

曲线拟合的方法

The Method of Curve Fitting

耿爱成 GENG Ai-cheng

(沈阳工程学院基础教学部 沈阳 110136)

(Department of Preparatory Courses, Shenyang Institute of Engineering, Shenyang 110136, China)

摘要: 在科学实验和工程实际应用中,经常需要通过一组测试数据来分析自变量和因变量之间的近似函数关系。这篇文章详细介绍了借助 matlab 软件来实现曲线拟合的各种方法,使得曲线拟合的计算变得简单。

Abstract: In the practical application of scientific experiment and engineering, a set of test datas are often used to analyze the approximate function relationship between the independent variable and the dependent variable. This paper introduces the various methods of using matlab software to realize curve fitting, which makes the calculation of curve fitting more simple.

关键词: 曲线拟合;最小二乘法;matlab

Key words: curve fitting; least squares method; matlab

中图分类号: TP29

文献标识码: A

文章编号: 1006-4311(2017)31-0182-03

DOI:10.14018/j.cnki.cn13-1085/n.2017.31.080

0 引言

在工程实际应用和科学实验中通过测量得到的一组离散的数据点,为了从中找到两个变量之间的内在规律性,也就是求自变量和因变量之间的近似程度比较好的函数关系式,这类问题有插值法和曲线拟合法。当个别数据的误差较大时,插值效果显然是不理想的。

此外,由实验或观测提供的数据个数往往很多,如果用插值法,势必得到次数较高的插值多项式,会出现龙格现象。为此,希望从给定的数据出发构造一个近似函数,不要求该函数完全通过所有的数据点,只要求所得的近似曲线能反映数据的基本趋势,使求得的逼近函数与已知函数从总体上来说偏差的平方和最小,这就是最小二乘法。

1 最小二乘曲线拟合的方法

1.1 应用 polyfit 函数或 polytool 函数进行代数多项式拟合

调用格式 $p = \text{polyfit}(x, y, n)$ $p = \text{polytool}(x, y)$

其中 x 和 y 分别为自变量和因变量的数据向量, n 为拟合多项式的次数, $n=1$ 时就是线性拟合,而返回的 p 为所求拟合多项式的系数按照降幂排列得到的行向量。

例题 1. 某种铝合金当含铝量为 x 时,其熔解温度为 y ,由实验测得 x 与 y 的数据如表 1 所示,用一次多项式 $y = ax + b$ 来拟合该数据。

表 1

$x/\%$	36.9	46.7	63.7	77.8	84.0	87.5
$y/^\circ\text{C}$	181	197	235	270	283	292

解: 在 matlab 命令窗口中输入:

```
>>x=[36.9 46.7 63.7 77.8 84.0 87.5]
```

```
>>y=[181 197 235 270 283 292]
```

作者简介:耿爱成(1979-)男,满族,辽宁开原人,讲师,硕士。

```
>>plot(x,y,'r*','markersize',10)
```

```
>>p=polyfit(x,y,1)
```

```
>>x1=36:0.1:88
```

```
>>y1=polyval(p,x1)
```

```
>>hold on
```

```
>>plot(x1,y1)
```

```
>>a=p(1)
```

```
>>b=p(2)
```

最后得到所求拟合直线为 $y = 2.2337x + 95.3524$ 。

如果在 matlab 命令窗口输入命令

```
>>polytool(x, y)
```

则会出现一个交互式窗口。

在该窗口的上面可以选择拟合多项式的次数,在下面输入自变量数值后左侧会给出相应函数值预测值,从左下角的 Export 中可以导出所需要拟合多项式的系数等信息。polytool 比 polyfit 使用起来更方便些。

1.2 应用 lsqcurvefit 和 lsqnonlin 进行非线性曲线拟合

调用格式 $x = \text{lsqcurvefit}('fun', x0, xdata, ydata)$

$x = \text{lsqnonlin}('fun', x0)$

其中 fun 是一个事先建立的拟合函数的 M-文件, $x0$ 为迭代初值, $xdata$ 和 $ydata$ 为已知数据向量。

例题 2 用表 2 中一组数据拟合函数 $y = ce^{ax}$ 中的参数 a 和 c 。

表 2

x_i	0	1	2	3	4
y_i	1.5	2.5	3.5	5.0	7.5

解法 1. 用命令 lsqcurvefit

①编写 M-文件 curvefun1.m

```
function f=curvefun1(x,tdata)
```

```
f= x(1)*exp(x(2)*tdata)
```

%其中 $x(1)=c$; $x(2)=a$;

②输入命令

```
>>xdata=0:4
>>ydata=[1.5,2.5,3.5,5.0,7.5]
>> x0=[0.5,0.5]
>>x=lsqcurvefit('curvefun1',x0,xdata,ydata)
```

③运算结果为

x=1.6109 0.3836

④结论 c=1.6109 , a=0.3836

解法 2 用命令 lsqnonlin

①编写 M-文件 curvefun2.m

```
function f=curvefun2(x)
xdata=0:4;
ydata=[1.5,2.5,3.5,5.0,7.5];
f= x(1)*exp(x(2)*xdata)- ydata
```

