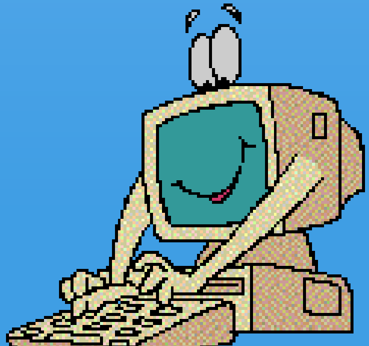
数值计算

第四章 插值与拟合



4.1 引例及问题综述

4.1.1 问题综述

在科学研究和工程中，常常会遇到计算函数$y=f(x)$的函数值、导数值、零点、极值或积分值等一类问题。然而函数关系往往是很复杂的，甚至没有明显的解析表达式。

例如，根据观测或实验得到一系列的数据，确定了与自变量的某些点相应的函数值。

x	x_0	x_1	…	x_n
$f(x)$	$f(x_0)$	$f(x_1)$	…	$f(x_n)$

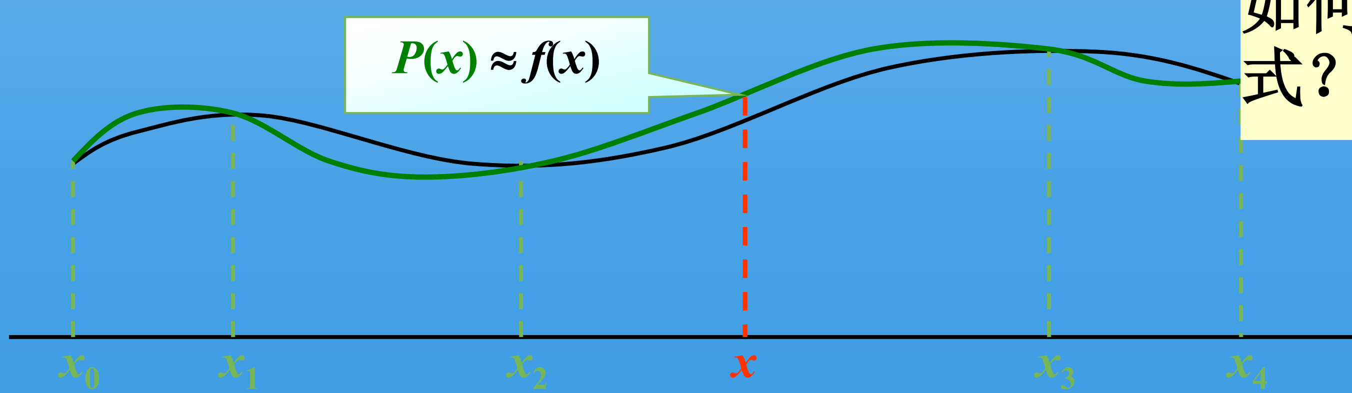
当函数比较复杂或根本无法写出解析式时，往往根据观测数据构造一个适当的简单的函数近似地代替要寻求的函数。

当近似的指标含义不同时，就构成了插值与拟合两种研究方法。

思路/* Lagrange Polynomial */

当精确函数 $y = f(x)$ 非常复杂或未知时，在一系列节点 $x_0 \ldots x_n$ 处测得函数值 $y_0 = f(x_0), \ldots y_n = f(x_n)$，由此构造一个简单易算的近似函数 $P(x) \approx f(x)$，满足条件$P(x_i) = f(x_i)\ \ (i = 0, \ldots n)$。这里的 $P(x)$ 称为$f(x)$ 的插值函数。最常用的插值函数是 …?

多项式



如何求多项式？

插值区间

定义4.1 设函数 $y=f(x)$ 在区间[a, b]上有定义，且已知 $f(x)$ 在 [a, b]上n+1个互异节点：x_0，x_1，…，x_n 上的函数值$f(x_0)$，$f(x_1)$，…，$f(x_n)$。

插值节点

若存在一个$f(x)$的近似函数$P(x)$，满足

$$P(x_i)=f(x_i)\quad(i=0,1,2,\cdots,n)\qquad(4.8)$$

插值函数

被插函数(原函数)

插值条件

误差函数 $R(x)=f(x)-P(x)$

插值余项

通常满足同一个插值条件的插值函数有许多类型，如多项式、三角函数类型、指数函数类型等等。

由于多项式或分段多项式计算简单，分析容易，而往往被选为插值函数。

当插值函数是多项式时称为**代数插值**（或**多项式插值**）。

即若存在一个次数不超过n次的多项式$P_n(x)$，满足

$$P_n(x_i) = f(x_i) \quad (i = 0, 1, 2, \cdots, n) \qquad (4.9)$$

则称$P_n(x)$为$f(x)$ 的**n次插值多项式**。

本章只讨论代数插值。

满足条件：$P(x_i)=f(x_i)=y_i$

$x_i \neq x_j$互异的点

- 对于一般的n次代数插值多项式可以写成下列形式：
- $P_n(x)=a_0+a_1x+a_2x^2+\dots+a_nx^n$ (1)
- 使其在给定的n+1个互异的插值基点上满足插值原则
- $P_n(x_i)=y_i,\quad i=0,1,\dots,n$ (2)

  - 根据插值原则式(2),代数多项式(1)中的各个系数$a_0,a_1,\dots,a_n$应满足下列n+1阶线性方程组

$$\begin{cases} P_n(x_0)=a_0+a_1x_0+a_2x_0^2+\cdots+a_nx_0^n=y_0 \\ P_n(x_1)=a_0+a_1x_1+a_2x_1^2+\cdots+a_nx_1^n=y_1 \\ \vdots \\ P_n(x_n)=a_0+a_1x_n+a_2x_n^2+\cdots+a_nx_n^n=y_n \end{cases} \qquad (3)$$

主要内容

n次插值多项式$P_n(x)$是否存在？
如何求解？
插值误差或余项又如何估计？

4.2 拉格朗日插值

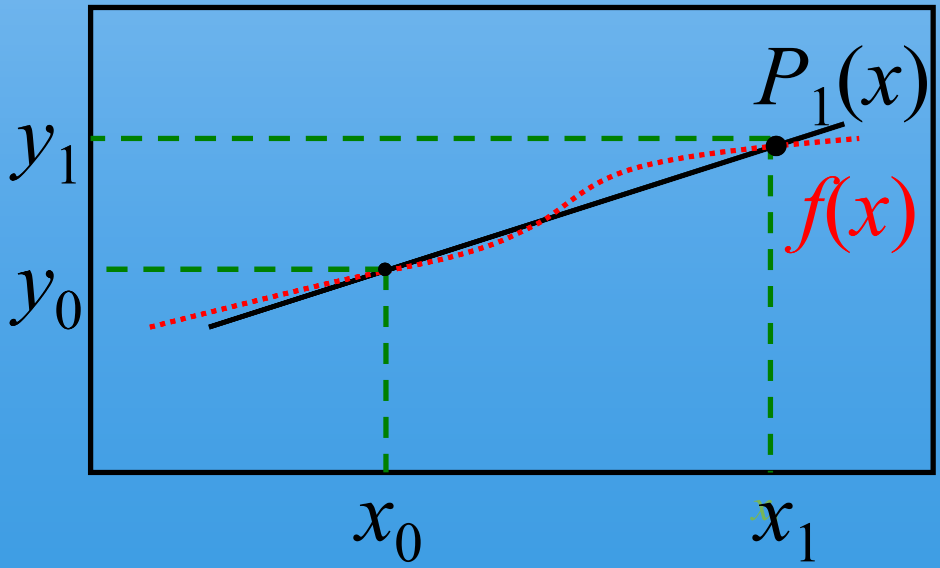
4.2.1 线性插值与抛物插值

$n=1$ **线性插值**

已知 x_0, x_1；y_0, y_1， 求 $P_1(x)=a_0+a_1x$ 使得

$P_1(x_0)=y_0$， $P_1(x_1)=y_1$

可见 $P_1(x)$ 是过 (x_0, y_0) 和 (x_1, y_1) 两点的直线。



线性插值示意图

当$i=j$时$\delta_{ij}=1$； 当$i\neq j$时$\delta_{ij}=0$

$n=1$ 线性插值

称为拉氏基函数（线性插值基函数） /* Lagrange Basis */，

满足条件 $l_i(x_j)=\delta_{ij}$ /* Kronecker Delta */

$P_1(x_0)=y_0$

可见 $P_1(x)$ 是过 (x_0, y_0) 和 (x_1, y_1) 两点的直线。

$$\Rightarrow P_1(x) = y_0 + \frac{y_1 - y_0}{x_1 - x_0}(x - x_0)$$

$$= \underbrace{\left(\frac{x-x_1}{x_0-x_1}\right)}_{l_0(x)} y_0 + \underbrace{\left(\frac{x-x_0}{x_1-x_0}\right)}_{l_1(x)} y_1 = \sum_{i=0}^{1} l_i(x) y_i$$

在节点上满足

$$\begin{cases} l_0(x_0)=1; l_0(x_1)=0 \\ l_1(x_0)=0; l_1(x_1)=1 \end{cases}$$

$n=2$ 抛物线插值

已知 x_0, x_1, x_2；y_0, y_1, y_2， 求 $P_2(x)=a_0+a_1x+a_2x^2$ 使得

$P_2(x_0)=y_0$， $P_2(x_1)=y_1$ $P_2(x_2)=y_2$

可见 $P_2(x)$ 是过 (x_0, y_0)、 (x_1, y_1)、 (x_2, y_2)的抛物线 。

基函数方法 $y=P_2(x)=\sum l_i(x)y_i$ 基函数都是二次的函数

为了满足插值条件在节点上有：

$$\begin{cases} l_0(x_0)=1; l_0(x_1)=0; l_0(x_2)=0 \\ l_1(x_0)=0; l_1(x_1)=1; l_1(x_2)=0 \\ l_2(x_0)=0; l_2(x_1)=0; l_2(x_2)=1 \end{cases}$$ 在节点上满足

每个基函数有两个零点

$$\begin{cases} l_0(x)=A(x-x_1)(x-x_2) \\ l_1(x)=B(x-x_0)(x-x_2) \\ l_2(x)=C(x-x_0)(x-x_1) \end{cases}$$

在节点上条件

$$\begin{cases} l_0(x_0)=1; l_0(x_1)=0; l_0(x_2)=0 \\ l_1(x_0)=0; l_1(x_1)=1; l_1(x_2)=0 \\ l_2(x_0)=0; l_2(x_1)=0; l_2(x_2)=1 \end{cases}$$

求系数

$$\begin{cases} A=\dfrac{1}{(x_0-x_1)(x_0-x_2)} \\ B=\dfrac{1}{(x_1-x_0)(x_1-x_2)} \\ C=\dfrac{1}{(x_2-x_0)(x_2-x_1)} \end{cases} \Rightarrow \begin{cases} l_0(x)=\dfrac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} \\ l_1(x)=\dfrac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} \\ l_2(x)=\dfrac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \end{cases}$$

基函数方法 $y=P_2(x)=\sum l_i(x)y_i$

$n \geq 1$

希望找到$l_i(x)$，$i = 0, \ldots, n$ 使得 $l_i(x_j)=\delta_{ij}$；然后令

$P_n(x) = \sum_{i=0}^{n} l_i(x)\, y_i$，则显然有$P_n(x_i) = y_i$。

$l_i(x)$

每个

$l_i(x_i)=1$ $C_i = \prod_{j \neq i} \frac{1}{(x_i - x_j)}$

与节点有关，而与 f 无关

Lagrange Polynomial

$$l_i(x) = \prod_{\substack{j \neq i \\ j=0}}^{n} \frac{(x - x_j)}{(x_i - x_j)} \quad \Rightarrow \quad L_n(x) = \sum_{i=0}^{n} l_i(x) y_i$$

4.2.2 拉格朗日插值多项式

由一次、二次插值多项式的表示方法很容易推广到一般情形。

如何构造通过$n+1$个节点$x_0< x_1<\ldots< x_n$的n次插值多项式$L_n(x)$，满足

$$L_n(x_j)=y_j \quad (j=0,1,2,\cdots,n) \qquad (4.15)$$

定义4.2 若n次多项式$l_j(x)$ $(j=0,1,2,\ldots,n)$在$n+1$个节点$x_0< x_1<\ldots< x_n$上满足条件

$$l_j(x_k)=\begin{cases}1 & k=j\\ 0 & k\neq j\end{cases} \quad (j,k=0,1,2,\cdots,n) \qquad (4.16)$$

则称这$n+1$个n次多项式$l_j(x)$ $(j=0,1,2,\ldots,n)$为节点$x_0< x_1<\ldots< x_n$上的**n次插值基函数。**

类似于$n=2$时的推导方法，可得n次插值基函数。

$$l_k(x)=\frac{(x-x_0)(x-x_1)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_0)(x_k-x_1)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}$$

$$(k=0,1,2,\cdots,n) \qquad (4.17)$$

显然满足条件（4.16）$l_j(x_k)=\begin{cases}1 & k=j\\ 0 & k\neq j\end{cases} \quad (j,k=0,1,2,\cdots,n)$

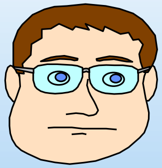
⇨ 满足条件（4.15）的插值多项式$L_n(x)$可表示为

$$L_n(x)=\sum_{k=0}^{n} y_k l_k(x) \qquad (4.18)$$

n次拉格朗日(Lagrange)插值多项式

⇨ $$L_n(x_j)=\sum_{k=0}^{n} y_k l_k(x_j)=y_j \quad (j=0,1,2,\cdots,n)$$

4.2.3 插值多项式的存在唯一性



是不是唯一？

由插值条件$P_n(x_j)=f(x_j)\quad(j=0,1,\ldots,n)$知，插值多项式$p_n(x)$的系数$a_i\,(i=0,1,\ldots,n)$满足线性方程组

$$\begin{cases} a_0+a_1x_0+\cdots+a_nx_0^n=y_0 \\ a_0+a_1x_1+\cdots+a_nx_1^n=y_1 \\ \qquad\cdots\cdots\cdots\cdots\cdots\cdots \\ a_0+a_1x_n+\cdots+a_nx_n^n=y_n \end{cases}$$

