数值计算方法

杨鲤铭

Tel：15850687079

Email：lmyang@nuaa.edu.cn

南京航空航天大学

航空学院

2023-6

数值计算方法

机械工业出版社

马东升 董宁 编著

最终成绩比例

期末考试：60%

平时出勤和作业：30%

上机作业报告：10%（第三章）

目录

1 数值计算引论

1.2 误差的来源

●产生误差的原因

实际过程	数学模型	模型误差	● 数学建模需要忽略次要因素 ● 数学模型的近似性
	实验测量	观测误差	● 观测设备精度有限 ● 观测者人为因素
	算法精度	截断误差	● 数值算法阶数有限
	数据表示	舍入误差	● 计算机数据位数有限

●截断误差

- 大多数工程问题为连续的动力学过程
- 数值算法只能将无穷阶的连续过程近似为有限阶的有限过程

$$f(x)=\underbrace{f(x_0)+\sum_{k=1}^{N}\frac{1}{k!}f^{(k)}(x_0)\Delta x^k}_{\text{数值近似}}+\underbrace{\sum_{k=\mathrm{N}+1}^{\infty}\frac{1}{k!}f^{(k)}(x_0)\Delta x^k}_{\text{截断误差}}$$

1.2 误差的来源

●舍入误差

- ●计算机运算字长有限
- ●数据的表示和存储格式均有位数限制
- ●无限位浮点数无法精确表示
- ●随着运算进行，舍入误差可能累积甚至放大发散

```
In [5]:

In [6]: import numpy as np

In [7]: print("pi: %s" % np.pi)
pi: 3.141592653589793

In [8]:
```

$$a = A + a'$$

$$b = B + b'$$

$$A >> a', B >> b'$$

问： $ab = AB + Ab' + Ba'$ 是否成立？

1.3 近似数的误差表示

绝对误差

相对误差

有效数字

1.3.1 绝对误差

●绝对误差

定义1-1

设x*为准确值x的一个近似值，称

$$e(x^*) = x - x^*$$

为近似值x*的绝对误差，简记为e*.

1.3.1 绝对误差

- **绝对误差** $e(x^*) = x - x^*$

基本性质

- **e^*正负性，有量纲，不可确知**
- **弱近似值，$x^* < x$, $e^* > 0$;反之，强近似值**
- **精度由$|e^*|$表征**
- **$|e^*|$ 越小越好**

1.3.1 绝对误差

●误差限

定义1-2

如果满足

$$\left|e(x^*)\right| = \left|x - x^*\right| \le \varepsilon(x^*)$$

称 $\varepsilon(x^*)$ **为x*近似x的绝对误差限，简称误差限，简记为** ε^*

1.3.1 绝对误差

●误差限

$$|e(x^*)| = |x - x^*| \leq \varepsilon(x^*)$$

基本性质

- **误差限的有界性**
- **误差限不唯一**
- **误差限的设置取决于精度要求**
- **按四舍五入，取最小位的1/2**

$$x \in [x^* - \varepsilon^*, x^* + \varepsilon^*]$$

$$e^* \in [-\varepsilon^*, \varepsilon^*]$$

1.3.1 绝对误差

四舍五入的绝对误差限

$$x = \pm 0.x_1 x_2 \ldots \times 10^m,\ \ x_1 \neq 0,\ m, n \in Z,\ n \geq 1$$

$$x^* = \begin{cases} \pm 0.x_1 x_2 \ldots x_n \times 10^m & x_{n+1} \leq 4 \\ \pm 0.x_1 x_2 \ldots x_{n\text{-}1} (x_n + 1) \times 10^m & x_{n+1} \geq 5 \end{cases}$$

$$\left| x - x^* \right| \leq \frac{1}{2} \times 10^{m-n}$$

- **用四舍五入得到的近似数的误差限是末位的半个单位**

1.3.1 绝对误差

- **用四舍五入得到的近似数的误差限是末位的半个单位**
- **例题：求以下近似数的误差限** $\pi \approx 3.1416$

- **法1 由上述性质** $\varepsilon^* = \frac{1}{2} \times 10^{-4}$

- **法2** $\varepsilon^* = \frac{1}{2} \times 10^{m-n}$

$m-n=1-5=-4$

1.3.2 相对误差

定义1-2

设x*为准确值x的一个近似值，称

$$\frac{x - x^*}{x}$$

为近似值x*的**相对误差**.

