以在直观判断的基础上，选几种曲线分别拟合，然后比较，看哪条曲线的最小二乘指标 J 最小。

7.2 线性最小二乘法的 MATLAB 实现

7.2.1 解超定线性方程组拟合参数

要拟合（1）式中的参数 $a_1,\cdots,a_m$，把观测值代入（1）式，在上面的记号下，得到线性方程组

$$RA=Y.$$

则 MATLAB 中拟合参数向量 A 的命令为 A=R\Y。

例 7.1 为了测量刀具的磨损速度，我们做这样的实验：经过一定时间（如每隔一小时），测量一次刀具的厚度，得到一组实验数据 $(t_i,y_i)(i=1,2,\cdots,8)$ 如表 7.1 所示。试根据实验数据建立 y 与 t 之间的经验公式 $y=at+b$.

表 7.1 实验数据观测值

t_i	0	1	2	3	4	5	6	7
y_i	27.0	26.8	26.5	26.3	26.1	25.7	25.3	24.8

解 拟合参数 a,b 的准则是最小二乘准则，即求 a,b，使得

$$\delta(a,b)=\sum_{i=0}^{7}(at_i+b-y_i)^2$$

达到最小值，由极值的必要条件，得

$$\begin{cases}\dfrac{\partial\delta}{\partial a}=2\sum\limits_{i=0}^{7}(at_i+b-y_i)t_i=0,\\ \dfrac{\partial\delta}{\partial b}=2\sum\limits_{i=0}^{7}(at_i+b-y_i)=0,\end{cases}$$

化简，得到正规方程组

$$\begin{cases}a\sum\limits_{i=0}^{7}t_i^2+b\sum\limits_{i=0}^{7}t_i=\sum\limits_{i=0}^{7}y_it_i,\\ a\sum\limits_{i=0}^{7}t_i+8b=\sum\limits_{i=0}^{7}y_i.\end{cases}$$

解之，得 a,b 的估计值分别为

$$\hat{a}=\frac{\sum\limits_{i=0}^{7}(t_i-\bar{t})(y_i-\bar{y})}{\sum\limits_{i=0}^{7}(t_i-\bar{t})^2},$$

$$\hat{b}=\bar{y}-\hat{a}\bar{t},$$

其中其中 $\bar{t}=\frac{1}{8}\sum\limits_{i=0}^{7}t_i$，$\bar{y}=\frac{1}{8}\sum\limits_{i=0}^{7}y_i$ 分别为 t_i 的均值和 y_i 的均值。

利用给定的观测值和 MATLAB 软件，求得 a,b 的估计值为 $\hat{a}=-0.3036$，$\hat{b}=27.1250$.

```
clc, clear
t=[0 1    2    3    4    5    6    7]';a=[t,ones(8,1)];
y=[27.0   26.8 26.5 26.3 26.1 25.7 25.3 24.8]';
tb=mean(t); yb=mean(y);
ahat=sum((t-tb).*(y-yb))/sum((t-tb).^2) %编程计算
bhat=yb-ahat*tb
[c,cint,r,rint,stats]=regress(y,a)   %利用回归分析
```

```
cs=a\y   %解超定的线性方程组
```

例 7.2 在研究某单分子化学反应速度时，得到数据$(t_i, y_i)(i=1,2,\cdots,8)$如表 7.2 所示。其中$t$表示从实验开始算起的时间，$y$表示时刻$t$反应物的量。试根据上述数据定出经验公式$y=ke^{mt}$，其中$k,m$是待定常数。

表 7.2 反应物的观测值数据

t_i	3	6	9	12	15	18	21	24
y_i	57.6	41.9	31.0	22.7	16.6	12.2	8.9	6.5

解 对$y=ke^{mt}$两边取对数，得$\ln y=\ln k+mt$，记$\ln k=b$，则有$\ln y=b+mt$，我们使用线性最小法拟合参数b,m，即求b,m的估计值使得

$$\sum_{i=1}^{8}(b+mt_i-\ln y_i)^2$$

达到最小值。

利用 MATLAB 软件，求得a,m的估计值分别为

$$\hat{b}=4.3640, \hat{m}=-0.1037,$$

从而k的估计值为$\hat{k}=78.5700$，即所求的经验公式为

$$y=78.5700e^{-0.1037t}.$$

```
clc, clear
t=[3    6    9    12   15   18   21   24]';
y=[57.6 41.9    31.0    22.7    16.6    12.2    8.9 6.5]';
a=[ones(8,1),t];
cs=a\log(y), cs(1)=exp(cs(1))
```

例 7.3 某天文学家要确定一颗小行星绕太阳运行的轨道，他在轨道平面内建立以太阳为原点的直角坐标系，两坐标轴上的单位长度取为 1 天文测量单位（1 天文测量单位为地球到太阳的平均距离：1.496×10^8千米）。在 5 个不同的时间对小行星作了 5 次观察，测得轨道上 5 个点的坐标数据见**表 7.3**。由开普勒第一定律知，小行星的轨道为一椭圆，其一般方程可表示为