其系数行列式（记为V）是n+1阶范德蒙德（Vandermonde)行列式：

$$\mathbf{V}=\begin{vmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{vmatrix}=\prod_{i=1}^{n}\prod_{j=0}^{i-1}(x_i-x_j)$$

$$\mathbf{V}=\begin{vmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{vmatrix}=\prod_{i=1}^{n}\prod_{j=0}^{i-1}(x_i-x_j)$$

因x_0、 x_1、 …、 x_n 是区间[a, b]上不同的点，上式右端乘积中的每一个因子x_i-x_j≠0，于是$\mathbf{V}\neq 0$，方程组的解存在且唯一。

$$\begin{cases} a_0+a_1x_0+\cdots+a_nx_0^n=y_0 \\ a_0+a_1x_1+\cdots+a_nx_1^n=y_1 \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ a_0+a_1x_n+\cdots+a_nx_n^n=y_n \end{cases}$$

定理4.1： 若x_0、 x_1、…、 x_n是互异的节点，且$f(x_j)$ (j = 0,1,…,n)已知，则存在唯一不超过n的多项式$P_n(x)$，使得

$$P_n(x_j) = f(x_j) \quad (j = 0,1,\ldots,n)$$

证：存在性

$$L_n(x) = \sum_{k=0}^{n} y_k l_k(x) \qquad (4.18)$$

n次拉格朗日(Lagrange)插值多项式

唯一性

设另有n次多项式$P_n(x)$ 满足：$P_n(x_j)=f(x_j)$ ($j = 0,1,\ldots,n$)

令$h(x)=P_n(x)-L_n(x)$，则$h(x)$是一个次数不超过n的多项式，且

$h(x_j)=P_n(x_j)-L_n(x_j)=f(x_j) - f(x_j) =0$ ($j = 0,1,\ldots,n$)

$$h(x_j)=P_n(x_j)-L_n(x_j)=f(x_j)-f(x_j)=0 \quad (j=0,1,\ldots,n)$$

即$h(x)$有$n+1$个互异的根。
由代数基本定理知n次代数方程有且仅有n个根，
因此

$$h(x)\equiv 0$$

故$P_n(x)=L_n(x)$。定理得证。

注：若不将多项式次数限制为 n ，则插值多项式不唯一。

例如 $P(x)=L_n(x)+p(x)\prod_{i=0}^{n}(x-x_i)$ 也是一个插值多项式，其中$p(x)$可以是任意多项式。

例如：过两点的直线只有一条，但过两点的抛物线有无限多条！

例4.1 已知函数$f(x)$在节点-1，0，1处的值分别为0.3679，1.000，2.7182，求出拉格朗日插值多项式。

解法一 设所求的多项式为 $L_2(x)=a_0+a_1x+a_2x^2$

把已知条件代入得到：

$$\begin{cases} a_0+a_1(-1)+a_2(-1)^2=0.3679 \\ a_0+a_1(0)+a_2(0)^2=1.000 \\ a_0+a_1(1)+a_2(1)^2=2.7182 \end{cases}$$

求出系数得到：

$$L_2(x)=1+1.1751x+0.5431x^2$$

解法二 ？

$$L_n(x)=\sum_{k=0}^{n}y_k l_k(x) \qquad (4.18)$$

$$l_k(x)=\frac{(x-x_0)(x-x_1)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_0)(x_k-x_1)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}$$

$$(k=0,1,2,\cdots,n) \qquad (4.17)$$

解法二 利用公式(4.17)

$$l_k(x)=\frac{(x-x_0)(x-x_1)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_0)(x_k-x_1)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}$$

$$(k=0,1,2,\cdots,n) \tag{4.17}$$

$$l_0(x)=\frac{(x-0)(x-1)}{(-1-0)(-1-1)}=\frac{x(x-1)}{2}$$

$$l_1(x)=\frac{[x-(-1)](x-1)}{[0-(-1)](0-1)}=-(x+1)(x-1)$$

$$l_2(x)=\frac{[x-(-1)](x-0)}{[1-(-1)](1-0)}=\frac{x(x+1)}{2}$$

代入(4.18)得 $$L_n(x)=\sum_{k=0}^{n} y_k l_k(x) \qquad (4.18)$$

$$L_2(x)=0.3679 l_0(x)+1.0000 l_1(x)+2.7182 l_2(x)$$

即

$$L_2(x)=1+1.1751x+0.5431x^2$$

两种不同方法结论完全一样。——插值多项式是存在且唯一的。

思考：

1. 拉格朗日方法似乎更麻烦？那为什么要用拉格朗日方法？用解线性方程组的方法不是更好吗？

2. 用拉格朗日方法解决实际差值问题时，有必要计算出a_i吗？

4.2.4 插值余项

在插值区间$[a, b]$上用n次插值多项式$L_n(x)$近似代替$f(x)$，除了在插值节点$x_j(j=0,1,\ldots,n)$上没有误差外，

$$L_n(x_j)=y_j \quad (j=0,1,2,\cdots,n) \qquad (4.15)$$

在其它点上一般是存在有误差的。

若$x \neq x_j(j=0,1,2,\cdots,n)$，则一般

$$L_n(x) \neq f(x)$$

若记$\mathrm{R}_n(x)=f(x)-L_n(x)$，

则$\mathrm{R}_n(x)$就是用$L_n(x)$近似代替$f(x)$时的**截断误差**（不考虑舍入误差）。

称**$\mathrm{R}_n(x)$为插值多项式$L_n(x)$的余项**，并且可根据下面定理来估计它的大小。

定理4.2 设$f(x)$在包含n+1个互异节点x_0、x_1、…、x_n的区间$[a, b]$上具有n阶连续导数，且在（a, b）内存在n+1阶导数，则对任意$x\in[a, b]$，必存在一点ξ $\in(a,b)$，使得

$$R_n(x)=f(x)-L_n(x)=\frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x),$$

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

$\xi\in(a,b)$ 与x的位置有关。

证明:

当$x=x_j(j=0,1,\cdots n)$时，$R_n(x_j)=f(x_j)-L_n(x_j)=0,$

$\omega_{n+1}(x_j)=\prod_{i=0}^{n}(x_j-x_i)=0;$ 显然下面的结论成立

$$R_n(x)=f(x)-L_n(x)=\frac{f^{n+1}(\xi)}{(n+1)!}\omega_{n+1}(x)$$

其中，$\omega_{n+1}(x)=\prod_{i=0}^{n}(x-x_i)$

当$x \neq x_j (j=0,1,\cdots n)$时，

$x_j (j=0,1,\cdots n)$是$R_n(x_j)=f(x_j)-L_n(x_j)=0$的$n+1$个零点，

设：$R_n(x)=K(x)(x-x_0)(x-x_1)\cdots(x-x_n)=K(x)\omega_{n+1}(x)$

$$\omega_{n+1}(x)=\prod_{i=0}^{n}(x-x_i)$$

其中，$K(x)$是待定的函数，它与x的位置有关。

设一个辅助函数：$\varphi(t)=R_n(t)-K(x)\omega_{n+1}(t)$

$\varphi(t)$至少有$n+2$个互异的零点，它们是x_0，x_1，$\cdots$，x_n，x

$R_n(x)$ 至少有 $n+1$ 个根 $\longrightarrow$ $R_n(x) = K(x)\prod_{i=0}^{n}(x - x_i)$

任 注意这里是对 t 求导 $\varphi(t) = R_n(t) - K(x)\prod_{i=0}^{n}(t - x_i)$

$\varphi(x)$有 $n+2$ 个不同的根 $x_0 \dots x_n\ x$ $\longrightarrow$ $\varphi^{(n+1)}(\xi_x) = 0,\quad \xi_x \in (a, b)$

$$\|$$

$$f^{(n+1)}(\xi_x) - \cancel{L_n^{(n+1)}}(\xi_x) = K(x)(n+1)! \longleftarrow R_n^{(n+1)}(\xi_x) - K(x)\ (n+1)!$$

$$\longrightarrow\quad K(x) = \frac{f^{(n+1)}(\xi_x)}{(n+1)!}$$

$$\boxed{R_n(x) = \frac{f^{(n+1)}(\xi_x)}{(n+1)!}\prod_{i=0}^{n}(x - x_i)}$$

注： ☞ 通常不能确定 ξ_x, 而是估计 $\left|f^{(n+1)}(x)\right| \le M_{n+1}$, $\forall x \in (a,b)$

将 $\dfrac{M_{n+1}}{(n+1)!}\prod_{i=0}^{n}|x-x_i|$ 作为误差估计上限。

☞当 $f(x)$ 为任一个次数$\le n$ 的多项式时， $f^{(n+1)}(x) \equiv 0$, 可知 $R_n(x) \equiv 0$ ，即插值多项式对于次数$\le n$ 的多项式是精确的。

$$R_n(x) = f(x) - L_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x)$$

如果$f^{(n+1)}(x)$在$[\boldsymbol{a}, \mathrm{b}]$上有界，即存在常数M，使

$\max\limits_{a \le x \le b}\left|f^{(n+1)}(x)\right| \le M$, 则必有

$$\left|R_n(x)\right| \le \frac{M}{(n+1)!}\left|x - x_0\right|\left|x - x_1\right|\cdots\left|x - x_n\right|$$

由此可知，x（称**插值点**）邻近节点x_0、x_1、…、x_n中某点时，插值多项式的误差$\left|R_n(x)\right|$可能很小；不邻近任一节点时，误差可能很大。

例：已知 $\sin\frac{\pi}{6}=\frac{1}{2}$，$\sin\frac{\pi}{4}=\frac{1}{\sqrt{2}}$，$\sin\frac{\pi}{3}=\frac{\sqrt{3}}{2}$

分别利用 $\sin x$ 的1次、2次 Lagrange 插值计算 $\sin 50°$ 并估计误差。

$50^0=\frac{5\pi}{18}$

解：$n=1$ 分别利用x_0, x_1 以及 x_1, x_2 计算

x_0 x_1 x_2

- 利用 $x_0=\frac{\pi}{6}, x_1=\frac{\pi}{4}$ ➡ $L_1(x)=\frac{x-\pi/4}{\pi/6-\pi/4}\times\frac{1}{2}+\frac{x-\pi/6}{\pi/4-\pi/6}\times\frac{1}{\sqrt{2}}$

si … x, $f^{(2)}(\xi_x)=-\sin\xi_x$，$\xi_x\in(\frac{\pi}{6},\frac{\pi}{3})$

… $\frac{\pi}{4})$

内插通常优于外推。选择要计算的 x 所在的区间的端点，插值效果较好。

➡ … $\sin 50° = 0.7660444\ldots$

外推 /* extrapolation */ 的实…误差 ≈ -0.01001

- 利用 $x_1=\frac{\pi}{4}, x_2=\frac{\pi}{3}$ ➡ $\sin 50° \approx$ 0. … 8, $0.00538<\tilde{R}_1\left(\frac{5\pi}{18}\right)<0.00660$

内插 /* interpolation */ 的实际误差 ≈ 0.00596

$$l_i(x)=\prod_{\substack{j=0\\ j\neq i}}^{n}\frac{(x-x_j)}{(x_i-x_j)} \Longrightarrow L_n(x)=\sum_{i=0}^{n} l_i(x)\,y_i$$

$n=2$

$$L_2(x)=\frac{(x-\frac{\pi}{4})(x-\frac{\pi}{3})}{(\frac{\pi}{6}-\frac{\pi}{4})(\frac{\pi}{6}-\frac{\pi}{3})}\times\frac{1}{2}+\frac{(x-\frac{\pi}{6})(x-\frac{\pi}{3})}{(\frac{\pi}{4}-\frac{\pi}{6})(\frac{\pi}{4}-\frac{\pi}{3})}\times\frac{1}{\sqrt{2}}+\frac{(x-\frac{\pi}{6})(x-\frac{\pi}{4})}{(\frac{\pi}{3}-\frac{\pi}{6})(\frac{\pi}{3}-\frac{\pi}{4})}\times\frac{\sqrt{3}}{2}$$

$$\sin 50^0 \approx L_2(\frac{5\pi}{18}) \approx 0.76543$$

$$R_2(x)=\frac{-\cos\xi_x}{3\,!}(x-\frac{\pi}{6})(x-\frac{\pi}{4})(x-\frac{\pi}{3})\,; \qquad \frac{1}{2}<\cos\xi_x<\frac{\sqrt{3}}{2}$$

$$\Rightarrow \quad 0.00044 < R_2\left(\frac{5\pi}{18}\right) < 0.00077 \qquad \sin 50° = 0.7660444\ldots$$

2次插值的实际误差 ≈ 0.00061

高次插值通常优于
插值

但绝对不是次数越
高就越好……

线性插值与抛物线插值的截断误差：

$$R_n(x)=\frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x)$$

$$R_1(x)=\tfrac{1}{2}f''(\xi)(x-x_0)(x-x_1)$$

$$R_2(x)=\tfrac{1}{6}f'''(\xi)(x-x_0)(x-x_1)(x-x_2)$$

例4.2 对函数 $f(x)=\int_0^x e^{-t^2}dt$ ，假设在区间[1，2]上以 x_0=1，x_1=2为插值节点作线性插值计算$f(x)$的近似值，问会有多大的误差？

