18. 1stOpt 特殊约束拟合案例演示

拟合计算中经常会遇到两类问题，一是各类带约束条件的拟合问题，约束条件有时看上去还较为复杂，无法按常规方式处理；二是多因变量拟合时由于不同因变量间数据大小存在数量级差别，此时该如何兼顾每个因变量而使得整体及每个因变量拟合效果看上去都好？以 4 个实际案例展示 1stOpt 如何完美处理这类拟合问题。

18.1 案例一：过点及面积约束

拟合公式，数据及图形如下

$$y = y_0 + \left(\frac{a}{w\sqrt{\frac{\pi}{2}}}\right)\exp\left(-2\left(\frac{x-xc}{w}\right)^2\right) \tag{18-1}$$

表 18.1 数据：

x	0.5,1.5,2.5,3.5,4.5,5.5,6.5,7.5,8.5,9.5,10.5,11.5,12.5,13.5,14.5,15.5,16.5,17.5,18.5,19.5,20.5,21.5,22.5,23.5,24.5,25.5,26.5,27.5,28.5,29.5,30.5,31.5,32.5,33.5,34.5
y	0.0202,0.0292,0.0467,0.0658,0.0707,0.0856,0.0832,0.0865,0.0827,0.0782,0.0677,0.0597,0.0503,0.0401,0.0305,0.0269,0.0199,0.0133,0.0088,0.0079,0.0082,0.0037,0.0036,0.0026,0.0015,0.0016,0.0014,0.0011,0.0007,0.0002,0.0006,0.0006,0.0001,0.0002,0.0001

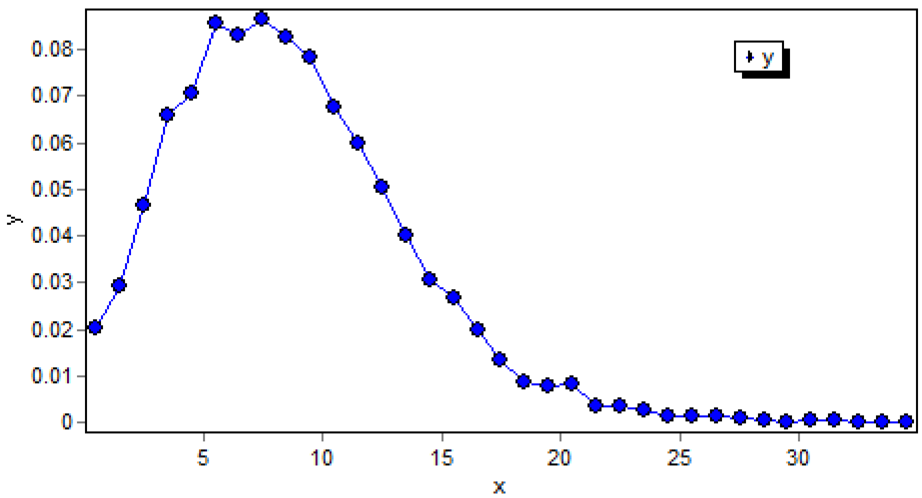

图 18.1 拟合数据原始图

约束条件：

1）拟合区间范围的面积等于 1；

2）拟合曲线必须通过第一个及最后一个点，即[0.5,0.0202]和[34.5,0.0001]两

个点。

如果不考虑约束条件，该问题就是一个简单的高斯峰拟合问题，代码如下：

代码 18-1

```
Function y=y0+(A/(w*sqrt(PI/2)))*exp(-2*((x-xc)/w)^2);
Data;
x=[0.5,1.5,2.5,3.5,4.5,5.5,6.5,7.5,8.5,9.5,10.5,11.5,12.5,13.5,14.5,15.5,16.5,17.5,18.5,19.5,20.5,21.5,22.5,23.5,24.5,25.5,26.5,27.5,28.5,29.5,30.5,31.5,32.5,33.5,34.5];
y=[0.0202,0.0292,0.0467,0.0658,0.0707,0.0856,0.0832,0.0865,0.0827,0.0782,0.0677,0.0597,0.0503,0.0401,0.0305,0.0269,0.0199,0.0133,0.0088,0.0079,0.0082,0.0037,0.0036,0.0026,0.0015,0.0016,0.0014,0.0011,0.0007,0.0002,0.0006,0.0006,0.0001,0.0002,0.0001];
```