$$a_1x^2+a_2xy+a_3y^2+a_4x+a_5y+1=0.$$

请根据观测数据建立行星运行轨道的方程，并画出轨道曲线。

表 7.3 小行星观察数据的坐标

	1	2	3	4	5
x 坐标	5.764	6.286	6.759	7.168	7.408
y 坐标	0.648	1.202	1.823	2.526	3.360

解 将天文学家所测的轨道上 5 个点的坐标数据(x_i, y_i)（$i=1,2,\cdots,5$）代入椭圆轨道方程，可得下面的线性方程组

$$\begin{cases}a_1x_1^2+a_2x_1y_1+a_3y_1^2+a_4x_1+a_5y_1=-1,\\ a_1x_2^2+a_2x_2y_2+a_3y_2^2+a_4x_2+a_5y_2=-1,\\ a_1x_3^2+a_2x_3y_3+a_3y_3^2+a_4x_3+a_5y_3=-1,\\ a_1x_4^2+a_2x_4y_4+a_3y_4^2+a_4x_4+a_5y_4=-1,\\ a_1x_5^2+a_2x_5y_5+a_3y_5^2+a_4x_5+a_5y_5=-1.\end{cases}$$

解上述线性方程组，得

$$a_1=0.0508, \quad a_2=-0.0702, \quad a_3=0.0381, \quad a_4=-0.4531, \quad a_5=0.2643,$$

即小行星轨道的椭圆方程为

$$0.0508x^2-0.0702xy+0.0381y^2-0.4531x+0.2643y+1=0\,.$$

小行星的运行轨道曲线如图 7.1 所示。

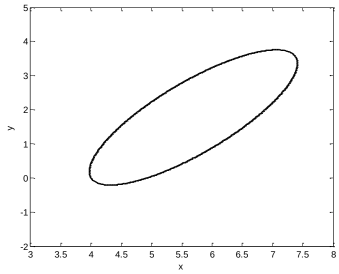

图 7.1　小行星运行轨道图

计算及画图的 MATLAB 程序如下

```
clc, clear
x0=[5.764   6.286   6.759   7.168   7.408]';
y0=[0.648   1.202   1.823   2.526   3.360]';
a=[x0.^2   x0.*y0   y0.^2   x0 y0]; b=-ones(5,1);
cs=a\b
fxy=@(x,y)[x.^2 x.*y y.^2 x y]*cs+1; %定义椭圆方程的匿名函数
h=ezplot(fxy,[3,8,-2,5]), title('')
set(h,'Color','k','LineWidth',2) %设置线的颜色为黑色，否则打印时很不清晰
```

7.2.2　求解约束线性最小二乘问题的 lsqlin 函数

在最小二乘意义下解约束线性方程组

$$Cx=d$$

$$\text{s.t.}\quad\begin{cases}Ax\le b,\\ Aeq\cdot x=beq,\\ lb\le x\le ub.\end{cases}$$

即求解数学规划问题

$$\min\quad \frac{1}{2}\|Cx-d\|_2^2$$

$$\text{s.t.}\quad\begin{cases}Ax\le b,\\ Aeq\cdot x=beq,\\ lb\le x\le ub.\end{cases}$$

的 MATLAB 函数 lsqlin 的调用格式为

```
x=lsqlin(C,d,A,b,Aeq,beq,lb,ub)
```

例 7.4　已知 x,y 的观测值见表 7.4。用给定数据拟合函数 $y=ae^x+b\ln x$，且满足 $a>0$，$b>0$，$a+b\le 1$。

表 7.4　x,y 的观测值

x	3	5	6	7	4	8	5	9
y	4	9	5	3	8	5	8	5

```
clc, clear
a0=[3    5    6    7    4    8    5    9
4    9    5    3    8    5    8    5];
x0=a0(1,:)'; y0=a0(2,:)';
C=[exp(x0),log(x0)]; d=y0;
```

A=[1 1]; b=1; %线性不等式的约束矩阵和常数项列
lb=zeros(2,1); %参数向量的下界
cs=lsqlin(C,d,A,b,[],[],lb) %拟合参数
求解结果为 $y = 0.0005e^x + 0.9995\ln x$.

7.2.3　多项式拟合

MATLAB 多项式拟合的函数为 polyfit，调用格式为

p=polyfit(x,y,n)　　%拟合 n 次多项式，返回值 p 是多项式对应的系数，排列次序为从高次幂系数到低次幂系数。

计算多项式 p 在 x 处的函数值命令为
y=polyval(p,x)

例 7.5　利用三次多项式 $y = x^3 + x^2 + 2x + 3$ ，产生一组数据 (x_i, y_i) ，再在 y_i 上添加服从正态分布 $N(1, 2)$ 的随机扰动。利用产生的数据拟合三次多项式。

```
clc, clear
p1=[1 1 2 3]; %定义已知多项式
x0=linspace(2,4,20) %取区间[2,4]上均匀分布的 20 个点
y0=polyval(p1,x0) %计算对应的函数值
ya=y0+normrnd(1,sqrt(2),1,20) %加上正态分布随机扰动的数据
p2=polyfit(x0,ya,3)    %利用扰动数据拟合三次多项式
save mydata75 x0 ya    %保存数据，供下面继续使用
```

7.3　fittype 和 fit 函数

函数 fittype 和 fit 配对，既可以做线性拟合，也可以做非线性拟合，我们这里首先介绍一下这两个函数的调用格式，然后举几个拟合的例子。

函数 fittype 定义要拟合的函数，它的调用格式为：
aFittype=fittype(libraryModeName)
aFittype=fittype(expression,Name,Value)
aFittype=fittype(linearModeTerm,Name,Value)
aFittype=fittype(anonymousFunction,Name,Value)