解：作线性插值多项式$L_1(x)$，并取 $f(x)\approx L_1(x)$

由插值余项公式有

$$R_1(x)=f(x)-L_1(x)=\tfrac{1}{2}f''(\xi)(x-x_0)(x-x_1)$$

$$(x_0<\xi<x_1)$$

由于

$$f'(x)=e^{-x^2},f''(x)=-2xe^{-x^2},f'''(x)=2e^{-x^2}(2x^2-1)$$

当 $x \in (1,2)$ 时，$f'''(x) > 0$ 所以在$(1,2)$上 $f''(x) > 0$ 单调递增，

$$f''(1) \le f''(x) \le f''(2)$$

而

$$f''(1) = -0.73576, f''(2) = -0.07326$$

可得

$$\left|f''(x)\right| \le \left|f''(1)\right| = 0.73576$$

最大误差满足

$$\max_{1\le x\le 2}\left|R_1(x)\right| \le \tfrac{1}{2}\max_{1\le x\le 2}\left|f''(x)\right|\square\max_{1\le x\le 2}\left|(x-x_0)(x-x_1)\right|$$

$$\le \tfrac{1}{2}\times 0.73576\times\tfrac{1}{4} = 0.09197$$

拉格朗日插值多项式

$$L_n(x)=\sum_{i=0}^{n} l_i(x)y_i=\sum_{i=0}^{n} y_i\frac{(x-x_0)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)}{(x_i-x_0)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)}$$

$$=\sum_{i=0}^{n}\Big(\prod_{\substack{j=0\\ j\neq i}}^{n}\frac{x-x_j}{x_i-x_j}\Big)y_i$$

插值余项

$$R_n(x)=\frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x)$$

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

4.3 差商与牛顿插值公式

4.3.1 差商及其性质

当插值节点增加或减少时，拉格朗日插值法的全部插值基函数$l_k(x)$ (k=0,1,…,n)均要随之变化，整个公式也将改变，这不满足实际工程中的要求。

为了克服这个缺点，希望能给出一个构造$L_k(x)$的方法，它只需要对$L_{k-1}(x)$作一个简单的修正。为此，设

$$g(x)=L_k(x)-L_{k-1}(x)$$

显然$g(x)$是一个次数不超过k次的多项式，而且有

$$g(x_j)=L_k(x_j)-L_{k-1}(x_j)=f(x_j)-f(x_j)=0$$

$$(j=0,1,\cdots,k-1)$$

$$g(x_j) = L_k(x_j) - L_{k-1}(x_j) = f(x_j) - f(x_j) = 0$$

$$(j = 0, 1, \cdots, k-1)$$

这样$g(x)$有k个零点$x_0, x_1, \ldots, x_{k-1}$，因而$g(x)$可表示为

$$g(x) = a_k(x - x_0)(x - x_1)\cdots(x - x_{k-1})$$

其中a_k为待定常数。这样由$g(x) = L_k(x) - L_{k-1}(x)$可知

$$L_k(x) = L_{k-1}(x) + a_k(x - x_0)(x - x_1)\cdots(x - x_{k-1})$$

利用这个公式反复递推，可得插值多项式便于计算的形式：

$$\begin{aligned} L_n(x) = {} & a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \cdots \\ & + a_n(x - x_0)(x - x_1)\cdots(x - x_{n-1}) \end{aligned}$$

$$L_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \cdots$$
$$+ a_n(x - x_0)(x - x_1)\cdots(x - x_{n-1})$$

其中a_k (k =0, 1, …, n) 为待定常数。可由插值条件来确定

$$L_n(x_j) = f(x_j) \quad (j = 0,1,2,\cdots,n)$$

为满足插值条件，$L_n(x)$需而且只需满足方程组

$$\begin{cases} f(x_0) = L_n(x_0) = a_0 \\ f(x_1) = L_n(x_1) = a_0 + a_1(x_1 - x_0) \\ f(x_2) = L_n(x_2) = a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) \\ \qquad\qquad\vdots \\ f(x_n) = L_n(x_n) = a_0 + a_1(x_n - x_0) + a_2(x_n - x_0)(x_n - x_1) \\ + \cdots + a_n(x_n - x_0)(x_n - x_1)\cdots(x_n - x_{n-1}) \end{cases}$$

由此显然

$$a_0 = f(x_0)$$

$$a_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

$$f(x_2) = L_n(x_2) = a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1)$$

$$a_2 = \frac{f(x_2) - a_0 - a_1(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)}$$

$$a_2 = \frac{f(x_2)-a_0-a_1(x_2-x_0)}{(x_2-x_0)(x_2-x_1)}$$

$$= \frac{f(x_2)-f(x_1)+f(x_1)-f(x_0)-a_1(x_2-x_0)}{(x_2-x_0)(x_2-x_1)}$$

$$= \frac{f(x_2)-f(x_1)+a_1(x_1-x_0)-a_1(x_2-x_0)}{(x_2-x_0)(x_2-x_1)}$$

$$= \frac{\dfrac{f(x_2)-f(x_1)}{x_2-x_1}-a_1}{(x_2-x_0)} = \frac{\dfrac{f(x_2)-f(x_1)}{x_2-x_1}-\dfrac{f(x_1)-f(x_0)}{x_1-x_0}}{(x_2-x_0)}$$

依次递推，可以得到$a_3,\ldots,a_n$。

为写出系数a_k的一般形式，需引进差商定义。

定义4.3 设已知函数$f(x)$的n+1个互异节点$x_0,x_1,\ldots,x_n$上的函数值为$f(x_j)$ (j=0,1,…,n)，称

$$\frac{f(x_j)-f(x_i)}{x_j-x_i}$$

为$f(x)$关于节点x_i, x_j的**1阶差商或均差，**

记作$f[x_i, x_j]$，即

$$f[x_i,x_j]=\frac{f(x_j)-f(x_i)}{x_j-x_i}$$

称一阶差商$f[x_i, x_j]$和$f[x_j, x_k]$的差商 $\dfrac{f[x_j,x_k]-f[x_i,x_j]}{x_k-x_i}$

为$f(x)$关于节点x_i, x_j, x_k 的2阶差商或均差，

记作$f[x_i, x_j, x_k]$，即

$$f[x_i,x_j,x_k]=\frac{f[x_j,x_k]-f[x_i,x_j]}{x_k-x_i}$$

一般地，称k-1阶差商的差商为k阶差商或均差，即

$$f[x_0,x_1,\cdots,x_{k-1},x_k]$$
$$=\frac{f[x_1,x_2,\cdots,x_{k-1},x_k]-f[x_0,x_1,\cdots,x_{k-2},x_{k-1})}{x_k-x_0}$$

实际计算时，通常将差商排成表格形式, 称为**差商表。**
例如给定函数表

x	x_0	x_1	x_2	x_3
$f(x)$	$f(x_0)$	$f(x_1)$	$f(x_2)$	$f(x_3)$

相应的差商表为 $f[x_i,x_j]=\dfrac{f(x_j)-f(x_i)}{x_j-x_i}$　$f[x_i,x_j,x_k]=\dfrac{f[x_j,x_k]-f[x_i,x_j]}{x_k-x_i}$

x_k	$f(x_k)$	一阶差商	二阶差商	三阶差商
x_0	$f(x_0)$	$f[x_0,x_1]$	$f[x_0,x_1,x_2]$	$f[x_0,x_1,x_2,x_3]$
x_1	$f(x_1)$	$f[x_1,x_2]$	$f[x_1,x_2,x_3]$	
x_2	$f(x_2)$	$f[x_2,x_3]$		
x_3	$f(x_3)$			

$$f[x_0,x_1,\cdots,x_{k-1},x_k]=\frac{f[x_1,x_2,\cdots,x_{k-1},x_k]-f[x_0,x_1,\cdots,x_{k-2},x_{k-1}]}{x_k-x_0}$$

X和Y全抄下，分子前面下减上，分母X大减小！

4.3.2 牛顿插值公式

当把条件 $L_n(x_j)=f(x_j)$ (j=0,1,2,...,n)依次代入公式

$$L_n(x)=a_0+a_1(x-x_0)+a_2(x-x_0)(x-x_1)+\cdots$$
$$+a_n(x-x_0)(x-x_1)\cdots(x-x_{n-1})$$

得

$$\begin{cases} f(x_0)=L_n(x_0)=a_0 \\ f(x_1)=L_n(x_1)=a_0+a_1(x_1-x_0) \\ f(x_2)=L_n(x_2)=a_0+a_1(x_2-x_0)+a_2(x_2-x_0)(x_2-x_1) \\ \qquad\qquad\qquad\qquad\vdots \\ f(x_n)=L_n(x_n)=a_0+a_1(x_n-x_0)+a_2(x_n-x_0)(x_n-x_1) \\ +\cdots+a_n(x_n-x_0)(x_n-x_1)\cdots(x_n-x_{n-1}) \end{cases}$$

$$a_k = f[x_0, x_1, \cdots, x_{k-1}, x_k] \quad (k = 0, 1, 2, \cdots, n)$$

将a_k代入 $L_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \cdots$

$$+ a_n(x - x_0)(x - x_1)\cdots(x - x_{n-1})$$

可得n次差值多项式$L_n(x)$。这里将$L_n(x)$特记为$N_n(x)$，并称之为n次**牛顿（Newton）插值多项式**，即

$$N_n(x) = f(x_0) + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \cdots$$
$$+ f[x_0, x_1, \cdots, x_n](x - x_0)(x - x_1)\cdots(x - x_{n-1})$$

$$N_n(x)=f(x_0)+f[x_0,x_1](x-x_0)+f[x_0,x_1,x_2](x-x_0)(x-x_1)+\cdots$$
$$+f[x_0,x_1,\cdots,x_n](x-x_0)(x-x_1)\cdots(x-x_{n-1})$$

例4.3 给定如下数据

x_k	-2	0	1	3
$f(x_k)$	1	-1	2	4

试求$f(x_k)$的3次Newton插值多项式。

$$N_n(x)=f(x_0)+f[x_0,x_1](x-x_0)+f[x_0,x_1,x_2](x-x_0)(x-x_1)+\cdots$$
$$+f[x_0,x_1,\cdots,x_n](x-x_0)(x-x_1)\cdots(x-x_{n-1})$$

解 先构造差商表

x_k	$f(x_k)$	一阶差商	二阶差商	三阶差商
-2	1	-1	4/3	-2/5
0	-1	3	-2/3	
1	2	1		
3	4			

$$N_3(x)=1-(x+2)+\tfrac{4}{3}(x+2)x-\tfrac{2}{5}(x+2)x(x-1)$$

X和Y全抄下，分子前面下减上，分母X大减小！

4.3.3 牛顿插值余项

牛顿插值多项式$N_n(x)$与拉格朗日插值多项式$L_n(x)$满足同样的插值条件$L_n(x_j)=f(x_j)$ ($j = 0,1,\ldots,n$)，实质是同一多项式，因此余项相同，均可用公式表示为：

$$R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x)$$

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

定理4.3 满足插值条件$N_n(x_k) = f(x_k)$ ($k= 0,1,\ldots,n$)的n次Newton插值多项式$N_n(x)$的余项为

$$R_n(x) = f(x) - N_n(x) = f[x_0, x_1, \cdots, x_n, x]\omega_{n+1}(x)$$

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

$$R_n(x)=f(x)-N_n(x)=f[x_0,x_1,\cdots,x_n,x]\omega_{n+1}(x)$$

需要说明的是，式中的n+1阶差商$f[x_0,x_1,\ldots,x_n,x]$与$f(x)$的值（它正是我们要计算的）有关，故不可能准确地计算出$f[x_0,x_1,\ldots,x_n,x]$的精确值，只能对它作出一种估计。

4.3.4 差分以及等距节点牛顿插值多项式

上面讨论的是节点任意分布的Newton插值多项式。在实际应用中，有时碰到等距节点的情况，即节点为

$$x_i = x_0 + ih \ (i=0,1,\dots,n)$$

这时利用节点等距的特点，可以使**Newton公式简化。**

定义4.4 设函数$f(x)$在等距节点x_i上的函数值为$f(x_i)=f_i$ $(i=0,1,\dots,n)$，则称 $\boldsymbol{f_{i+1}-f_i}$ 为$f(x)$在x_i处以h为步长的**1阶向前差分**，**简称1阶差分**，记作 Δf_i

$$\Delta f_i = f_{i+1} - f_i$$

于是，函数$f(x)$在各节点处的一阶差分依次为 $\Delta f_0 = f_1 - f_0$ ，$\Delta f_1 = f_2 - f_1$ ，…，$\Delta f_{n-1} = f_n - f_{n-1}$ 。

又称一阶差分的差分

$$\Delta^2 f_k = \Delta(\Delta f_k) = \Delta f_{k+1} - \Delta f_k$$ **为二阶差分。**

一般地，称

$$\Delta^k f_i = \Delta^{k-1} f_{i+1} - \Delta^{k-1} f_i$$

为$f(x)$在点x_i处以h为步长的k阶向前差分，简称k阶差分

为了便于计算与应用，通常采用表格形式计算差分

x_k	f_k	Δf_k	$\Delta^2 f_k$	$\Delta^3 f_k$	$\Delta^4 f_k$
x_0	f_0	Δf_0	$\Delta^2 f_0$	$\Delta^3 f_0$	$\Delta^4 f_0$
x_1	f_1	Δf_1	$\Delta^2 f_1$	$\Delta^3 f_1$	
x_2	f_2	Δf_2	$\Delta^2 f_2$		
x_3	f_3	Δf_3			
x_4	f_4				

