

数学建模与数学实验

插值

应用数学系

陈六新



引言

背景

函数的数值逼近是一个很广泛的领域。(数值分析中研究历史最长的学科之一)。



图像处理

图像插值是数字图像处理领域的一项重要任务，是图像缩放、旋转、几何矫正等图像操作的基础。

当被逼近的信息
是离散数据时

本次主要介绍一维插值、二位插值；其中包括代数多项式插值和样条函数插值，双线性插值等。

目的

1. 了解插值的基本内容。
2. 掌握用数学软件包求解插值问题。

内容

- [1] 一维插值
- [2] 二维插值
- [3] 插值案例

一 维插值

一、插值的定义

二、插值的方法

拉格朗日插值

分段线性插值

三次样条插值

三、用Matlab解插值问题

插值法

引言

一、多项式插值问题的定义

设给出 $f(x) \in [a, b]$ 的一系列函数值表

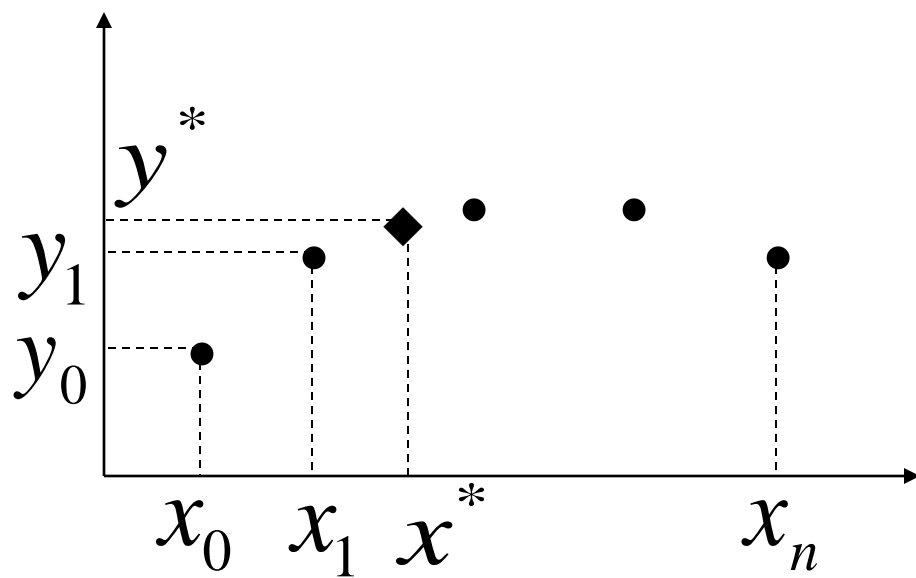
x	x_0	x_1	x_2	\dots	x_n
y	y_0	y_1	y_2	\dots	y_n

其中 $a \leq x_0 < x_1 < \dots < x_n \leq b$

计算 $f(\tilde{x})$?

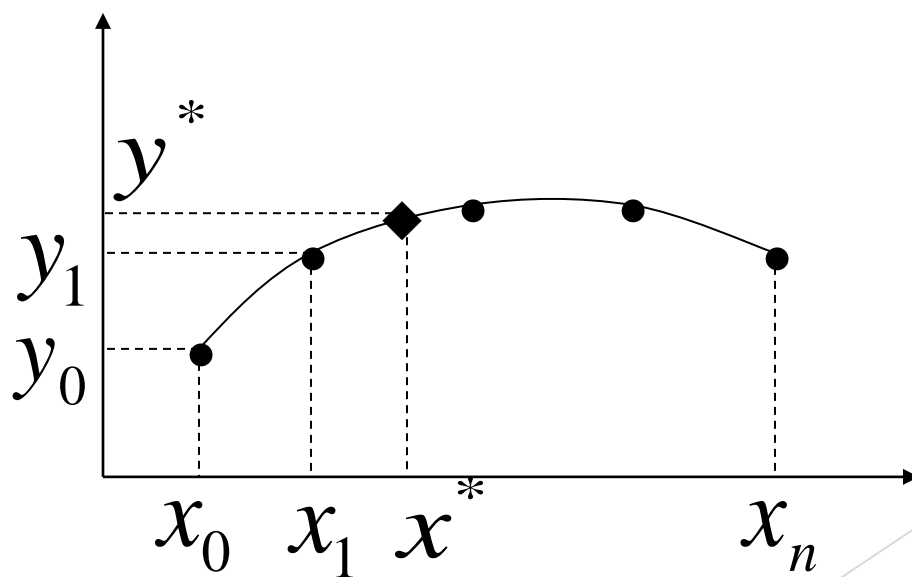
插值法

已知 $n + 1$ 个节点 (x_j, y_j) $j = 1, 2, \dots, n$, 其中 x_j 互不相同, 不妨设 $a = x_1 < x_2 < \dots < x_n = b$, 求任一插值点 x^* ($\neq x_j$) 处的插值 y^* .



节点可视为由 $y = g(x)$ 产生, $g(x)$ 的表达式复杂, 或无封闭形式, 或未知。

构造一个(相对简单的)函数 $y = f(x)$,通过全部节点,即 $y_j = f(x_j), j = 0, 1, 2, \dots, n$,再用 $y = f(x)$ 计算插值,即 $y^* = f(x^*)$ 。



返回

求 $f(x)$ 的插值函数的几何意义

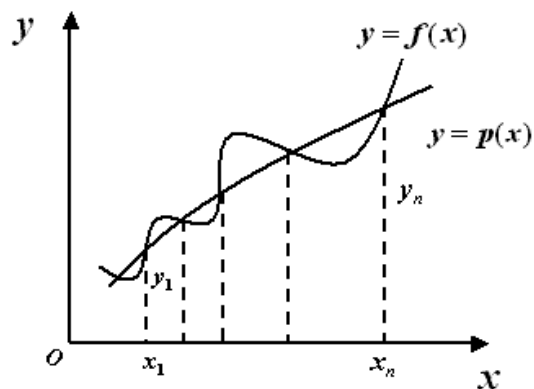


图 2-1

定义 1 对于已知满足 $y_i = f(x_i) (i = 0, 1, \dots, n)$ (1)

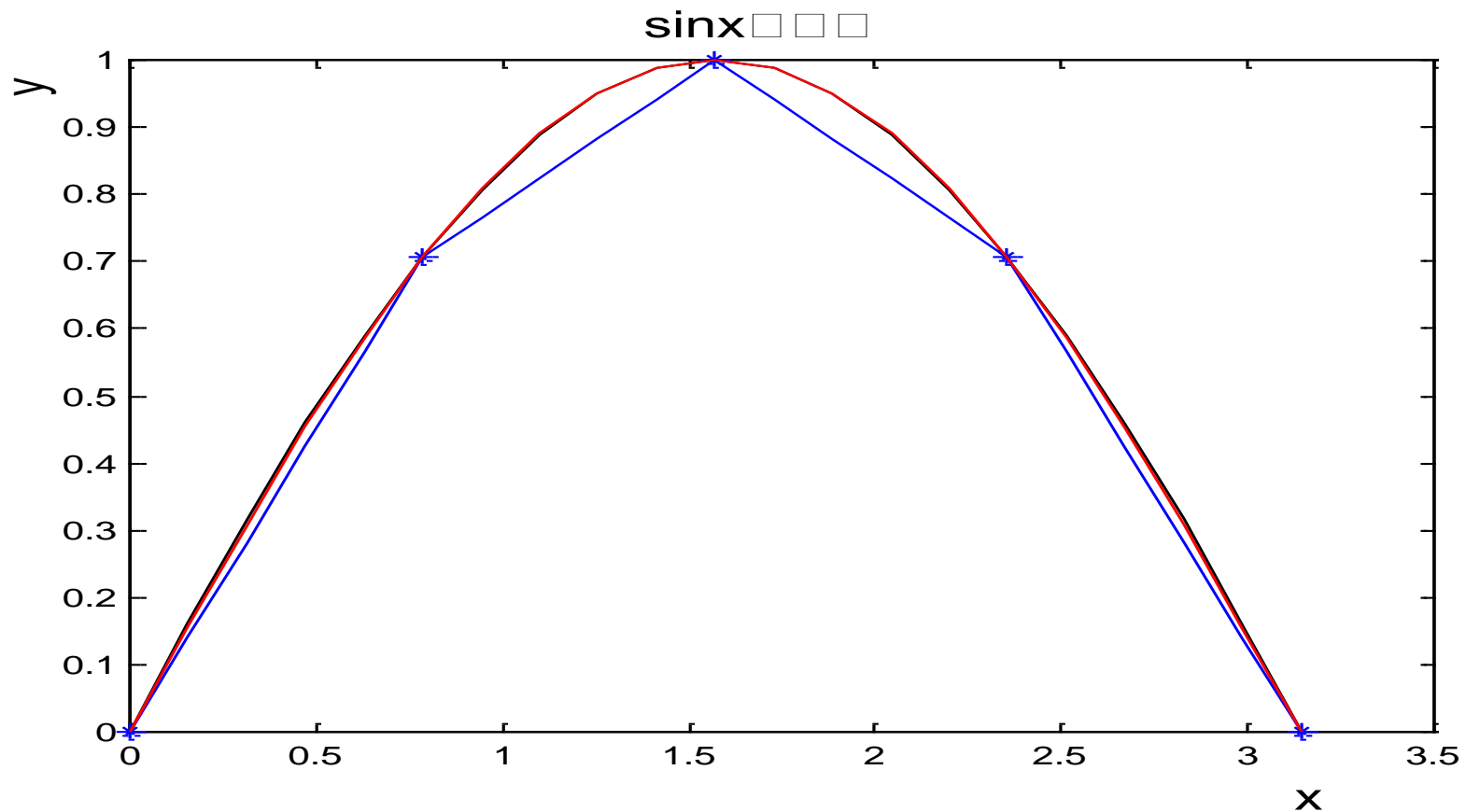
的函数 $y = f(x)$, 设若存在一个次数不超过 n 次的多项式

$p_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$ (其中 a_i 为实数) 满足条件

$$\underline{p_n(x_i) = y_i} \quad (i = 0, 1, \dots, n) \quad (2)$$

则称 $p_n(x)$ 为函数 $f(x)$ 的 n 次插值多项式.

