第五章　曲线拟合

- 最小二乘拟合曲线
- 曲线拟合
- 样条函数插值

5.1最小二乘拟合曲线

- 给定数据对$(x_1,y_1),\ldots,(x_N,y_N)$，寻找函数$y=f(x)$；

  设$y=f(x)=Ax+B$;

  同时 $f(x_k)=y_k+e_k$

  e_k：测量误差

- 目标：寻找“最佳”逼近表达式。

- 误差指标
- 最大误差： $$E_\infty(f)=\max_{1\le k\le N}\{|f(x_k)-y_k|\}$$
- 平均误差： $$E_1(f)=\frac{1}{\mathrm{N}}\sum_{k=1}^{N}|f(x_k)-y_k|$$
- 均方根误差： $$E_2(f)=\left(\frac{1}{\mathrm{N}}\sum_{k=1}^{N}|f(x_k)-y_k|^2\right)^{1/2}$$

- 定理（最小二乘拟合曲线）设N个点 $\{(x_k, y_k)\}_{k=1}^{N}$ 横坐标 $\{x_k\}_{k=1}^{N}$ 是确定的，最小二乘拟合曲线 y=Ax+B的系数是下列线性方程组的解，方程称为正规方程：

$$\mathrm{A}\sum_{i=1}^{N} x_i^2 + B\sum_{i=1}^{N} x_i = \sum_{\mathrm{i}=1}^{\mathrm{N}} y_i x_i$$

$$\mathrm{A}\sum_{i=1}^{N} x_i + BN = \sum_{\mathrm{i}=1}^{\mathrm{N}} y_i$$

标准是

$$\min E_2(f) = \left(\frac{1}{\mathrm{N}} \sum_{k=1}^{N} \left| Ax_k + B - y_k \right|^2 \right)^{1/2}$$

证明：

$$E=\sum_{i=1}^{N}(\mathrm{A}x_i+B-y_i)^2$$

$$\frac{\partial E}{\partial A}=0, \quad 2\sum_{i=1}^{N}(\mathrm{A}x_i+B-y_i)x_i=0$$

$$\frac{\partial E}{\partial B}=0, \quad 2\sum_{i=1}^{N}(\mathrm{A}x_i+B-y_i)=0$$

$$\sum_{i=1}^{N}(\mathrm{A}x_i+B-y_i)x_i=0 \Rightarrow \mathrm{A}\sum_{i=1}^{N}x_i^2+B\sum_{i=1}^{N}x_i=\sum_{\mathrm{i}=1}^{\mathrm{N}}y_ix_i$$

$$\sum_{i=1}^{N}(\mathrm{A}x_i+B-y_i)=0 \Rightarrow \mathrm{A}\sum_{i=1}^{N}x_i+BN=\sum_{\mathrm{i}=1}^{\mathrm{N}}y_i$$

最小二乘法

```
function [A,B]=lsline(x,y)
%x:1*N; y:1*N
xmean=mean(x);
ymean=mean(y);
sumx2=(x-xmean)*(x-xmean)';
sumxy=(y-ymean)*(x-xmean)';
A=sumxy/sumx2;
B=ymean-A*xmean;
```

- 幂函数拟合y=Ax^M, M为已知常数

系数

$$A=\left(\sum_{k=1}^{N} x_k^M y_k\right) \Big/ \left(\sum_{k=1}^{N} x_k^{2M}\right)$$

- 5.2曲线拟合
- y=Ce^{Ax}的线性化拟合

取对数，有ln(y)=Ax+ln(C)；

令Y=ln(y)，B=ln(C);