基本性质

- **无量纲，用百分比表示**
- **由于精确值未知，常用下式代替**

$$e_r(x^*) = \frac{x - x^*}{x^*}$$

1.3.2 相对误差

相对误差（限）

设x*为准确值x的一个近似值，若

$$\left|e_r(x^*)\right| = \left|\frac{e^*}{x^*}\right| = \left|\frac{x - x^*}{x^*}\right| \le \frac{\varepsilon^*}{\left|x^*\right|} = \varepsilon_r^*$$

则称 ε_r^* 为近似值x*的**相对误差（限）**.

1.3.3 有效数字

定义1-2

设x*为准确值x的一个近似值，满足

$$x^{*}=0.x_1x_2\ldots x_n\times 10^{m},\ x_1\neq 0,\ m,n\in Z,\ n\geq 1$$

若绝对误差

$$\left|x-x^{*}\right|\leq \frac{1}{2}\times 10^{m-n}$$

则称 x*为x的**有n位有效数字的近似值，准确到第n位。**

$x_1x_2\ldots x_n$ **为x*的有效数字**

1.3.3 有效数字

基本性质

- **有效数字由绝对误差决定**
- **若近似值x^*的绝对误差(限)是某位的半个单位，则说x^* 精确到该位，若从该位到 x^* 的左面第一位非零数字一共有n位，则称近似值x^*有n位有效数字。**

1.3.3 有效数字

- 例题

求 3.142 和 3.141 作为圆周率 π 的近似值有几位有效数字?

解:

$$|\pi - 3.142| = 0.000407\ldots < 0.0005 = \frac{1}{2}\times 10^{-3}$$

$$m = 1, m - n = -3, n = 4$$

有4位有效数字

$$|\pi - 3.141| = 0.00059\ldots < 0.005 = \frac{1}{2}\times 10^{-2}$$

$$m = 1, m - n = -2, n = 3$$

有3位有效数字，3.14 是有效数字，千分位的1 不是有效数字

1.3.3 有效数字

- 注释
  - 用四舍五入得到的近似数的误差限是末位的半个单位；近似数的误差限是末位的半个单位，则有n位有效数字；因此用四舍五入得到的近似数是有效数字
  - 有效数字位数相同的两个近似数，绝对误差不一定相同
  - 把任何数乘以10^p(p=0，+1，+2，...）等于移动该数的小数点，这样并不影响其有效数字的位数
  - 在公式运算中，要先区分准确量和近似数。准确数有无穷多位有效数字
  - 有效数字位与小数点后有多少位数无直接关系

1.3.4 有效数字与相对误差

- 绝对误差与有效数字位数

$$\varepsilon^* = \frac{1}{2} \times 10^{m-n}$$

- 绝对误差与相对误差

$$\varepsilon_{\mathrm{r}} = \frac{\varepsilon^*}{\left|x^*\right|}$$

- 有效数字位数与相对误差——两个定理

1.3.4 有效数字与相对误差

定理 1-1 若近似数 $x* = \pm 0.x_1x_2\cdots x_n \times 10^m$ 具有 n 位有效数字，则其相对误差 $\varepsilon_r^* \le \dfrac{1}{2x_1}\times 10^{-(n-1)}$

证 因 $x* = \pm 0.x_1x_2\cdots x_n \times 10^m$，有 $|x*| \ge x_1 \times 10^{m-1}$，

又，$x*$ 有 n 位有效数字，即 $|x - x*| \le \dfrac{1}{2}\times 10^{m-n}$，因此

$$|e_r^*| = \frac{|x - x*|}{|x*|} \le \frac{\frac{1}{2}\times 10^{m-n}}{x_1 \times 10^{m-1}} = \frac{1}{2x_1}\times 10^{-(n-1)}$$，即 $\varepsilon_r^* = \dfrac{1}{2x_1}\times 10^{-(n-1)}$

