数据拟合

数据拟合

用插值的方法对一函数进行近似，要求所得到的插值多项式经过已知插值节点；在n比较大的情况下，插值多项式往往是高次多项式，这也就容易出现振荡现象（龙格现象），即虽然在插值节点上没有误差，但在插值节点之外插值误差变得很大，从“整体”上看，插值逼近效果将变得“很差”。

具体所谓数据拟合是求一个简单的函数，例如是一个低次多项式，不要求通过已知的这些点，而是要求在整体上“尽量好”的逼近原函数。这时，在每个已知点上就会有误差，数据拟合就是从整体上使误差，尽量的小一些。

1 多项式拟合

- n次多项式： $g(x)=c_1x^n+c_2x^{n-1}+\cdots+c_{n+1}$
- 曲线与数据点的残差为：

$$r_i=y_i-g(x_i),\quad i=1,2,\cdots,L$$

- 残差的平方和为：

$$R=\sum_{i=1}^{L}r_i^2$$

- 为使其最小化，可令R关于 c_j 的偏导数为零，即：

$$\frac{\partial R}{\partial c_j}=0,\quad j=1,2,\cdots,n+1$$

- 或 $$\sum_{j=1}^{n+1}\left(\sum_{i=1}^{L} x_i^{2n+2-j-k}\right)c_j=\sum_{i=1}^{L} x_i^{n+1-k} y_i,\ k=1,2,\cdots,n+1$$
- 或矩阵形式：

$$\begin{bmatrix} \sum_{i=1}^{L} x_i^{2n} & \sum_{i=1}^{L} x_i^{2n-1} & \cdots & \sum_{i=1}^{L} x_i^{n} \\ \sum_{i=1}^{L} x_i^{2n-1} & \cdots & \cdots & \sum_{i=1}^{L} x_i^{n-1} \\ \cdots & \cdots & \cdots & \cdots \\ \sum_{i=1}^{L} x_i^{n} & \cdots & \cdots & \sum_{i=1}^{L} x_i^{0} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_{n+1} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{L} x_i^{n} y_i \\ \sum_{i=1}^{L} x_i^{n-1} y_i \\ \cdots \\ \sum_{i=1}^{L} y_i \end{bmatrix}$$

多项式拟合MATLAB命令：polyfit

格式：p=polyfit(x,y,n)

其中:

x 和 y 为原始的样本点构成的向量

n 为选定的多项式阶次

p 为多项式系数按降幂排列得出的行向量

例　已知的数据点来自 $f(x) = (x^2-3x+5)e^{-5x}\sin x$，用多项式拟合的方法在不同的阶次下进行拟合

拟合该数据的 3 次多项式:

```
x0=0:.1:1; y0=(x0.^2-3*x0+5).*exp(-5*x0).*sin(x0);
p3=polyfit(x0,y0,3)
vpa(poly2sym(p3),10)    % 显示为多项式
```

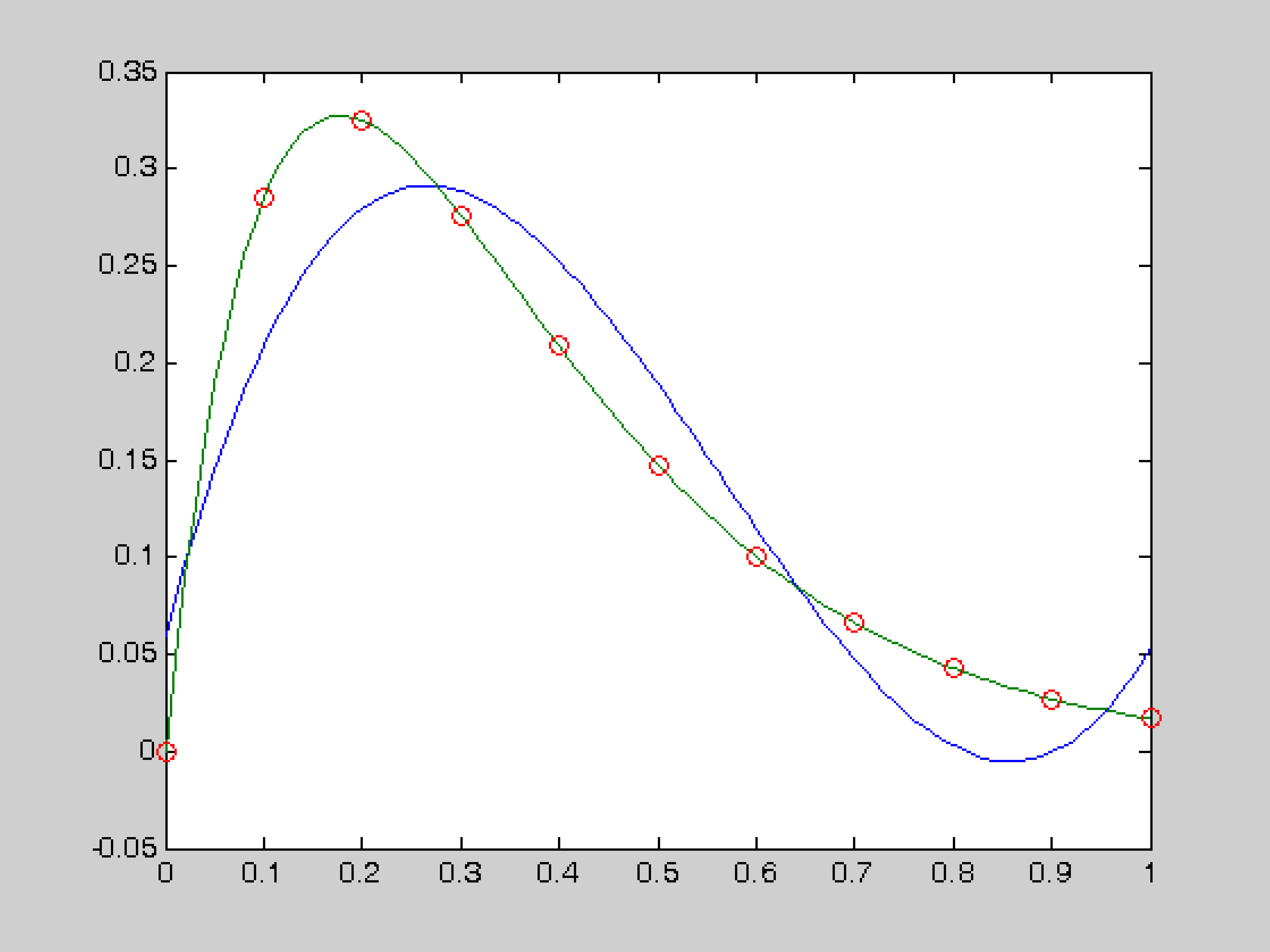
```
p3 =
   2.8400   -4.7898    1.9432    0.0598

ans =
2.839962923*x^3-4.789842696*x^2+1.943211631*x+.5975248921e-1
```