插值
条件



如函数 $y = \sin x$, 若给定 $[0, \pi]$ 上5个等分点

其插值函数的图像如图

多项式插
值常用

在区间 $[a, b]$ 上, 根据函数表构造一个 **次数不超过 n** 的代数多项式

$$p_n(x) = a_0 + a_1x + a_2x^2 + \cdots + a_nx^n,$$

使
$$p_n(x_i) = f(x_i) = y_i \quad (i = 0, 1, \cdots, n)$$

插值多项式的 **插值余项** 或截断误差

$$R_n(x) = f(x) - p_n(x) \quad (3)$$

定理 1 满足插值条件 (2) 的 n 次插值多项式是 **存在且唯一** 的.

二、插值多项式的误差估计

定理 2 设 $f^{(n)}(x) \in C[a, b]$ ，任意 $x \in (a, b)$ ， $f^{(n+1)}(x)$ 存在， x_0, x_1, \dots, x_n 为 $n+1$ 个互异插值节点， $p_n(x)$ 为 $f(x)$ 在 $[a, b]$ 上的 n 次插值多项式，则对任意 $x \in [a, b]$ 有 **余项**

$$R_n(x) = f(x) - p_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \omega_{n+1}(x), \quad \xi \in (a, b) \quad (4)$$

其中 $\omega_{n+1}(x) = (x - x_0)(x - x_1) \cdots (x - x_n) = \prod_{i=0}^n (x - x_i)$.

误差
估计

插值法

2 拉格朗日多项式插值

一、线性插值

已知

x	x_0	x_1
$f(x)$	y_0	y_1

线性
插值

求作一次多项式 $L_1(x) = a_0 + a_1x$ ，使它满足条件： $L_1(x_0) = y_0$ ， $L_1(x_1) = y_1$ 。

将已知条件代入，得线性方程组

$$\begin{cases} y_0 = a_0 + a_1x_0 \\ y_1 = a_0 + a_1x_1 \end{cases}$$

$$L_1(x) = y_0 + \frac{y_1 - y_0}{x_1 - x_0}(x - x_0) \quad (\text{点斜式})$$

$$= \frac{x - x_1}{x_0 - x_1} y_0 + \frac{x - x_0}{x_1 - x_0} y_1 \quad (\text{两点式})$$

记
$$l_0(x) = \frac{x - x_1}{x_0 - x_1}, \quad l_1(x) = \frac{x - x_0}{x_1 - x_0}$$

线性插
值基函数

可将求得的一次插值多项式改写为

$$L_1(x) = \underline{l_0(x)y_0 + l_1(x)y_1}$$

二、抛物线插值

已知

x	x_0	x_1	x_2
$f(x)$	y_0	y_1	y_2

求二次多项式 $L_2(x) = a_0 + a_1x + a_2x^2$ 满足 $L(x_i) = y_i$ ($i = 0, 1, 2$).

解 设 $L_2(x) = l_0(x)y_0 + l_1(x)y_1 + l_2(x)y_2$,

其中 $l_i(x)$ ($i = 0, 1, 2$) 为二次插值基函数.

插值基函数的性质

$$l_i(x_j) = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases} \quad (i, j = 0, 1, 2) \quad (9)$$

求出 $l_i(x)$ ($i = 0, 1, 2$) 后得

$$\begin{aligned} L_2(x) &= y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \\ &= \sum_{j=0}^2 y_j \prod_{i=0}^2 \left(\frac{x-x_i}{x_j-x_i} \right) \quad (i \neq j) \end{aligned} \quad (10)$$

例 已知 $\sqrt{1} = 1$, $\sqrt{4} = 2$, $\sqrt{9} = 3$, 用二次拉格朗日插值公式求 $\sqrt{5}$ 的近似值.

解 取 $x_0 = 1, x_1 = 4, x_2 = 9$, 则 $y_0 = 1, y_1 = 2, y_2 = 3$

所以 $\sqrt{5} \approx L_2(5)$

$$= 1 \times \frac{(5-4)(5-9)}{(1-4)(1-9)} + 2 \times \frac{(5-1)(5-9)}{(4-1)(4-9)} + 3 \times \frac{(5-1)(5-4)}{(9-1)(9-4)} = 2.267$$

注:与准确值 $\sqrt{5} = 2.23606 \cdots$ 比较, 计算结果具有两位有效数字.

三、拉格朗日插值

定义 称 n 次多项式 $l_0(x), l_1(x), l_2(x), \dots, l_n(x)$ 为在 $n+1$ 个节点 x_i ($i = 0, 1, \dots, n$) 上的 n 次插值基函数, 其中

$$l_k(x) = \frac{(x-x_0) \cdots (x-x_{k-1})(x-x_{k+1}) \cdots (x-x_n)}{(x_k-x_0) \cdots (x_k-x_{k-1})(x_k-x_{k+1}) \cdots (x_k-x_n)} = \prod_{\substack{j=0 \\ j \neq k}}^n \frac{x-x_j}{x_k-x_j}$$

对于给定的 $n+1$ 个互异插值节点 x_0, x_1, \dots, x_n ,

$$L_n(x) = \sum_{k=0}^n y_k l_k(x) \quad (13)$$

称为插值问题的 n 次多项式插值函数.

简称为拉格朗日插值多项式.

$$\omega_{n+1}(x) = \prod_{j=0}^n (x - x_j) \implies L_n(x) = \sum_{i=0}^n f(x_i) \frac{\omega_{n+1}(x)}{(x - x_i) \omega'_{n+1}(x_i)}$$

Lagrange多项式插值方法小结

拉格朗日插值公式具有形式对称，便于编程计算的特点。

缺点

1. 插值基函数计算复杂；
节点变化时需重算。

2. 高次插值的精度不一定高。

$$\begin{aligned} l_k(x) &= \prod_{j=0, j \neq k}^n \frac{x - x_j}{x_k - x_j} \\ &= \frac{\omega_{n+1}(x)}{(x - x_k)\omega'_{n+1}(x_k)} \quad (k = 0, 1, 2, \dots, n) \end{aligned}$$

例
$$g(x) = \frac{1}{1+25x^2}, \quad -1 \leq x \leq 1$$

采用拉格朗日多项式插值：选取不同插值节点个数 $n+1$ ，其中 n 为插值多项式的次数，当 n 分别取2,4,6,8,10时，绘出插值结果图形。

To Matlab
lch(larg1)

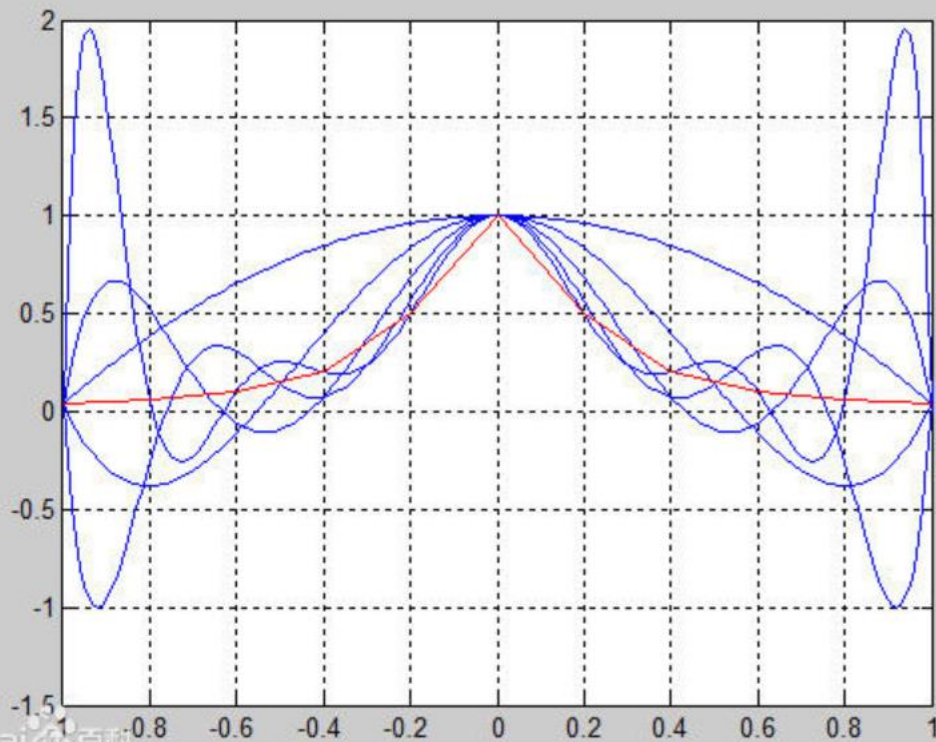
拉格朗日多项式插值的这种振荡现象叫 **Runge现象**。



返回

例 $g(x) = \frac{1}{1+25x^2}, -1 \leq x \leq 1$

拉格朗日多项式插值的
这种振荡现象叫 **Runge现象**。



下面是MATLAB中演示对
 $f(x)=1/(1+25x^2)$ 插值的代码
%演示龙格函数的插值情况
for i=3:2:11

x=linspace(-1,1,i);

y=1./(1+25*x.^2);

p=polyfit(x,y,i-1);

xx=-1:0.01:1;

yy=polyval(p,xx);

plot(xx,yy,'b');