1.3.4 有效数字与相对误差

- 注记
  - 定理给出的是一个充分条件，而不是必要条件。
  - 定理的逆命题不成立。即若相对误差满足定理， 近似数不一定有n位有效数字，可能少于n。
  - 实际应用时，为了使相对误差满足一定的要求，可以用定理1-1来确定所取的近似数应该具有多少位有效数字（可能位数会偏多，属于保守估计）。

1.3.4 有效数字与相对误差

定理 1-2 若近似数 $x^* = \pm 0.x_1x_2\cdots x_n \times 10^m$ 的相对误差限

$$\varepsilon_r^* \le \frac{1}{2(x_1+1)} \times 10^{-(n-1)}$$

则它至少具有 n 位有效数字。

证 $x^* = \pm 0.x_1x_2\cdots x_n \times 10^m$ 有 $\left|x^*\right| \le (x_1+1)\times 10^{m-1}$

则

$$\left|x-x^*\right| = \frac{\left|x-x^*\right|}{\left|x^*\right|}\left|x^*\right| \le \frac{1}{2(x_1+1)} \times 10^{-(n-1)} \times (x_1+1)\times 10^{m-1} = \frac{1}{2}\times 10^{m-n},$$

即它至少具有 n 位有效数字。

1.3.4 有效数字与相对误差

- 注记

  - 定理给出的是一个充分条件，而不是必要条件。
  - 定理的逆命题不成立。即若近似数有n位有效数字，相对误差不一定满足定理。
  - 实际应用时，为了使取的近似数具有n位有效数字，要求所取近似数的相对误差要满足定理的要求 。

1.3.4 有效数字与相对误差（两定理比较）

例　为了使近似数 x^*的相对误差限不超过 0.25%，问 x^*要取几位有效数字？

解　根据定理 1-1，有

$$\frac{1}{2x_1}\times 10^{-(n-1)}=0.25\%$$

x_1的取值范围是 1 到 9，由于 x_1未给出，取

$$x_1=1，n= 3.301$$

$$x_1=9，n= 2.347$$

按最不利情况，x^*要取 4 位有效数字。

1.3.4 有效数字与相对误差（两定理比较）

例　已知近似数 x^*的相对误差限不超过 0.25%，问 x^*至少有几位有效数字？

解　根据定理 1-2，有

$$\frac{1}{2\left(x_1+1\right)}\times 10^{-(n-1)}=0.25\%$$

x_1 的取值范围是 1 到 9，由于 x_1 未给出，取

$x_1=1$，$n=$ 3.000

$x_1=9$，$n=$ 2.301

按最不利情况，x^*至少有 2 位有效数字。

1.4.1 函数运算误差

对$f(x)$在近似值x^*附近泰勒展开

$$f(x)=f(x^*)+f'(x^*)(x-x^*)+\frac{f''(\xi)}{2}(x-x^*)^2$$

ξ 介 于 x和 x^*之 间 ， 取 绝 对 值 得

$$\left|f(x)-f(x^*)\right|\le\left|f'(x^*)\right|\varepsilon^*+\frac{f''(\xi)}{2}(\varepsilon^*)^2$$

式 中 ε^*为 近 似 数 x^*的 绝 对 误 差 限 。

$$\varepsilon\left(f\left(x^*\right)\right)\approx\left|f'\left(x^*\right)\right|\varepsilon^*$$

$$\varepsilon_r\left(f\left(x^*\right)\right)\approx\left|\frac{f'\left(x^*\right)}{f\left(x^*\right)}\right|\varepsilon^*$$

$$\varepsilon\left(y^*\right)=\sum_{i=1}^{n}\left|\left(\frac{\partial f}{\partial x_i}\right)^*\right|\varepsilon^*$$

$$\varepsilon_r\left(y^*\right)=\sum_{i=1}^{n}\left|\left(\frac{\partial f}{\partial x_i}\right)^*\right|\frac{\varepsilon^*}{y^*}$$