- 绘制拟合曲线：

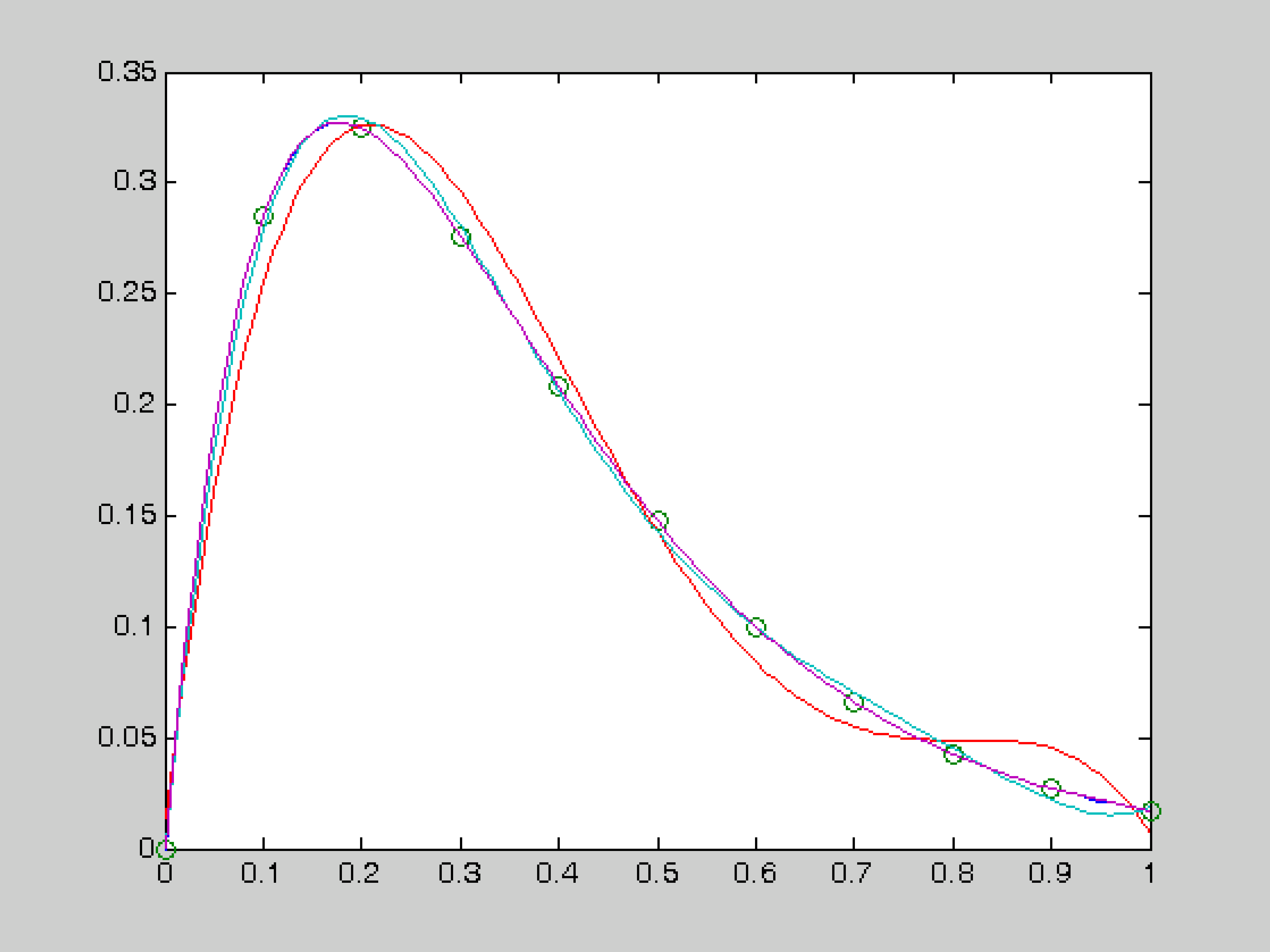
x=0:.01:1; ya=(x.^2-3*x+5).*exp(-5*x).*sin(x);

y1=polyval(p3,x); plot(x,y1,x,ya,x0,y0,'o')



- 就不同的次数进行拟合：

```
p4=polyfit(x0,y0,4); y2=polyval(p4,x);
p5=polyfit(x0,y0,5); y3=polyval(p5,x);
p8=polyfit(x0,y0,8); y4=polyval(p8,x);
plot(x,ya,x0,y0,'o',x,y2,x,y3,x,y4)
```



- 拟合最高次数为8的多项式：

```
>> vpa(poly2sym(p8),5)
ans =
-8.2586*x^8+43.566*x^7-101.98*x^6+140.22*x^5-125.29*x^4+74.450*x^3-
    27.672*x^2+4.9869*x+.42037e-6
```