函数 fit 的调用格式为：
fitobject=fit(x,y, aFittype)　%x 和 y 分别为自变量和因变量的观测值
fitobject=fit([x,y],z,aFittype) %[x,y]为自变量的观测值的两列矩阵，z 为因变量的观测列向量，这里是拟合二元函数。
fitobject=fit(x,y, aFittype,fitOptions)
fitobject=fit(x,y, aFittype,Name,Value)

7.3.1 使用库模型定义拟合函数

1.三次多项式函数

```
f=fittype(‘poly3’)
f =
        Linear model Poly3:
        f(p1,p2,p3,p4,x) = p1*x^3 + p2*x^2 + p3*x + p4
```

例 7.6（续**例 7.5**）　使用**例 7.5** 的数据，拟合三次多项式

```
clc, clear
load mydata75
ft=fittype('poly3')    %定义拟合函数类型
f=fit(x0',ya',ft)    %注意这里已知数据必须是列向量
```

2.一阶傅里叶级数

```
f=fittype('fourier1')
f =
```

```
General model Fourier1:
f(a0,a1,b1,w,x) =   a0 + a1*cos(x*w) + b1*sin(x*w)
```

例 7.7　利用函数 $y=1+2\cos(3x)+3\sin(3x)$，产生一组数据 (x_i, y_i)，再在 y_i 上加上白噪声扰动，即加上服从标准正态分布的随机数。利用产生的数据拟合一阶傅里叶级数。

```
clc, clear
x0=linspace(1,20,41); x0=x0'; y0=1+2*cos(3*x0)+3*sin(3*x0);
ya=y0+normrnd(0,1,41,1); %添加白噪声
save mydata77 x0 ya   %保存数据，供下面使用
f1=fit(x0,y0,'fourier1')   %利用无噪声数据拟合
f2=fit(x0,ya,'fourier1')   %利用噪声数据拟合
```

例 7.8（续例 7.7）　利用例 7.7 的数据，拟合一个 8 阶傅里叶级数。

```
clc, clear
load mydata77   %加载数据 x0,ya
f=fit(x0,ya,'fourier8')   %拟合 8 阶傅里叶级数
```

MATLAB 工具箱中库模型中的函数类是很丰富的，我们就不一一举例了。可以在命令窗口中输入 doc fittype 打开帮助页面，再点击“Model Names and Equations”链接，就可以看到库函数中的所有函数类。

7.3.2　自定义一元、二元函数模型

1.线性拟合（关于未知参数是线性的）

```
g=fittype('a*u+b*exp(n*u)','problem','n','independent','u')
g =
     General model:
     g(a,b,n,u) = a*u+b*exp(n*u)
```

其中属性'problem'的取值 n 为已知的可变参数，u 为函数的自变量。

例 7.9　用模拟数据拟合函数 $y=ax+be^{2x}$。

```
clc, clear
x0=linspace(0,1,20); y0=2*x0+3*exp(2*x0); %生成无扰动的模拟数据
ya=y0+normrnd(0,1,1,20); %生成扰动的模拟数据
save mydata711 x0 y0 ya   %保存数据供下面例子使用
ft=fittype('a*x+b*exp(n*x)','problem','n','independent','x')
f1=fit(x0',y0',ft,'problem',2,'Start',rand(1,2))   %无扰动数据的拟合
f2=fit(x0',ya',ft,'problem',2,'Start',rand(1,2))   %扰动数据的拟合
```

```
g=fittype('a+b*log(x)+c*y','dependent',{'z'},'independent',{'x', 'y'},'coefficients',{'a','b', 'c'})
```

其中属性'dependent'指的是因变量，它的取值为'z'，属性'independent'指的是自变量，它的取值为'x', 'y'，这里使用了细胞数组，属性'coefficients'指的是待定参数,它的取值为'a','b', 'c'。

例 7.10　用模拟数据拟合函数 $z=a+b\ln x+cy$。

```
clc, clear
x0=linspace(1,10,21); y0=linspace(3,20,21);
z0=1+2*log(x0)+3*y0; %产生 z=1+2lnx+3y 的无扰动数据
za=z0+normrnd(0,1,1,21); %产生扰动数据
g=fittype('a+b*log(x)+c*y','dependent',{'z'},'independent',{'x', 'y'},'coefficients',{'a','b', 'c'})
f1=fit([x0',y0'],z0',g,'Start',rand(1,3))   %利用无扰动数据拟合
f2=fit([x0',y0'],za',g,'Start',rand(1,3))   %利用扰动数据拟合
```

说明：（1）定义拟合函数类型时，只要说明自变量属性'independent'和因变量属性'dependent'；未知参数的属性'coefficients'可以不说明。自变量属性'independent'的默认值为'x'，可以不说明；因变量属性'dependent'的默认值'y'也可以不说明。

例 7.10 的程序可以改写为

```
clc, clear
x0=linspace(1,10,21); y0=linspace(3,20,21);
z0=1+2*log(x0)+3*y0; %产生 z=1+2lnx+3y 的无扰动数据
za=z0+normrnd(0,1,1,21); %产生扰动数据
g=fittype('a+b*log(x)+c*y','dependent',{'z'},'independent',{'x', 'y'})
f1=fit([x0',y0'],z0',g,'Start',rand(1,3))    %利用无扰动数据拟合
f2=fit([x0',y0'],za',g,'Start',rand(1,3))    %利用扰动数据拟合
```