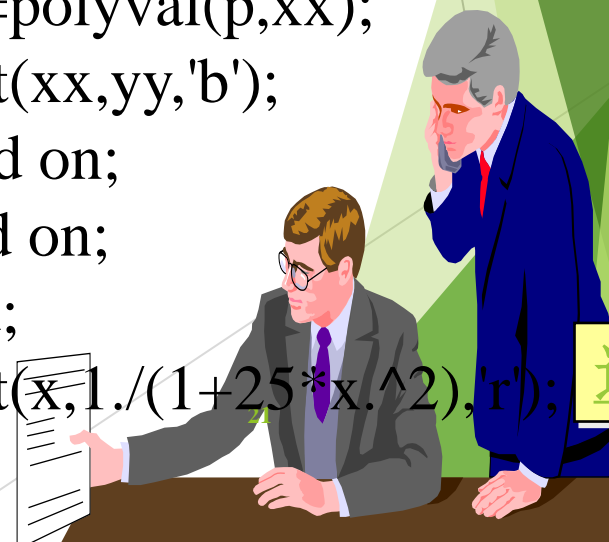
hold on;

grid on;

end;

plot(x,1./(1+25*x.^2),'r');

返回



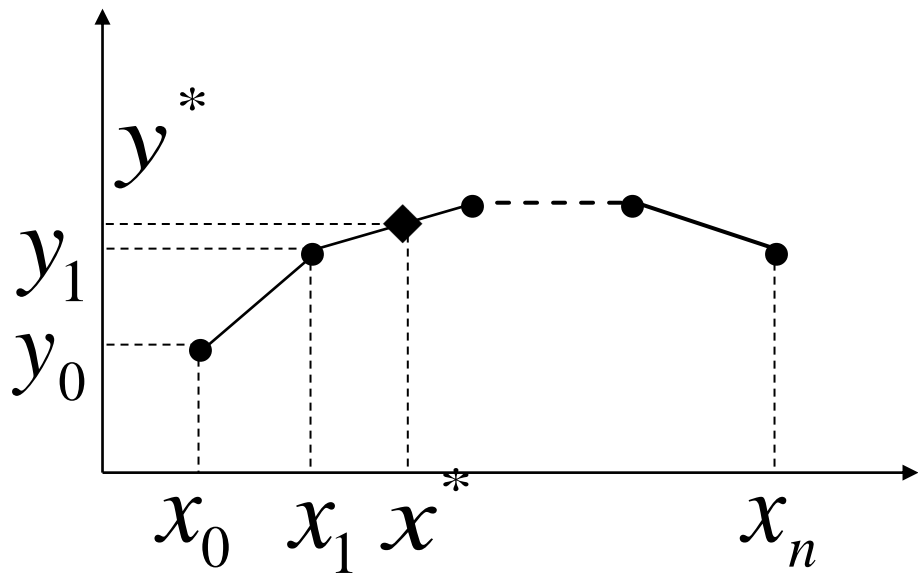
分段线性插值

将每两个相邻的节点用直线连接起来，如此形成的一条折线就是分段插值函数，记作 $I_n(x)$ 。它满足 $I_n(x_i) = y_i$ ，且 $I_n(x)$ 在每一个小区间 (x_i, x_{i+1}) $i = 0, 1, \dots, n-1$ 上是线性函数。

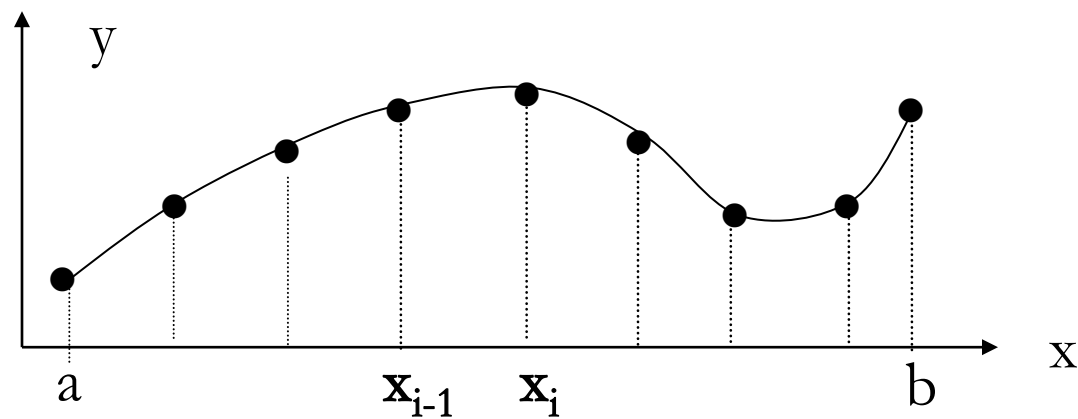
$I_n(x)$ 可以表示为 $I_n(x) = \sum_{i=0}^n l_i(x) * y_i$ ，其中

$$l_i(x) = \begin{cases} \frac{(x - x_{i-1})}{(x_i - x_{i-1})}, & x \in (x_{i-1}, x_i), i \neq 0 \\ \frac{(x - x_{i+1})}{(x_i - x_{i+1})}, & x \in (x_i, x_{i+1}), i \neq n \\ 0, & \text{orelse} \end{cases}$$

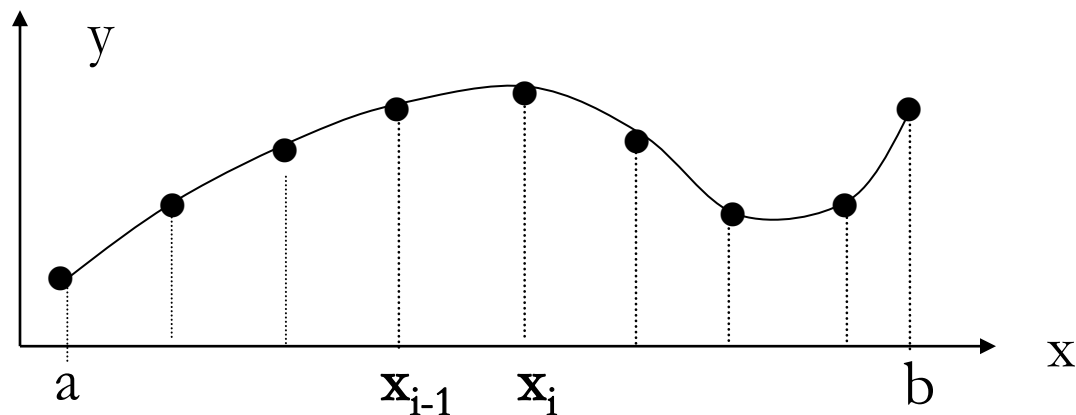
计算量与 n 无关; n 越大，误差越小.



样条插值



样条插值的概念



许多工程技术中提出的计算问题是对插值函数的光滑性有较高要求，如飞机机翼外形，内燃机的进、排气门的凸轮曲线，都要求曲线具有较高的光滑程度，不仅要连续，而且要有连续的曲率，这就导致了样条函数的产生。



样条是指飞机或轮船等的制造过程中为描绘出光滑的外形曲线(放样)所用的工具.

样条**本质上**是一段接一段的**多项式拼合而成**的曲线.在拼接处,不仅函数是连续的,而且一阶和二阶导数也是连续的.

定义 对区间 $[a,b]$ 的一个划分 $\Delta: a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b$, 函数

$S(x)$ 满足:

(1) $S(x)$ 在 $[x_i, x_{i+1}]$, $(i = 0, 1, \cdots, n-1)$ 上为 k 次多项式,

(2) 在区间 $[a,b]$ 上 $S(x)$ 及其 $1, 2, \cdots, k-1$ 阶导数连续, 则称 $S(x)$ 为区间 $[a,b]$ 上对应于划分 Δ 的 k 次多项式样条函数.

且若对于样条节点 x_i 上给定的函数值 $f(x_i) = y_i$ $(i = 0, 1, \cdots, n)$, 有

$S(x_i) = y_i$, $i = 0, 1, \cdots, n$, 则称 $S(x)$ 为 k 次样条插值函数.

例 已知函数 $S(x) = \begin{cases} x^3 + x^2, & 0 \leq x \leq 1 \\ 2x^3 + bx^2 + cx - 1, & 1 \leq x \leq 2 \end{cases}$

是以 0,1,2 为节点的**三次样条函数**，求系数 b, c 的值。

解 由定义知 $S(x) \in C^2[0,2]$ ，故 $S(x)$ 和 $S'(x)$ 分别在 $x_1 = 1$ 处连续

$$\begin{cases} 2 + b + c - 1 = 2 \\ 6 + 2b + c = 5 \end{cases} \Rightarrow b = -2, c = 3$$

$$\text{设 } S(x) = \begin{cases} S_1(x), & x \in [x_0, x_1] \\ S_2(x), & x \in [x_1, x_2] \\ \dots\dots\dots \\ S_n(x), & x \in [x_{n-1}, x_n] \end{cases}$$

其中

$$S_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i \quad (i = 1, \dots, n)$$

$S(x)$ 在 $[a, b]$ 上有 $4n$ 个待定参数.