则Y=AX+B;由 $\{(x_k, y_k)\}_{k=1}^{N}$，变换为

$$(X_k, Y_k) = (x_k, \ln(y_k))$$

由最小二乘求得A,B；则C=e^{B}

- $y=Ce^{Ax}$的非线性最小二乘法

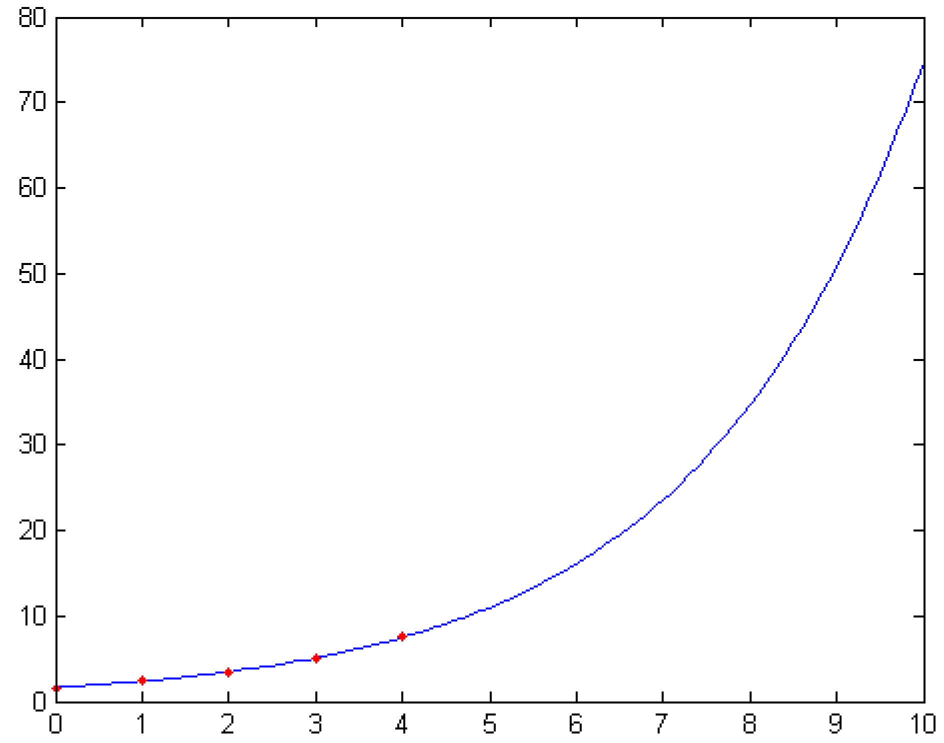
给定点 $\{(x_k, y_k)\}_{k=1}^{N}$

最小化 $$E(A,C)=\sum_{i=1}^{N}\left(Ce^{Ax_i}-y_i\right)^2$$

利用 $\begin{cases}\dfrac{\partial E}{\partial A}=0\\[2ex]\dfrac{\partial E}{\partial C}=0\end{cases}$ ，类似可求得正规方程。

- 例 根据5个数据点(0.1.5)、(1,2.5)、(2,3.5)、(3,5.0)、(4,7.5)，利用最小二乘法求指数拟合

$y=Ce^{Ax}$



1

- exm5_5.m

数据线性化变量替换

函数	线性变换形式	变量与常数的变化
$y=\frac{A}{x}+B$	$y=A\frac{1}{x}+B$	$X=\frac{1}{x}, Y=y$
$y=Ce^{Ax}$	$\ln y=Ax+\ln C$	$X=x, Y=\ln y$
p187	…	…

线性最小二乘法

已知某函数的线性组合为：

$$f(x) = c_1 f_1(x) + c_2 f_2(x) + c_3 f_3(x) + \cdots + c_M f_M(x)$$

其中 $f_1(x), f_2(x), \cdots, f_M(x)$ 为M个已知线性独立函数

$c_1, c_2, \cdots, c_M$ 为待定系数。

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

求解最小误差平方和：

$$E(c_1, c_2, \cdots, c_M) = \sum_{k=1}^{N} (f(x) - y_k)^2$$

令：$\partial E / \partial c_i = 0, i = 1, 2, \cdots, M$

$$\sum_{k=1}^{N}\left(\left(\sum_{j=1}^{M} c_j f_j(x_k)\right) - y_k\right)\left(f_i(x_k)\right) = 0, i = 1, \cdots, M$$

- 得到 $M \times M$ 线性方程组

$$\sum_{j=1}^{M}\left(\sum_{k=1}^{N} f_i(x_k) f_j(x_k)\right) c_j = \sum_{k=1}^{N} f_i(x_k) y_k, i = 1, \cdots, M$$

矩阵形式 $$\sum_{j=1}^{M}\left(\sum_{k=1}^{N} f_i(x_k) f_j(x_k)\right) c_j = \sum_{k=1}^{N} f_i(x_k) y_k , i = 1, \cdots, M$$

$$F = \begin{bmatrix} f_1(x_1) & f_2(x_1) & \cdots & f_M(x_1) \\ f_1(x_2) & f_2(x_2) & \cdots & f_M(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \\ f_1(x_N) & f_2(x_N) & & f_M(x_N) \end{bmatrix}, y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ \\ y_N \end{bmatrix}$$

$$C = [c_1, c_2, \cdots, c_M]^T$$

有 $F^T F C = F^T y$ 令 $A = F^T F, B = F^T y$

当M很小时，由线性方程组有效求解C $$AC = B$$

例：假设测出一组（x_i,y_i),已知函数原型为

$$y(x)=c_1+c_2e^{-3x}+c_3\cos(-2x)e^{-4x}+c_4x^2$$

用已知数据求出待定系数c_i的值。

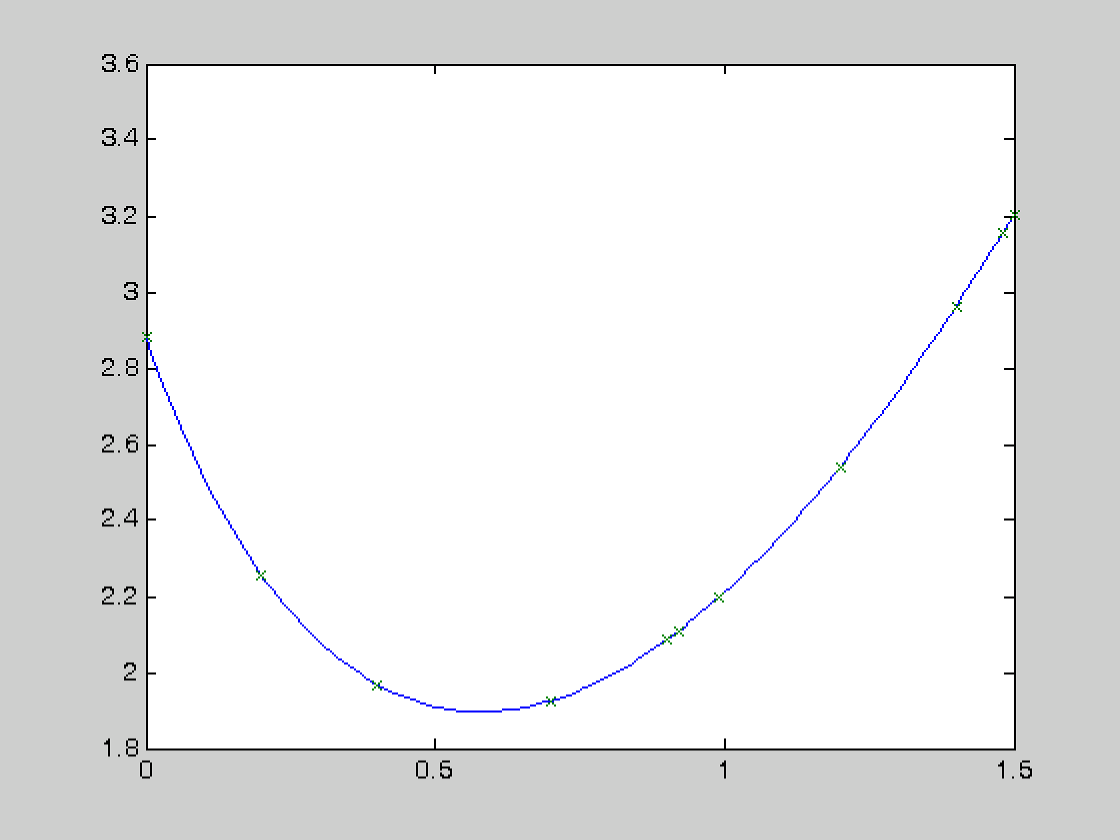
x_i	0	0.2	0.4	0.7	0.9	0.92	0.99	1.2	1.4	1.48	1.5
y_i	2.88	2.2576	1.9683	1.9258	2.0862	2.109	2.1979	2.5409	2.9627	3.155	3.2052

```
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];
F=[ones(size(x))， exp(-3*x), cos(-2*x).*exp(-4*x)， x.^2];
A=F'*F;  B=F'*y;c=uptrbk(A,B); c1=c'
c1 =1.2200    2.3397   -0.6797    0.8700
```