（2）利用上述自定义形式的函数不能拟合三元以上的函数。

2.非线性拟合（关于未知参数是非线性的）

例 7.11 用**表 7.5** 的数据拟合函数 $y = ae^{bx_1} + cx_2^2$ 。

表 7.5 x_1, x_2, y **的观测值**

x_1	6	2	6	7	4	2	5	9
x_2	4	9	5	3	8	5	8	2
y	5	2	1	9	7	4	3	3

```
clc, clear
d=[6 2    6    7    4    2    5    9
4    9    5    3    8    5    8    2
5    2    1    9    7    4    3    3];
x1=d(1,:)'; x2=d(2,:)'; y=d(3,:)';
g=fittype('a*exp(b*x1)+c*x2^2','independent',{'x1', 'x2'}) %因变量'dependent'属性值'y'可以不说明
f=fit([x1,x2],y,g,'Start',rand(3,1))
```

求得 $a = 5.09$， $b = -0.0026$， $c = -0.0215$ 。

7.3.3 自定义一元线性模型的第 2 种表示方式（关于未知参数是线性的）

拟合函数 $y = ax + be^{2x}$ ，基函数为 x ， e^{2x} 。

拟合函数 $y = ax + b\sin x + c$ ，基函数为 x ， $\sin x$ ， 1。

例 7.12 利用模拟数据拟合函数 $y = ax + b\sin x + c$ 。

```
clc, clear
x0=linspace(2,10,21);
y0=2*x0+3*sin(x0)+5; %计算 y=2x+3sin(x)+5 的函数值
ya=y0+normrnd(0,1,1,21);    %产生扰动数据
g=fittype({'x','sin(x)','1'})
f1=fit(x0',y0',g)    %利用无扰动数据拟合
f2=fit(x0',ya',g)    %利用扰动数据拟合
```

7.3.4 由函数文件定义的分段线性模型

例 7.13（MATLAB 工具箱的帮助例程）利用给定数据拟合分段线性函数

$$y = \begin{cases} a + bx, & x < k, \\ c + dx, & x \ge k. \end{cases}$$

首先定义分段线性函数 piecewiseLine(x,a,b,c,d,k)如下：

```
function y = piecewiseLine(x,a,b,c,d,k)
y = zeros(size(x));
for i = 1:length(x)
      if x(i) < k,
```

```
            y(i) = a + b* x(i);
        else
            y(i) = c + d* x(i);
        end
    end
    end
```

然后利用给定的数据，拟合参数 a,b,c,d,k，程序如下：

```
clc, clear
x = [0.81;0.91;0.13;0.91;0.63;0.098;0.28;0.55;0.96;0.96;0.16;0.97;0.96];
y = [0.17;0.12;0.16;0.0035;0.37;0.082;0.34;0.56;0.15;-0.046;0.17;-0.091;-0.071];
ft = fittype( 'piecewiseLine( x, a, b, c, d, k )' )
f = fit( x, y, ft, 'Start', [1, 0, 1, 0, 0.5] )
plot( f, x, y )
```

7.3.5 利用匿名函数生成函数模型

例 7.18（续例 7.17） 利用匿名函数生成函数模型，拟合例 7.17 中的分段线性函数。

```
clc, clear
g1=@(a,b,c,d,k,x)(a+b*x).*(x<k)+(c+d*x).*(x>=k); %定义匿名函数
g2=fittype(g1) %生成 fittype 类型的函数类, ,'independent'的默认属性值'x'可以不说明
x = [0.81;0.91;0.13;0.91;0.63;0.098;0.28;0.55;0.96;0.96;0.16;0.97;0.96];
y = [0.17;0.12;0.16;0.0035;0.37;0.082;0.34;0.56;0.15;-0.046;0.17;-0.091;-0.071];
f = fit( x, y, g2, 'Start', [1, 0, 1, 0, 0.5] )  %函数拟合
plot( f, x, y )  %画图
```

例 7.14（续例 7. 13） 用 MATLAB 的匿名函数拟合例 7. 13 中的函数。

```
clc, clear
d=[62   6   7   4   2   5   9
4   9   5   3   8   5   8   2
5   2   1   9   7   4   3   3];
x1=d(1,:)'; x2=d(2,:)'; y=d(3,:)';
g1=@(a,b,c,x1,x2)a*exp(b*x1)+c*x2.^2;  %定义匿名函数
g2=fittype(g1,'independent',{'x1','x2'})
f=fit([x1,x2],y,g2,'Start',rand(3,1))
```

7.4 非线性拟合

非线性拟合的主要准则也是最小二乘准则，我们以一元函数为例说明。对于未知函数 $y=f(\theta,x)$，其中 θ 为未知的参数向量，函数关于 θ 是非线性的。给定 $y=f(\theta,x)$ 的一些观测值 (x_i,y_i)，$i=1,2,\cdots,n$，拟合参数 θ 的最小二乘准则，就是所确定参数 θ 的值要使在观测点上误差平方和

$$J(\theta)=\sum_{i=1}^{n}(y_i-\hat{y}_i)^2 \tag{5}$$

最小，这里 $\hat{y}_i=f(\theta,x_i)$。

求式（5）的最小值问题有很多算法，我们这里就不介绍了，例如 MATLAB 工具箱 fit 函数使用的默认算法是 Levenberg-Marquardt 算法，还有 Trust-Region 算法。