X和Y全抄下，分子前面下减上，分母就是1！

在等距节点$x_i=x_0+ih(i=0,1,\ldots,n)$ 情况下，可以利用差分表示牛顿插值多项式的系数，并将所得公式加以简化。

$$N_n(x)=a_0+a_1(x-x_0)+a_2(x-x_0)(x-x_1)+\cdots$$
$$+a_n(x-x_0)(x-x_1)\cdots(x-x_{n-1})$$

事实上，由插值条件$N_n(x_0)=f_0$立即可得

$$a_0=f_0$$

再由插值条件 $f_1=N_n(x_1)=a_0+a_1(x_1-x_0)$ 可得

$$a_1=\frac{f_1-f_0}{x_1-x_0}=\frac{\Delta f_0}{h}$$

由插值条件 $f_2 = N_n(x_2) = a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1)$

可得

$$a_2 = \frac{f_2 - f_0 - \frac{\Delta f_0}{h}(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} = \frac{f_2 - 2f_1 + f_0}{2h * h}$$

$$= \frac{f_2 - f_1 - (f_1 - f_0)}{2h * h} = \frac{\Delta^2 f_0}{2! \cdot h^2}$$

一般地，由插值条件$N_n(x_k)$=f_k可得 $a_k = \dfrac{\Delta^k f_0}{k! \cdot h^k}$ $(k = 1, 2, \cdots, n)$

于是，满足插值条件$N_n(x_i)$=f_i的插值多项式为

$$N_n(x) = f_0 + \frac{\Delta f_0}{h}(x - x_0) + \frac{\Delta^2 f_0}{2! \cdot h^2}(x - x_0)(x - x_1)$$

$$+ \cdots + \frac{\Delta^n f_0}{n! \cdot h^n}(x - x_0)(x - x_1)\cdots(x - x_{n-1})$$

令$x = x_0 + th$ ($t>0$), 并注意到$x_i=x_0+ih$($i=0,1,\ldots,n$), 则可简化为

$$N_n(x) = f_0 + \frac{\Delta f_0}{h}(x - x_0) + \frac{\Delta^2 f_0}{2!\cdot h^2}(x - x_0)(x - x_1)$$

$$+ \cdots + \frac{\Delta^n f_0}{n!\cdot h^n}(x - x_0)(x - x_1)\cdots(x - x_{n-1})$$

$$N_n(x_0 + th) = f_0 + t\Delta f_0 + \frac{t(t-1)}{2!}\Delta^2 f_0 + \cdots + \frac{t(t-1)\cdots(t-n+1)}{n!}\Delta^n f_0$$

$$= \sum_{k=0}^{n} \frac{\Delta^k f_0}{k!} \prod_{j=0}^{k-1}(t - j) \quad \text{其中} t = \frac{x - x_0}{h}, \Delta^0 f_0 = f_0$$

这个用向前差分表示的插值多项式，称为**牛顿向前插值公式。**

由插值余项公式，可得

$$R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x)$$

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

$$R_n(x_0+th) = \frac{t(t-1)\cdots(t-n)}{(n+1)!}h^{n+1}f^{(n+1)}(\xi) \qquad \xi \in (x_0, x_n)$$

例 从给定的正弦函数表出发计算sin(0.12)，并估计截断误差。

x	$\sin x$	Δf	$\Delta^2 f$	$\Delta^3 f$
0.1	<u>0.09983</u>	<u>0.09884</u>	<u>-0.00199</u>	<u>-0.00096</u>
0.2	0.19867	0.09685	-0.00295	-0.00094
0.3	0.29552	0.09390	-0.00389	-0.00091
0.4	0.38942	0.09001	-0.00480	
0.5	0.47943	0.08521		
0.6	0.56464			

取x_0=0.1,此时 $t=\dfrac{x-x_0}{h}=\dfrac{0.12-0.1}{0.1}=0.2$，构造差分表。

表中带下划线各数依次为sin$x$在x_0=0.1处的函数值和各阶差分。

 X和Y全抄下，分子前面下减上，分母就是1！

若用线性插值求sin(0.12)的近似值，则可得

$$\sin(0.12)\approx N_1(0.12)=0.09983+0.2\times 0.09884=0.11960$$

用二次插值得

$$\sin(0.12)\approx N_2(0.12)=0.09983+0.2\times 0.09884+\frac{0.2\times(0.2-1)}{2}\times(-0.00199)$$

$$=N_1(0.12)+0.00016=0.11976$$

用三次插值得

$$\sin(0.12)\approx N_3(0.12)$$

$$=N_2(0.12)+\frac{0.2\times(0.2-1)\times(0.2-2)}{6}\times(-0.00096)$$

$$=0.11971$$

$$N_n(x_0+th)=f_0+t\Delta f_0+\frac{t(t-1)}{2!}\Delta^2 f_0+\cdots+\frac{t(t-1)\cdots(t-n+1)}{n!}\Delta^n f_0$$

因$N_3(0.12)$与$N_2(0.12)$很接近，且由差分表可以看出，三阶差分接近于常数（即$\Delta^4 f_0$接近于零），故取$N_3(0.12)=0.11971$作为$\sin(0.12)$的近似值，此时由余项公式可知其截断误差

$$\left|R_3(0.12)\right| \le \left|\frac{0.2\times(0.2-1)\times(0.2-2)\times(0.2-3)}{24}\right| \times (0.1)^4 \times \sin(0.4)$$

$$< 0.000002$$

$$R_n(x_0+th) = \frac{t(t-1)\cdots(t-n)}{(n+1)!} h^{n+1} f^{(n+1)}(\xi) \qquad \xi \in (x_0, x_n)$$

n次牛顿（Newton）插值多项式

$$N_n(x) = f(x_0) + f[x_0,x_1](x-x_0) + f[x_0,x_1,x_2](x-x_0)(x-x_1) + \cdots + f[x_0,x_1,\cdots,x_n](x-x_0)(x-x_1)\cdots(x-x_{n-1})$$

差商表：$\boxed{\text{X和Y全抄下，分子前面下减上，分母X大减小！}}$

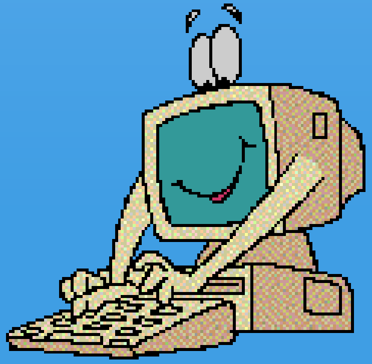
牛顿向前插值公式

$$N_n(x_0+th) = f_0 + t\Delta f_0 + \frac{t(t-1)}{2!}\Delta^2 f_0 + \cdots + \frac{t(t-1)\cdots(t-n+1)}{n!}\Delta^n f_0$$

$$= \sum_{k=0}^{n} \frac{\Delta^k f_0}{k!} \prod_{j=0}^{k-1}(t-j) \quad \text{其中} t = \frac{x-x_0}{h}, \Delta^0 f_0 = f_0$$

差分表：$\boxed{\text{X和Y全抄下，分子前面下减上，分母就是1！}}$

4.5 分段线性插值



4.5.1 高次插值的病态分析

适当地提高插值多项式的次数，有可能提高计算结果的准确程度。

但是决不可由此得出结论，认为插值多项式的次数越高越好，利用被插函数节点信息越多，误差越小。

由插值多项式的截断误差公式可见：

$$R_n(x) = f(x) - p_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x)$$

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

截断误差同$f^{(n+1)}(\xi)$与$\omega_{n+1}(x)$有关，其绝对值不一定随次数n增加而减小。

龙格现象

考虑一个典型的例子，设 $f(x)=\dfrac{1}{1+x^2} \qquad (-5\le x\le 5)$

取等距节点 $x_k=-5+\dfrac{10k}{n} \qquad (k=0,1,\cdots,n)$

所构造的拉格朗日插值多项式为

$$L_n(x)=\sum_{i=0}^{n}\frac{1}{1+x_i^2}\Box\prod_{\substack{j=0\\ j\ne i}}^{n}\frac{x-x_j}{x_i-x_j}$$

$$L_n(x)=\sum_{i=0}^{n}l_i(x)y_i=\sum_{i=0}^{n}y_i\frac{(x-x_0)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)}{(x_i-x_0)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)}$$
$$=\sum_{i=0}^{n}\Big(\prod_{\substack{j=0\\ j\ne i}}^{n}\frac{x-x_j}{x_i-x_j}\Big)y_i$$

在[-5，5]上取一点 $\tilde{x}=5-\frac{5}{n}$

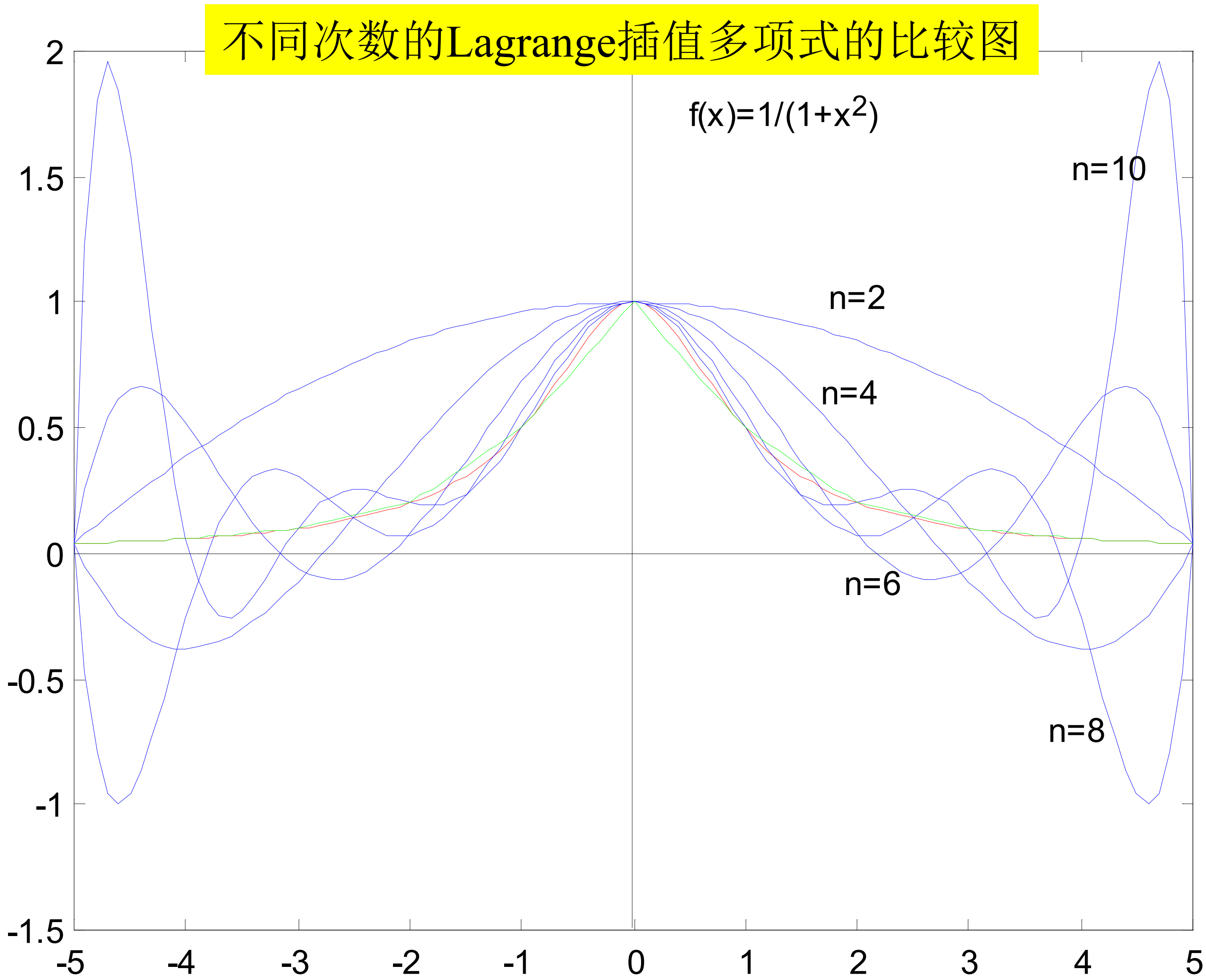
计算出n=2,4,…,20的 $L_n(\tilde{x})$和$R(\tilde{x})$

n	$f(\tilde{x})$	$L_n(\tilde{x})$	$R(\tilde{x})$
2	0.137931	0.759615	-0.621684
4	0.066390	-0.356826	0.423216
6	0.054463	0.607879	-0.553416
8	0.049651	-0.831017	0.880668
10	0.047059	1.578721	-1.531662
12	0.045440	-2.75500	2.800440
14	0.044334	50.332742	-5.288409
16	0.043530	-10.173867	10.217397
18	0.042920	20.123671	-20.080751
20	0.042440	-39.952449	39.994889

这个例子最早是在20世纪初由龙格(Runge)研究的。

这种节点变密但误差增大的现象称为龙格(Runge)现象。

随着n的增加，$|R_n(\tilde{x})|$ 不但没有减少，反而成倍地增加。



不同次数的Lagrange插值多项式的比较图
f(x)=1/(1+x2)
n=10
n=2
n=4
n=6
n=8
2
1.5
1
0.5
0
-0.5
-1
-1.5
-5
-4
-3
-2
-1
0
1
2
3
4
5

Runge现象

4.5.2 分段线性插值

当插值节点很多时，使用高次插值多项式未必能够得到好的效果。

我们介绍分段插值法，就是将插值区间分成若干个子区间，然后在每个子区间上使用线性插值多项式，

分段线性插值:通过插值点用折线段连接起来逼近$f(x)$。

设$f(x)$在$n+1$个节点 $a=x_0<x_1<x_2<\ldots<x_n=b$上的函数值为

$$f(x_k)\ (k=0,1,2,\ldots,n),$$

将$[a,b]$分成n个小区间，在每个小区间$[x_k, x_{k+1}]$上作线性插值，得

$$L_1(x)=f(x_k)\frac{x-x_{k+1}}{x_k-x_{k+1}}+f(x_{k+1})\frac{x-x_k}{x_{k+1}-x_k}\quad (x_k\le x\le x_{k+1})$$