$4n - 2$
约束

$$\text{条件} \begin{cases} S(x_i - 0) = S(x_i + 0) \\ S'(x_i - 0) = S'(x_i + 0) \\ S''(x_i - 0) = S''(x_i + 0) \end{cases} \quad (i = 1, 2, \dots, n-1) \text{ 给出了 } 3(n-1) \text{ 个约束};$$

插值条件 $S(x_i) = y_i \quad (i = 0, 1, 2, \dots, n)$ 给出了 $n+1$ 个约束.

为了确定 $S(x)$ ，在区间 $[a, b]$ 的端点处补充两个条件，称为边界条件。

常用的边界条件有下列 3 种。

① 给定端点的一阶导数值 $S'(x_0) = y'_0, \quad S'(x_n) = y'_n$;

② 给定端点的二阶导数值 $S''(x_0) = y''_0, \quad S''(x_n) = y''_n$;

特别地， $S''(x_0) = S''(x_n) = 0$ 的边界条件称为自然边界条件；

③ 周期边界条件 $S(x_0) = S(x_n), \quad S'(x_0) = S'(x_n), \quad S''(x_0) = S''(x_n)$ 。

二、三次样条插值函数的构造

记 $S'(x_j) = m_j$ ($j = 0, 1, \dots, n$)

则在区间 $[x_j, x_{j+1}]$ 上满足: ($j = 0, 1, \dots, n-1$)

$$S(x_j) = y_j, \quad S(x_{j+1}) = y_{j+1}; \quad S'(x_j) = m_j, \quad S'(x_{j+1}) = m_{j+1}$$

当 $x \in [x_j, x_{j+1}]$ 时, 有 $S(x) = S_j(x)$

$$S_j(x) = \frac{x - x_{j+1}}{x_j - x_{j+1}} y_j + \frac{x - x_j}{x_{j+1} - x_j} y_{j+1} + (px + q)(x - x_j)(x - x_{j+1})$$

($j = 0, 1, \dots, n-1$), 其中 p, q 为待定系数.

$$S_j(x) = \frac{x - x_{j+1}}{x_j - x_{j+1}} y_j + \frac{x - x_j}{x_{j+1} - x_j} y_{j+1} \\ + (px + q)(x - x_j)(x - x_{j+1})$$

$$S'_j(x) = \frac{1}{h_j} (y_{j+1} - y_j) + p(x - x_j)(x - x_{j+1}) + (px + q)(2x - x_j - x_{j+1})$$

因为 $S'_j(x_j) = m_j$, $S'_j(x_{j+1}) = m_{j+1}$

所以 $m_j = \frac{1}{h_j} (y_{j+1} - y_j) - (px_j + q)h_j$

$$m_{j+1} = \frac{1}{h_j} (y_{j+1} - y_j) + (px_{j+1} + q)h_j$$

$$\Rightarrow p, q$$

其中 $h_j = x_{j+1} - x_j$

于是

$$\begin{aligned} S_j(x) = & \left(1 + 2\frac{x-x_j}{h_j}\right)\left(\frac{x-x_{j+1}}{h_j}\right)^2 y_j + \left(1 - 2\frac{x-x_{j+1}}{h_j}\right)\left(\frac{x-x_j}{h_j}\right)^2 y_{j+1} \\ & + (x-x_j)\left(\frac{x-x_{j+1}}{h_j}\right)^2 m_j + (x-x_{j+1})\left(\frac{x-x_j}{h_j}\right)^2 m_{j+1} \quad (2.40) \end{aligned}$$

$$j = 0, 1, 2, \dots, n-1$$

由二阶导数在**内节点**处连续 $S_j''(x_j + 0) = S_{j-1}''(x_j - 0)$

可导出关于参数 m_j 的方程组

三转角方程

$$\beta_j m_{j-1} + 2m_j + \alpha_j m_{j+1} = d_j, \quad j = 1, 2, \dots, n-1 \quad (2.41)$$

其中 $\alpha_j = \frac{h_{j-1}}{h_{j-1} + h_j}, \quad \beta_j = 1 - \alpha_j = \frac{h_j}{h_{j-1} + h_j}$

$$d_j = 3 \left[\frac{\beta_j}{h_{j-1}} (y_j - y_{j-1}) + \frac{\alpha_j}{h_j} (y_{j+1} - y_j) \right]$$

边界条件 $m_0 = y'_0, \quad m_n = y'_n$

令 $S''(x_j) = y_j'' = M_j$, $j = 0, 1, \dots, n$.

当 $x \in [x_j, x_{j+1}]$ 时, 直线 $S''(x) = S_j''(x)$ 过 (x_j, M_j) 和 (x_{j+1}, M_{j+1}) 两点, 从而有

$$S''(x) = S_j''(x) = M_{j+1} \frac{x - x_j}{h_j} - M_j \frac{x - x_{j+1}}{h_j}, \text{ 其中 } h_j = x_{j+1} - x_j.$$

对上式连续积分两次

$$S'_j(x) = M_{j+1} \frac{(x - x_j)^2}{2h_j} - M_j \frac{(x - x_{j+1})^2}{2h_j} + c_{1j}$$

$$S_j(x) = M_{j+1} \frac{(x - x_j)^3}{6h_j} + M_j \frac{(x_{j+1} - x)^3}{6h_j} + c_{1j}(x - x_j) + c_{2j}, \quad j = 0, 1, \dots, n-1$$

$$\text{其中} \begin{cases} c_{1j} = \frac{y_{j+1} - y_j}{h_j} - \frac{h_j}{6}(M_{j+1} - M_j) \\ c_{2j} = y_j - \frac{h_j^2}{6}M_j \end{cases}$$

于是

$$\begin{aligned} S_j(x) &= M_{j+1} \frac{(x-x_j)^3}{6h_j} + M_j \frac{(x_{j+1}-x)^3}{6h_j} + \left[\frac{y_{j+1}-y_j}{h_j} - \frac{h_j}{6}(M_{j+1}-M_j) \right] (x-x_j) + \left(y_j - \frac{h_j^2}{6} M_j \right) \\ &= M_{j+1} \frac{(x-x_j)^3}{6h_j} + M_j \frac{(x_{j+1}-x)^3}{6h_j} + \left(y_{j+1} - \frac{M_{j+1}h_j^2}{6} \right) \frac{x-x_j}{h_j} - \left(y_j - \frac{M_jh_j^2}{6} \right) \frac{x-x_{j+1}}{h_j}, \quad j=0,1,\dots,n-1 \end{aligned}$$

对上式求导，并利用一阶导数的连续性： $S'_{j-1}(x_j-0) = S'_j(x_j+0)$ ， $j=1,\dots,n-1$ 得

$$-\frac{h_j}{2}M_j + \frac{y_{j+1}-y_j}{h_j} + \frac{h_j}{6}(M_j-M_{j+1}) = \frac{h_{j-1}}{2}M_j + \frac{y_j-y_{j-1}}{h_{j-1}} + \frac{h_{j-1}}{6}(M_{j-1}-M_j)$$

或

$$\frac{h_{j-1}}{6}M_{j-1} + \frac{h_j+h_{j-1}}{3}M_j + \frac{h_j}{6}M_{j+1} = \frac{y_{j+1}-y_j}{h_j} - \frac{y_j-y_{j-1}}{h_{j-1}}, \quad j=1,\dots,n-1$$

则可得方程组

$$\mu_j M_{j-1} + 2M_j + \lambda_j M_{j+1} = g_j, \quad j=1,2,\dots,n-1 \quad (2.44)$$

$$\mu_j M_{j-1} + 2M_j + \lambda_j M_{j+1} = g_j, \quad j=1,2,\cdots,n-1$$

三弯矩方程

$$\text{其中} \begin{cases} \mu_j = \frac{h_{j-1}}{h_j + h_{j-1}} \\ \lambda_j = 1 - \mu_j = \frac{h_j}{h_j + h_{j-1}} \\ g_j = \frac{6}{h_j + h_{j-1}} \left(\frac{y_{j+1} - y_j}{h_j} - \frac{y_j - y_{j-1}}{h_j} \right) \end{cases}$$

它含有 $n+1$ 个参数 M_0, \cdots, M_n ，利用前面任意一组边界条件即可得到参数。如

给定边界条件 $S'(x_0) = y'_0, \quad S'(x_n) = y'_n$ 得

$$\begin{cases} 2M_0 + M_1 = \frac{6}{h_0} \left(\frac{y_1 - y_0}{h_0} - y'_0 \right) \\ M_{n-1} + 2M_n = \frac{6}{h_{n-1}} \left(y'_n - \frac{y_n - y_{n-1}}{h_{n-1}} \right) \end{cases}$$

用MATLAB作插值计算

一维插值函数：

$y_i = \text{interp1}(x, y, x_i, \text{'method'})$

xi处的插
值结果

插值节点

被插值点

插值方法

'nearest' : 最邻近插值
'linear' : 线性插值;
'spline' : 三次样条插值;
'cubic' : 立方插值。
缺省时: 分段线性插值。

注意：所有的插值方法都要求x是单调的，并且xi不能够超过x的范围。

例：在1-12的11小时内，每隔1小时测量一次温度，测得的温度依次为：5, 8, 9, 15, 25, 29, 31, 30, 22, 25, 27, 24。试估计每隔1/10小时的温度值。

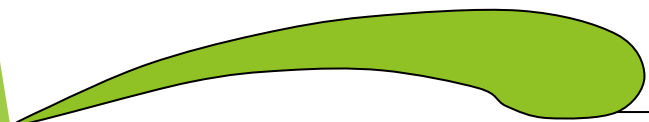
```
hours=1:12;  
temps=[5 8 9 15 25 29 31 30 22 25 27 24];  
h=1:0.1:12;  
t=interp1(hours,temps,h,'spline'); %(直接输出数据将是很多的)  
plot(hours,temps,'+',h,t,hours,temps,'r:') %作图  
xlabel('Hour'),ylabel('Degrees Celsius')
```