MATLAB 非线性拟合的主要命令有 fit（要用 fittype 定义函数类），lsqcurvefit，nlinfit 等命令。fit 函数使用很方便，但只能拟合一元和二元函数，lsqcurvefit 可以拟合任意多个自变量的函数，并且可以约束未知参数的下界和上界；nlinfit 函数无法约束参数的界限，我们这里就不介绍了。

前面已经给出 fit 非线性拟合的例子，这里就不给出 fit 的例子了。下面首先给出 lsqcurvefit 的用法说明，再举几个用 lsqcurvefit 拟合函数的例子。

要拟合函数 $y=f(\theta,x)$，给定 x 的观测值 $xdata$， y 的观测值 $ydata$，求参量 θ，使得误差平方和最小，即

$$\min_{\theta}\|f(\theta,xdata)-ydata\|_2^2=\sum_i\left(f(\theta,xdata_i)-ydata_i\right)^2.$$

MATLAB 中的函数为

theta=lsqcurvefit(fun,theta0,xdata,ydata,lb,ub,options)

其中 fun 是定义函数 $f(\theta,x)$ 的 M 文件，theta0 是 θ 的初始值，lb 是参数 θ 的下界，ub 是参数 θ 的上界，options 参数可以对计算过程的一些算法等属性进行设置，返回值 theta 是所求参数 θ 的值。

例 7.15　已知 x, y 的观测值见表 7.6。用 lsqcurvefit 拟合函数 $y=\frac{a}{e^{bx}}$。

表 7.6　x, y 的观测值

x	6	2	6	7	4	2	5	9
y	4	9	5	3	8	5	8	2

```
clc, clear
a=[6    2    6    7    4    2    5    9
4    9    5    3    8    5    8    2];
x0=a(1,:)'; y0=a(2,:)';
yx=@(cs,x)cs(1)./exp(cs(2)*x); %定义所拟合函数的匿名函数
cs=lsqcurvefit(yx,rand(2,1),x0,y0)
```

注：（1）程序中定义匿名函数时，必须使用“./”，否则会出错。

（2）对于一些非线性拟合，有时感觉像在凑数，MATLAB 每次运行时答案都是不唯一的，这时可能使用 Lingo 软件进行非线性拟合，效果会好些。本例的 Lingo 拟合程序如下：

```
model:
sets:
num/1..8/:x0,y0;
endsets
data:
x0=6    2    6    7    4    2    5    9;
y0=4    9    5    3    8    5    8    2;
enddata
min=@sum(num:(y0-a/@exp(b*x0))^2);
@free(a); @free(b);
end
```

例 7.16　利用表 7.7 的数据，拟合函数 $y=a\sin(x_1)+e^{bx_2}+\cos(cx_3)$。

表 7.7　x_1, x_2, y 的观测值

x_1	6	2	6	7	4	2	5	9
x_2	4	9	5	3	8	5	8	2
x_3	2	5	6	3	6	6	8	7
y	5	2	1	9	7	4	3	3

```
clc, clear
a=[62    6    7    4    2    5    9
4    9    5    3    8    5    8    2
2    5    6    3    6    6    8    7
5    2    1    9    7    4    3    3];
zd=a([1:3],:)'; %自变量的观测值，每一列是一个变量的取值
yd=a(4,:)';    %因变量的观测值
fx=@(t,x)t(1)*sin(x(:,1))+exp(t(2)*x(:,2))+cos(t(3)*x(:,3));
```

option=optimset('MaxFunEvals',3000)
abc=lsqcurvefit(fx,rand(3,1),zd,yd,[],[],option)

7.5 曲线和曲面拟合的用户图形界面解法

可以使用 cftool 进行曲线和曲面拟合。具体执行步骤如下：

（1）把数据导入到工作空间；

（2）运行 cftool，打开用户图形界面窗口；

（3）选择适当的模型进行拟合；

（4）把计算结果输出到 MATLAB 工作空间。

可以通过帮助（运行 doc cftool）熟悉该命令的使用细节。

例 7.17（续例 7.11） 使用用户图形界面，用**表** 7.5 的数据拟合函数 $y=ae^{bx_1}+cx_2^2$。

解 （1）首先运行如下MATLAB程序，把数据加载到MATLAB工作空间。

```
clc, clear
d=[62    6    7    4    2    5    9
4    9    5    3    8    5    8    2
5    2    1    9    7    4    3    3];
x0=d(1,:)'; y0=d(2,:)'; z0=d(3,:)';
```