②输入命令

```
>>x0=[0.5,0.5]
>>x=lsqnonlin('curvefun2',x0)
```

③运算结果

x =1.6109 0.3836

④结论 c=1.6109 , a=0.3836

可以看出,两个命令的计算结果是相同的。

1.3 用 cftool 命令进行曲线拟合

Matlab 的曲线拟合工具箱有强大的图形拟合功能,是一个可视化的图形界面。下面以例题 1 为例介绍一下拟合工具箱 cftool 的使用方法。在 matlab 命令窗口中输入以下命令

```
>>x=[36.9 46.7 63.7 77.8 84.0 87.5]
>>y=[181 197 235 270 283 292]
>>cftool
```

就可以得到一个拟合工具箱界面。

单击“Data”按钮会弹出“Data”窗口,在 Xdata 和 Ydata 的下拉菜单中读入事先定义的数据向量 x 和 y,如果数据还有权重方面的限制,那么可以在 weights 的下拉菜单中读入权重的向量,单击“Creat data set”按钮,然后再单击“Close”按钮就可以退出“Data”窗口,返回到拟合工具箱界面,单击“Fitting”按钮会弹出“Fitting”对话框,单击“New fit”,在“Fit Name”中输入“拟合直线”,在“Type of fit”中选择“polynomial”,在“polynomial”的下面可以根据需要选择多项式的次数(系统提供了 1 次到 9 次多项式),这里我们选择 linear polynomial,然后单击“Apply”,就会得到拟合函数的参数及残差。

1.4 用 matlab 中的回归命令实现曲线拟合

1.4.1 多元线性回归命令 regress()

调用格式 [b,bint,r,rint,stats]=regress(Y,X,alpha)

其中 b 为回归系数的估计值(第一个为常数项);

bint 为回归系数的区间估计;

r 残差,rint 残差的置信区间;

stats 用于检验回归模型的统计量,有 4 个数值:相关系数 r^2 、F 值、与 F 对应的概率 p 和残差的方差(前两个越大越好,后两个越小越好);

alpha 显著性水平(缺省时为 0.05,即置信水平为 95%)

下面对例 1 用 regress 进行拟合。

①在 matlab 输入数据:

```
>>x=[36.9 46.7 63.7 77.8 84.0 87.5]
>>X=[ones(6,1) x]
>>Y=[181 197 235 270 283 292]'
```

②回归分析及检验

```
>>[b,bint,r,rint,stats]=regress(Y,X)
```

得结果:

```
b =
    95.3524
    2.2337
bint =
    84.7246 105.9803
    2.0791 2.3883
stats =1.000 1609.8 0.0000 6.700
```

因为 $r^2=1$, $F=1609.8$, $p=0$, 所以回归模型 $y=2.2337x+95.3524$ 成立。与用 polyfit 得到的结果一样。

③残差分析,作残差图

```
>>Rcplot(r,rint)
```

从残差图可以看出,除第一个数据外,其余数据的残差距离零点均较近,且残差的置信区间均包含零点,这说明回归模型能很好的符合原始数据,而第一个数据可视为异常点。

1.4.2 多元二项式回归命令 rstool()

调用格式 rstool(x,y,'model',alpha)

其中 x 表示 n 行 m 列矩阵, y 表示 n 维列向量, alpha 为显著性水平(缺省时为 0.05), model 表示由下列 4 个模型中选择 1 个(用字符串输入,默认为线性模型):

linear(线性) $y=\beta_0+\beta_1x_1+\cdots+\beta_mx_m$

purequadratic(纯二次) $y=\beta_0+\beta_1x_1+\cdots+\beta_mx_m+\sum_{j=1}^m\beta_{jj}x_{jj}^2$

interaction(交叉) $y=\beta_0+\beta_1x_1+\cdots+\beta_mx_m+\sum_{1\leq j\neq k\leq m}\beta_{jk}x_jx_k$

quadratic(完全二次) $y=\beta_0+\beta_1x_1+\cdots+\beta_mx_m+\sum_{1\leq j,k\leq m}\beta_{jk}x_jx_k$

$\beta_{jk}x_jx_k$

例题:设某商品的需求量与消费者的平均收入、商品价格的统计数据如表 3,建立回归模型。

表 3

需求量	100	75	80	70	50	65	90	100	110	60
收入	1000	600	1200	500	300	400	1300	1100	1300	300
价格	5	7	6	6	8	7	5	4	3	9

解:选择纯二次模型直接用多元二次项回归:

```
>>x1=[1000,600,1200,500,300,400,1300,1100,1300,300];
>>x2=[5,7,6,6,8,7,5,4,3,9];
>>y=[100,75,80,70,50,65,90,100,110,60]';
>>x=[x1',x2'];
>>rstool(x,y,'purequadratic')
```

在所得到的下方的文本框中输入 x_1 x_2 的值后,则在画面左边的“Predicted Y”下方显示预测的数值。通过画面左下方的 Export 菜单可以把 beta rmse 和 residuals 的值都传送到 matlab 工作区中。