To MATLAB

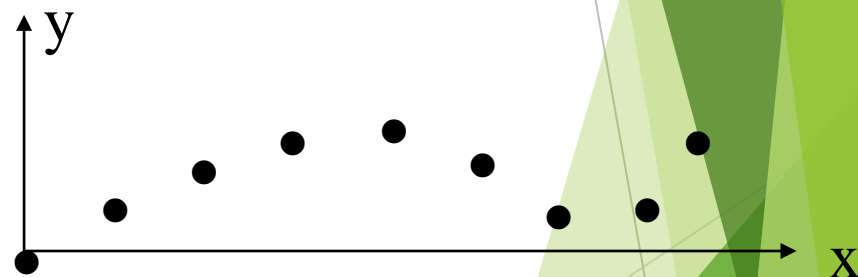
39 (temp)

例 已知飞机下轮廓线上数据如下，求 x 每改变0.1时的 y 值。

X	0	3	5	7	9	11	12	13	14	15
Y	0	1.2	1.7	2.0	2.1	2.0	1.8	1.2	1.0	1.6



机翼下
轮廓线



To MATLAB(plane)

返回

二维插值

一、二维插值定义

二、网格节点插值法

最邻近插值

分片线性插值

双线性插值

三、用Matlab解插值问题

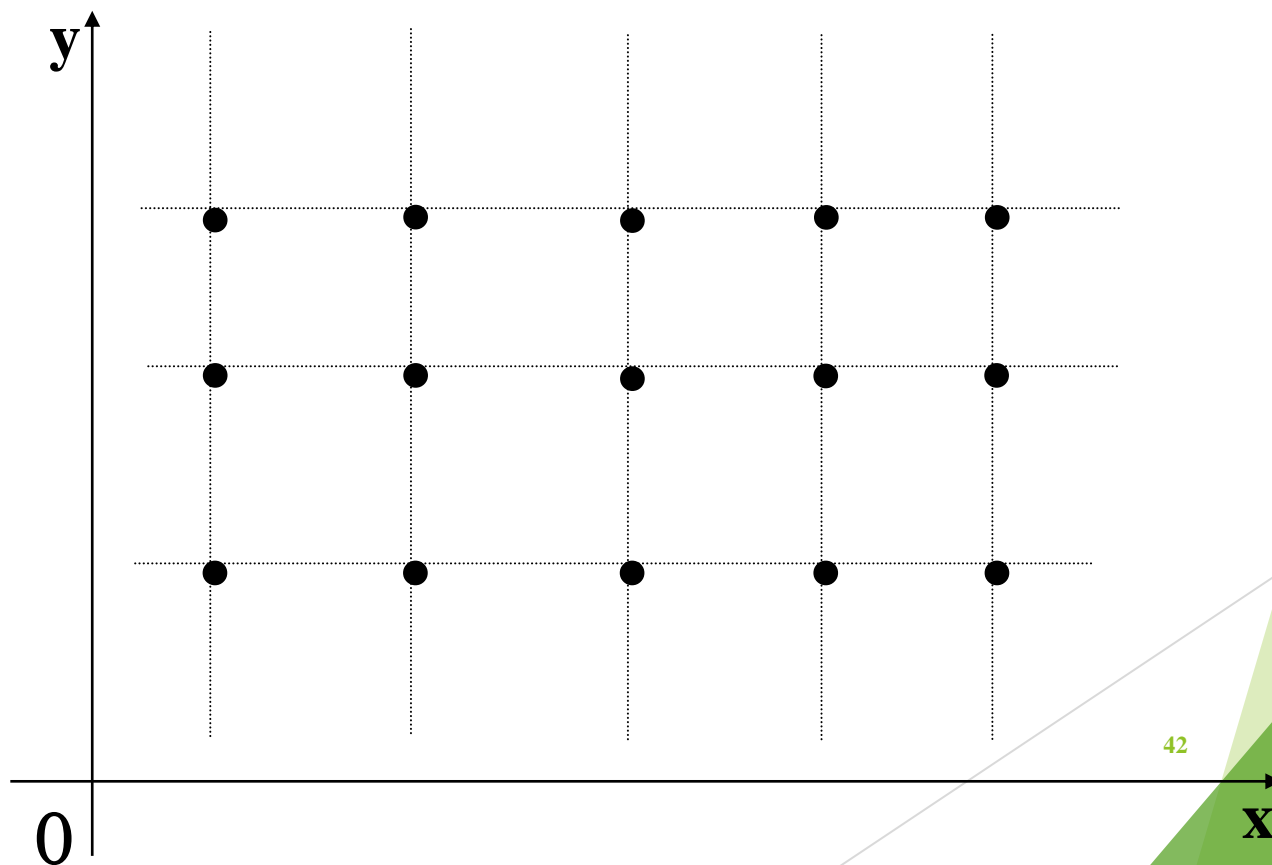
网格节点数据的插值

散点数据的插值

二维插值的定义

若节点是二维的，插值函数就是二元函数，即曲面。如在某个区域测量了若干个节点的高程(节点值)，为了画出较精确的等高线，首先要先插入更多的节点，计算这些点的高程(插值)。

第一种 (网格节点)：



已知 $m \times n$ 个节点

$$(x_i, y_j, z_{ij}) \quad (i = 1, 2, \dots, m; j = 1, 2, \dots, n)$$

其中 x_i, y_j 互不相同, 不妨设

$$a = x_1 < x_2 < \dots < x_m = b$$

$$c = y_1 < y_2 < \dots < y_n = d$$

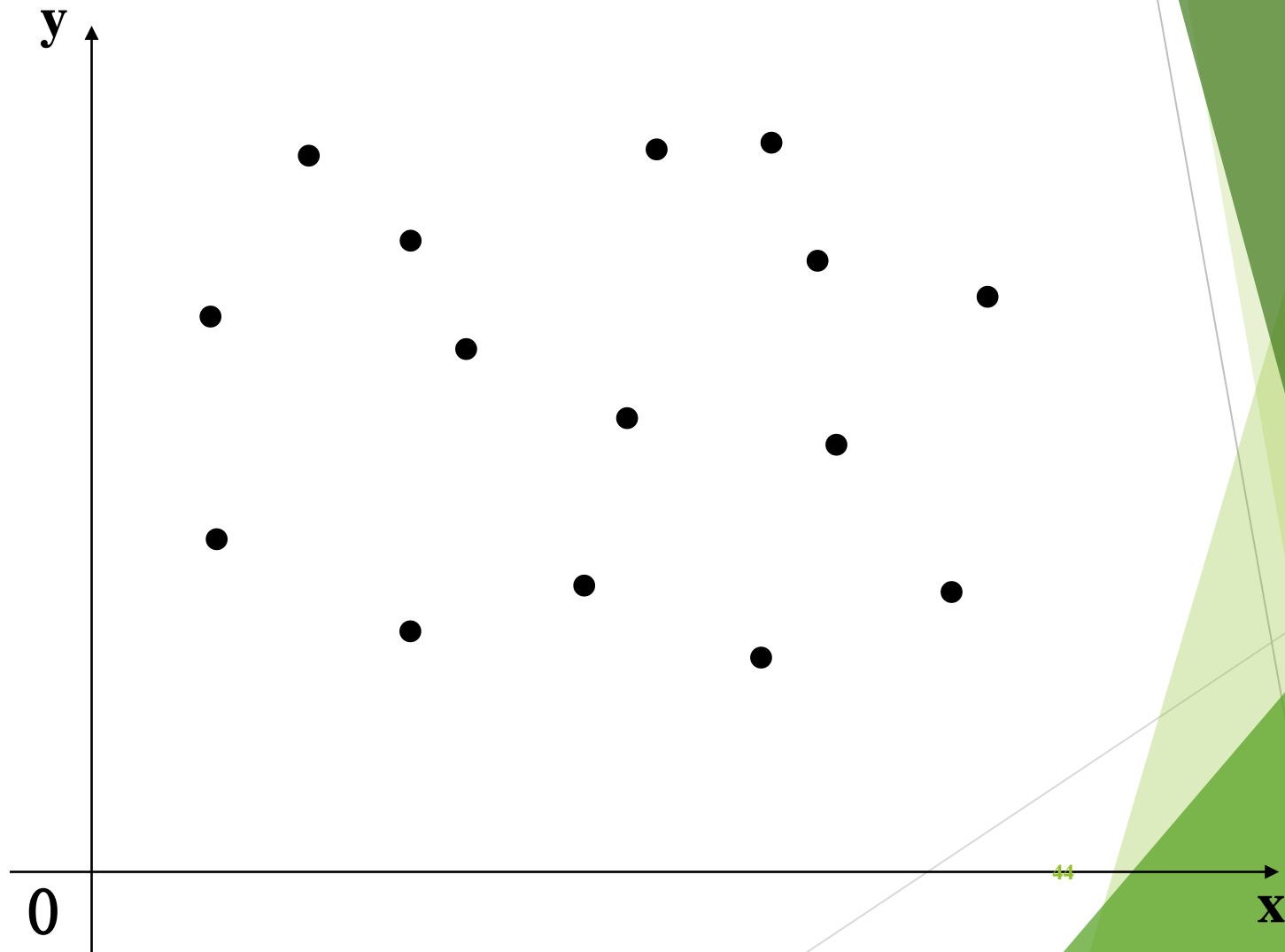
构造一个二元函数 $z = f(x, y)$, 通过全部已知节点, 即

$$f(x_i, y_j) = z_{ij}$$

$$(i = 0, 1, \dots, m; j = 0, 1, \dots, n)$$

再用 $f(x, y)$ 计算插值, 即 $z^* = f(x^*, y^*)$.