linear_ls.m

图形显示

```
x0=[0:0.01:1.5]';
F1=[ones(size(x0)) exp(-3*x0), cos(-2*x0).*exp(-4*x0) x0.^2];
y1=F1*c;
plot(x0,y1,x,y,'x')
```



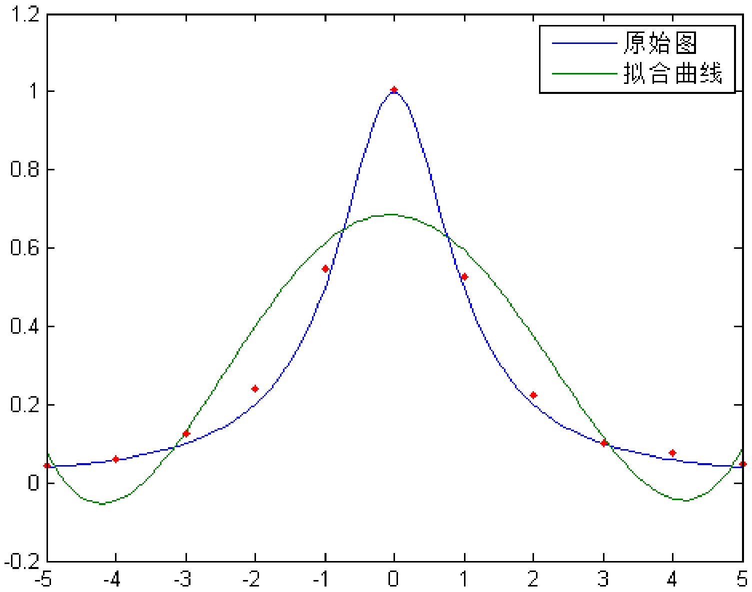
- 多项式拟合
- 取函数集合 $\left\{f_j(x)=x^{j-1}\right\}, j=1,\cdots,M+1$

  函数 $f(x)=c_1+c_2x+\cdots+c_{M+1}x^M$

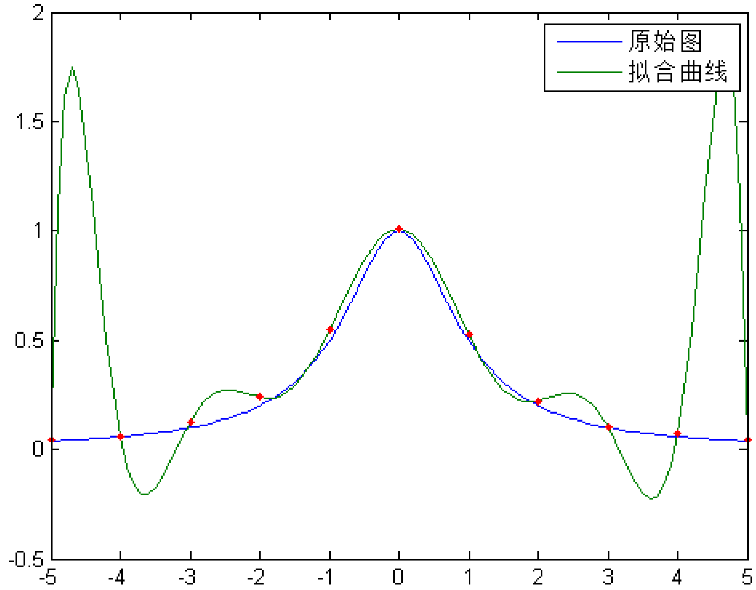
```
function C = lspoly(X,Y,M)
    n=length(X);
    B=zeros(1:M+1);
    F=zeros(n,M+1);
    for k=1:M+1
        F(:,k)=X'.^(k-1);
    end
    A=F'*F;
    B=F'*Y';
   C=A\B;
   C=flipud(C);
```