可以很容易得到如下结果：

```
Root of Mean Square Error (RMSE): 0.0032191714956919
Sum of Squared Residual: 0.000362707279153634
Correlation Coef. (R): 0.994594533528291
R-Square: 0.989218286124359
Adjusted R-Square: 0.988544429007132
Determination Coef. (DC): 0.989218286124359
Chi-Square: 0.0148447776456218
F-Statistic: 948.079535194921

Parameter   Best Estimate
----------  -------------
y0      0.00179285323600204
a       0.986550230935031
w       9.313041405745
xc      7.65038707294
```

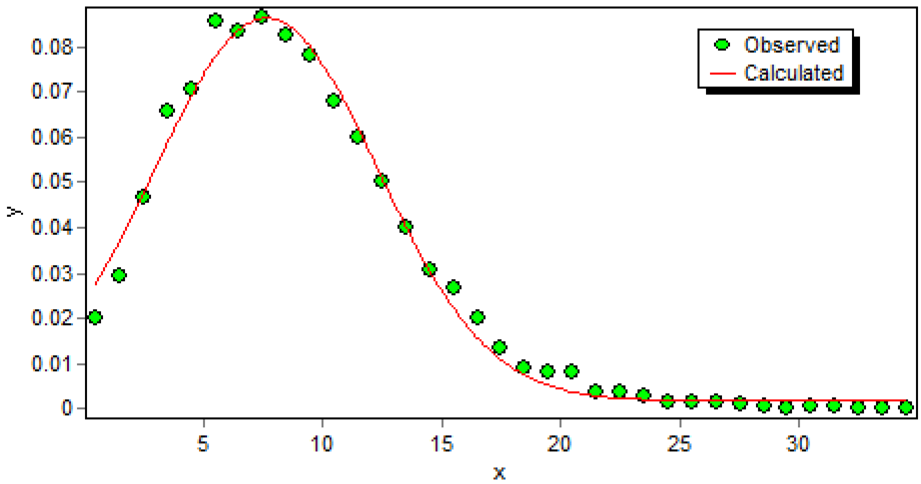

图 18.2 无约束拟合结果图-1

增加过点约束：两个过点坐标用 xx=[0.5,34.5]，yy=[0.0202,0.0001]表示，则有：

$$\begin{cases} yy_1 = y_0 + \left(\frac{a}{w\sqrt{\frac{\pi}{2}}}\right)\exp\left(-2\left(\frac{xx_1 - xc}{w}\right)^2\right) \\ yy_2 = y_0 + \left(\frac{a}{w\sqrt{\frac{\pi}{2}}}\right)\exp\left(-2\left(\frac{xx_2 - xc}{w}\right)^2\right) \end{cases} \tag{18-2}$$

增加拟合曲线与 x 轴包络面积为 1 的约束：

$$\int_{0.5}^{34.5}\left(y_0 + \left(\frac{a}{w\sqrt{\frac{\pi}{2}}}\right)\exp\left(-2\left(\frac{s - xc}{w}\right)^2\right)\right)\mathrm{d}s = 1 \tag{18-3}$$