- Taylor幂级数展开：

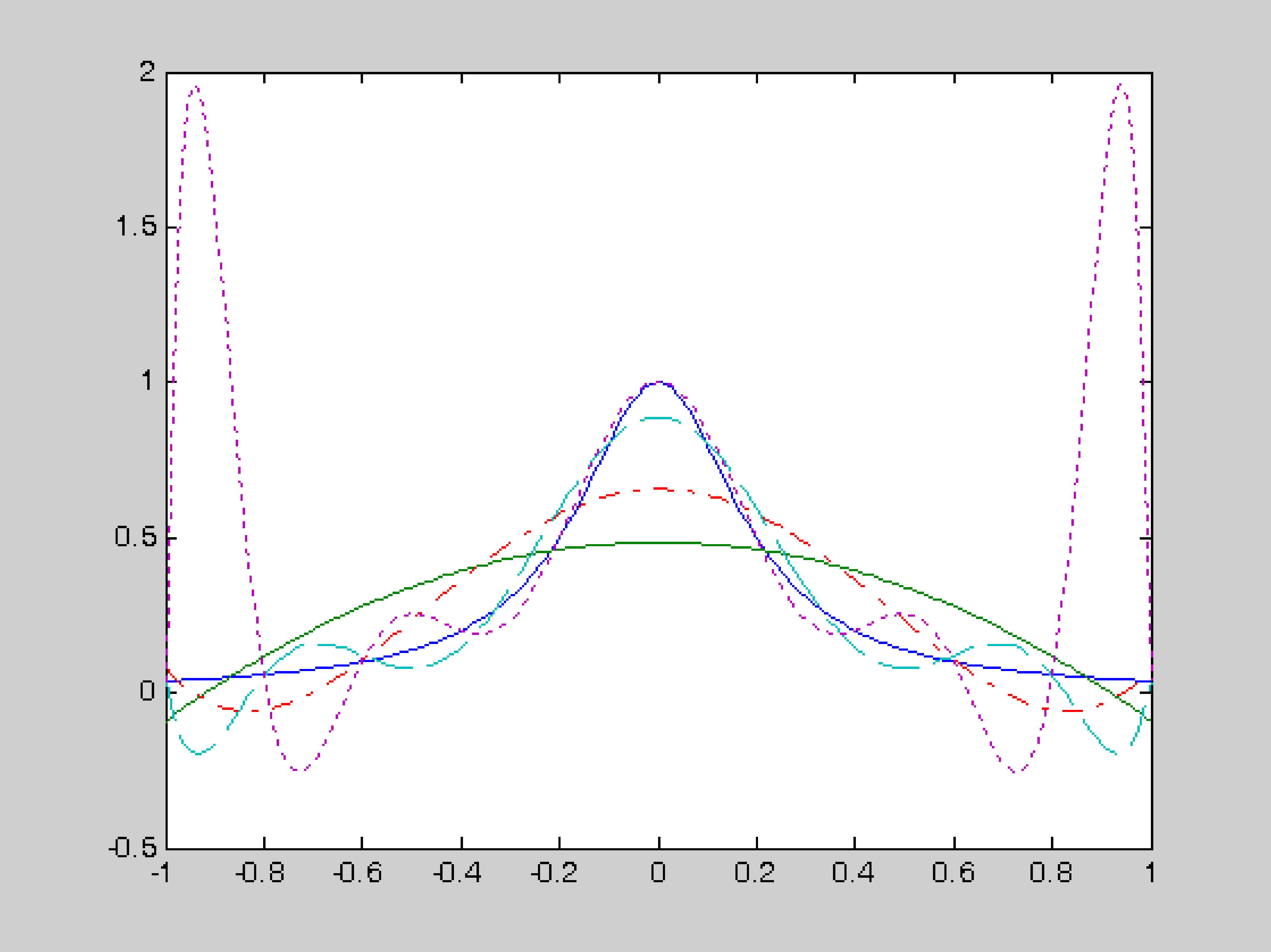
```
>> syms x; y=(x^2-3*x+5)*exp(-5*x)*sin(x);
>> vpa(taylor(y,9),5)
ans =
5.*x-28.*x^2+77.667*x^3-142.*x^4+192.17*x^5-204.96*x^6+179.13*x^7-131.67*x^8
```

- 多项式表示数据模型是不唯一的，即是两个多项式函数完全不同。在某一区域内其曲线可能特别近似。

例 对$f(x) = 1/(1+25x^2), -1 \le x \le 1$进行多项式拟合

多项式拟合的效果并不一定总是很精确的。

```
x0=-1+2*[0:10]/10; y0=1./(1+25*x0.^2);
x=-1:.01:1; ya=1./(1+25*x.^2);
p3=polyfit(x0,y0,3);
y1=polyval(p3,x);
p5=polyfit(x0,y0,5);
y2=polyval(p5,x);
p8=polyfit(x0,y0,8);
y3=polyval(p8,x);
p10=polyfit(x0,y0,10);
y4=polyval(p10,x);
plot(x,ya,x,y1,x,y2,'-.',x,y3,'--',x,y4,':')
```

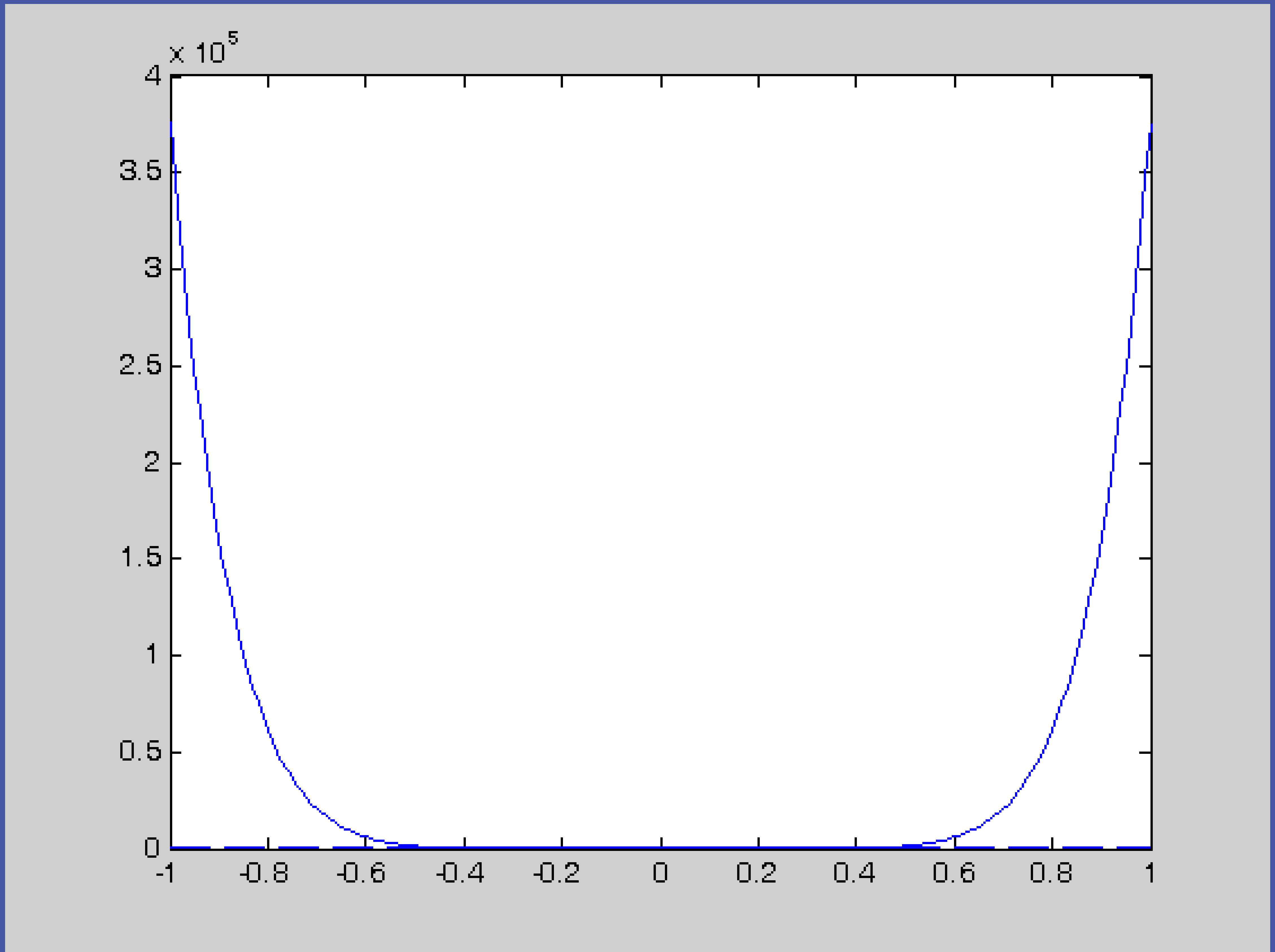


- 用Taylor幂级数展开效果将更差。

```
>> syms x; y=1/(1+25*x^2); p=taylor(y,x,10)
p =
1-25*x^2+625*x^4-15625*x^6+390625*x^8
```