1.4.2 算术运算误差

例 1-15 正方形的边长约为 100 cm，怎样测量才能使其面积误差不超过 1 cm^2？

解 设正方形边长为 x cm，测量值为 x^* cm，面积

$$y=f(x)=x^2$$

由于 $f'(x)=2x$，记自变量和函数的绝对误差分别是 e^* 和 $e(y^*)$，则

$$e^*=x-x^*$$

$$e(y^*)=y-y^*\approx f'(x^*)(x-x^*)=2x^*e^*=200e^*$$

现要求 $|e(y^*)|\approx 200e^*<1$，于是

$$|e^*|\leqslant\left(\frac{1}{200}\right)\mathrm{cm}=0.005\ \mathrm{cm}$$

要使正方形面积误差不超过 1 cm^2，测量边长时绝对误差应不超过 0.005 cm。

2 非线性方程的数值解法

单根和重根

定义　若 $f(x)$ 满足 $f(x)=(x-x^*)^m\varphi(x)$，其中 $\varphi(x^*)\neq 0$，则称 x^* 是方程 $f(x)=0$ 的 m 重根和函数 $f(x)$ 的 m 重零点。当 $m=1$ 时，x^* 是方程的单根和函数 $f(x)$ 的单零点。

定义　设 $f(x)$ 在 x^* 的某邻域存在 **m** 阶连续导数，如果有 $f(x^*)=f'(x^*)=\cdots=f^{(m-1)}(x^*)=0$，但 $f^{(m)}(x^*)\neq 0$，则称 x^* 是方程 $f(x)=0$ 的 m 重根和函数 $f(x)$ 的 m 重零点。当 $m=1$ 时，x^* 是方程的单根和函数 $f(x)$ 的单零点。

例　$x=0$ 是 $f(x)=\mathrm{e}^{2x}-1-2x-2x^2=0$ 的几重根？

解　$f(x)=\mathrm{e}^{2x}-1-2x-2x^2$，$f(0)=0$

$f'(x)=2\mathrm{e}^{2x}-2-4x$，$f'(0)=0$

$f''(x)=4\mathrm{e}^{2x}-4$，$f''(0)=0$

$f'''(x)=8\mathrm{e}^{2x}$，$f'''(0)=8\neq 0$，因此 $x=0$ 是方程 $f(x)=\mathrm{e}^{2x}-1-2x-2x^2=0$ 的 3 重根。

有根区间和隔根区间

定理 2-1（介值定理） 若函数 $f(x)$ 在 $[a,b]$ 连续，且 $f(a)f(b)<0$，则方程 $f(x)=0$ 在 (a,b) 内至少有一个实根。将 $[a,b]$ 称为 $f(x)$ 的有根区间。

定理2-2 函数 $f(x)$ 在 $[a,b]$ 上单调连续，且 $f(a)f(b)<0$，则方程 $f(x)=0$ 在区间 $[a,b]$ 上有且仅有一个实根 x^*。$[a,b]$ 称为 $f(x)$ 的隔根区间。

2.1.3 区间二分法

定理2-2 函数$f(x)$在$[a,b]$上单调连续，且$f(a)f(b)<0$，则方程$f(x)=0$在区间$[a,b]$上有且仅有一个实根x^*。

二分过程结束的原则：

先给定精度要求ε(绝对误差限)，

（1）事先由ε估计出二分的最小次数k，取$x^* \approx x_k$

$$由 \left| x^* - x_k \right| \le \frac{b-a}{2^{k+1}} < \varepsilon \text{ 得 } 2^{k+1} > \frac{b-a}{\varepsilon}, \quad k > \frac{\lg(b-a) - \lg\varepsilon}{\lg 2} - 1$$

（2）当$|b_{k+1} - a_{k+1}| < \varepsilon$时结束二分计算，取$x^* \approx x_k$；

证明非线性方程 $f(x)=1-x-\sin x=0$ 在 $[0,1]$ 内有唯一根，使用二分法求误差不大于 $\frac{1}{2}\times 10^{-4}$ 的根要二分多少次?