代码 18-2

```
Constant xx=[0.5,34.5],yy=[0.0202,0.0001];
Function y=y0+(A/(w*sqrt(PI/2)))*exp(-2*((x-xc)/w)^2);
          yy1=y0+(a/(w*sqrt(pi/2)))*exp(-2*((xx1-xc)/w)^2);
          yy2=y0+(a/(w*sqrt(pi/2)))*exp(-2*((xx2-xc)/w)^2);
          Int(y0+(A/(w*sqrt(PI/2)))*exp(-2*((s-xc)/w)^2),s=0.5,34.5)=1;
Data;
x=[0.5,1.5,2.5,3.5,4.5,5.5,6.5,7.5,8.5,9.5,10.5,11.5,12.5,13.5,14.5,15.5,16.5,17.5,18.5,19.5,20.5,21.5,22.5,23.5,2
4.5,25.5,26.5,27.5,28.5,29.5,30.5,31.5,32.5,33.5,34.5];
y=[0.0202,0.0292,0.0467,0.0658,0.0707,0.0856,0.0832,0.0865,0.0827,0.0782,0.0677,0.0597,0.0503,0.0401,0.0
305,0.0269,0.0199,0.0133,0.0088,0.0079,0.0082,0.0037,0.0036,0.0026,0.0015,0.0016,0.0014,0.0011,0.0007,0.
0002,0.0006,0.0006,0.0001,0.0002,0.0001];
```

拟合过点约束也可以用关键字“PassPoint”简单设定，代码如下：

代码 18-3

```
PassPoint = [[0.5,0.0202],[34.5,0.0001]];
Function y=y0+(A/(w*sqrt(PI/2)))*exp(-2*((x-xc)/w)^2);
          Int(y0+(A/(w*sqrt(PI/2)))*exp(-2*((s-xc)/w)^2),s=0.5,34.5)=1;
Data;
x=[0.5,1.5,2.5,3.5,4.5,5.5,6.5,7.5,8.5,9.5,10.5,11.5,12.5,13.5,14.5,15.5,16.5,17.5,18.5,19.5,20.5,21.5,22.5,23.5,2
4.5,25.5,26.5,27.5,28.5,29.5,30.5,31.5,32.5,33.5,34.5];
y=[0.0202,0.0292,0.0467,0.0658,0.0707,0.0856,0.0832,0.0865,0.0827,0.0782,0.0677,0.0597,0.0503,0.0401,0.0
305,0.0269,0.0199,0.0133,0.0088,0.0079,0.0082,0.0037,0.0036,0.0026,0.0015,0.0016,0.0014,0.0011,0.0007,0.
0002,0.0006,0.0006,0.0001,0.0002,0.0001];
```

上述两段代码效果一样。

结果

```
Root of Mean Square Error (RMSE): 0.0059126710489067
Sum of Squared Residual: 0.00122358876264028
Correlation Coef. (R): 0.987152527692732
R-Square: 0.974470112930151
Adjusted R-Square: 0.972874494988285
Determination Coef. (DC): 0.963628014383883
F-Statistic: 338.131307433748
```

Parameter	Best Estimate
y0	9.99985139768279E-5
a	1.03738725820506
w	8.77141961898219
xc	8.21299279668913