第二种（散乱节点）：



已知n个节点

$$(x_i, y_i, z_i) \quad (i = 1, 2, \dots, n)$$

其中 (x_i, y_i) 互不相同,

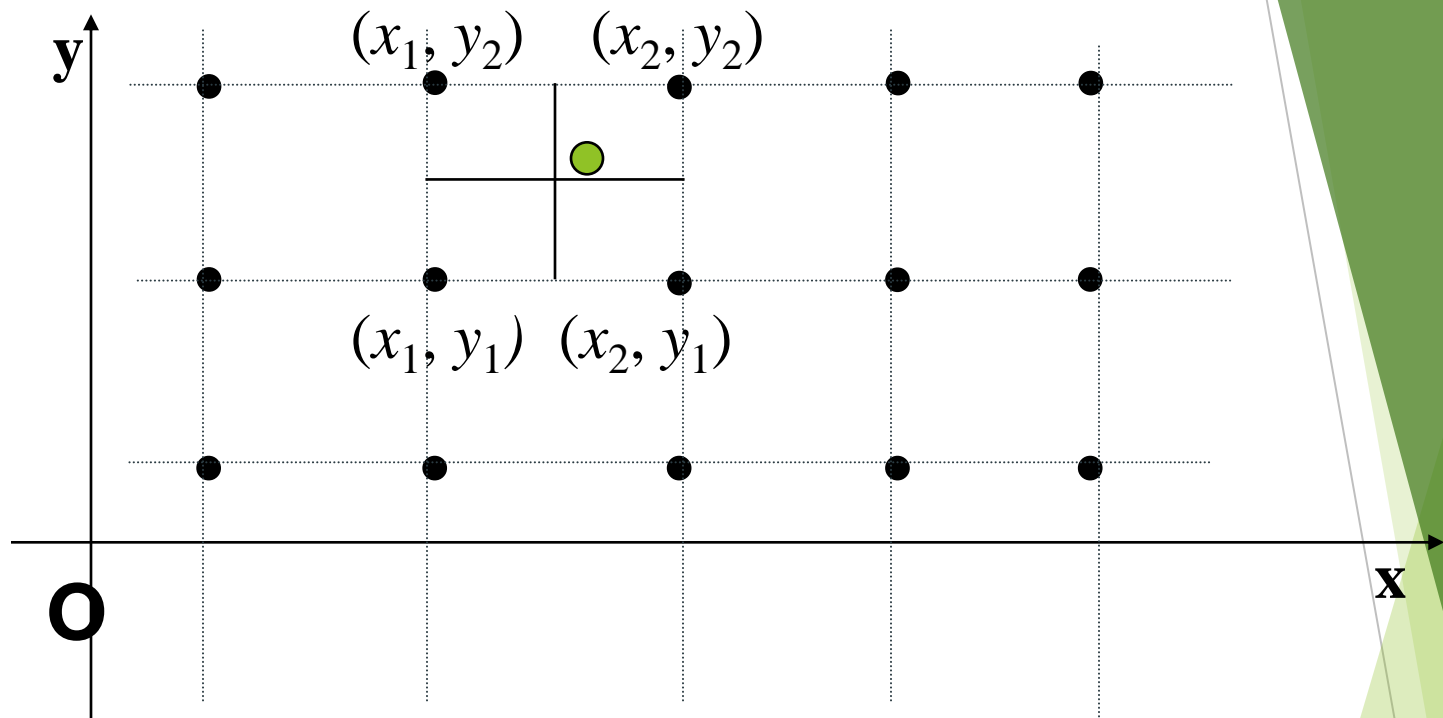
构造一个二元函数 $z = f(x, y)$, 通过全部已知节点, 即

$$f(x_i, y_i) = z_i \quad (i = 0, 1, \dots, n)$$

再用 $f(x, y)$ 计算插值, 即 $z^* = f(x^*, y^*)$.

返回

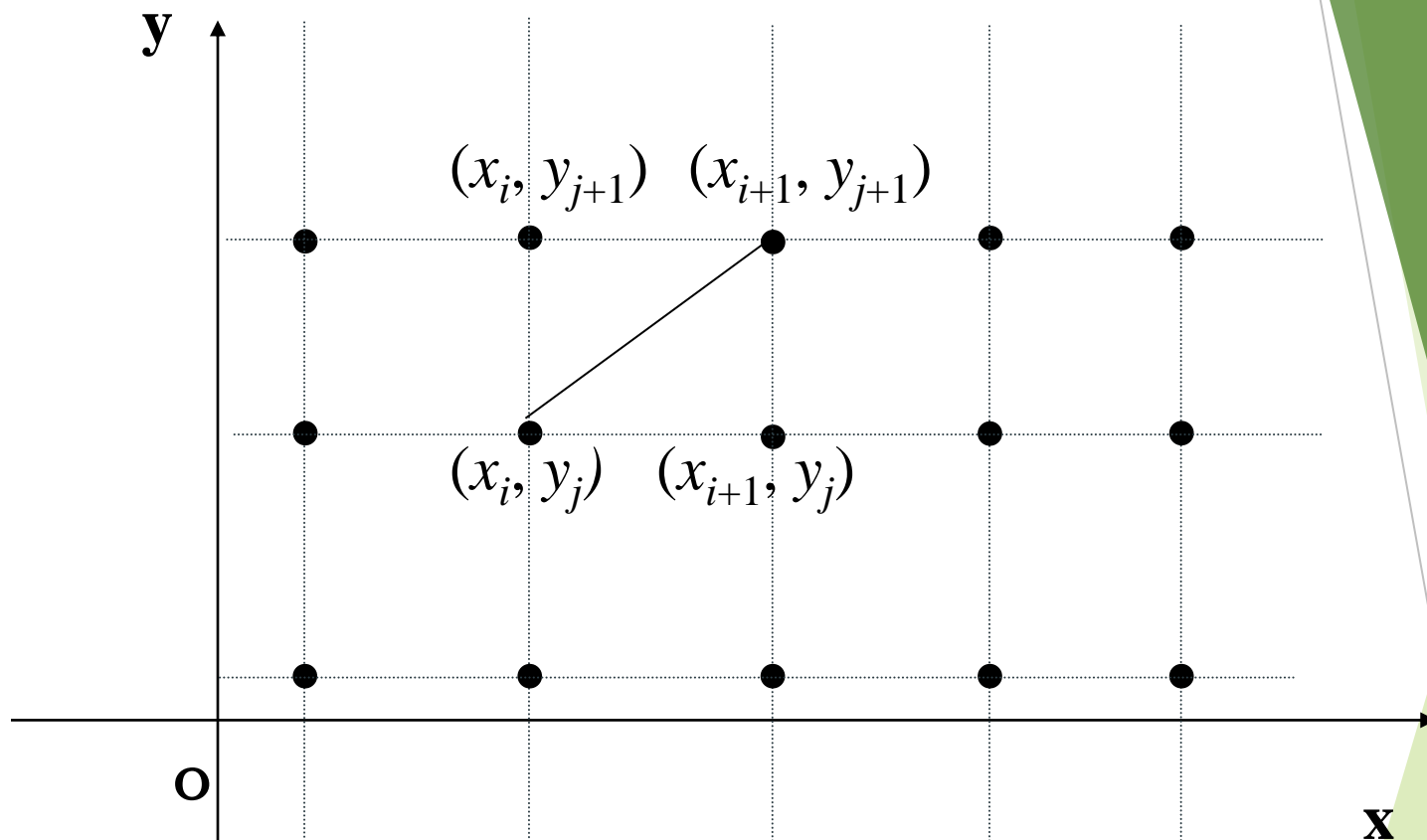
最邻近插值



二维或高维情形的最邻近插值，与被插值点最邻近的节点的函数值即为所求。

注意：最邻近插值一般不连续。具有连续性的最简单的插值是分片线性插值。

分片线性插值



将四个插值点（矩形的四个顶点）处的函数值依次简记为：

$$f(x_i, y_j)=f_1, \quad f(x_{i+1}, y_j)=f_2, \quad f(x_{i+1}, y_{j+1})=f_3, \quad f(x_i, y_{j+1})=f_4$$