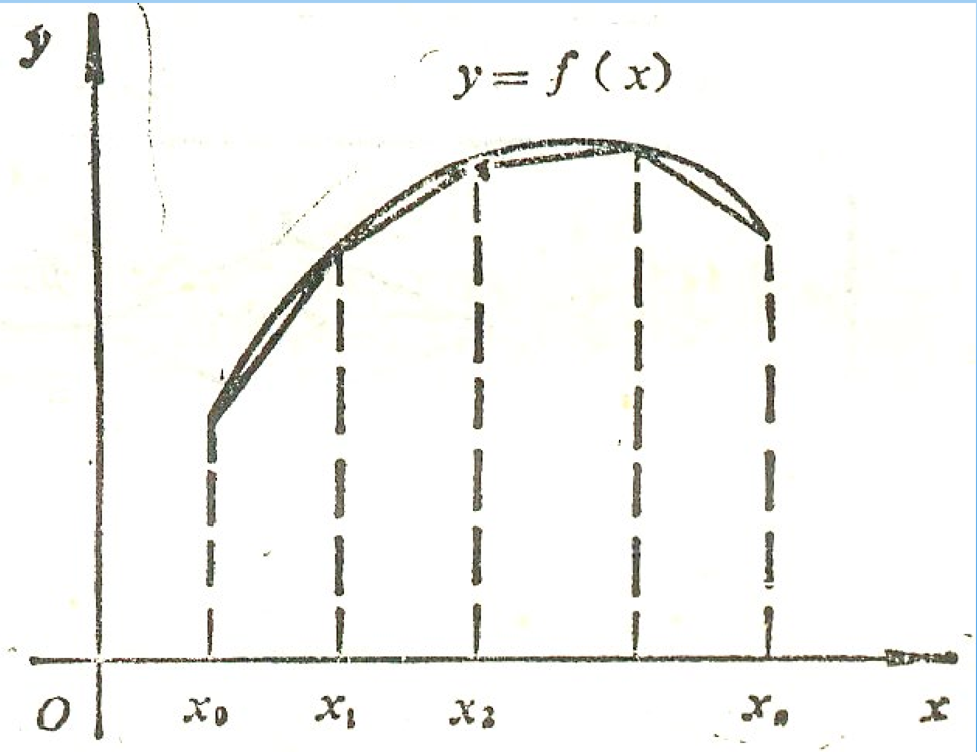
若要计算点$x\neq x_k$处函数值$f(x)$的近似值。可先选取两个节点x_k与x_{k+1}，使$x\in[x_k, x_{k+1}]$；然后在小区间上作线性插值，计算函数值。

$$f(x)\approx L_1(x)=f(x_k)\frac{x-x_{k+1}}{x_k-x_{k+1}}+f(x_{k+1})\frac{x-x_k}{x_{k+1}-x_k}$$

$$x\in[x_k,x_{k+1}]$$

$$L_n(x)=\sum_{i=0}^{n}l_i(x)y_i=\sum_{i=0}^{n}y_i\frac{(x-x_0)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)}{(x_i-x_0)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)}$$
$$=\sum_{i=0}^{n}\Big(\prod_{\substack{j=0\\ j\neq i}}^{n}\frac{x-x_j}{x_i-x_j}\Big)y_i$$

在几何上就是用通过曲线n+1个点(x_k, y_k)的折线，去近似代替曲线 。故分段线性插值又称**折线插值**。



分段线性插值的余项估计式：

$$f(x)-L_1(x)=\frac{1}{2}f''(\xi_k)(x-x_k)(x-x_{k+1})\quad(\xi_k\in(x_k,x_{k+1}))$$

注意在$[x_{i-1}, x_i]$上$\left|(x-x_k)(x-x_{k+1})\right|\le\frac{1}{4}h_k^2$，$h_k=x_{k+1}-x_k$

X轴之下的开口向上的抛物线

推得

$$\max_{x_k\le x\le x_{k+1}}\left|f(x)-L_1(x)\right|=\max_{x_k\le x\le x_{k+1}}\left|\frac{1}{2}f''(\xi_k)(x-x_k)(x-x_{k+1})\right|$$

$$\le\frac{1}{8}h_k^2\max_{x_k\le x\le x_{k+1}}\left|f''(x)\right|$$

$$R_1(x)=\frac{1}{2}f''(\xi)(x-x_0)(x-x_1)$$

$$\max_{a\le x\le b}\left|f(x)-L(x)\right| = \max_{x_0\le x\le x_n}\left|f(x)-L(x)\right|$$

$$= \max_{0\le k\le n-1}\max_{x_k\le x\le x_{k+1}}\left|f(x)-L(x)\right|$$

$$= \max_{0\le k\le n-1}\max_{x_k\le x\le x_{k+1}}\left|f(x)-L_1(x)\right|$$

$$\le \max_{0\le k\le n-1}\tfrac{1}{8}h_k^2\max_{x_k\le x\le x_{k+1}}\left|f''(x)\right|$$

$$\le \tfrac{1}{8}h^2\max_{a\le x\le b}\left|f''(x)\right| \qquad (h=\max_{0\le k\le n-1}h_k)$$

可见，分段线性插值的余项只依赖于二阶导数的界。

这表明，只要小区间长度h_k充分小，便可保证 $L_1(x)$ 充分靠近$f(x)$，即 $h=\max\limits_{1\le k\le n}h_k \to 0$时分段线性插值函数$L_1(x)$收敛于被插函数$f(x)$。

4.7　曲线拟合的最小二乘法

在科学实验和生产实践中，经常要从一组实验数据（x_i,y_i）(i=1,2,…,m)出发，寻求函数$y=f(x)$的一个近似表达式$y=p(x)$（称为经验公式）。从几何上看，就是希望根据给定的m个点（x_i,y_i），求曲线$y=f(x)$的一条近似曲线$y=p(x)$。因此，这是一个曲线拟合的问题。

多项式插值虽然在一定程度上解决了由函数表求函数的近似表达式问题，但用它来解决这里提出的问题，是有明显缺陷的。

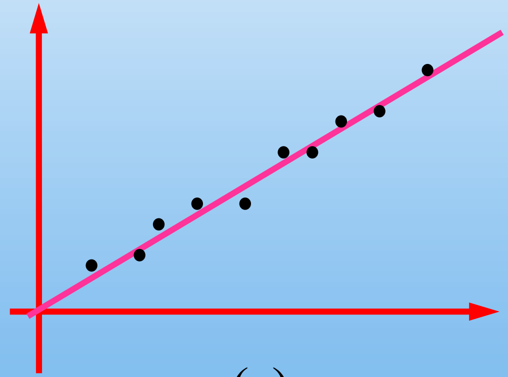
多项式插值的缺陷

由实验提供的数据通常带有测试误差。如果要求近似曲线$y=p(x)$严格地通过所给的每个数据点（x_i,y_i），就会使曲线保留着原有的测试误差。当个别数据的误差较大时，插值效果显然是不理想的。

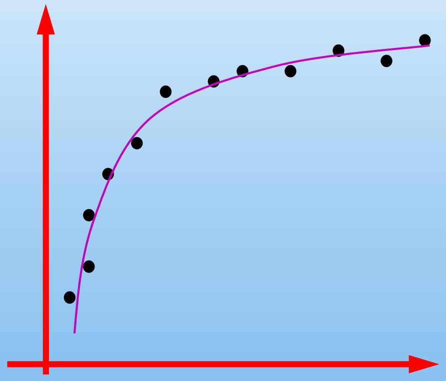
怎样从给定的一组实验数据出发，在某个函数类φ中寻求一个“最好”的函数$p(x)$来拟合这组数据，是一个值得讨论的课题。

随着拟合效果“好”、“坏”标准的不同，解决此类问题的方法也就不同。这里，将介绍一种最常用的曲线拟合方法，即最小二乘法。

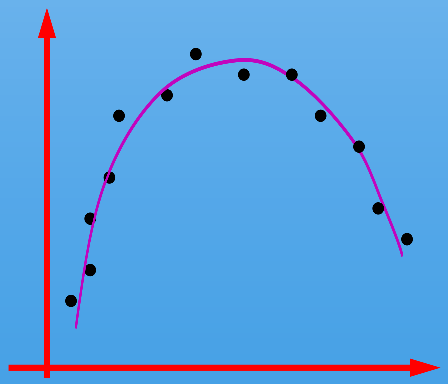
例



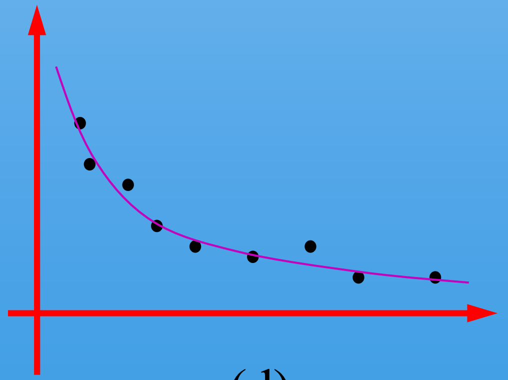
(a)



(c)

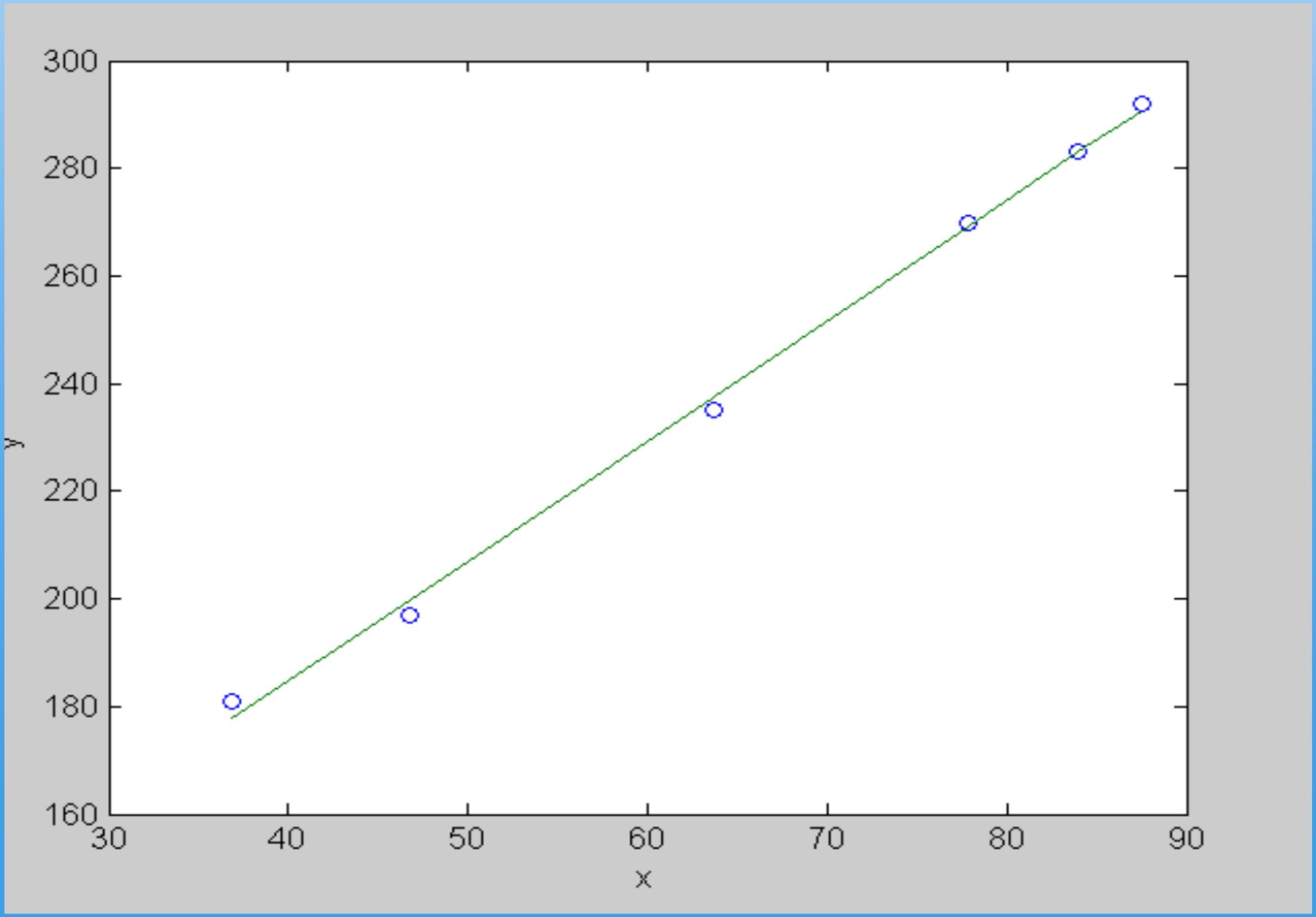


(b)



(d)