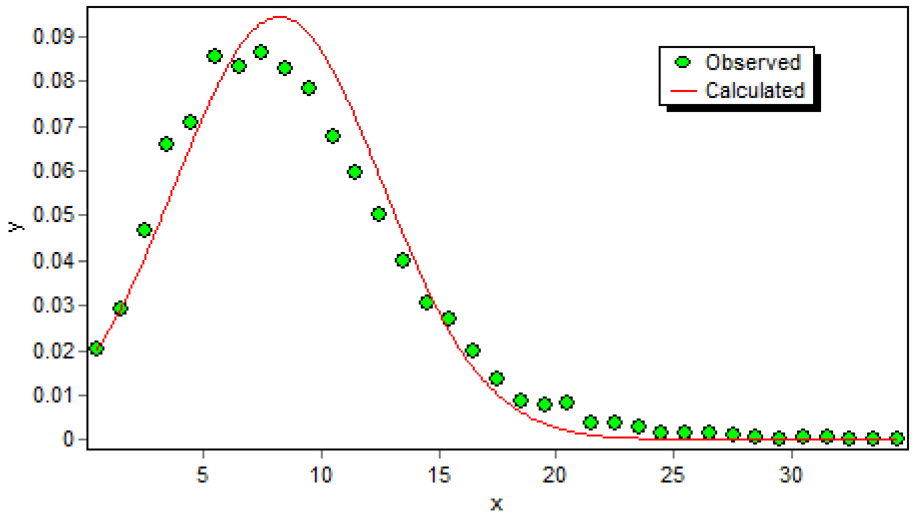

图 18.3 约束拟合结果图

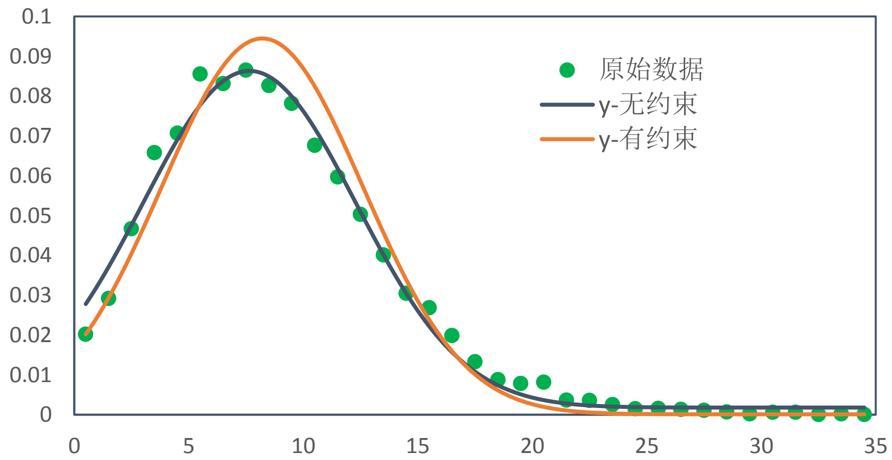

图 18.4 案例一有约束与无约束拟合对比结果图

从计算结果可以看出，无约束时的残差平方和为 0.0003627，增加约束后变为 0.0012235，也即整体拟合效果变差，这也是为满足约束条件而付出的代价。

18.2 案例二：计算因变量不为负约束

拟合公式，数据及图形如下

$$y = c\left(1 + \mathrm{erf}\left(\frac{(x-b)\exp\left(-a\frac{x}{b}\right)}{\sqrt{2}ab}\right)\right) + px \tag{18-4}$$

表 18.2 数据：

x	0,50,100,150,200,250,300,350,400,450,500,550,600,650,700,750,800,850,900,950,1000,1050,1100,1150,1200,1250,1300,1350,1400,1450,1500,1550,1600
y	0,0,0,0,0,0,0,0,0,0.04,0.08,0.14104,0.18635,0.24,0.3,0.38272,0.44474,0.5573,0.64532,0.69053,0.76975,0.82394,0.86369,0.90259,0.93255,0.96256,0.98994,0.99023,0.99621,0.99821,1,1,1