（2）在MATLAB命令窗口输入cftool，并回车运行。

（3）在打开的用户图形界面输入如图7.2所示的参数及函数，得到的结果显示在图2所示窗口的左边。

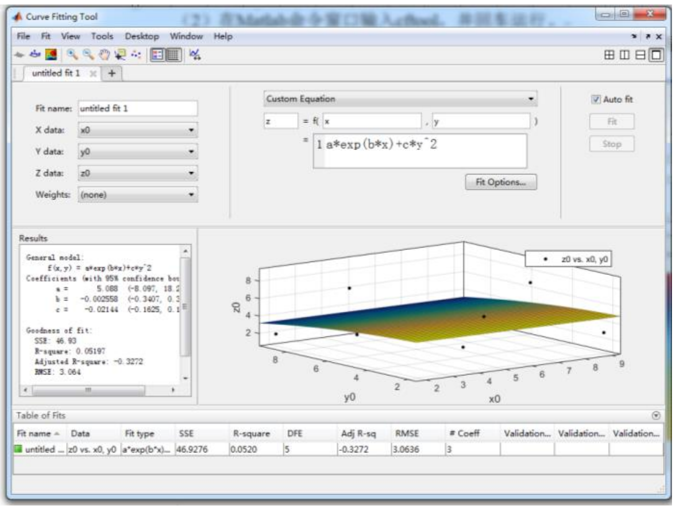

图 7.2 用户图形界面示意图

例 7.18 利用数据 x=1:50;y=rand(1,50); 拟合 8 阶傅里叶级数

$$y=a_0+\sum_{k=1}^{8}[a_k\cos(kwx)+b_k\sin(kwx)],$$

即待定参数 $a_0,a_1,\cdots,a_8,b_1,b_2,\cdots,b_8,w$。

解 我们使用用户图形界面进行拟合。

（1）第一步在命令窗口中运行

```
x=1:50;y=rand(1,50);
```

即把数据加载到工作空间。

（2）运行 cftool，打开用户图形界面。进行如图 7.3 所示的操作，首先在左上方选择有关的数据，然后选择拟合的函数为傅里叶级数。

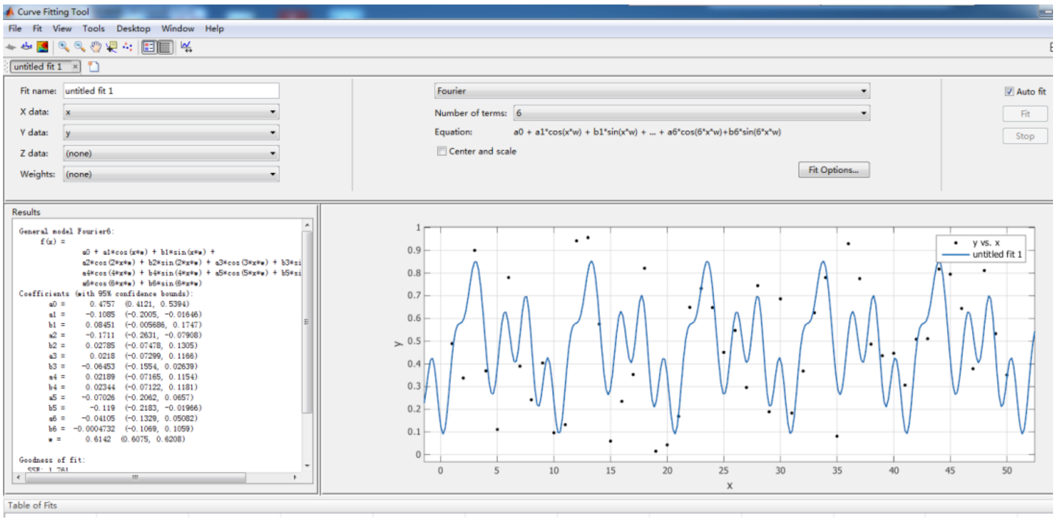

图 7.3 傅里叶级数的拟合图示

例 7.19 求 $y=(x-1)^3$ 与数据 x=1:50;y=rand(1,50); 所拟合 8 阶傅里叶级数

$$y=a_0+\sum_{k=1}^{8}[a_k\cos(kwx)+b_k\sin(kwx)]$$

的交点坐标。

解 （1）把例 7.18 所拟合的模型以名称“ft”输出到工作空间。

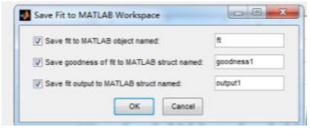

图 7.4 拟合结果输出示意图

（2）画出两个函数交点的示意图，并求交点，计算的 MATLAB 程序如下

```
fplot(ft,[-1,3]), hold on
yy=@(x)(x-1).^3; fplot(yy,[-1,3])
y3=@(x)[ft(x)-yy(x)];
x=fsolve(y3,rand)   %求交点的 x 坐标
y=ft(x)   %求交点的 y 坐标
```

求得的交点坐标为 (1.7754, 0.4663) 。

7.6 拟合和统计等工具箱中的一些检验参数解释

下面对MATLAB拟合和统计等工具箱中的一些检验参数给出解释。

（1）SSE（误差平方和）The sum of squares due to error

该统计参数计算的是拟合数据和原始数据对应点的误差平方和，计算公式是

$$\mathrm{SSE}=\sum_{i=1}^{n}(y_i-\hat{y}_i)^2 .$$

SSE越接近于0，说明模型选择和拟合效果好，数据预测也成功。下面的指标MSE和RMSE与指标SSE有关联，它们的校验效果是一样的。

（2）MSE（方差）Mean squared error

该统计参数是预测数据和原始数据对应点误差平方和的均值，也就是SSE/$(n-m)$，这里 n 是观测数据的个数，m 是拟合参数的个数，和SSE没有太大的区别，计算公式是

$$\mathrm{MSE}=\mathrm{SSE}/(n-m)=\frac{1}{n-m}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2 .$$

（3）RMSE（剩余标准差）Root mean squared error

该统计参数，也叫回归系统的拟合标准差，是MSE的平方根，计算公式是

$$\mathrm{RMSE}=\sqrt{\frac{1}{n-m}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}\ .$$