多项式插值：必须通过所有的数据点。
最小二乘法：不需要通过所有的数据点，而是数据点误差之和最小。



什么是最小二乘法

在一般情况下，不能要求近似曲线$y=p(x)$严格地通过所有数据点（x_i,y_i），亦即不能要求拟合函数在x_i处的残差（亦称偏差）

$$\delta_i = p(x_i) - y_i \quad (i = 1,2,\cdots,m)$$

都严格地等于零。但是，为了使近似曲线能尽量反映所给数据点的变化趋势，要求$|\delta_i|$都较小还是需要的。达到这一目标的途径很多，常见的有：

1）选取$p(x)$,使残差绝对值之和最小，即

$$\sum_{i=1}^{m} |\delta_i| = \sum_{i=1}^{m} |p(x_i) - y_i| = \min$$

（2）选取$p(x)$, 使残差最大绝对值最小，即

$$\max_{1\le i\le m}|\delta_i| = \max_{1\le i\le m}|p(x_i)-y_i| = \min$$

（3）选取$p(x)$,使残差平方和s最小，即

$$s=\sum_{i=1}^{m}\delta_i^2=\sum_{i=1}^{m}\left[p(x_i)-y_i\right]^2=\min$$

为了便于计算、分析与应用，较多地根据**“使残差平方和最小”的原则（称为最小二乘原则）**来选取拟合曲线$y=p$（x）。

按最小二乘原则选择拟合曲线的方法，称为**最小二乘法**（Least-Squares Method）。

p（x）称为m个数据（x_i,y_i）$(i=1,2,\ldots,m)$的**最小二乘法拟合函数**。$y\approx p(x)$近似反映了变量x和y之间的函数关系 $y=f(x)$，称为**经验公式或数学模型**。$f(x)$称为**被拟合函数**。

定义4.7 设$x_1, x_2, \ldots, x_n$为互不相同的点，$\varphi_0(x), \varphi_1(x), \cdots, \varphi_m(x)$是$m+1$个已知函数。如果存在不全为零的常数$C_0, C_1, \cdots, C_m$，使得

$$C_0\varphi_0(x_j)+C_1\varphi_1(x_j)+\cdots+C_m\varphi_m(x_j)=0 \quad (j=1,2,\cdots,n),$$

则称函数$\varphi_0(x), \varphi_1(x), \cdots, \varphi_m(x)$（关于节点$x_1, x_2, \ldots, x_n$）是**线性相关**的，否则称为**线性无关**。

简单的说，如果线性相关，则其中一定有某一函数可以用其他函数的线性关系来表示

例如，在$(-\infty,+\infty)$中函数系

$$\left\{x^j\right\}_{j=0}^{n}, \left\{\cos kx\right\}_{k=0}^{n}, \left\{\sin kx\right\}_{k=1}^{n}$$

以及 $1, \cos x, \sin x, \cos 2x, \cdots, \cos nx, \sin nx$

都是线性无关的

定义4.8 给定数据(x_j,y_j)(j=1,2,…,n)。

x	x_1	x_2	…	x_n
$y=f(x)$	y_1	y_2	…	y_n

设拟合函数的形式为

$$P(x)=a_0\varphi_0(x)+a_1\varphi_1(x)+\cdots+a_m\varphi_m(x) \qquad (4.62)$$

其中$\varphi_k(x)(k=0,\cdots,m)$为已知的线性无关函数，求系数$a_0,a_1,\cdots,a_m$，使得

$$\varphi(a_0,a_1,\cdots,a_m)=\sum_{j=1}^{n}\left[P(x_j)-y_j\right]^2=\sum_{j=1}^{n}\left[\sum_{k=0}^{m}a_k\varphi_k(x_j)-y_j\right]^2 \qquad (4.63)$$

最小

如果

$$\sum_{j=1}^{n}\left[\sum_{k=0}^{m} a_k^{*}\varphi_k(x_j)-y_j\right]^2=\min_{\substack{a_k\in R\\0\le k\le m}}\sum_{j=1}^{n}\left[\sum_{k=0}^{m} a_k\varphi_k(x_j)-y_j\right]^2 \quad (4.64)$$

则称相应的

$$P(x)=a_0^{*}\varphi_0(x)+a_1^{*}\varphi_1(x)+\cdots+a_m^{*}\varphi_m(x)$$

为**最小二乘拟合函数**

特别地，若

$$P(x)=a_0^{*}+a_1^{*}x+a_2^{*}x^2\cdots+a_m^{*}x^m$$

则称$P(x)$为$\boldsymbol{m}$**次最小二乘拟合多项式**。

用最小二乘法解决实际问题包含两个基本环节：

（1）先根据所给数据点的变化趋势与问题实际背景确定线性无关的函数 $\varphi_k(x)(k=0,\cdots,m)$，即确定 $P(x)$ 所具有的形式。

确定$\boldsymbol{\varphi_k(x)}$的形式

↓

确定$\boldsymbol{a_k}^*$

（2）然后按最小二乘原则(残差平方和S最小)

$$S=\sum_{j=1}^{n}\delta_j^2=\sum_{j=1}^{n}\left[P(x_j)-y_j\right]^2=\min$$

求取最小二乘解 $P^*(x)$，即确定其系数 $a_k^*(k=0,1,\cdots,m)$。

则可以得到拟合曲线

$$P(x)=a_0^*\varphi_0(x)+a_1^*\varphi_1(x)+\cdots+a_m^*\varphi_m(x)$$

$$=\sum_{k=0}^{m}a_k^*\varphi_k(x)$$

最小二乘解的求法

用求多元函数极值的方法求最小点$(a_0^*, a_1^*, \cdots a_m^*)$

最小二乘解$(a_0^*, a_1^*, \cdots a_m^*)$应满足条件

$$\sum_{j=1}^{n}\left[\sum_{k=0}^{m} a_k^*\varphi_k(x_j) - y_j\right]^2 = \min_{\substack{a_k \in R \\ 0 \le k \le m}} \sum_{j=1}^{n}\left[\sum_{k=0}^{m} a_k\varphi_k(x_j) - y_j\right]^2 \qquad (4.64)$$

得: 点$(a_0^*, a_1^*, \cdots a_m^*)$是多元函数

$$\varphi(a_0, a_1, \cdots, a_m) = \sum_{j=1}^{n}\left[P(x_j) - y_j\right]^2 = \sum_{j=1}^{n}\left[\sum_{k=0}^{m} a_k\varphi_k(x_j) - y_j\right]^2 \quad (4.63)$$

的极小点。

由多元函数取极值的必要条件知,

$(a_0^*, a_1^*, \cdots a_m^*)$ 满足方程组 $\dfrac{\partial\varphi(a_0, a_1, \cdots, a_m)}{\partial a_k} = 0 \qquad (k = 0, 1, \cdots, m)$

将式(4.63)两边对a_k求偏导，并令

$$\frac{\partial\varphi}{\partial a_k}=\frac{\partial}{\partial a_k}\{\sum_{j=1}^{n}[P(x_j)-y_j]^2\}=2\sum_{j=1}^{n}[P(x_j)-y_j]\frac{\partial P(x_j)}{\partial a_k}$$

$$=2\sum_{j=1}^{n}\left\{[\sum_{i=0}^{m}a_i^*\varphi_i(x_j)-y_j]\varphi_k(x_j)\right\}=0 \qquad (k=0,1,\ldots,m)$$

即

$$\sum_{j=1}^{n}\varphi_k(x_j)\left[a_0^*\varphi_0(x_j)+a_1^*\varphi_1(x_j)+\cdots+a_m^*\varphi_m(x_j)-y_j\right]=0$$

亦即

$$a_0^*\sum_{j=1}^{n}\varphi_k(x_j)\varphi_0(x_j)+a_1^*\sum_{j=1}^{n}\varphi_k(x_j)\varphi_1(x_j)+\cdots+a_m^*\sum_{j=1}^{n}\varphi_k(x_j)\varphi_m(x_j)=\sum_{j=1}^{n}\varphi_k(x_j)y_j$$

$$\varphi(a_0,a_1,\cdots,a_m)=\sum_{j=1}^{n}\left[P(x_j)-y_j\right]^2=\sum_{j=1}^{n}\left[\sum_{k=0}^{m}a_k\varphi_k(x_j)-y_j\right]^2 \quad (4.63)$$

$$a_0^*\sum_{j=1}^{n}\varphi_k(x_j)\varphi_0(x_j)+a_1^*\sum_{j=1}^{n}\varphi_k(x_j)\varphi_1(x_j)+\cdots+a_m^*\sum_{j=1}^{n}\varphi_k(x_j)\varphi_m(x_j)=\sum_{j=1}^{n}\varphi_k(x_j)y_j$$

即

$$\sum_{i=0}^{m}\left[\sum_{j=1}^{n}\varphi_i(x_j)\varphi_k(x_j)\right]a_i^*=\sum_{j=1}^{n}y_j\varphi_k(x_j)\quad(k=0,1,\cdots,m)\qquad(4.65)$$

为了将上式表达得更简洁，引进**内积记号**，在线性代数中，R^n中两个向量$\mathbf{x}=(x_1,\cdots,x_n)^T$和$\mathbf{y}=(y_1,\cdots,y_n)^T$的内积定义为

$$(\mathbf{x},\mathbf{y})=x_1y_1+\cdots+x_ny_n$$

将它加以推广有下面的定义。

定义4.9 设$u(x)$与$v(x)$是两个已知函数，记

$$\mathbf{u}=(u(x_1),u(x_2),\cdots,u(x_n))^T,\mathbf{v}=(v(x_1),v(x_2),\cdots,v(x_n))^T,$$ 令

$$(\mathbf{u},\mathbf{v})=\sum_{j=1}^{n}u(x_j)v(x_j) \qquad (4.66)$$

称之为**u和v的内积**。

内积的性质：

1) $(\mathbf{u},\mathbf{u})\geq 0$。 $(\mathbf{u},\mathbf{u})=0$当且仅当$\mathbf{u}=0$;

2) $(\mathbf{u},\mathbf{v})=(\mathbf{v},\mathbf{u})$;

3) $(\mathbf{u}+\mathbf{v},\mathbf{w})=(\mathbf{u},\mathbf{w})+(\mathbf{v},\mathbf{w})$;

其中$\mathbf{w}=(w(x_1),w(x_2),\cdots,w(x_n))^T$

4) $(\alpha\mathbf{u},\mathbf{v})=\alpha(\mathbf{u},\mathbf{v})$（$\alpha$为任意实数）

利用内积的定义，则式

$$a_0^*\sum_{j=1}^{n}\varphi_k(x_j)\varphi_0(x_j)+a_1^*\sum_{j=1}^{n}\varphi_k(x_j)\varphi_1(x_j)+\cdots+a_m^*\sum_{j=1}^{n}\varphi_k(x_j)\varphi_m(x_j)=\sum_{j=1}^{n}\varphi_k(x_j)y_j$$

$$\Longrightarrow a_0^*(\varphi_k,\varphi_0)+a_1^*(\varphi_k,\varphi_1)+\cdots a_m^*(\varphi_k,\varphi_m)=(y,\varphi_k)$$

其中 $$(k=0,1,\cdots,m)$$

$$\varphi_0=\begin{bmatrix}\varphi_0(x_1)\\\varphi_0(x_2)\\\vdots\\\varphi_0(x_{n-1})\\\varphi_0(x_n)\end{bmatrix}\ \varphi_1=\begin{bmatrix}\varphi_1(x_1)\\\varphi_1(x_2)\\\vdots\\\varphi_1(x_{n-1})\\\varphi_1(x_n)\end{bmatrix},\cdots,\varphi_m=\begin{bmatrix}\varphi_m(x_1)\\\varphi_m(x_2)\\\vdots\\\varphi_m(x_{n-1})\\\varphi_m(x_n)\end{bmatrix},y=\begin{bmatrix}y_1\\y_2\\\vdots\\y_{n-1}\\y_n\end{bmatrix}$$

$$a_0^*(\varphi_k,\varphi_0)+a_1^*(\varphi_k,\varphi_1)+\cdots a_m^*(\varphi_k,\varphi_m)=(y,\varphi_k)$$

$$(k=0,1,\cdots,m)$$

写成矩阵形式即

$$\begin{bmatrix} (\varphi_0,\varphi_0) & (\varphi_0,\varphi_1) & \cdots & (\varphi_0,\varphi_m) \\ (\varphi_1,\varphi_0) & (\varphi_1,\varphi_1) & \cdots & (\varphi_1,\varphi_m) \\ \cdots & \cdots & \cdots & \cdots \\ (\varphi_m,\varphi_0) & (\varphi_m,\varphi_1) & \cdots & (\varphi_m,\varphi_m) \end{bmatrix} \begin{bmatrix} a_0^* \\ a_1^* \\ \vdots \\ a_m^* \end{bmatrix} = \begin{bmatrix} (y,\varphi_0) \\ (y,\varphi_1) \\ \vdots \\ (y,\varphi_m) \end{bmatrix} \qquad (4.67)$$