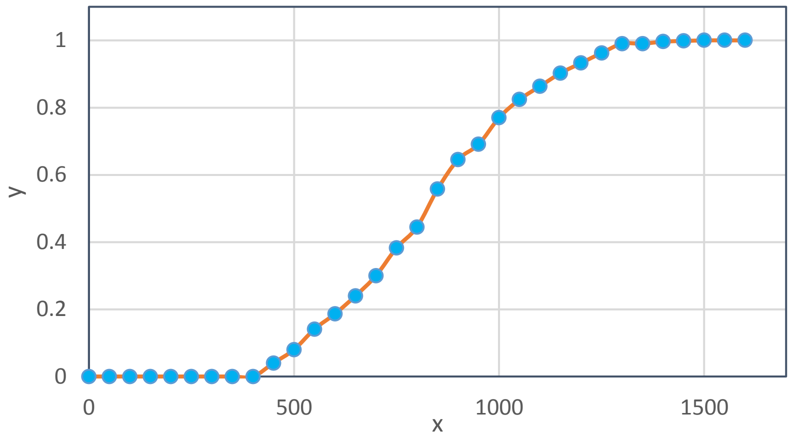

图 18.5 拟合数据原始图-2

如果按正常拟合，代码如下：

代码 18-4

```
Function y=c*(1+Erf(((x-b)*exp(-a*(x/b)))/(sqrt(2)*a*b)))+p*x;
Data;
x=[0,50,100,150,200,250,300,350,400,450,500,550,600,650,700,750,800,850,900,950,1000,1050,1100,1150,1200,1250,1300,1350,1400,1450,1500,1550,1600];
y=[0,0,0,0,0,0,0,0,0,0.04,0.08,0.14104,0.18635,0.24,0.3,0.38272,0.44474,0.5573,0.64532,0.69053,0.76975,0.82394,0.86369,0.90259,0.93255,0.96256,0.98994,0.99023,0.99621,0.99821,1,1,1];
```

结果

```
Root of Mean Square Error (RMSE): 0.0101815847189418
Sum of Squared Residual: 0.00342093402383665
Correlation Coef. (R): 0.999698630469334
R-Square: 0.999397351762262
Adjusted R-Square: 0.99935717521308
Determination Coef. (DC): 0.999393308805306
Chi-Square: 0.0131149956613571
F-Statistic: 15972.5002973265

Parameter   Best Estimate
----------  -------------
```

```
c       0.53619795114083
b       821.962754103782
a       0.24418464093796
p       -3.35403403119747E-5

====== Output Results ======

No.   Observed y Calculated y
1     0      2.26110298049807E-5
2     0      -0.0015960577397312
3     0      -0.00309672530690528
4     0      -0.00429927695643287
5     0      -0.00483150785762665
6     0      -0.00401405737178218
7     0      -0.000749614273653904
8     0      0.00652907426603518
9     0      0.0197907917125968
10    0.04   0.0411793977373915
11    0.08   0.072645487866163
….
```

从上面结果可看出，前七个计算 y 值中有 6 个小于 0，如果要求所有计算 y 值均大于 0，这种约束该如何加呢？有多种方式可以实现：

1). “For”语句添加前 10 或 20 个点计算 y 值均大于 0 的约束

代码 18-5

```
Constant xx=Variable(x), yy=Variable(y);
Function y=c*(1+Erf(((x-b)*exp(-a*(x/b)))/(sqrt(2)*a*b)))+p*x;
        for(i=1:20,xx)(c*(1+Erf(((xx-b)*exp(-a*(xx/b)))/(sqrt(2)*a*b)))+p*xx>=0);
Data;
x=[0,50,100,150,200,250,300,350,400,450,500,550,600,650,700,750,800,850,900,950,1000,1050,1100,1150,1200,1250,1300,1350,1400,1450,1500,1550,1600];
y=[0,0,0,0,0,0,0,0,0,0.04,0.08,0.14104,0.18635,0.24,0.3,0.38272,0.44474,0.5573,0.64532,0.69053,0.76975,0.82394,0.86369,0.90259,0.93255,0.96256,0.98994,0.99023,0.99621,0.99821,1,1,1];
```

2). “MinIn”语句表示求一组数据中的最小值，即添加前 10 或 20 个点计算 y 值中的最小值大于 0 的约束，该约束满足，其余点一定满足

代码 18-6

```
Constant xx=Variable(x), yy=Variable(y);
Function y=c*(1+Erf(((x-b)*exp(-a*(x/b)))/(sqrt(2)*a*b)))+p*x;
        MinIn(i=1:20,xx)(c*(1+Erf(((xx-b)*exp(-a*(xx/b)))/(sqrt(2)*a*b)))+p*xx)>=0;
Data;
x=[0,50,100,150,200,250,300,350,400,450,500,550,600,650,700,750,800,850,900,950,1000,1050,1100,1150,1200,1250,1300,1350,1400,1450,1500,1550,1600];
y=[0,0,0,0,0,0,0,0,0,0.04,0.08,0.14104,0.18635,0.24,0.3,0.38272,0.44474,0.5573,0.64532,0.69053,0.76975,0.82394,0.86369,0.90259,0.93255,0.96256,0.98994,0.99023,0.99621,0.99821,1,1,1];
```