分两片的函数表达式如下：

第一片(下三角形区域)： (x, y) 满足

$$y \leq \frac{y_{j+1} - y_j}{x_{i+1} - x_i} (x - x_i) + y_j$$

插值函数为：

$$f(x, y) = f_1 + (f_2 - f_1)(x - x_i) + (f_3 - f_2)(y - y_j)$$

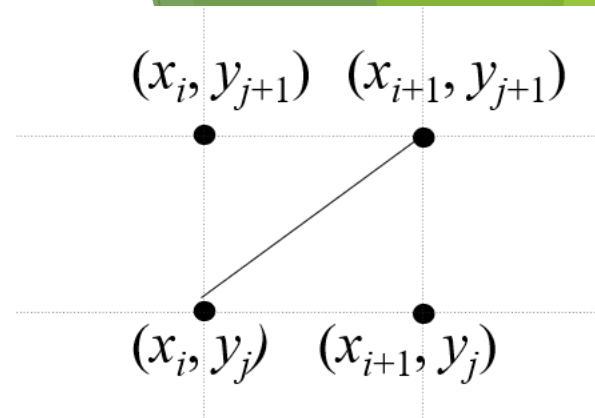
第二片(上三角形区域)： (x, y) 满足

$$y > \frac{y_{j+1} - y_j}{x_{i+1} - x_i} (x - x_i) + y_j$$

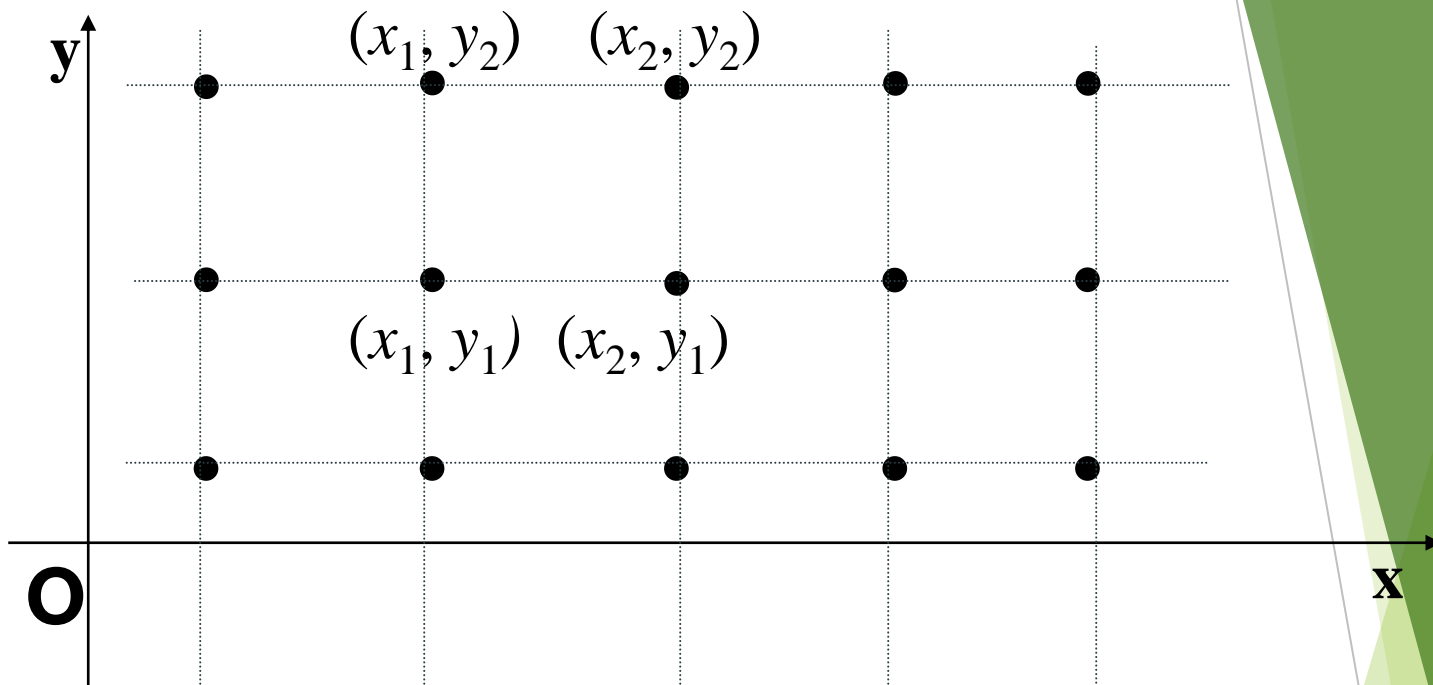
插值函数为：

$$f(x, y) = f_1 + (f_4 - f_1)(y - y_j) + (f_3 - f_4)(x - x_i)$$

注意： (x, y) 当然应该是在插值节点所形成的矩形区域内。
显然，分片线性插值函数是连续的；



双线性插值



双线性插值是一片一片的空间二次曲面构成。双线性插值函数的形式如下：

$$f(x, y) = (ax + b)(cy + d)$$

其中有四个待定系数，利用该函数在矩形的四个顶点（插值节点）的函数值，得到四个代数方程，正好确定四个系数。

用MATLAB作网格节点数据的插值

`z=interp2(x0,y0,z0,x,y,'method')`

被插值点
的函数值

插值
节点

被插值点

插值方法

'nearest'	最邻近插值
'linear'	双线性插值
'cubic'	双三次插值
缺省时,	双线性插值

要求 x_0, y_0 单调； x, y 可取为矩阵，或 x 取行向量， y 取为列向量， x, y 的值分别不能超出 x_0, y_0 的范围。

例：测得平板表面3*5网格点处的温度分别为：

82 81 80 82 84

79 63 61 65 81

84 84 82 85 86

试作出平板表面的温度分布曲面 $z=f(x,y)$ 的图形。

1.先在三维坐标画出原始数据，画出粗糙的温度分布曲图。

输入以下命令：

```
x=1:5;
```

```
y=1:3;
```

```
temps=[82 81 80 82 84;79 63 61 65 81;84 84 82 85 86];
```

```
mesh(x,y,temps)
```

2. 以平滑数据,在x、y方向上每隔0.2个单位的地方进行插值.

再输入以下命令:

```
xi=1:0.2:5;
```

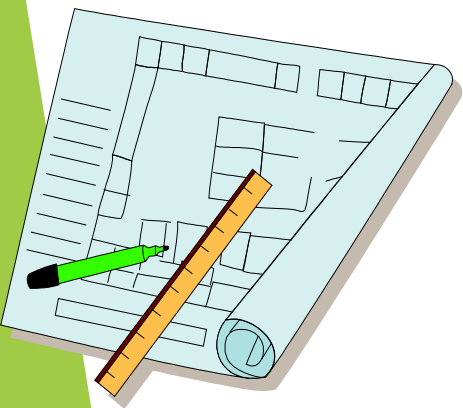
```
yi=1:0.2:3;
```

```
zi=interp2(x,y,temps,xi',yi,'cubic');
```

```
mesh(xi,yi,zi)
```

画出插值后的温度分布曲面图.

To MATLAB
(wendu)



例 山区地貌：

在某山区测得一些地点的高程如下表。平面区域为

$$1200 \leq x \leq 4000, 1200 \leq y \leq 3600$$

试作出该山区的地貌图和等高线图，并对几种插值方法进行比较。

<div>X \ Y</div>	1200	1600	2000	2400	2800	3200	3600	4000
1200	1130	1250	1280	1230	1040	900	500	700
1600	1320	1450	1420	1400	1300	700	900	850
2000	1390	1500	1500	1400	900	1100	1060	950
2400	1500	1200	1100	1350	1450	1200	1150	1010
2800	1500	1200	1100	1550	1600	1550	1380	1070
3200	1500	1550	1600	1550	1600	1600	1600	1550
3600	1480	1500	1550	1510	1430	1300	1200	980

通过此例对最近邻点插值、双线性插值方法和双三次插值方法的插值效果进行比较。

[To MATLAB \(moutain\)](#)

返回

用MATLAB作散点数据的插值计算

插值函数griddata格式为：

cz=griddata (x, y, z, cx, cy, 'method')

被插值点的
函数值

插值
节点

被插值点

插值方法

'nearest' 最邻近插值
'linear' 双线性插值
'cubic' 双三次插值
'v4'- Matlab提供的插值方法
缺省时，双线性插值

要求cx取行向量，cy取为列向量。

例 在某海域测得一些点(x,y)处的水深z由下表给出，船的吃水深度为5英尺，在矩形区域 $(75, 200) \times (-50, 150)$ 里的哪些地方船要避免进入。

x	129	140	103.5	88	185.5	195	105
y	7.5	141.5	23	147	22.5	137.5	85.5
z	4	8	6	8	6	8	8
x	157.5	107.5	77	81	162	162	117.5
y	-6.5	-81	3	56.5	-66.5	84	-33.5
z	9	9	8	8	9	4	9

1. 输入插值基点数据;
2. 在矩形区域 $(75, 200) \times (-50, 150)$ 作二维插值.
三次插值法(*hd1*)
3. 作海底曲面图;
4. 作出水深小于5的海域范围, 即 $z=5$ 的等高线.

To MATLAB *hd1*

返回

黄河小浪底调水调沙问题

2004年6月7日黄河进行了第三次调水调沙试验，特别是首次由小浪底、三门峡和万家寨三大水库联合调度，采用接力式防洪预泄放水，形成人造洪峰进行调沙试验获得成功。整个试验为期20多天，小浪底从6月19日开始预泄放水，直到7月13日结束并恢复正常供水。

小浪底水利工程按照设计拦沙量为75.5亿立方米，在这之前，小浪底共积泥沙14.14亿吨。

这次调水调沙试验一个重要目的就是由小浪底上游的三门峡和万家寨水库泄洪，在小浪底形成人造洪峰，冲刷小浪底库区沉积的泥沙，在小浪底水库开闸泄洪以后，从6月27日开始三门峡水库和万家寨水库陆续开闸放水，人造洪峰与6月29日先后到达小浪底，7月3日达到最大流量2700立方米/秒，使小浪底的排沙量也不断增加。

返回

To MATLAB and 1

观察数据

日期	6.29		6.30		7.1		7.2		7.3		7.4	
时间	8:00	20:00	8:00	20:00	8:00	20:00	8:00	20:00	8:00	20:00	8:00	20:00
水流量	1800	1900	2100	2200	2300	2400	2500	2600	2650	2700	2720	2650
含沙量	32	60	75	85	90	98	100	102	108	112	115	116
日期	7.5		7.6		7.7		7.8		7.9		7.10	
时间	8:00	20:00	8:00	20:00	8:00	20:00	8:00	20:00	8:00	20:00	8:00	20:00
水流量	2600	2500	2300	2200	2000	1850	1820	1800	1750	1500	1000	900
含沙量	118	120	118	105	80	60	50	30	26	20	8	5