法方程组或正规方程组, 其中系数矩阵是对称的。

当函数$\varphi_0(x),\varphi_1(x),\cdots,\varphi_m(x)$线性无关时，方程组是对称正定的，因此有唯一解。

求出(4.67)的解后，代入(4.62)即可得最小二乘拟合函数。

$$P(x)=a_0\varphi_0(x)+a_1\varphi_1(x)+\cdots+a_m\varphi_m(x) \qquad (4.62)$$

代数多项式拟合

作为曲线拟合的一种常用的情况，若讨论的是代数多项式拟合，

即取 $\varphi_0(x)=1$，$\varphi_1(x)=x$，$\cdots$，$\varphi_m(x)=x^m$

则由 $(\mathbf{u},\mathbf{v})=\sum_{j=1}^{n}u(x_j)v(x_j)$ 知

$$(\varphi_i,\varphi_k)=\sum_{j=1}^{n}x_j^i x_j^k=\sum_{j=1}^{n}x_j^{i+k}\qquad(i,k=0,1,\cdots,m)$$

$$(y,\varphi_k)=\sum_{j=1}^{n}x_j^k y_j\qquad(k=0,1,\cdots,m)$$

$$\begin{bmatrix}(\varphi_0,\varphi_0) & (\varphi_0,\varphi_1) & \cdots & (\varphi_0,\varphi_m)\\ (\varphi_1,\varphi_0) & (\varphi_1,\varphi_1) & \cdots & (\varphi_1,\varphi_m)\\ \cdots & \cdots & \cdots & \cdots\\ (\varphi_m,\varphi_0) & (\varphi_m,\varphi_1) & \cdots & (\varphi_m,\varphi_m)\end{bmatrix}\begin{bmatrix}a_0^*\\ a_1^*\\ \vdots\\ a_m^*\end{bmatrix}=\begin{bmatrix}(y,\varphi_0)\\ (y,\varphi_1)\\ \vdots\\ (y,\varphi_m)\end{bmatrix}\qquad(4.67)$$

$$\sum_{j=1}^{n}x_j^0$$

故相应的法方程组为

$$\begin{bmatrix} n & \sum_{j=1}^{n}x_j & \cdots & \sum_{j=1}^{n}x_j^m\\ \sum_{j=1}^{n}x_j & \sum_{j=1}^{n}x_j^2 & \cdots & \sum_{j=1}^{n}x_j^{m+1}\\ \cdots & \cdots & \cdots & \cdots\\ \sum_{j=1}^{n}x_j^m & \sum_{j=1}^{n}x_j^{m+1} & \cdots & \sum_{j=1}^{n}x_j^{2m}\end{bmatrix}\begin{bmatrix}a_0\\ a_1\\ \vdots\\ a_m\end{bmatrix}=\begin{bmatrix}\sum_{j=1}^{n}y_j\\ \sum_{j=1}^{n}x_jy_j\\ \vdots\\ \sum_{j=1}^{n}x_j^m y_j\end{bmatrix}$$

$$P(x) = a_0 + a_1 x + \cdots + a_m x^m$$

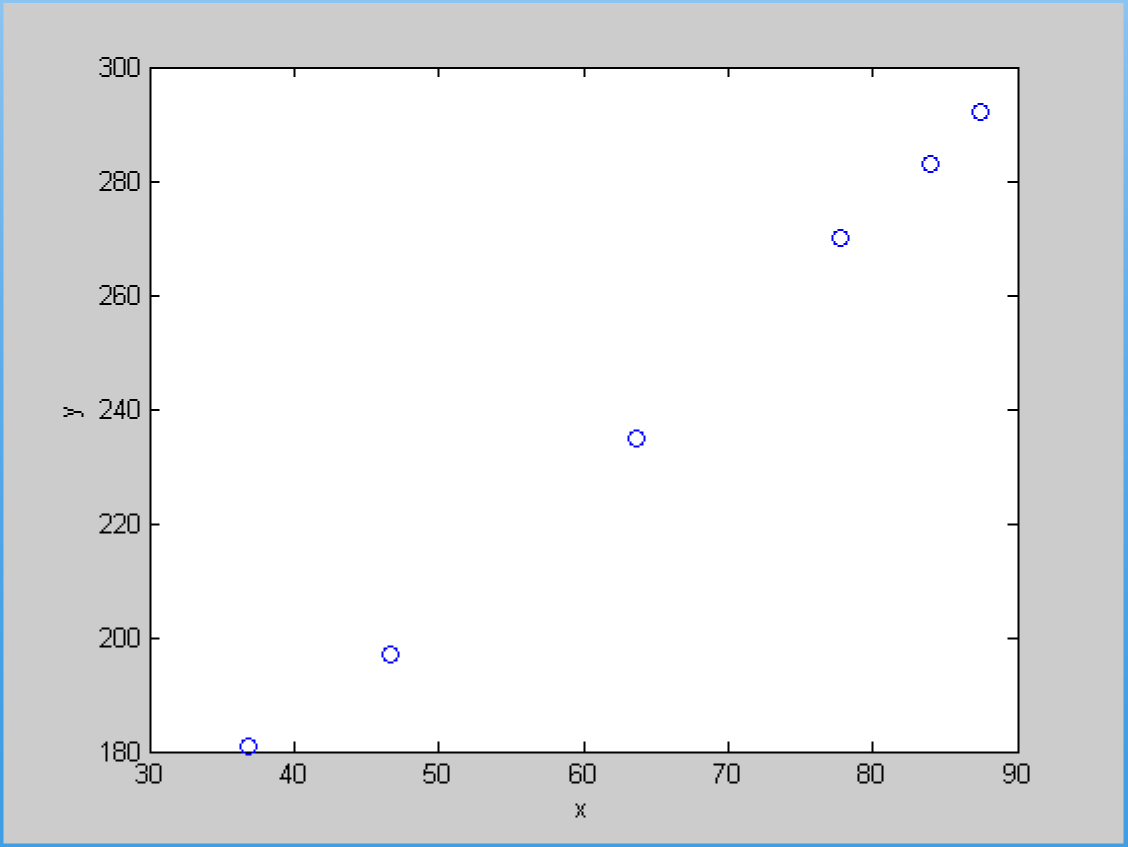
$$\begin{bmatrix} n & \sum_{j=1}^{n} x_j & \cdots & \sum_{j=1}^{n} x_j^m \\ \sum_{j=1}^{n} x_j & \sum_{j=1}^{n} x_j^2 & \cdots & \sum_{j=1}^{n} x_j^{m+1} \\ \cdots & \cdots & \cdots & \cdots \\ \sum_{j=1}^{n} x_j^m & \sum_{j=1}^{n} x_j^{m+1} & \cdots & \sum_{j=1}^{n} x_j^{2m} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{n} y_j \\ \sum_{j=1}^{n} x_j y_j \\ \vdots \\ \sum_{j=1}^{n} x_j^m y_j \end{bmatrix}$$

例 1 某种铝合金的含铝量为 x%，其熔解温度为 y℃，由实验测得 x 与 y 的数据如表所示。试用最小二乘法建立 x 与 y 之间的经验公式。

i	x_i	y_i	x_i^2	x_iy_i
1	36.9	181	1361.61	6678.9
2	46.7	197	2180.89	9199.9
3	63.7	235	4057.69	14969.5
4	77.8	270	6052.84	21006.0
5	84.0	283	7056.00	23772.0
6	87.5	292	7656.25	25550.0
Σ	396.6	1458	28365.28	101176.3

解 根据前面讨论，解决问题的过程如下：

（1）将表中给出的数据点（x_i, y_i）（i=1, 2, ⋯, 6）描绘在坐标纸上，如图所示。



（2）确定拟合曲线的形式。由图可以看出，六个点位于一条直线的附近，故可以选用线性函数（直线）来拟合这组实验数据，即令

$$P(x) = a_0 + a_1 x + \cdots + a_m x^m$$

$$\varphi(x) = a + bx$$

其中 a、b 为待定常数。

（3）建立法方程组。由于问题归结为一次多项式拟合问题，相应的法方程组形如

$$\begin{bmatrix} 6 & \sum_{i=1}^{6} x_i \\ \sum_{i=1}^{6} x_i & \sum_{i=1}^{6} x_i^2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{6} y_i \\ \sum_{i=1}^{6} x_i y_i \end{bmatrix}$$

$$\begin{bmatrix} n & \sum_{j=1}^{n} x_j & \cdots & \sum_{j=1}^{n} x_j^m \\ \sum_{j=1}^{n} x_j & \sum_{j=1}^{n} x_j^2 & \cdots & \sum_{j=1}^{n} x_j^{m+1} \\ \cdots & \cdots & \cdots & \cdots \\ \sum_{j=1}^{n} x_j^m & \sum_{j=1}^{n} x_j^{m+1} & \cdots & \sum_{j=1}^{n} x_j^{2m} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{n} y_j \\ \sum_{j=1}^{n} x_j y_j \\ \vdots \\ \sum_{j=1}^{n} x_j^m y_j \end{bmatrix}$$

（4）经过计算，即得确定待定数 a、b 的法方程组

$$\begin{cases} 6a + 396.6b = 1458 \\ 396.6a + 28365.28b = 101176.3 \end{cases}$$

解得

$$a=95.3524, \qquad b=2.2337$$

将所得结果代入 $\varphi(x) = a + bx$ 即得经验公式

$$y = 95.3524+2.2337x$$

(5) 检验：

所得经验公式能否较好地反映客观规律，还需通过实践来检验。

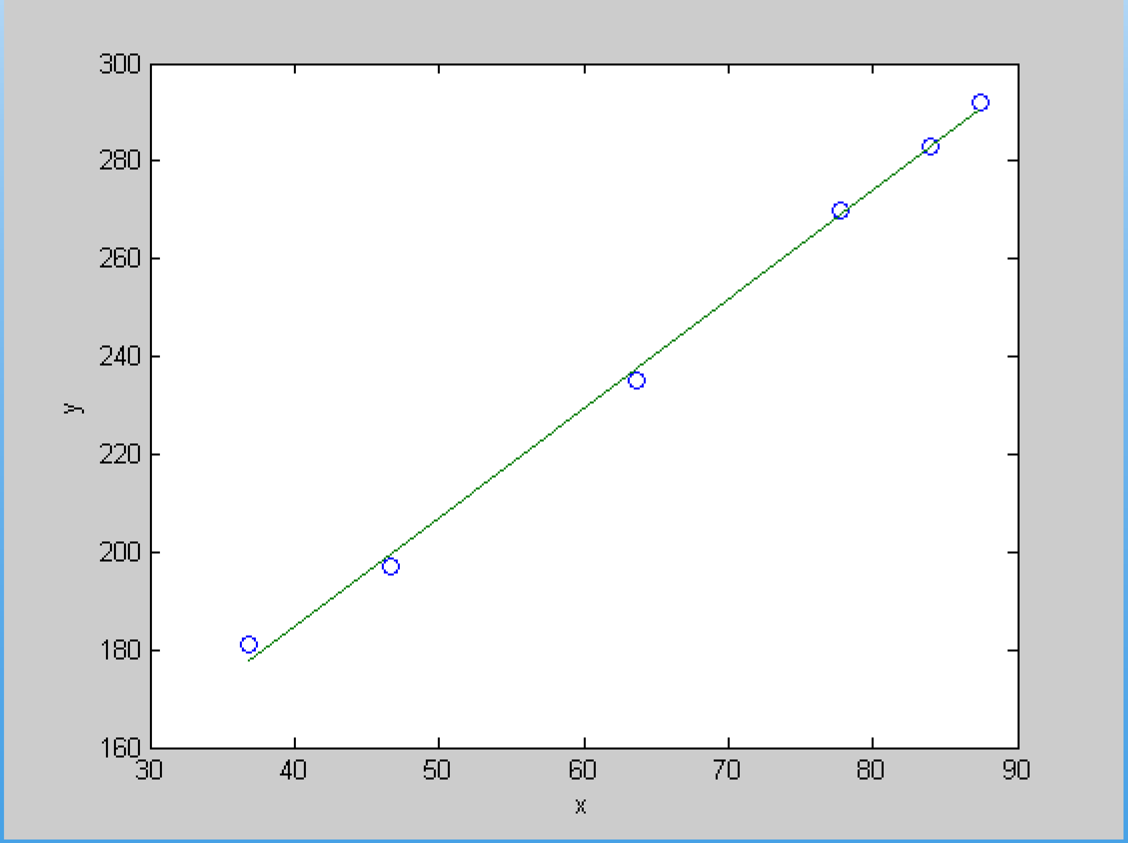
由经验公式算出的函数值（称为拟合值）

$$\tilde{y}_i = 95.3524 + 2.2337 x_i \qquad (i = 1, 2, \cdots, n)$$

与实测值之间有一定偏差。

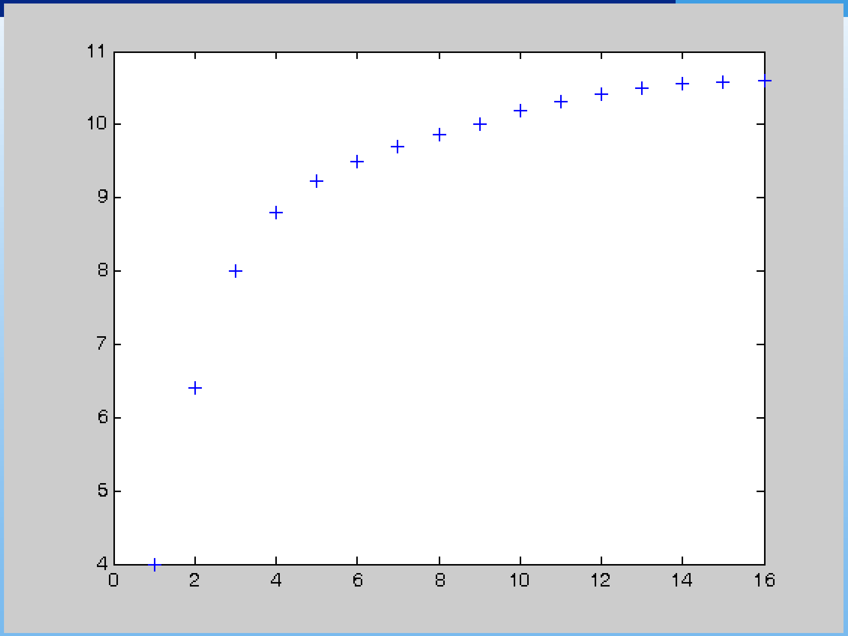
i	1	2	3	4	5	6
x_i	36.9	46.7	63.7	77.8	84.0	87.5
$\tilde{y}_i$	177.8	199.67	237.64	269.13	282.98	290.80
y_i	181	197	235	270	283	292
δ_i	−3.22	2.67	2.64	−0.87	−0.02	−1.20
δ_i^2	10.37	7.13	6.97	0.76	0.0004	1.44
$\sum\delta_i^2$	26.6704					