（4）R-square（判断系数，拟合优度）Coefficient of determination

在讲判断系数之前，我们需要介绍另外两个参数SSR和SST，因为判断系数就是由它们两个决定的。

对总平方和 $\mathrm{SST}=\sum_{i=1}^{n}(y_i-\bar{y})^2$ 进行分解，有

$$\mathrm{SST}=\mathrm{SSE}+\mathrm{SSR}\ ,\quad \mathrm{SSR}=\sum_{i=1}^{n}(\hat{y}_i-\bar{y})^2$$

其中 $\bar{y}=\frac{1}{n}\sum_{i=1}^{n}y_i$ ，SSE 是误差平方和，反映随机误差对 y 的影响，SSR 称为回归平方和，反映自变量对 y 的影响。

判断系数定义为

$$R^2=\frac{\mathrm{SSR}}{\mathrm{SST}}=\frac{\mathrm{SST}-\mathrm{SSE}}{\mathrm{SST}}=1-\frac{\mathrm{SSE}}{\mathrm{SST}}\,.$$

（5）调整的判断系数

统计学家主张在回归建模时，应采用尽可能少的自变量，不要盲目地追求判定系数 R^2 的提高。其实，当变量增加时，残差项的自由度就会减少。而自由度越小，数据的统计趋势就越不容易显现。为此，又定义一个调整判定系数

$$\bar{R}^2=1-\frac{\mathrm{SSE}/(n-m)}{SST/(n-1)}\,.$$

$\bar{R}^2$ 与 R^2 的关系是

$$\bar{R}^2=1-(1-R^2)\frac{n-1}{n-m}\,.$$

当 n 很大、m 很少时，$\bar{R}^2$ 与 R^2 之间的差别不是很大；但是，当 n 较少，而 m 又较大时，$\bar{R}^2$ 就会远小于 R^2 。

7.7　应用举例

例 7.20　某地区用水管理机构需要对居民的用水速度（单位时间的用水量）和日总用水量进行估计。现有一居民区，其自来水是由一个圆柱形水塔提供，水塔高 12.2m，塔的直径为 17.4m。水塔是由水泵根据水塔中的水位自动加水。按照设计，当水塔中的水位降至最低水位，约 8.2m 时，水泵自动启动加水；当水位升高到最高水位，约 10.8m 时，水泵停止工作。

表 7.8 给出的是 28 个时刻的数据，但由于水泵正向水塔供水，有 4 个时刻无法测到水位（表 7.8 中为-）。

表 7.8　水塔中水位原始数据

时刻（t）/h	0	0.92	1.84	2.95	3.87	4.98	5.90
水位/m	9.68	9.48	9.31	9.13	8.98	8.81	8.69
时刻（t）/h	7.01	7.93	8.97	9.98	10.92	10.95	12.03
水位/m	8.52	8.39	8.22	–	–	10.82	10.5
时刻（t）/h	12.95	13.88	14.98	15.9	16.83	17.93	19.04
水位/m	10.21	9.94	9.65	9.41	9.18	8.92	8.66
时刻（t）/h	19.96	20.84	22.01	22.96	23.88	24.99	25.91
水位/m	8.43	8.22	–	–	10.59	10.35	10.18

试建立数学模型，来估计居民的用水速度和日总用水量。

上面的问题，可以用插值方法求解，也可以用拟合方法求解，下面我们用拟合方法求

解上述问题。

要估计在任意时刻（包括水泵灌水期间）t 居民的用水速度和日总用水量，分如下三步。

（1）水塔中水的体积的计算

计算水的流量，首先需要计算出水塔中水的体积

$$V=\frac{\pi}{4}D^2h,$$

式中，D 为水塔的直径，h 为水塔中的水位高度。

（2）水塔中水流速度的估计

居民的用水速度就是水塔中的水流速度，水流速度应该是水塔中水的体积对时间的导数，但由于没有每一时刻水体积的具体数学表达式，只能用差商近似导数。

由于在两个时段，水泵向水塔供水，无法确定水位的高度，因此在计算水塔中水流速度时要分三段计算。第一段从 0h 到 8.97h，第二时段，从 10. 95h 到 20.84h，第三段，从 23.88h 到 25.91h。

上面计算仅给出流速的离散值，流速的散点图见图 7.5 中的“*”点，总共有 20 个数据点，因为已知数据有 24 个有效数据，计算数值导数时，这两组无效数据的左右两边又产生 4 个无效数据。如果需要得到流速的连续型曲线，可以拟合多项式曲线，这里我们分三段进行三次多项式拟合，应用前 6 个数据点拟合三次多项式，即在时间区间[0, 4.98]上拟合三次多项式；应用第 6 个数据点到第 10 个数据点，即在时间区间[4.98, 12.03]，拟合第二个三次多项式；应用第 10 个数据点到第 20 个数据点，总共 11 个数据点，即在时间区间[12.03,25.91]，拟合第三个三次多项式。拟合得到的分段三次多项式曲线见图 7.5。