解 $f(x)=1-x-\sin x=0$ 是连续函数，且

1、判断是否连续函数

$$f(0)=1>0, f(1)=-\sin 1<0$$

2、判断是否有根区间

又 $f'(x)=-1-\cos x<0 \qquad x\in[0,1]$

3、判断是否隔根区间

所以，$f(x)$ 在 $x\in[0,1]$ 上有且仅有一个根。

又 $\left|x^*-x\right|\le \frac{1}{2^{k+1}}(b-a)\le \frac{1}{2}\times 10^{-4}$

4、确定二分次数

$$k\ge \frac{\lg(1-0)+4}{\lg 2}=13.82$$

所以需二分 14 次。

方程 $f(x)=0$ 化为等价形式的方程 $x=\varphi(x)$,

构造迭代格式 $x_{k+1}=\varphi(x_k)$, $k=0$, 1 , 2 , ……

取初始近似根x_0 , 进行迭代计算 $x_1=\varphi(x_0)$, $x_2=\varphi(x_1)$, ……

则有x_1 , x_2 , …… , x_k , ……,得到迭代序列$\{x_k\}$。如果这个序列有极限，则迭代公式是收敛的。这时

$$\lim_{k\to\infty} x_k = x^*$$

则 $x^*=\varphi(x^*)$, x^* 为不动点，等价地有 $f(x^*)=0$, x^* 即为方程的根。连续函数 $\varphi(x)$ 称为迭代函数。

实际计算到 $|x_k - x_{k-1}|<\varepsilon$ （ε是预定的精度），取$x^*\approx x_k$。

2.2.2 迭代的收敛性

迭代法的收敛条件及误差估计式

定理2-3（区间收敛性） 设函数 $\varphi(x)$ 在$[a, b]$上具有连续的一阶导数，且满足

(1) 对 $x \in [a, b]$,有 $\varphi(x) \in [a, b]$

(2) 存在$0< L < 1$，使对任意$x \in [a, b]$有

$$|\varphi'(x)| \le L < 1$$

则方程 $x=\varphi(x)$ 在 $[a, b]$ 上的根 x^* 存在且唯一；

对初值 $x_0 \in [a, b]$, 迭代过程 $x_{k+1}=\varphi(x_k)$ 均收敛于方程的根x^*。

例　用迭代法求方程 $x^3-2x-5=0$ 的最小正根。

解　试凑正根所在的区间

x	0	1	2	3	4	200	20000
$f(x)$	-5	-6	-1	16	51	+	+

取正根区间[2,3]，迭代格式 $x_{k+1}=\frac{1}{2}(x_k^3-5)$ ，迭代函数导数 $\varphi'(x)=\frac{3}{2}x^2, \max\limits_{2\le x\le 3}|\varphi'(x)|=13.5>1$，不满足收敛定理。

步骤1：找到隔根区间，确定有根

步骤2：建立迭代格式，判断收敛性

将原方程改写成

$$x=\sqrt[3]{(2x+5)}\text{，}\quad \varphi(x)=\sqrt[3]{2x+5},\varphi'(x)=\frac{2}{3}(2x+5)^{-\frac{2}{3}},$$

且 $\max\limits_{2\le x\le 3}|\varphi'(x)|=0.154<1$，迭代格式 $x_{k+1}=\sqrt[3]{2x_k+5}$

收敛。取初值 $x_0=2.5$。进行迭代，结果如下所示。

迭代法的局部收敛性

定义 2-2 如果存在 x^* 的某个邻域 Δ：$\left|x-x^*\right| \le \delta$，使迭代过程 $x_{k+1}=\varphi(x_k)$ 对于任意初值 $x_0 \in \Delta$ 均收敛，则称迭代过程 $x_{k+1}=\varphi(x_k)$ 在根 x^* 附近具有局部收敛性。

定理 2-5 设 $\varphi(x)$ 在 $x=\varphi(x)$ 的根 x^* 附近一阶导连续，且有

$$\left|\varphi'(x^*)\right| < 1$$

则迭代格式 $x_{k+1}=\varphi(x_k)$ 在根 x^* 附近具有局部收敛性。

局部收敛性可以克服全局收敛性中映内性条件不易验证的缺陷

由于在实际应用中根 x^* 事先不知道，故条件

$$|\varphi'(x^*)| < 1$$

无法验证。但已知根的初值x_0在根 x^*邻域，又根据$\varphi'(x)$的连续性，则可采用

$$|\varphi'(x_0)| < 1$$

来代替$|\varphi'(x^*)| < 1$，判断迭代的收敛性。

例 求方程 $x=\mathrm{e}^{-x}$在$x=0.5$附近的一个根，按5位小数计算，结果的精度要求为$\varepsilon=10^{-3}$.