- 多项式拟合效果

```
>> x1=-1:0.01:1;
ya=1./(1+25*x1.^2);
y1=subs(p,x,x1);
plot(x1,ya,'--',x1,y1)
```



2 函数线性组合的曲线拟合方法

已知某函数的线性组合为:

$$g(x) = c_1 f_1(x) + c_2 f_2(x) + c_3 f_3(x) + \cdots + c_n f_n(x)$$

其中 $f_1(x), f_2(x), \cdots, f_n(x)$ 为已知函数 $c_1, c_2, \cdots, c_n$ 为待定系数

假设已经测出数据 $(x_1, y_1), (x_2, y_2), \cdots, (x_M, y_M)$

则可以建立如下的线性方程:

$$\boldsymbol{A}\boldsymbol{c} = \boldsymbol{y}$$

其中

$$A=\begin{bmatrix} f_1(x_1) & f_2(x_1) & \cdots & f_n(x_1) \\ f_1(x_2) & f_2(x_2) & \cdots & f_n(x_2) \\ \cdots & \cdots & \cdots & \cdots \\ f_1(x_M) & f_2(x_M) & \cdots & f_n(x_M) \end{bmatrix}, y=\begin{bmatrix} y_1 \\ y_2 \\ \cdots \\ y_M \end{bmatrix}$$

且 $\boldsymbol{c}=[c_1,c_2,\cdots,c_n]^T$

该方程的最小二乘解为:

$$\boldsymbol{c}=\boldsymbol{A}\backslash\boldsymbol{y}$$

例 假设测出一组 (x_i, y_i)，已知函数原型为

$$y(x) = c_1 + c_2 e^{-3x} + c_3 \cos(-2x) e^{-4x} + c_4 x^2,$$

用已知数据求出待定系数 c_i 的值

x_i	0	0.2	0.4	0.7	0.9
y_i	2.88	2.2576	1.9683	1.9258	2.0862

x_i	0.92	0.99	1.2	1.4	1.48	1.5
y_i	2.109	2.1979	2.5409	2.9627	3.155	3.2052

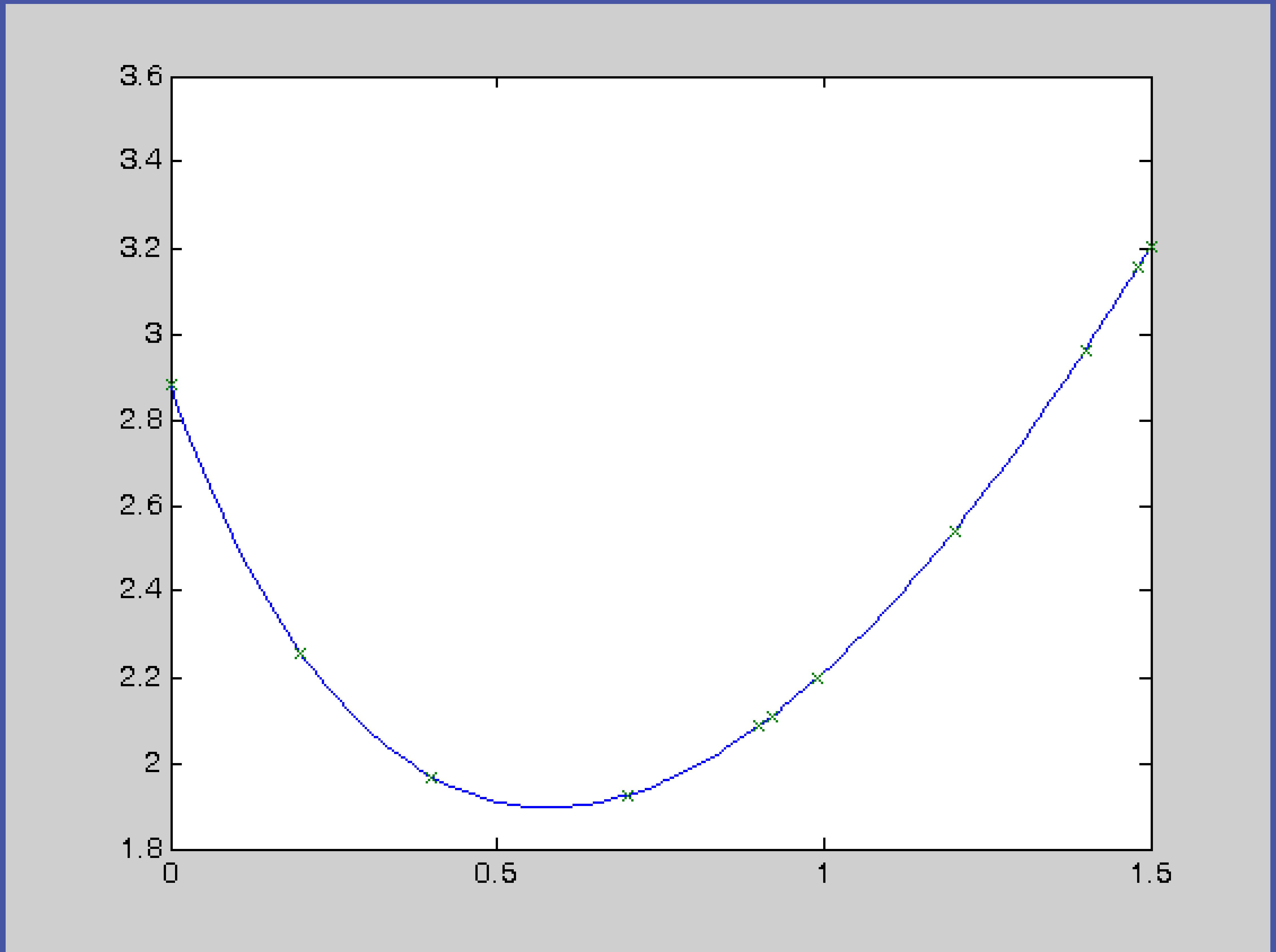
直接拟合 c_i 参数:

```
>> x=[0,0.2,0.4,0.7,0.9,0.92,0.99,1.2,1.4,1.48,1.5]';
y=[2.88;2.2576;1.9683;1.9258;2.0862;2.109;
   2.1979;2.5409;2.9627;3.155;3.2052];
A=[ones(size(x)),exp(-3.*x), cos(-2.*x).*exp(-4.*x),x.^2];
c=A\y; c1=c'

c1 =
   1.2200    2.3397   -0.6797    0.8700
```