- 例：对如下函数进行多项式拟合

$$f(x)=\frac{1}{1+x^2}$$



2



3

- （ls_poly.m）

多项式摆动（Runge现象）

- 问题的提出：根据区间[a,b]上给出的节点做插值多项式p(x)的近似值，一般总认为p(x)的次数越高则逼近f(x)的精度就越好，但事实并非如此；若数据不具有多项式特性时，求出的曲线可能产生大的振荡；称为多项式摆动
- 为解决Rung问题，引入分段插值；

- 5.3样条函数插值
- 设N+1个点$\{(x_k,y_k)\}_{k=0}^{N}$，将图形分段，每段为一个低阶多项式$S_k(x)$，分别位于区间$[x_k,x_{k+1}]$、$[x_{k+1},x_{k+2}]$，在共同结点$[x_{k+1},y_{k+1}]$处连接，函数集合$\{S_k(x)\}$形成一个分段多项式曲线：S(x)。

- 分段线性插值

$$S_k(x)=y_k+d_k(x-x_k)$$

- 斜率 $d_k=(y_{k+1}-y_k)/(x_{k+1}-x_k)$
- 线性样条函数：

$$S(x)=\begin{cases} y_0+d_0(x-x_0) & x\in[x_0,x_1] \\ \vdots & \\ y_k+d_k(x-x_k) & x\in[x_k,x_{k+1}] \\ \vdots & \\ y_{N-1}+d_{N-1}(x-x_{N-1}) & x\in[x_{N-1},x_N] \end{cases}$$

- 得到一系列折线段

- 分段三次样条曲线
- 定义：设N+1个点$\{(x_k, y_k)\}_{k=0}^{N}$，$a = x_0 < \cdots < x_N = b$

  如果存在N个三次多项式$S_k(x)$，满足：
- 1.$S(x) = S_k(x) = s_{k,0} + s_{k,1}(x - x_k) + s_{k,2}(x - x_k)^2 + s_{k,3}(x - x_k)^3$

  $$x \in [x_k, x_{k+1}], k = 0, \ldots, N-1$$
- 2. $S_k(x_k) = y_k, k = 0, \ldots, N$ 插值
- 3. $S_k(x_{k+1}) = S_{k+1}(x_{k+1}), k = 0, \ldots, N-2$ 节点连续
- 4. $S_k'(x_{k+1}) = S_{k+1}'(x_{k+1}), k = 0, \ldots, N-2$ 光滑连续
- 5. $S_k''(x_{k+1}) = S_{k+1}''(x_{k+1}), k = 0, \ldots, N-2$ 二阶导数连续

- 存在性
- 4N个系数（自由度），约束条件N+1+3(N-1) =4N-2；剩余两个端点约束。
- $S(x)$ 为分段三次多项式，二阶导数在区间$[x_0,x_N]$是分段线性，可作拉格朗日线性插值

$$S_k''(x) = S''(x_k)\frac{x - x_{k+1}}{x_k - x_{k+1}} + S''(x_{k+1})\frac{x - x_k}{x_{k+1} - x_k}$$

记

$$m_k = S''(x_k), m_{k+1} = S''(x_{k+1}), h_k = x_{k+1} - x_k$$

$$x \in [x_k, x_{k+1}], k = 0, \ldots, N-1$$

$$S_k''(x)=\frac{m_k}{h_k}(x_{k+1}-x)+\frac{m_{k+1}}{h_k}(x-x_k)$$

积分两次，并引入两个积分常数

$$S_k(x)=\frac{m_k}{6h_k}(x_{k+1}-x)^3+\frac{m_{k+1}}{6h_k}(x-x_k)^3+p_k(x_{k+1}-x)+q_k(x-x_k)$$

- 由条件2、3 $y_k=S_k(x_k)$ ； $y_{k+1}=S_k(x_{k+1})$ 得到

$$y_k=\frac{m_k}{6}h_k^2+p_k h_k \quad ; \quad y_{k+1}=\frac{m_{k+1}}{6}h_k^2+q_k h_k$$

- 求解 p_k,q_k 得到三次多项式

$$S_k(x)=\frac{m_k}{6h_k}(x_{k+1}-x)^3+\frac{m_{k+1}}{6h_k}(x-x_k)^3$$

$$+\left(\frac{y_k}{h_k}-\frac{m_k h_k}{6}\right)(x_{k+1}-x)+\left(\frac{y_{k+1}}{h_k}-\frac{m_{k+1}h_k}{6}\right)(x-x_k)$$

- 简化为只包含未知系数$\{m_k\}, k=0,\ldots,N$，计算一阶导数

$$S_k'(x) = -\frac{m_k}{2h_k}(x_{k+1}-x)^2 + \frac{m_{k+1}}{2h_k}(x-x_k)^2 - \left(\frac{y_k}{h_k} - \frac{m_k h_k}{6}\right) + \left(\frac{y_{k+1}}{h_k} - \frac{m_{k+1}h_k}{6}\right)$$

- 则 $S_k'(x_k) = -\frac{m_k h_k}{3} - \frac{m_{k+1}h_k}{6} + d_k$ $S_{k-1}'(x_k) = \frac{m_k h_{k-1}}{3} + \frac{m_{k-1}h_{k-1}}{6} + d_{k-1};$

$$d_k = \frac{y_{k+1}-y_k}{h_k},$$

- 利用条件4，得到

$$h_{k-1}m_{k-1} + 2(h_{k-1}+h_k)m_k + h_k m_{k+1} = u_k$$

$$u_k = 6(d_k - d_{k-1}), k = 1,\ldots,N-1$$

- 端点约束
- 未知数为 $\{m_k\}, k=0,\ldots,N$，方程组有**N-1**个，通过指定端点$\mathbf{m_0}$、$\mathbf{m_N}$，可求得全部未知值。
- 1.[紧压样条]已知 $S'(x_0), S'(x_N)$