解 迭代公式 $x_{k+1}=\mathrm{e}^{-x_k}$，取$\varphi(x)=\mathrm{e}^{-x}$,

$$|\varphi'(0.5)|=|-\mathrm{e}^{-0.5}|=\mathrm{e}^{-0.5}=<0.61<1$$

迭代公式 $x_{k+1}=\mathrm{e}^{-x_k}$收敛。

2.2.3 迭代过程的收敛速度

定义 2-3 设迭代过程 $x_{k+1}=\varphi(x_k)$ 收敛于 $x=\varphi(x)$ 的根 x^*，迭代误差 $e_k=x_k-x^*$，如果存在常数 p（$p\ge 1$）和不等于零的常数 c 使

$$\lim_{k\to\infty}\frac{|e_{k+1}|}{|e_k|^p}=c$$

则称序列 x_k 是 p 阶收敛的。c 称渐进误差常数。特别地，$p=1$ 称为线性收敛，$p=2$ 称为平方收敛，$p>1$ 时称为超线性收敛。

定理 2–6 对迭代过程 $x_{k+1}=\varphi(x_k)$，若 $\varphi^{(p)}(x)$ 在所求根 x^* 邻域连续，且

$$\varphi'(x^*)=\varphi''(x^*)=\cdots=\varphi^{(p-1)}(x^*)=0,\quad \varphi^{(p)}(x^*)\neq 0$$