- 图形显示

```
>> x0=[0:0.01:1.5]';
A1=[ones(size(x0)),exp(-3*x0), cos(-2*x0).*exp(-4*x0), x0.^2];
y1=A1*c;
plot(x0,y1,x,y,'x')
```



例　　假设测出一组实际数据，对其进行函数拟合

x_i	1.1052	1.2214	1.3499	1.4918	1.6487
y_i	0.6795	0.6006	0.5309	0.4693	0.4148

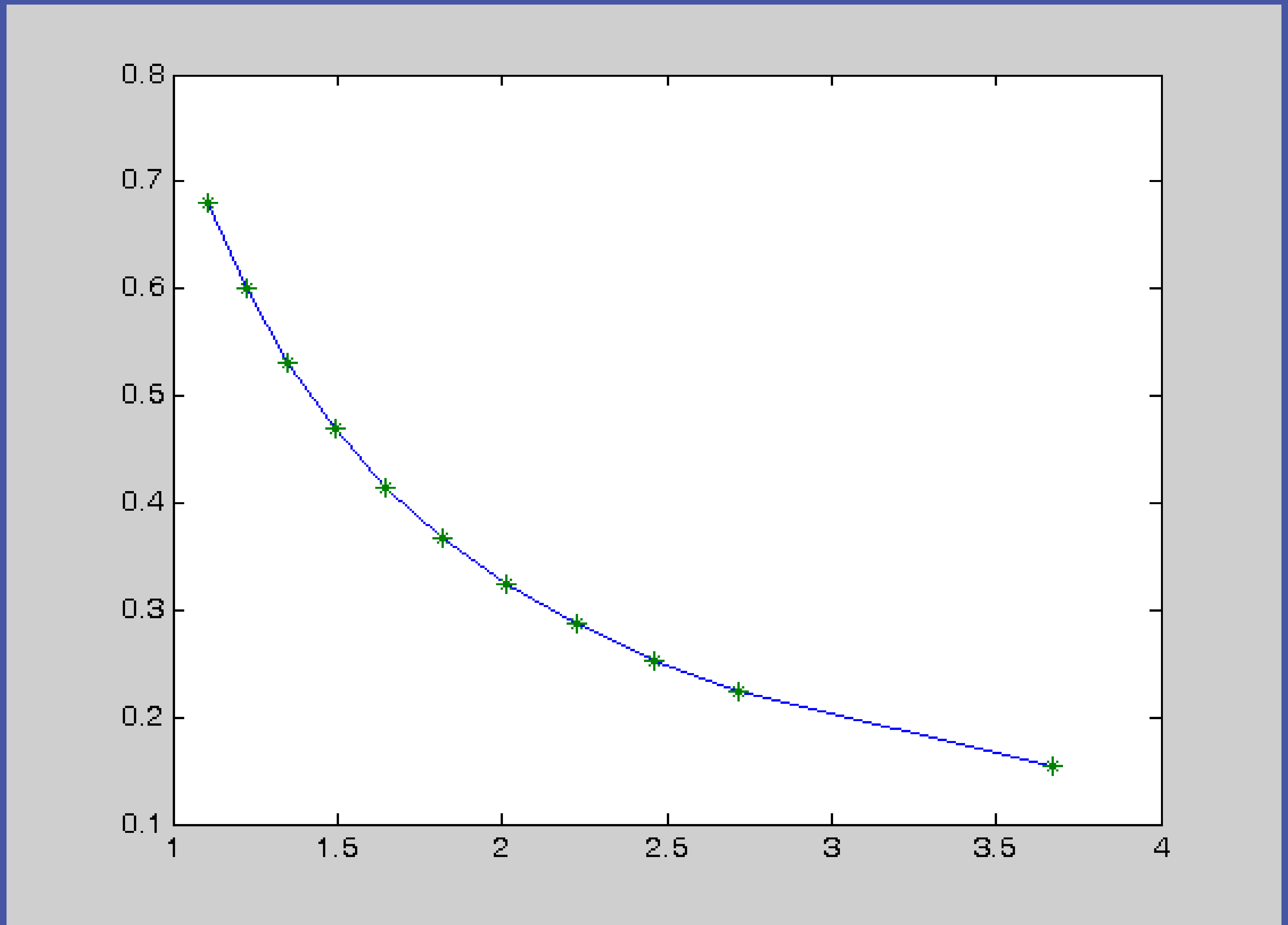
x_i	1.8221	2.0138	2.2255	2.4596	2.7183	3.6693
y_i	0.3666	0.3241	0.2865	0.2532	0.2238	0.1546