由 $S_k'(x) = -\frac{m_k h_k}{3} - \frac{m_{k+1} h_k}{6} + d_k$，可得到

$$m_0 = \frac{3}{h_0}(d_0 - S'(x_0)) - \frac{m_1}{2}$$

由 $S_{k-1}'(x) = \frac{m_k h_{k-1}}{3} + \frac{m_{k-1} h_{k-1}}{6} + d_{k-1}$; 可得到

$$m_N = \frac{3}{h_{N-1}}(S'(x_N) - d_{N-1}) - \frac{m_{N-1}}{2}$$

- .[紧压样条]的线性方程组

$$\left(\frac{3}{2}h_0+2h_1\right)m_1+h_1m_2=u_1-3\left(d_0-S'(x_0)\right)$$

$$h_{k-1}m_{k-1}+2(h_{k-1}+h_k)m_k+h_km_{k+1}=u_k$$

$$k=2,\ldots,N-2$$

$$h_{N-2}m_{N-2}+(2h_{N-2}+\frac{3}{2}h_{N-1})m_{N-1}=u_{N-1}-3(S'(x_N)-d_{N-1})$$

- 2.natural三次样条

$$m_0 = 0; m_N = 0$$

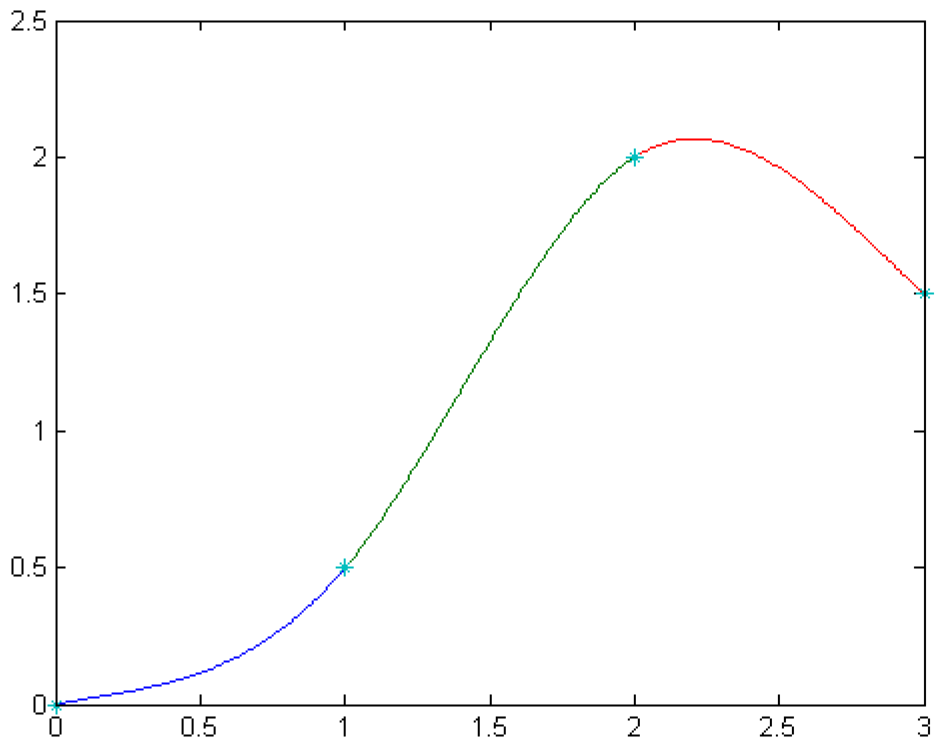
- 3.外推样条

$$m_0 = m_1 - \frac{h_0(m_2 - m_1)}{h_1} \qquad m_N = m_{N-1} - \frac{h_{N-1}(m_{N-1} - m_{N-2})}{h_{N-2}}$$

- 4.抛物线终结样条，在区间[x_0,x_1]内 $S'''(x) \equiv 0$ [x_{N-1},x_N]内 $S'''(x) \equiv 0$ $\quad m_0 = \mathrm{m}_1; m_N = m_{N-1}$

- 例5.12经过点[0 1 2 3； 0 0.5 2.0 1.5]，且一阶导数S’(0)=0.2,S’(3)=-1的三次紧压样条曲线

exm5_12.m



- 作业
- P181 #1(a) 5.1.3   第一题 (a)
- P183 #10   5.1.3   第10题
- P192 #1(a)   5.2.8   第一题 (a)

- 曲线拟合Matlab实现

- Polyfit
- Csapi
- spline

- 曲线拟合
- polyfit命令

Polyfit 多项式数据拟合.

- polyfit(X,Y,N) 寻找阶为N的数据拟合多项式P(X) 系数，用最小二乘法。
- [P,S] = POLYFIT(X,Y,N) returns the polynomial coefficients P and a structure S for use with POLYVAL to obtain error estimates on predictions.

P 为长度N+1 多项式系数，以幂降序排列

P(1)*X^N + P(2)*X^(N-1) +...+ P(N)*X + P(N+1).