3). 通过编程模式实现，下面代码为 Basic 语言的实现代码，基本思路是比较

求出所有计算 y 值中的最小值，再加上该最小值大于 0 的约束。

代码 18-7

```
Parameter c,b,a,p;
Variable x,y;
StartProgram [Basic];
Sub MainModel
    dim i as integer
    dim as double temd, miny
    for i = 0 to DataLength - 1
        temd = c*(1+Erf(((x(i)-b)*exp(-a*(x(i)/b)))/(sqrt(2)*a*b)))+p*x(i)
        y(i) = temd
        if i = 0 then
            miny = temd
        else
            miny = min(miny, temd)
        end if
    next
    ConstrainedResult = miny >= 0
End Sub
EndProgram;
Data;
x=[0,50,100,150,200,250,300,350,400,450,500,550,600,650,700,750,800,850,900,950,1000,1050,1100,1150,1200,1250,1300,1350,1400,1450,1500,1550,1600];
y=[0,0,0,0,0,0,0,0,0,0.04,0.08,0.14104,0.18635,0.24,0.3,0.38272,0.44474,0.5573,0.64532,0.69053,0.76975,0.82394,0.86369,0.90259,0.93255,0.96256,0.98994,0.99023,0.99621,0.99821,1,1,1];
```

上面三组代码均可得到下面相同的结果：

```
Root of Mean Square Error (RMSE): 0.0106513922337941
Sum of Squared Residual: 0.00374392116509828
Correlation Coef. (R): 0.999672131131761
R-Square: 0.999344369761517
Adjusted R-Square: 0.999300661078951
Determination Coef. (DC): 0.999336028117272
F-Statistic: 14485.4443017282

Parameter            Best Estimate
-------------------- -------------
c      0.509062807109692
b      822.626834869869
a      0.235420164228861
p      -8.54612338114129E-7

====== Output Results ======

No.    Observed yCalculated y
1      0      1.09933513903922E-5
2      0      1.02118292120978E-17
```

3	0	6.10177162872801E-5
4	0	0.000318304998845739
5	0	0.00104887433035054
6	0	0.00279553127075885
7	0	0.00649549782329825
8	0	0.0135749148933615
9	0	0.0259391509365413
…..		

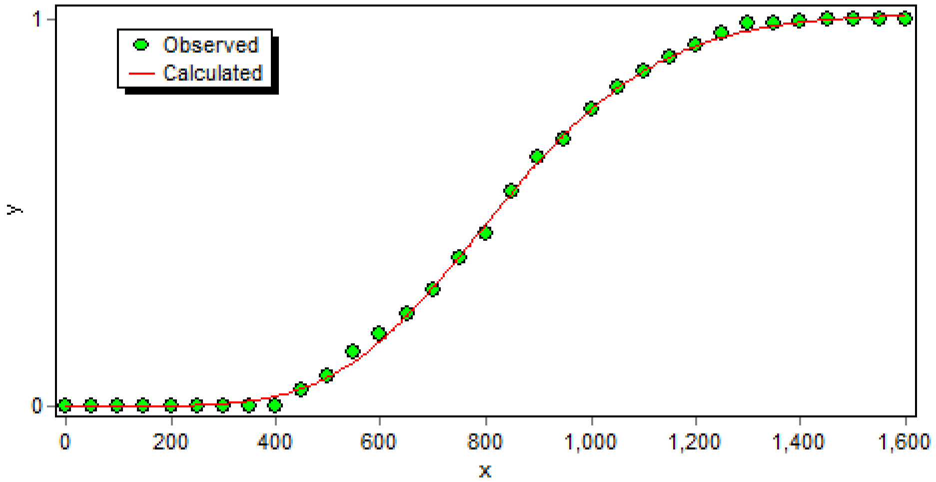

图 18.6 约束拟合对比图

计算结果完全满足计算 y 值均大于 0 的要求，同样，从计算结果也可看出相同的现象：增加约束前后的残差平方和（拟合优化计算的目标函数）分别为 0.0034209 和 0.0037439，满足约束条件的代价就是整体拟合度降低。