在 matlab 工作区中输入命令:

```
>>beta ,
```

得结果 beta =

110.5313

0.1464

-26.5709

-0.0001

1.8475

故回归模型为:

$$y=110.5362+0.1464x_1-26.5709x_2-0.0001x_1^2+1.8475x_2^2$$

1.4.3 非线性回归命令 nlinfit 和 nlintool

①确定回归系数的命令 nlinfit

调用格式 [beta,r,J]=nlinfit(x,y,'fun',beta0)

其中 beta 表示估计出的回归系数 (即参数的最优值) r 是各点处的拟合残差 J 为雅可比矩阵的数值 x y 为已知数据 x 为矩阵 y 为 n 维列向量,对一元非线性回归 x 为 n 维列向量 fun 是在 M 文件中定义的非线性函数,也可以用内联函数 inline 来定义 fun 函数 beta0 是函数中参数的初值。

②非线性回归交互命令 nlintool

调用格式 nlintool(x,y,'fun',beta0,alpha)

下面对例题 2 的数据用 nlintool 进行拟合

1)编写 M-文件 curvefun1.m

```
function f=curvefun1(x,tdata)
```

```
f= x(1)*exp(x(2)*tdata)
```

2)输入数据

```
>>x=0:4;
```

```
>>y=[1.5,2.5,3.5,5.0,7.5];
```

```
>> beta0=[0.5,0.5];
```

3)求回归系数

```
>> [beta,r,J]= nlinfit (x' y' , 'curvefun1' ,beta0) ;
```

```
>>beta
```

得到 beta =

1.6109 0.3836

4)预测及作图

```
>>[Y,DELTA]=nlpredci('curvefun1' x' beta r J)
```

```
>>plot(x,y, 'b+', x, Y, 'r')
```

如果输入命令

```
>>nlintool(x,y,'curvefun1',[0.5,0.5])
```

则会出现一个交互式对话框 通过画面左下角 Export 菜单可以把需要的数据导入 matlab 中。

1.5 图形界面拟合

除了直接通过命令方式来实现曲线拟合外,也可以在图像界面中进行曲线拟合。使用图像界面曲线拟合 非常直观,拟合后的数据结果可以保存在工作空间中。

下面以例 1 为例看看如何使用图形窗口进行曲线拟合。

在 matlab 命令窗口中输入命令:

```
x=[36.9 46.7 63.7 77.8 84.0 87.5];
```

```
y=[181 197 235 270 283 292];
```

```
plot(x,y,'r*','markersize',10)
```

在所得到的图形界面中选择 “Tools” 菜单中的 Basic Fitting(位于最下面第二个)后,系统会弹出一个曲线拟合窗口。在该窗口中根据需要选择曲线拟合的类型,可以选择是否显示方程,是否绘制残差,是否显示残差的范数。当然还可以把所得到的结果保存到工作空间中。如果此处选择 linear 方式进行拟合,拟合的效果跟 polyfit 函数拟合结果一样。

1.6 线性化变换的方法

在许多实际问题中变量之间的函数关系不是呈线性关系,但是可以把非线性的拟合函数通过线性变换的方法转换成多项式函数。例如在 $y=ce^{ax}$ 两边取对数,得 $\ln y = \ln c + ax$ 。令 $Y = \ln y$ $b = \ln c$ 就可以转化为线性函数 $Y = b + ax$ 。如果把例题 2 中问题用线性化的方法来解决拟合问题,应该在 matlab 命令窗口中输入:

```
>> x=0:4;
```

```
>> y=[1.5,2.5,3.5,5.0,7.5];
```

```
>> Y=log(y);
```

```
>> p=polyfit(x,Y,1);
```

```
>> a=p(1)
```

```
>> c=exp(p(2))
```

1.7 分段曲线拟合

对数据点的变化趋势比较,如果只采用一种函数曲线去拟合比较多的数据点,难以取得较好的拟合效果。有时候在进行曲线拟合时,如果前面的一些数据符合一种曲线的变化趋势,而后面数据符合另外一种曲线的变化趋势,为了有效地解决此类问题,此时要考虑分段曲线拟合。

2 结束语

本文详细地介绍了实现最小二乘曲线拟合的各种方法。拟合曲线的函数类型的确定是比较困难的。当人们系统地掌握了研究对象的各个因素间的关系后,一般采用机理分析法构建数学模型,通过曲线拟合来确定模型中各参数的取值。

在无法使用机理分析法进行建模分析的情况下,就必须根据观测数据绘制散点图,再对照散点图预测函数类型,继而结合上机实验和误差分析进行对比分析,最终确定拟合效果最佳的函数类型。

参考文献:

[1]关治,陆金甫.数值方法[M].北京:清华大学出版社,2011.

[2]陈岚峰,杨静瑜,崔崧,潘庆超,李柳.基于 MATLAB 的最小二乘曲线拟合仿真研究[J].沈阳师范大学学报(自然科学版),2014,32(1):75-79.

[3]唐家德.基于 MATLAB 的非线性曲线拟合[J].计算机与现代化,2008(6):15-19.