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。

POLYVAL

- POLYVAL 计算多项式值.
- Y = POLYVAL(P,X), when P is a vector of length N+1 whose elements are the coefficients of a polynomial, is the value of the polynomial evaluated at X.
- Y = P(1)*X^N + P(2)*X^(N-1) + ... + P(N)*X + P(N+1)
- If X is a matrix or vector, the polynomial is evaluated at all points in X. See also polyvalm for evaluation in a matrix sense.

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：　polyfit.m

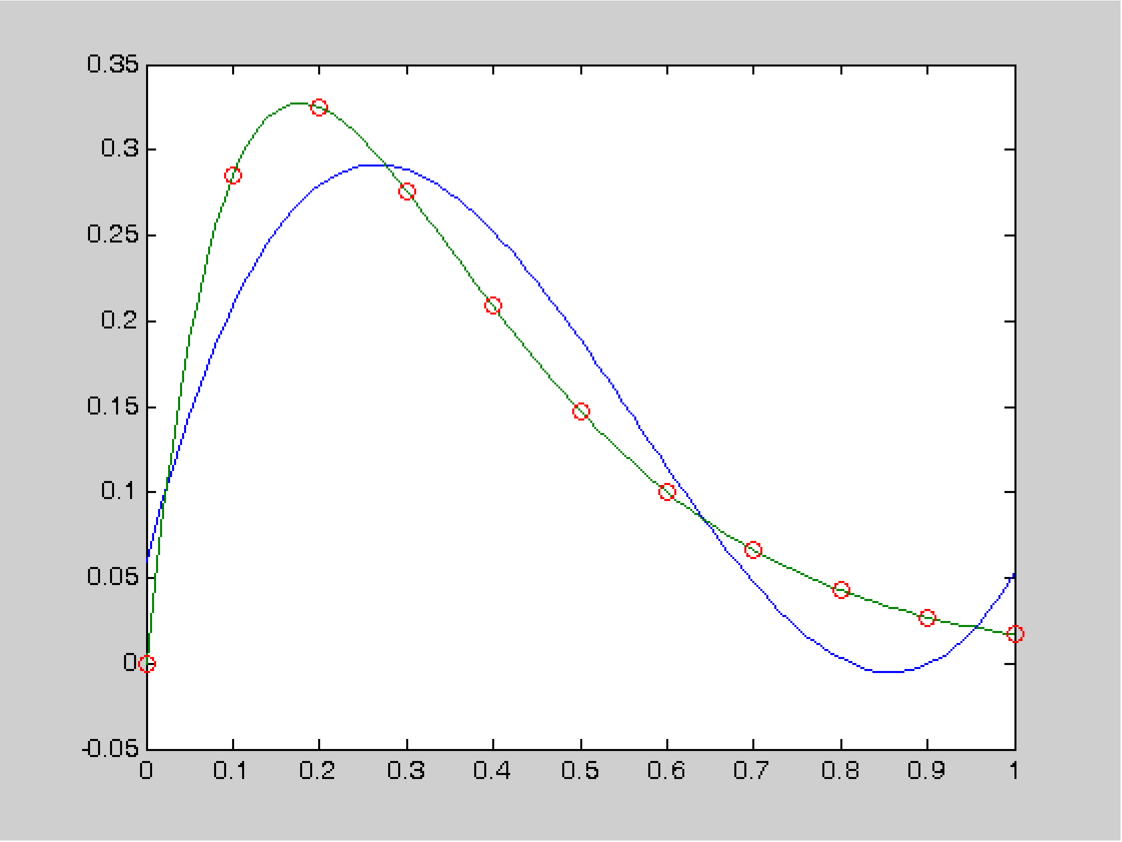
```
x0=0:.1:1; y0=(x0.^2-3*x0+5).*exp(-5*x0).*sin(x0);
p3=polyfit(x0,y0,3); vpa(poly2sym(p3),10)
% 多项式显示如下
ans =
2.839962923*x^3-4.789842696*x^2+1.943211631*x+0.05975248921
```

■ 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')