18.3 案例三：过点及斜率约束

拟合公式，数据及图形如下

$$y = c_1 \cdot \exp\left(c_2 \cdot (x - c_3)^2\right) + c_4 \cdot x + c_5 \cdot x^2 \qquad \text{(18-4)}$$

表 18.3 数据：

x	27.25,27.75,28.25,28.75,29.25,29.75,30.25,30.75,31.25,31.75,32.25
y	1,3,6,13,18,19,17,16,6,5,2

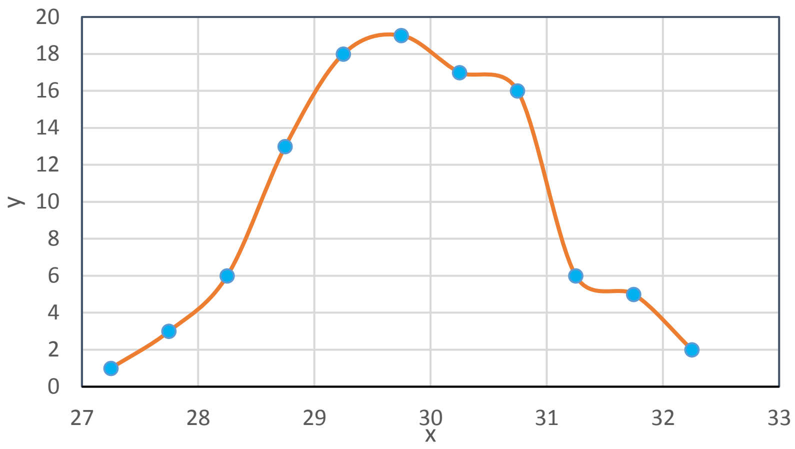

图 18.7 拟合数据原始图-3

约束条件：

1） 过 x=28.75 点时的斜率为 10.3；

2） 过 x= 29.8 点时的斜率为 0，也即该 x 值对应计算 y 值的最大点；

3） 过第一个点[27.25, 1]及最后一个点[32.25, 2]。

斜率也即拟合函数的一阶导数，在 1stOpt 中可用“diff()”函数表示，无约束及有约束代码分别如下。

代码 18-8：无约束

```
Function y=c1*exp(c2*(x-c3)^2)+c4*x+c5*x^2;
Data;
27.25,27.75,28.25,28.75,29.25,29.75,30.25,30.75,31.25,31.75,32.25;
1,3,6,13,18,19,17,16,6,5,2;
```

代码 18-9：有约束

```
Constant sx=[27.25,32.25],sy=[1,2];
Constant xx1=28.75, xx2=29.8;
Function y=c1*exp(c2*(x-c3)^2)+c4*x+c5*x^2;
        diff(c1*exp(c2*(xx1-c3)^2)+c4*xx1+c5*xx1^2,xx1)=10.3;
        diff(c1*exp(c2*(xx2-c3)^2)+c4*xx2+c5*xx2^2,xx2)=0;
        for(i=1:2)(c1*exp(c2*(sx[i]-c3)^2)+c4*sx[i]+c5*sx[i]^2=sy[i]);
Data;
27.25,27.75,28.25,28.75,29.25,29.75,30.25,30.75,31.25,31.75,32.25;
1,3,6,13,18,19,17,16,6,5,2;
```

结果：

无约束	Root of Mean Square Error (RMSE): 1.18364363344302 Sum of Squared Residual: 15.4111347608922 Correlation Coef. (R): 0.984101145693506 R-Square: 0.968455064955272 Adjusted R-Square: 0.96056883119409