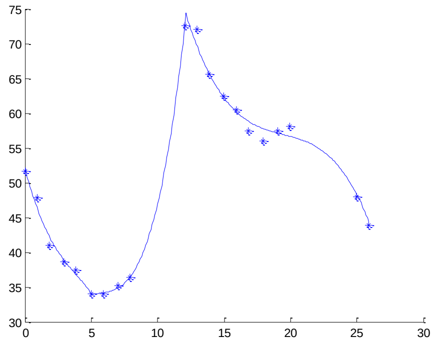

图 7.5　流速数据的散点图及拟合的分段三次多项式曲线

（3）日总用水量的计算

日用水量是对水流速度作积分，其积分区间是[0,24]，可以采用数值积分的方法计算。

这里用离散点数值积分求得的日总用水量为 1221.8 m^3，利用拟合的分段三次多项式的数值积分计算得到的日总用水量为 1312.7 m^3。

用 MATLAB 软件计算时，首先把原始数据粘贴到纯文本文件 data720.txt 中，并且把“-”替换为数值-1。计算的 MATLAB 程序如下：

```
clc, clear
a=load('data720.txt');
t0=a([1:2:end],:); t0=t0'; t0=t0(:); %提出时间数据，并展开成列向量
h0=a([2:2:end],:); h0=h0'; h0=h0(:); %提出高度数据，并展开成列向量
D=17.4;
V=pi/4*D^2*h0; %计算各时刻的体积
dv=gradient(V,t0); %计算各时刻的数值导数（导数近似值）
no1=find(h0==-1) %找出原始无效数据的地址
no2=[no1(1)-1:no1(2)+1,no1(3)-1:no1(4)+1] %找出导数数据的无效地址
```

```
t=t0; t(no2)=[]; %删除导数数据无效地址对应的时间
dv2=-dv; dv2(no2)=[]; %给出各时刻的流速
hold on, plot(t,dv2,'*') %画出流速的散点图
a1=polyfit(t(1:6),dv2(1:6),3); %拟合第一个多项式的系数
a2=polyfit(t(6:10),dv2(6:10),3); %拟合第二个多项式的系数
a3=polyfit(t(10:20),dv2(10:20),3); %拟合第三个多项式的系数
dvf1=polyval(a1,[t(1):0.1:t(6)]); %计算第一个多项式的函数值
dvf2=polyval(a2,[t(6):0.1:t(10)]); %计算第二个多项式的函数值
dvf3=polyval(a3,[t(10):0.1:t(end)]); %计算第三个多项式的函数值
tt=t(1):0.1:t(end); dvf=[dvf1,dvf2,dvf3];
plot(tt,dvf) %画出拟合的三个分段多项式曲线
I=trapz(tt(1:241),dvf(1:241)) %计算24小时内总流量的数值积分
I2=quadl(@(x)polyval(a1,x),t(1),t(6))+quadl(@(x)polyval(a2,x),t(6),t(10))+quadl(@(x)polyval(a3,x),t(10),t(end)) %第二种求日总流量的积分
```

习题7

7.1 某种合金的含铅量百分比（%）为 p，其熔解温度℃为 θ，由实验测得 p 与 θ 的数据如表7.9 所示，试用最小二乘法建立 θ 与 p 之间的经验公式 $\theta = ap + b$。

表 7.9 θ 与 p 的观测数据

p	36.9	46.7	63.7	77.8	84.0	87.5
θ	181	197	235	270	283	292

7.2 多项式 $f(x) = a_3x^3 + a_2x^2 + a_1x + a_0$，取 $a_3 = 8$，$a_2 = 5$，$a_1 = 2$，$a_0 = -1$，在 $[-6,6]$ 上等步长取 100 个点作为 x 的观测值，计算对应的函数值作为 y 的观测值；把得到的观测值记作 $(x_i, y_i), i = 1, 2, \cdots, 100$。

（1）利用观测值 $(x_i, y_i), i = 1, 2, \cdots, 100$，拟合三次多项式。

（2）把每个 y_i 加上白噪声，即加上一个服从标准正态分布的随机数，把得到的数据记作 $\tilde{y}_i (i = 1, 2, \cdots, 100)$，利用 $(x_i, \tilde{y}_i), i = 1, 2, \cdots, 100$，拟合三次多项式。

7.3 函数 $g(x) = \dfrac{10a}{10b + (a - 10b)\mathrm{e}^{-a\sin x}}$，取 $a = 1.1$，$b = 0.01$，计算 $x = 1, 2, \cdots, 20$ 时，$g(x)$ 对应的函数值，把这样得到的数据作为模拟观测值，记作 $(x_i, y_i), i = 1, 2, \cdots, 20$。利用 $(x_i, y_i), i = 1, 2, \cdots,$

（1）用 lsqcurve 拟合函数 $\hat{g}(x)$；

（2）用 fittype 和 fit 拟合函数 $\hat{g}(x)$；

（3）用 cftool 拟合函数 $\hat{g}(x)$。

7.4 （1）先生成 x_1, x_2, x_3 的模拟数据，其中

$x_1 = \mathrm{randi}([1,100],1,100)$，$x_2 = \mathrm{rand}(1,100)$，$x_3 = \mathrm{normrnd}(0,1,1,100)$。

（2）函数 $y = a\sin x_1 + b(\sin x_2)\cos(cx_3)$，取 $a = 2$，$b = -0.1$，$c = 8$，利用（1）中生成的模拟数据，计算对应的函数值，把 x_1, x_2, x_3, y 的数据作为模拟数据

（3）利用（1）（2）中生成的模拟数据，拟合 $y = a\sin x_1 + b(\sin x_2)\cos(cx_3)$ 中的参数 a, b, c。