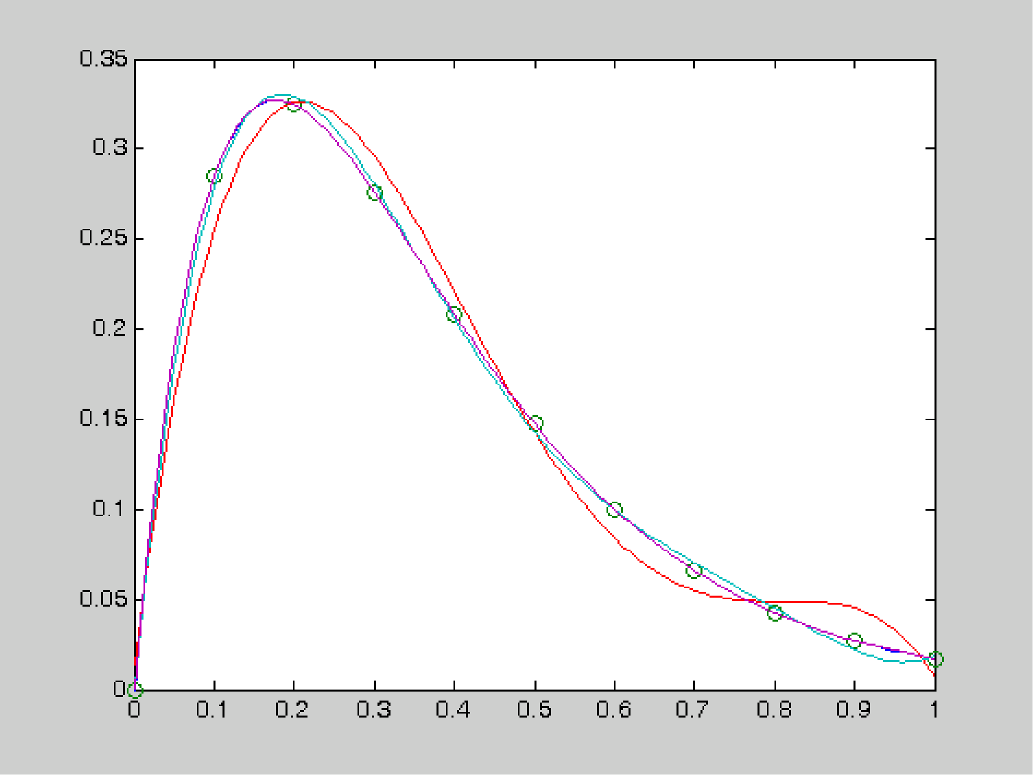
- 不同的次数进行拟合：

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)



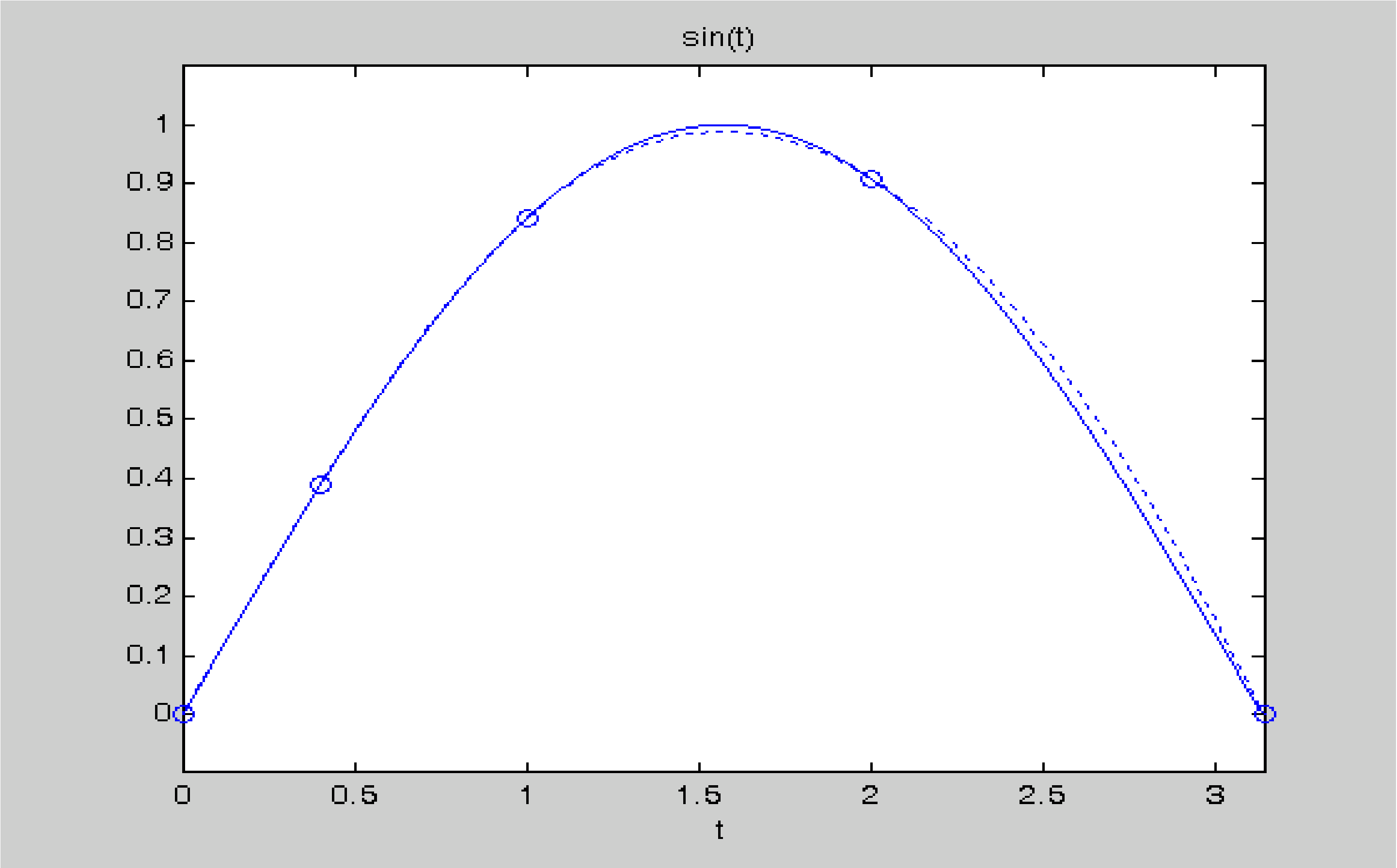
- 三次样条插值
- csapi Cubic spline interpolant with not-a-knot end condition（抛物线终结）
- PP = csapi(X,Y) returns the cubic spline interpolant (in ppform) to the given data (X,Y) using the not-a-knot end conditions.