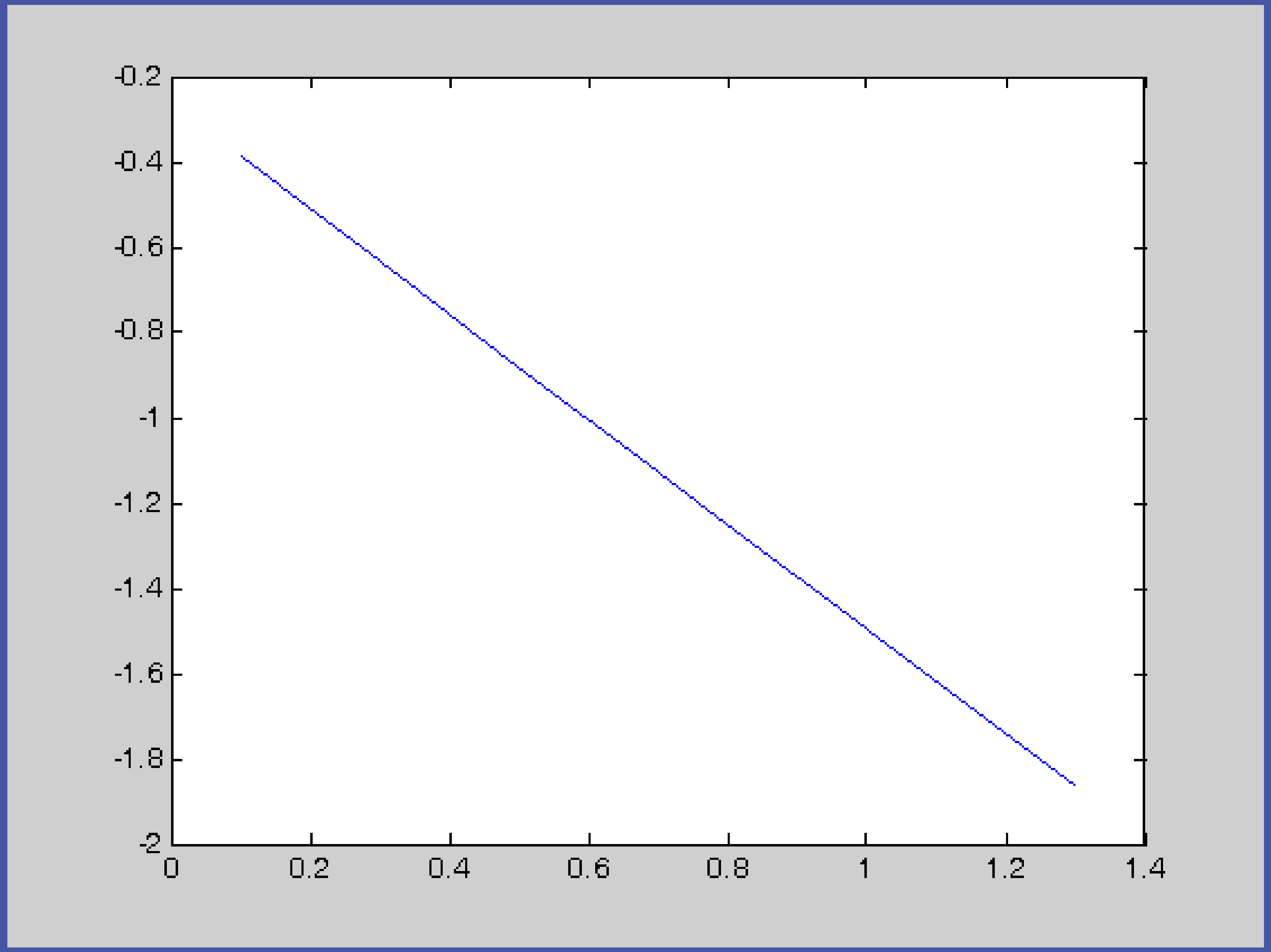
- 数据分析

```
>> x=[1.1052,1.2214,1.3499,1.4918,1.6487,1.8221,2.0138,...
    2.2255,2.4596,2.7183,3.6693];
y=[0.6795,0.6006,0.5309,0.4693,0.4148,0.3666,0.3241,...
    0.2864,0.2532,0.2238,0.1546];
plot(x,y,x,y,'*')
```

- 分别对*x*,*y*进行对数变换：

```
>> x1=log(x); y1=log(y);  plot(x1,y1)
```





用线性函数拟合的方法可以得出线性参数，

使得 $\ln y = a \ln x + b$，即 $y = \mathrm{e}^{b} x^{a}$

求解系数 a, b 及 e^{b}：

```
>> A=[x1', ones(size(x1'))]; c=[A\y1']'
c =
  -1.2339   -0.2630

>> exp(c(2))
ans =
   0.7687
```

拟合函数：$y(x) = 0.76871338819924 x^{-1.23389448522593}$

例 对 $f(x) = (x^2 - 3x + 5)\mathrm{e}^{-5x}\sin x$ 进行多项式 拟合，可以选择各个函数为

$$f_i(x) = x^{n+1-i},\quad i = 1, 2, \cdots, n,$$ 并观察多项式拟合的效果

```
>> x=[0:0.1:1]'; y=(x.^2-3*x+5).*exp(-5*x).*sin(x); n=8; A=[];
for i=1:n+1, A(:,i)=x.^(n+1-i); end
c=A\y; vpa(poly2sym(c),5)
```

```
ans =
-8.2586*x^8+43.566*x^7-101.98*x^6+140.22*x^5-125.29*x^4+74.450*x^3-
27.672*x^2+4.9869*x+.42037e-6
```

3 最小二乘曲线拟合

有一组数据 $x_i, y_i, i = 1, 2, \cdots, N$

满足某一函数原型 $\hat{y}(x) = f(\boldsymbol{a}, x)$，其中 $\boldsymbol{a}$ 为待定系数向量

则最小二乘曲线拟合的目标：

求出这一组待定系数的值，使得目标函数

$$J = \min_{\boldsymbol{a}} \sum_{i=1}^{N} [y_i - \hat{y}(x_i)]^2 = \min_{\boldsymbol{a}} \sum_{i=1}^{N} [y_i - f(\boldsymbol{a}, x_i)]^2$$

为最小

- 格式：

$$[a, j_m]=\text{lsqcurvefit}(\text{Fun},a_0,x,y)$$

其中： Fun 为原型函数的 MATLAB 表示，

可以是 M-函数或 inline() 函数

a_0 为最优化的初值

x, y 为原始输入输出数据向量

a 为返回的待定系数向量

J_m 为在此待定系数下的目标函数的值

例　　由下面的语句生成一组数据，

```
>> x=0:.1:10;
>> y=0.12*exp(-0.213.*x)+0.54*exp(-0.17.*x).*sin(1.23.*x);
```

并且该数据满足 $y(x) = a_1 e^{-a_2 x} + a_3 e^{-a_4 x}\sin(a_5 x)$， 其中 a_i 为待定系数，

采用最小二乘曲线拟合获得这些待定系数，使目标函数的值为最小。

编写函数：

```
>> f=inline('a(1).*exp(-a(2).*x)+a(3).*exp(-a(4).*x).*sin(a(5).*x)','a','x');
```

得出待定系数向量:

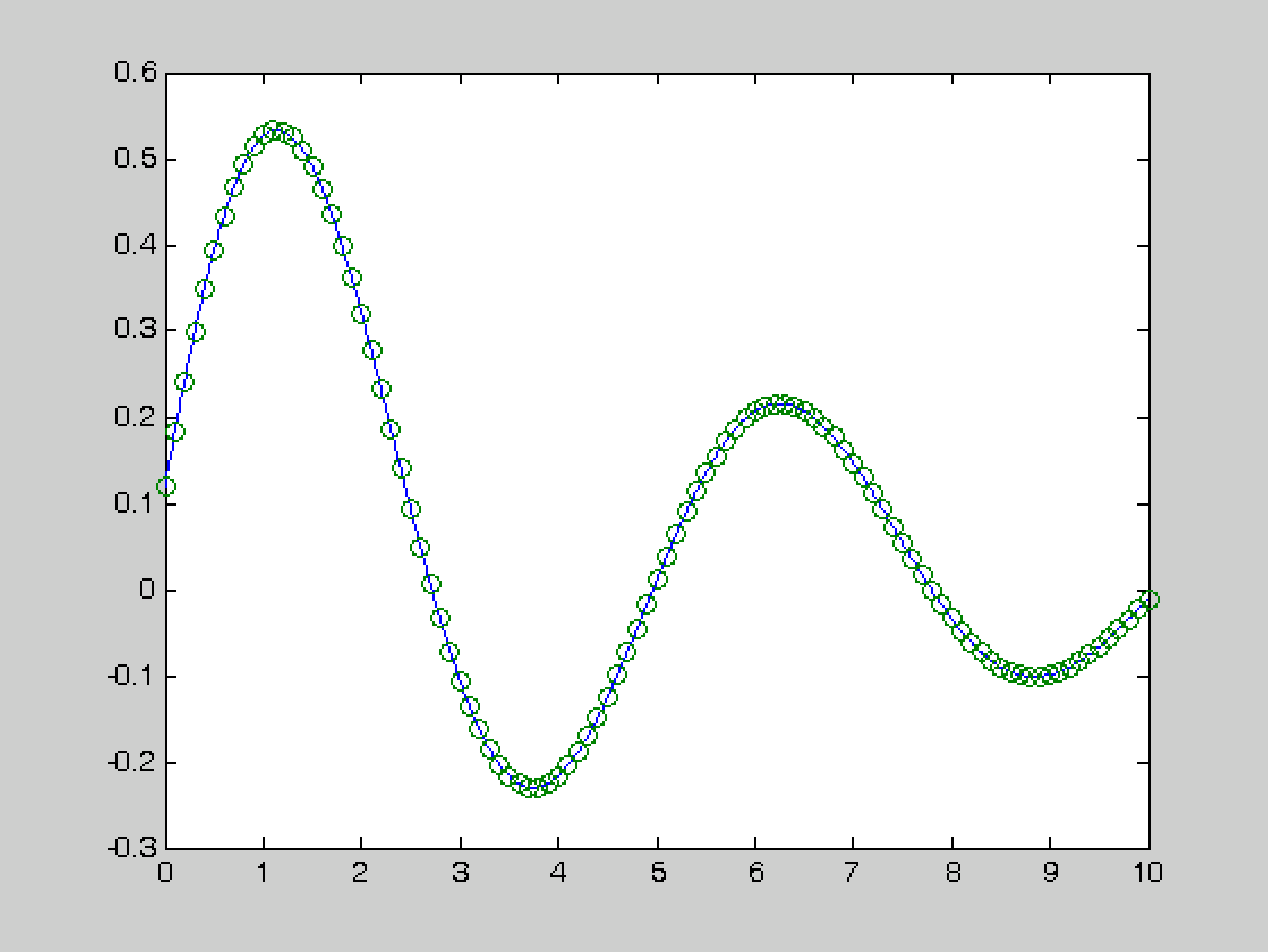
```
>> [xx,res]=lsqcurvefit(f,[1,1,1,1,1],x,y);xx',res
Optimization terminated successfully:
 Relative function value changing by less than OPTIONS.TolFun
ans =
    0.1197
    0.2125
    0.5404
    0.1702
    1.2300
res =
  7.1637e-007
```

修改最优化选项：

```
>> ff=optimset; ff.TolFun=1e-20; ff.TolX=1e-15; % 修改精度限制
[xx,res]=lsqcurvefit(f,[1,1,1,1,1],x,y,[],[],ff); xx',res % []变量界
Optimization terminated successfully:
 Relative function value changing by less than OPTIONS.TolFun
ans =
    0.1200
    0.2130
    0.5400
    0.1700
    1.2300
res =
  9.5035e-021
```

- 绘制曲线：

```
>> x1=0:0.01:10; y1=f(xx,x1); plot(x1,y1,x,y,'o')
```



例 已知数据可能满足 $y(x) = ax + bx^2 e^{-cx} + d$,

x_i	0.1	0.2	0.3	0.4	0.5
y_i	2.3201	2.6470	2.9707	3.2885	3.6008

x_i	0.6	0.7	0.8	0.9	1
y_i	3.9090	4.2147	4.5191	4.8232	5.1275

求满足数据的最小二乘解 a, b, c, d 的值

输入已知的参数:

```
>>  x=0.1:0.1:1;
>> y=[2.3201,2.6470,2.9707,3.2885,3.6008,3.9090,4.2147,4.5191,4.8232,5.1275];
```

令 $a_1 = a, a_2 = b, a_3 = c, a_4 = d$

则原型函数可以写成: $y(x) = a_1 x + a_2 x^2 \mathrm{e}^{-a_3 x} + a_4$

编写函数：

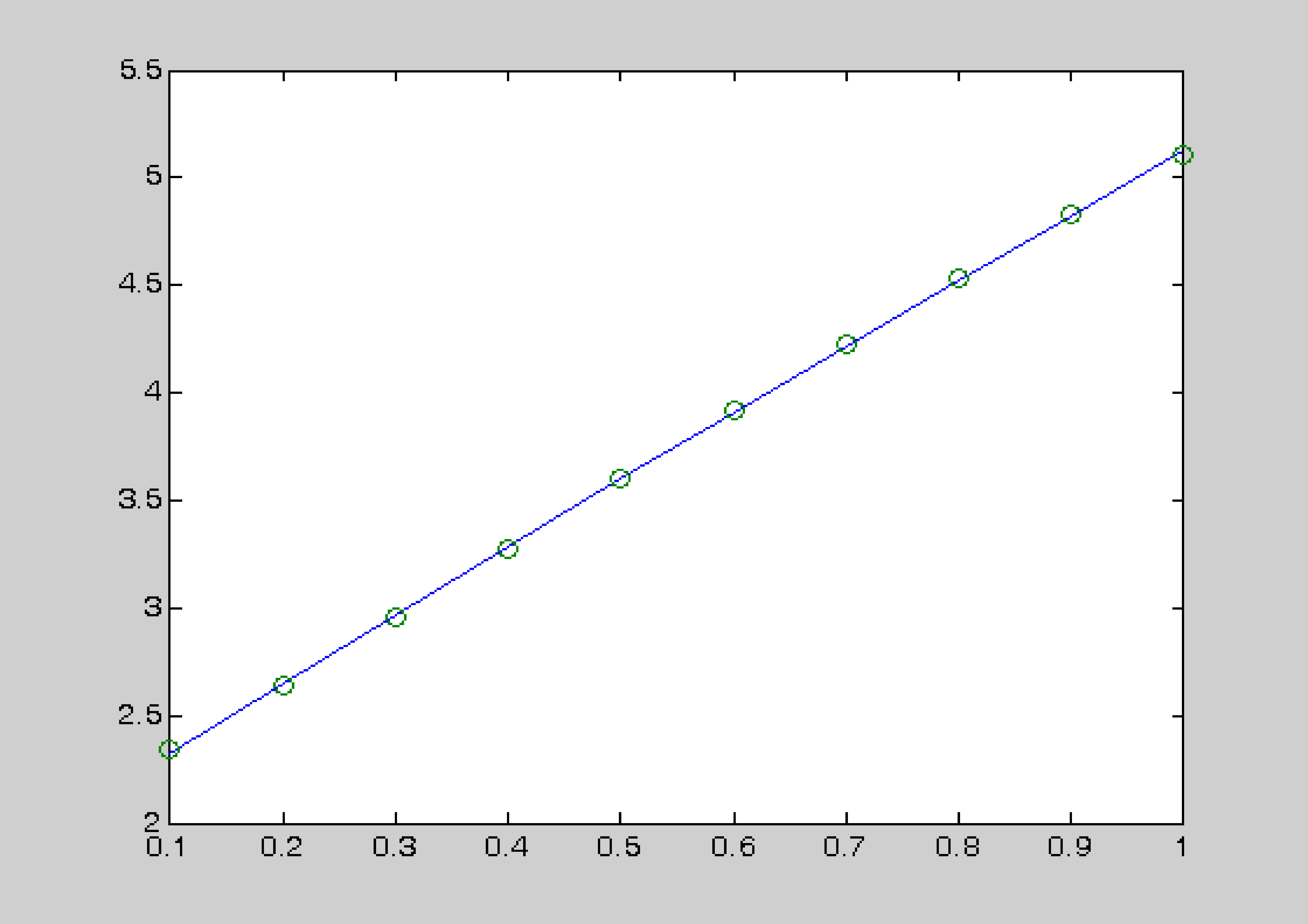
```
function y=c8f3(a,x)
y=a(1)*x+a(2)*x.^2.*exp(-a(3)*x)+a(4);
```