由上表可以看出，偏差的平方和$\sum_{i=1}^{6}\delta_i^2 = 26.6704$，它在一定程度上反映了所得经验公式的好坏。如果不满足要求，就要用改变函数类型或者增加实验数据等办法来建立新的经验公式。


300
280
260
240
220
200
180
160
y
30
40
50
60
70
80
90
x

例 2 在某化学反应里，测得生成物浓度 y%与时间 t（分）的数据如下试用最小二乘法建立 t 与 y 之间的经验公式。

t	1	2	3	4	5	6	7	8
y	4.0	6.4	8.0	8.80	9.22	9.50	9.70	9.86
t	9	10	11	12	13	14	15	16
y	10.0	10.2	10.32	10.42	10.50	10.55	10.58	10.60



解 将已知数据点(t_i, y_i) $(i=1, 2, \cdots, 16)$描绘在坐标纸上，如上图

由图及问题的物理背景可以看出，

拟合曲线 $y = \varphi(t)$ 应具有下列特点：

（1）曲线随着 t 的增加而上升，但上升速度由快到慢

当 t=0 时，反应尚未开始，即 y=0；

（2）当 $t\to\infty$时，y 趋于某一常数。故曲线应通过原点

（或者当 $t\to 0$ 时以原点为极限点），且有一水平渐近线。

具有上述特点的曲线很多。选用不同的数学模型，可以获得不同的拟合曲线与经验公式。下面提供两种方案。

方案 1：设想 $y=\varphi(t)$ 是双曲线型的，并且具有下面的形式

$$y=\frac{t}{at+b}$$

此时，若直接按最小二乘原则去确定参数 a 和 b，则问题归结为求二元函数

$$S(a,b)=\sum_{i=1}^{16}\left(\frac{t_i}{at_i+b}-y_i\right)^2$$ 的极小点。

$$\varphi(a_0,a_1,\cdots,a_m)=\sum_{j=1}^{n}\left[P(x_j)-y_j\right]^2=\sum_{j=1}^{n}\left[\sum_{k=0}^{m}a_k\varphi_k(x_j)-y_j\right]^2 \quad (4.63)$$

要求解 $S(a,b)=\sum_{i=1}^{16}(\frac{t_i}{at_i+b}-y_i)^2$ 的极小值问题将导致求解

非线性方程组

$$\begin{cases}\dfrac{\partial S}{\partial a}=-2\sum_{i=1}^{16}\dfrac{t_i^2}{(at_i+b)^2}(\dfrac{t_i}{at_i+b}-y_i)=0\\ \dfrac{\partial S}{\partial b}=-2\sum_{i=1}^{16}\dfrac{t_i}{(at_i+b)^2}(\dfrac{t_i}{at_i+b}-y_i)=0\end{cases}$$ 给计算带来了麻烦。

通过变量替换将它转化为关于待定参数的线性函数。

为此，将 $y=\frac{t}{at+b}$ 改写成：$\frac{1}{y}=a+\frac{b}{t}$。

于是，若引入新变量：

$y^{(1)}=\frac{1}{y}$，$t^{(1)}=\frac{1}{t}$，则 $\frac{1}{y}=a+\frac{b}{t}$ 就是：$y^{(1)}=a+bt^{(1)}$

t	1	2	3	4	5	6	7	8
y	4.0	6.4	8.0	8.80	9.22	9.50	9.70	9.86
t	9	10	11	12	13	14	15	16
y	10.0	10.2	10.32	10.42	10.50	10.55	10.58	10.60

由题中所给数据表可以算出新的数据表。这样，问题就归结为：根据新表，求形如 $y^{(1)} = a + bt^{(1)}$ 的最小二乘解。

i	1	2	3	…	16
$t_i^{(1)} = \dfrac{1}{t_i}$	1.00000	0.50000	0.33333	…	0.06250
$y_i^{(1)} = \dfrac{1}{y_i}$	0.25000	0.15625	0.12500	…	0.09434

参照例 1 的做法，解法方程组

$$\begin{bmatrix} 16 & \sum_{i=1}^{16} t_i^{(1)} \\ \sum_{i=1}^{16} t_i^{(1)} & \sum_{i=1}^{16} (t_i^{(1)})^2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{16} y_i^{(1)} \\ \sum_{i=1}^{16} t_i^{(1)} y_i^{(1)} \end{bmatrix}$$

即得

a=0.0801744603,　　b=0.1627225447

代入 $y = \dfrac{t}{at+b}$，得经验公式

$$y = \frac{t}{0.0801744603t + 0.1627225447}$$

方案 2：设想 $y=\varphi(t)$ 具有指数形式

$$y = ae^{b/t} \qquad (a>0, b<0)$$

为了在求取参数 a 和 b 时，避免求解一个非线性方程组，对上式两边取对数

$$\ln y = \ln a + \frac{b}{t}$$

此时若引入新变量

$$y^{(2)} = \ln y\text{，}\quad t^{(2)} = \frac{1}{t}$$

并记 A=ln$a$，B=b，则上式就是

$$y^{(2)} = A + Bt^{(2)}$$

由原始数据表可算出新的数据表

i	1	2	3	…	16
$t_i^{(2)}=1/t_i$	1. 00000	0. 50000	0. 33333	…	0. 06250
$y_i^{(2)}=\ln y_i$	1. 38629	1. 85630	2. 07944	…	2. 36085

于是将问题归结为：根据新数据表，求形如 $y^{(2)}=A+Bt^{(2)}$ 的最小二乘解。参照方案 1，写出相应的法方程组并解之，即得

A=2.427033135　　　B=-1.056683779

于是 $a=e^{A}$=11.325231，$b=$ B= -1.056683779

故得另一个经验公式

$$y=11.3252317e^{-\frac{1.056683779}{t}}$$

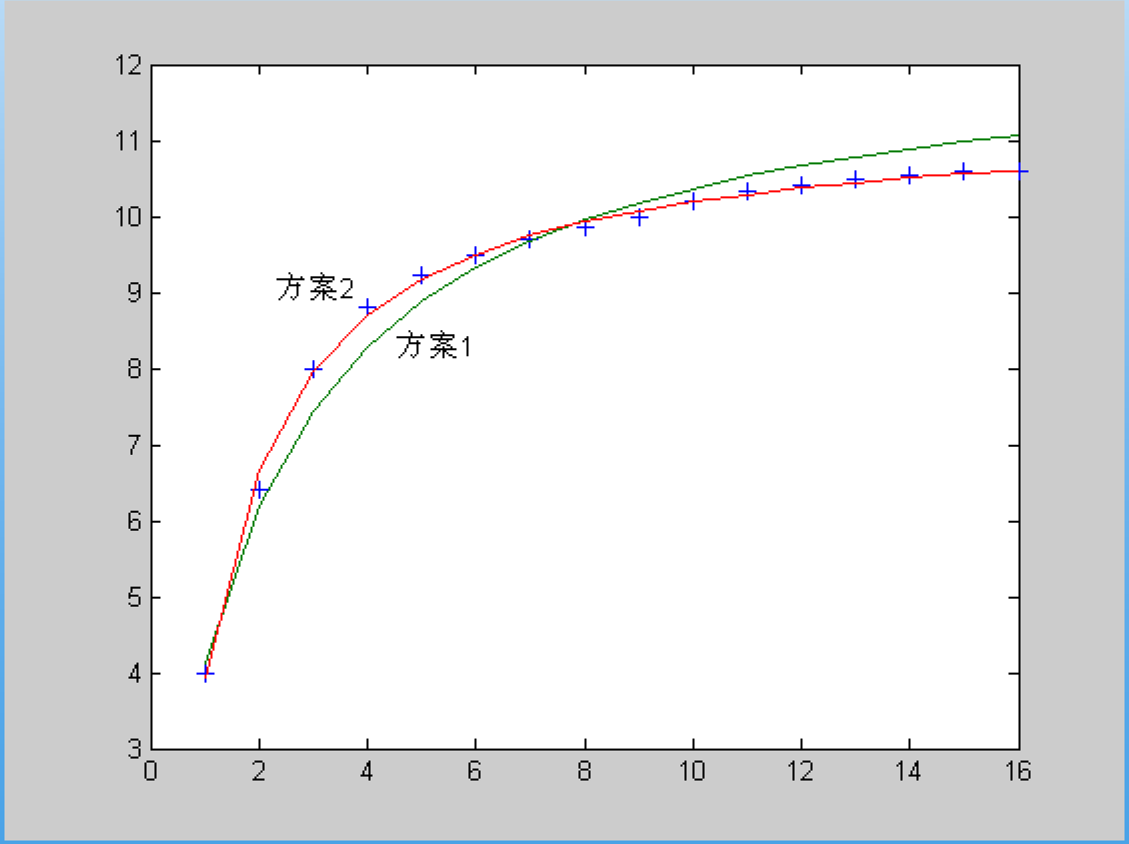
把两个不同的经验公式 $y=\dfrac{t}{at+b}$ 和 $y=ae^{b/t}$ 进行了比较，

从误差平方和与最大偏差绝对值这两个不同角度看，后者均优于前者。

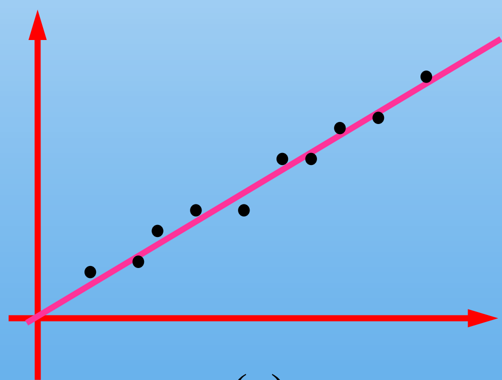
因此，在解决实际问题时，常常要经过反复分析，多次选择、计算与比较，

才能获得较好的数学模型。

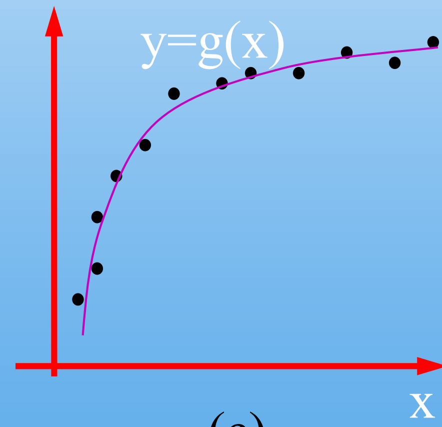
经验公式	误差平方和	最大偏差绝对值
$y=\dfrac{t}{at+b}$	1.562	0.560
$y=ae^{b/t}$	0.116	0.277


12
11
10
9
8
7
6
5
4
3
0
2
4
6
8
10
12
14
16
方案2
方案1

在作数据拟合时，选择合适的拟合数学公式是很重要的。通常最好是选用所给数据预先作出列表函数的曲线图，再根据曲线的大致形状，选择和确定合适的拟合数学公式形状，一味采用多项式拟合并不一定可取。



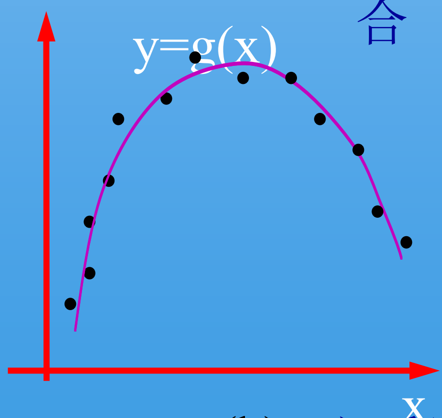
(a) 宜采用线性拟合



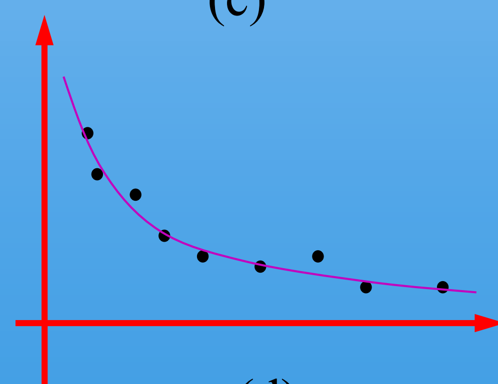
(c)

采用双曲线型函数$\varphi(x)=x/(ax+b)$

或指数型函数$\varphi(x)=ae^{-b/x}$



(b) 宜采用$\varphi(x)=a_0+a_1x+a_2x^2$拟合



(d)

采用

$\varphi(x)=1/(ax+b)$

$\varphi(x)=1/(a+bx^2)$

$\varphi(x)=ae^{-bx}$

等函数拟合

从函数角度看，插值法与最小二乘法都是一种根据函数表求函数的近似表达式的问题，属于函数逼近问题。

从几何上看，二者都是根据一列数据点求曲线的近似曲线问题，是曲线拟合问题。

插值法根据插值条件来选择近似函数

最小二乘法根据“偏差平方和最小”原则选择近似函数

上机作业要求：

P89 第一题

1. MATLAB程序（有注释）
2. 程序流程图
3. 详细的结果分析
4. 作业模板如下：

附录一：实验报告内容要求

实验报告内容要求

一、课题名称

二、班级、姓名

三、目的和意义

方法的理论意义和实用价值，如解超越方程的弦截法，改进了牛顿法，它适用于任意连续函数在大范围中求解，并且避免计算导数值，使其更具有实用性。

四、计算公式

五、结构程序设计

六、结果讨论和分析

如初值对结果的影响；不同方法的比较；该方法的特点和改进；整个实验过程中（包括程序编写，上机调试等）出现的问题及其处理等广泛的问题，以此扩大知识面和对实验环节的认识。