	Determination Coef. (DC): 0.968455064687418 Chi-Square: 0.805557960788299 F-Statistic: 46.0524812289073 Parameter Best Estimate ---------- ------------- c1 19.7455509330847 c2 -0.42509748880439 c3 29.7796718427271 c4 -0.225156178478047 c5 0.00752390287863714
有约束	Root of Mean Square Error (RMSE): 1.62579175391143 Sum of Squared Residual: 29.0751870979503 Correlation Coef. (R): 0.976052498044808 R-Square: 0.95267847893951 Adjusted R-Square: 0.940848098674388 Determination Coef. (DC): 0.94048621918916 Chi-Square: 1.65375020512162 F-Statistic: 18.9247208744977 Parameter Best Estimate ---------- ------------- c1 20.4486862040078 c2 -0.253700072099165 c3 29.7885521124898 c4 -0.301991952356591 c5 0.0070598577746585

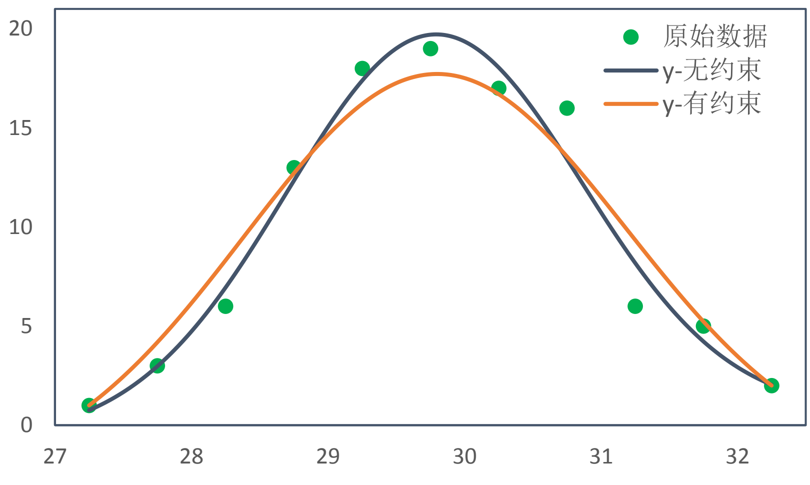

图 18.8 案例三有约束与无约束拟合对比结果图-2

前面求斜率函数“Diff()”可求出曲线函数的解析解，只适用于含初等数学函数的曲线拟合公式，如果含有特殊函数比如伽玛函数或曲线函数形式比较复杂而无法用“Diff()”函数得到一阶导数的解析表达式时，该如何处理？此时可以考虑用数

值微分去替代“Diff()”。

有很多种数值微分计算方式，在此采用对称差分计算公式如下：

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0 - h)}{2h} \tag{18-5}$$

f’(x0)表示 x0 点处的一阶导数（微分），h 为一很小的值，取 1E-5.

代码 18-10：数值微分约束

```
Constant h=1E-5;
ConstStr f(v)=c1*exp(c2*(v-c3)^2)+c4*v+c5*v^2;
Constant sx=[27.25,32.25],sy=[1,2];
Constant xx1=28.75, xx2=29.8;
Function y=f(x);
        (f(xx1+h)-f(xx1-h))/(2*h)=8.3;    //数值微分，xx1 点处斜率约束
        (f(xx2+h)-f(xx2-h))/(2*h)=0;      //数值微分，xx2 点处斜率约束
        for(i=1:2)(c1*exp(c2*(sx[i]-c3)^2)+c4*sx[i]+c5*sx[i]^2=sy[i]); //过点约束
Data;
27.25,27.75,28.25,28.75,29.25,29.75,30.25,30.75,31.25,31.75,32.25;
1,3,6,13,18,19,17,16,6,5,2;
```

运行该段代码可以得到与前面“Diff()”段几乎完全相同的结果，表明数值微分计算斜率约束问题可以满足精度要求，从而也可以广泛应用于无法求解一阶导数解析解的情况，更具普适性。

18.4 小结

1stOpt 可以方便处理各类特殊约束拟合问题，不仅实现效果好，运行代码也简单易懂。