求解：

```
>>  a=lsqcurvefit('c8f3',[1;2;2;3],x,y); a'
Maximum number of function evaluations exceeded;
  increase options.MaxFunEvals
ans =
   2.4575    2.4557    1.4437    2.0720
```

- 绘制曲线：

```
>> y1=c8f3(a,x); plot(x,y,x,y1,'o')
```



4. B样条函数及其MATLAB表示

建立函数：

- 格式

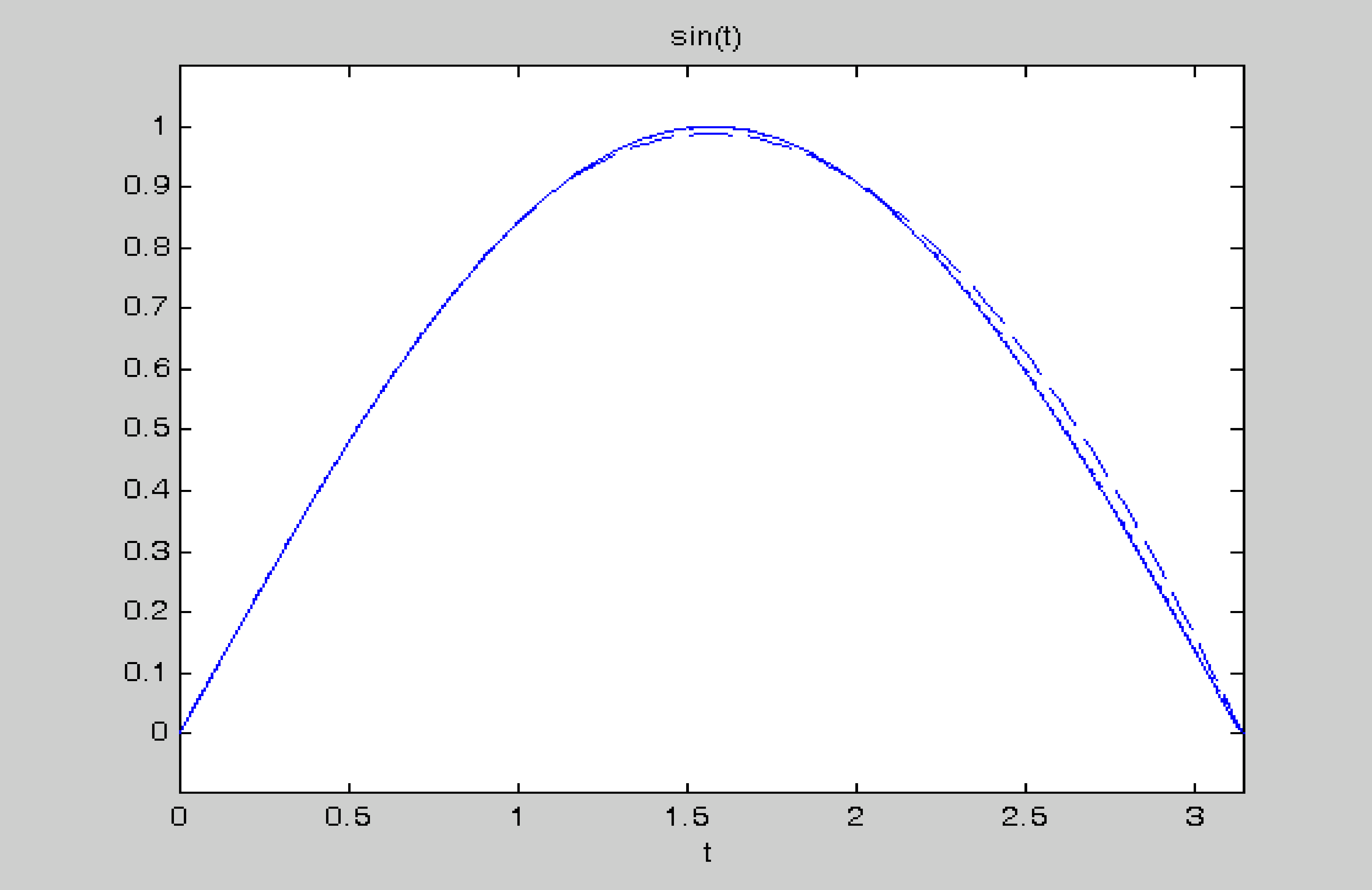
  S=spapi(k,x,y)

其中 k 为用户选定的 B 样条阶次， 一般选择 k=4,5

例　分别用 B 样条函数对 y=sin(t)和

$$f(x) = (x^2 - 3x + 5)e^{-5x}\sin x$$ 中给出的数据

进行 5 次 B 样条函数拟合，并与三次分段多项式样条函数拟合的结果相比较




```
x0=[0,0.4,1,2,pi];
y0=sin(x0);
ezplot('sin(t)',[0,pi]);
hold on
sp1=csapi(x0,y0)
fnplt(sp1,'r--');       % 三次分段多项式样条插值
sp2=spapi(5,x0,y0); fnplt(sp2,'b:')   % 5 次 B 样条插值
```

```
>> x=0:.12:1; y=(x.^2-3*x+5).*exp(-5*x).*sin(x);
>> ezplot('(x^2-3*x+5)*exp(-5*x)*sin(x)',[0,1]), hold on
>> sp1=csapi(x,y); fnplt(sp1,'--'); sp2=spapi(5,x,y);
   fnplt(sp2,'r:')
```