- 例： x0=[0,0.4,1,2,pi]; y0=sin(x0); csapi.m

```
sp=csapi(x0,y0), fnplt(sp,':'); hold on,
sp =
    form: 'pp'
    breaks: [0 0.4000 1 2 3.1416]
    coefs: [4x4 double]
    pieces: 4
    order: 4
    dim: 1
ezplot('sin(t)',[0,pi]); plot(x0,y0,'o')
sp.coefs
ans =
  -0.1627    0.0076    0.9965         0
  -0.1627   -0.1876    0.9245    0.3894
   0.0244   -0.4804    0.5238    0.8415
   0.0244   -0.4071   -0.3637    0.9093
```

在(0.4000, 1)区间内，插值多项式可以表示为:

$$S_2(x) = -0.1627(x-0.4)^3 - 0.1876(x-0.4)^2 + 0.9245(x-0.4) + 0.3894$$



- csape　以各种端点条件做三次样条插值.
- 格式：　PP　= csape(X,Y,CONDS)

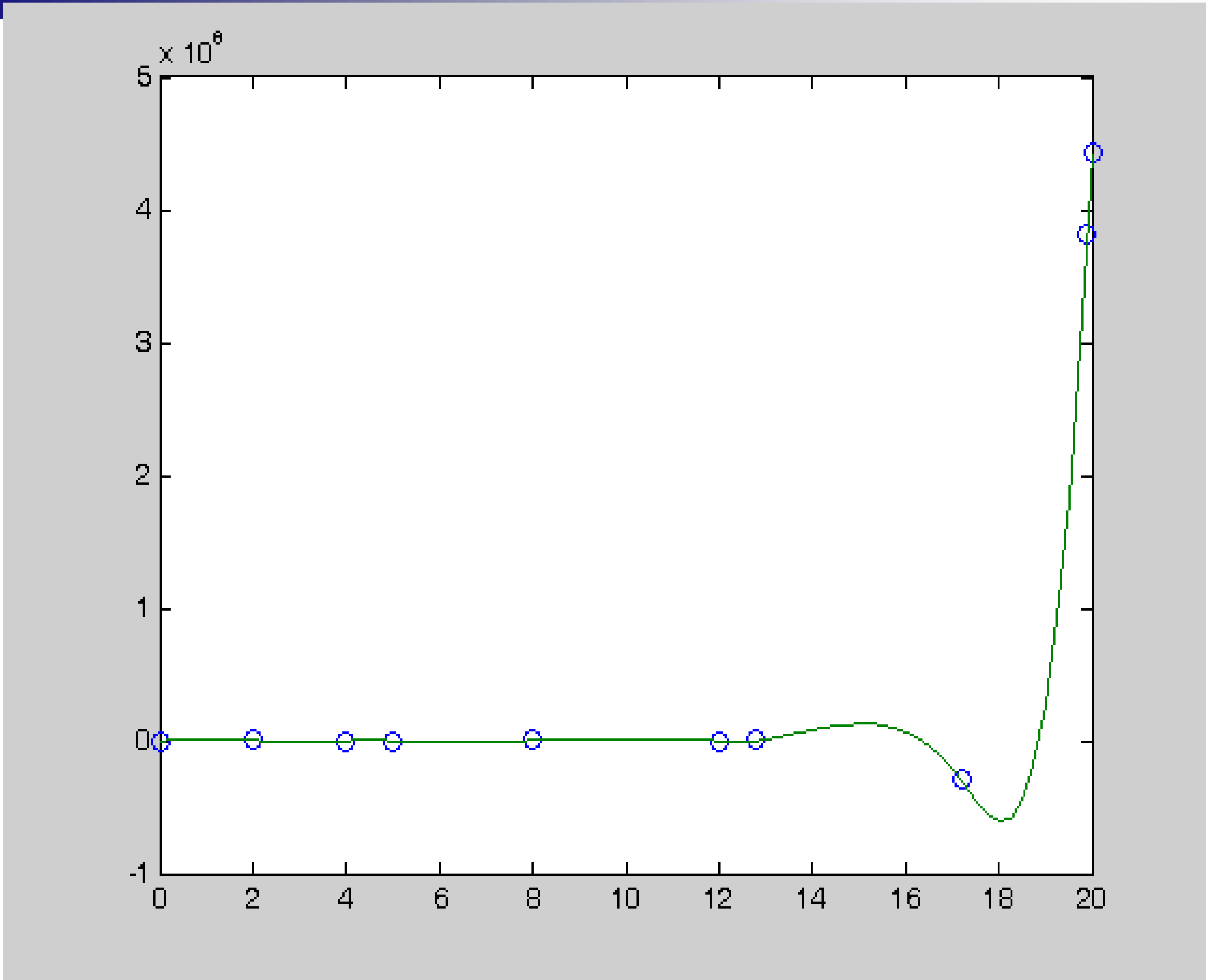
- 'complete'　: match endslopes (as given, with
- default as under *default*)
- 'not-a-knot'　: make spline C^3 across first and last interior
- break (ignoring given end condition values if any)
- 'periodic'　: match first and second derivatives at first data
- point with those at last data point
- (ignoring given end condition values if any)
- 'second'　: match end second derivatives (as given,
- with default [0 0], i.e., as in variational)
- 'variational' : set end second derivatives equal to zero
- (ignoring given end condition values if any)

函数spline

- **功能　三次样条数据插值**
- **格式　yy = spline(x,y,xx)**
- **(Ordinarily, the not-a-knot end conditions are used.)**
- **例**:对离散分布在y=exp(x)sin(x)函数曲线上的数据点进行样条插值计算:

yspline.m

```
x = [0 2 4 5 8 12 12.8 17.2 19.9 20];
y = exp(x).*sin(x);
xx = 0:.25:20;
yy = spline(x,y,xx);
plot(x,y,'o',xx,yy)
```



x 10^6
5
4
3
2
1
0
-1
0
2
4
6
8
10
12
14
16
18
20