现在，根据试验数据建立数学模型研究下列问题：

- (1) 给出估计任意时刻的排沙量及总排沙量的方法；
- (2) 确定排沙量与水流量的关系。

返回

To MATLAB 1

1. 模型假设

- 1) 水流量和排沙量都是连续的，不考虑上游泄洪所带来的含沙量和外界带来的含沙量。
- 2) 时间是连续变化的，所取时间点依次为1,2,3, ...,24,单位时间为12h。

模型建立

已知给定的观测时刻是等距的，以6月29日零时刻为开始时刻计时，则各次的观测时刻分别为

$$t_i = 3600(24i - 4), i = 1, 2, \dots, 24$$

其中，计时单位为秒，第1次观测时刻为 $t_1 = 28800$ ，最后1次观测时刻为 $t_{24} = 1022400$ 。

记第 i 次观测时的水流量为 v_i ，含沙量为 c_i ，则第 i 次观测的排沙量为 $y_i = c_i v_i$ 。有关数据如下：（排沙量：单位kg）

日期	1	2	3	4	5	6	7	8	9	10	11	12
时间	28800	72000	115200	158400	201600	244800	288000	331200	374400	417600	460800	504000
排沙量	576000	114000	157500	187000	207000	235200	250000	265200	286200	302400	312800	307400
日期	13	14	15	16	17	18	19	20	21	22	23	24
时间	547200	590400	633600	676800	720000	763200	806400	849600	892800	936000	979200	1022400
排沙量	306800	300000	271400	231000	160000	111000	91000	54000	45500	30000	8000	4500

模型建立

对于问题(1),根据所给问题的试验数据,要计算任意时刻的排沙量,就要确定出排沙量随时间变化的规律,可以通过插值来实现。考虑到实际中的排沙量应该是时间的连续函数,为了提高模型的精度,采用三次样条函数进行插值。

利用Matlab函数,求出三次样条函数,得到排沙量 $y = y(t)$ 与时间的关系,然后进行积分,就可以得到总的排沙量

$$z = \int_{t_1}^{t_{24}} y(t) dt$$

最后求出总的排沙量为 $1.844 * 10^9 t$;

返回

To MATLAB m1

计算的Matlab程序如下：

```
clc,clear
```

load data3.txt%把观测数据的日期和时间行删除，余下的数据保存在纯文本文件中

```
liu=data3([1,3],:);liu=liu' ;liu=liu(:);%提出水流量并按照顺序变成列向量
```

```
sha=data3([2,4],:);sha=sha' ;sha=sha(:);%提出含沙量并按照顺序变成列向量
```

```
y=sha*liu;y=y' ;%计算排沙量，并变成行向量
```

```
i=1:24;
```

```
t=(12*i-4)*3600;
```

```
t1=t(1);t2=t(end);
```

```
pp=csape(t,y);%进行三次样条插值
```

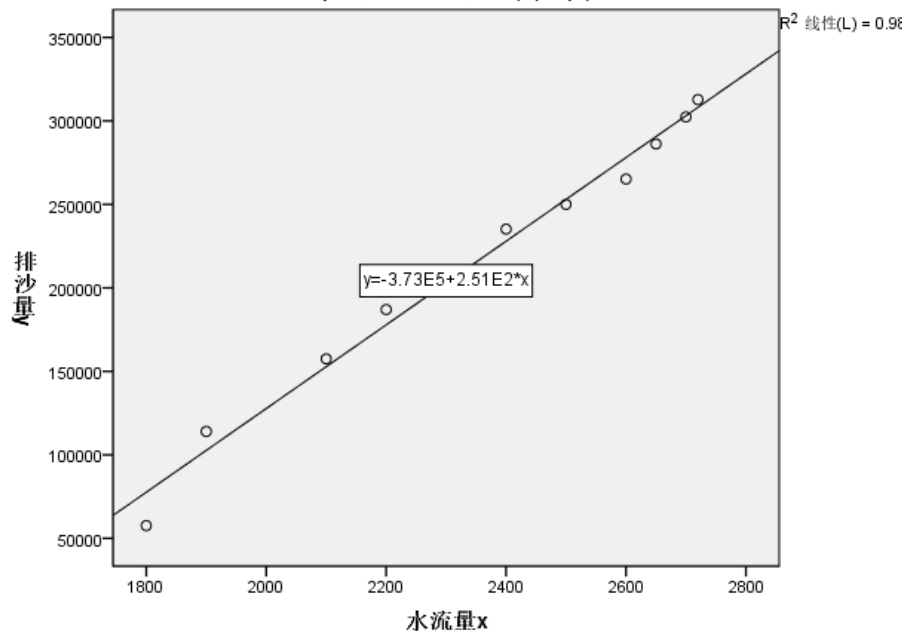
```
xsh=pp.coefs%求得多项式的系数矩阵，每一行是一个区间上的多项式系数
```

```
TL=quadl(@(tt)ppval(pp,tt),t1,t2)%求总含沙量的积分运算
```

模型建立

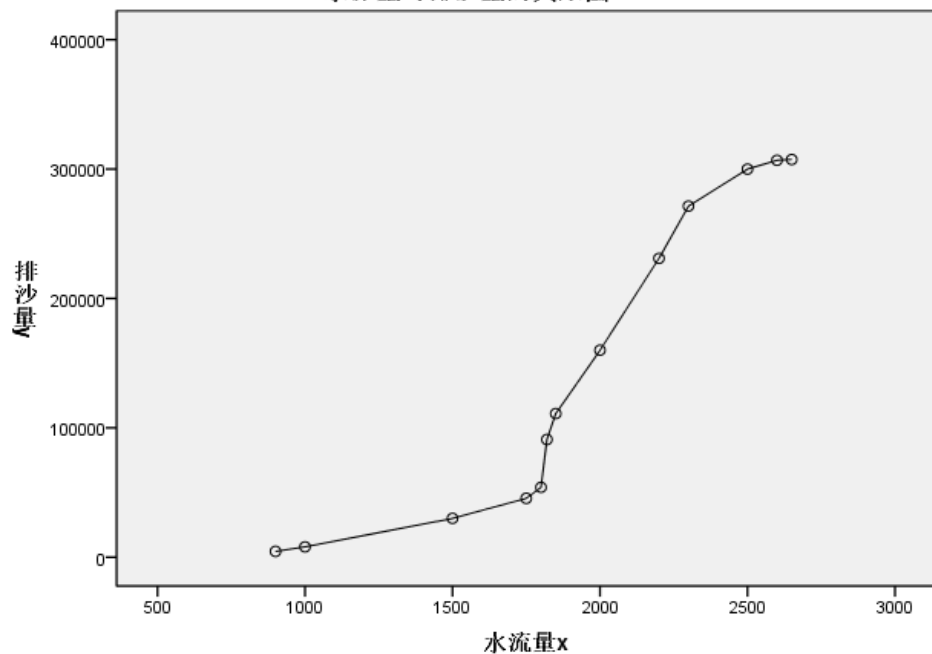
对于问题(2)，研究排沙量与水流量的关系，从试验数据可以看出，开始排沙量是随水流量的增加而增长，而后是随着水流量的减少而减少。显然，变化规律并非是线性的关系，为此，把问题分成两部分，从开始水流量增加到最大值 $2720 \text{ m}^3/\text{s}$ (即增长过程)为第一阶段，从水流量的最大值到结束为第二阶段，分别来研究水流量与排沙量的关系。

水流量与排沙量的关系图



第一阶段

水流量与排沙量的关系图



第二阶段

模型建立

从散点图可以看出，第一阶段基本上是线性关系，第二阶段为非线性关系，可以用多项式拟合，哪一个模型的剩余标准差最小就选取哪一个模型。最后求得第一阶段排沙量 y 与水流量 v 之间的关系为

$$y = 250.5655v - 373384.4661$$

第二阶段排沙量 y 与水流量 v 之间的关系为

$$y = -2.7693 * 10^{-7}v^4 + 0.0018v^3 - 4.092v^2 + 3891.0441v - 1322627.49668$$

返回

To MATLAB 11

计算的Matlab程序如下：

```
clc,clear
load data3.txt%把观测数据的日期和时间行删除，余下的数据保存在纯文本文件中
liu=data3([1,3],:);liu=liu' ;liu=liu(:);%提出水流量并按照顺序变成列向量
sha=data3([2,4],:);sha=sha' ;sha=sha(:);%提出含沙量并按照顺序变成列向量
y=sha*liu;%计算排沙量，这里是列向量
format long e
%以下是第一阶段的拟合
for j=1:2
nihe1{j}=polyfit(liu(1:11),y(1:11),j);%拟合多项式，系数从次数高次幂到低次幂
yhat1{j}=polyval(nihe1{j},liu(1:11));%求预测值
%以下求误差平方和与剩余标准差
cha1(j)=sum((y(1:11)-yhat1{j}).^2);rmsel(j)=sqrt(cha1(j)/(10-j));
end
nihe1{:}%显示细胞数组的所有元素
rmsel
```

计算的Matlab程序如下：

%以下是第二阶段的拟合

for j=1:3

nihe2{j}=polyfit(liu(12:24),y(12:24),j+1);%这里是使用细胞数组

yhat2{j}=polyval(nihe2{j},liu(12:24));%求预测值

%以下求误差平方和与剩余标准差

cha2(j)=sum((y(12:24)-yhat2{j}).^2);rmse2(j)=sqrt(cha2(j)/(11-j));

end

nihe2{:}%显示细胞数组的所有元素

rmse2

format%恢复默认的最小数的显示格式