则迭代格式 $x_{k+1}=\varphi(x_k)$ 在 x^* 领域是 p 阶收敛的。

2.3.1 牛顿迭代法

1．导出：非线性方程 $f(x)=0$ ，若 $f'(x_k)\neq 0$ ，建立迭代格式

$$x_{k+1}=x_k-\frac{f(x_k)}{f'(x_k)} \qquad k=0,1,\cdots$$

2．数学意义：把非线性方程线性化，用线性方程的解逐步逼近非线性方程的解。

$$f(x) = f(x_k) + f'(x_k)(x - x_k) + \frac{f''(x_k)}{2!}(x - x_k)^2 + \cdots$$

设 x_k 在 x 附近时，有 $f(x) \approx f(x_k) + f'(x_k)(x - x_k)$

对 $f(x^*) = 0$ 有 $f(x_k) + f'(x_k)(x^* - x_k) \approx 0$，解出

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \qquad k = 0, 1, \cdots$$

3.几何意义

过曲线上的点$p_k(x_k, f(x_k))$作切线，切线方程

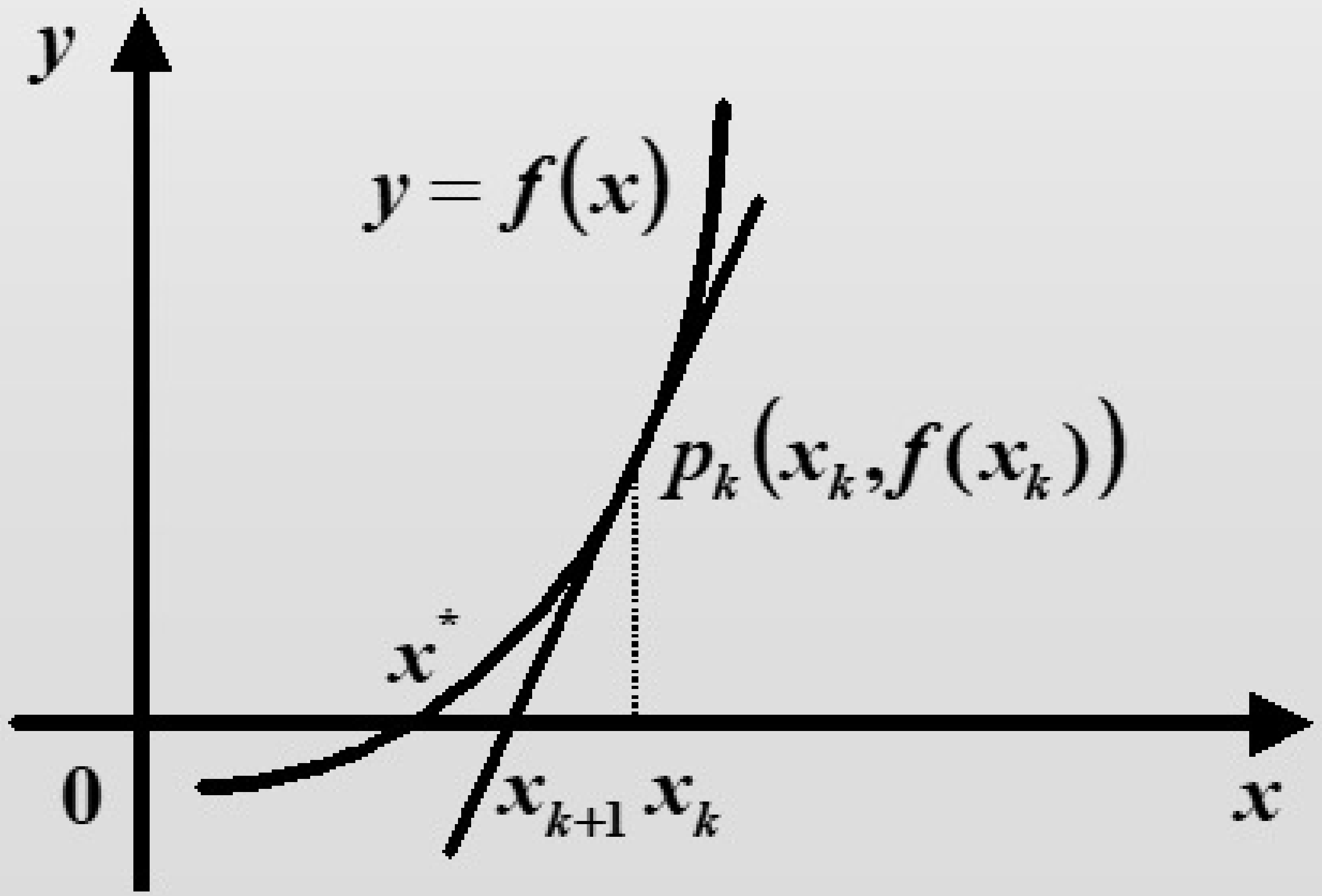
$$y=f(x_k)+f'(x_k)(x-x_k)$$

切线方程和横轴的交点$(x_{k+1}, 0)$，即

$$0=f(x_k)+f'(x_k)(x_{k+1}-x_k)$$

若$f'(x_k)\neq 0$，解出x_{k+1}, 则得*Newton*迭代公式

$$\boxed{x_{k+1}=x_k-\frac{f(x_k)}{f'(x_k)}, k=0,1,2,\cdots}$$



所以牛顿迭代法也称切线法

例　证明方程 $x^5+5x+1=0$ 在区间 $[-1,0]$ 内有唯一的实根，并用牛顿迭代法求这个根的近似值，使误差不超过 0.01。

解　令 $f(x)=x^5+5x+1$，则 $f(x)$ 在 $[-1,0]$ 上连续，又

$$f(-1)=-5<0, f(0)=1>0$$

故由介值定理知至少存在一个 $x^* \in [-1,0]$ 使 $f(x^*)=0$，又由

$f'(x)=5x^4+5>0$，知 $f(x)$ 在 $[-1,0]$ 上单调增加，因而方程

$x^5+5x+1=0$ 在区间 $[-1,0]$ 内有唯一实根。

下面用牛顿迭代法求这个根的近似值。

取 $x_0 = -1$ ，代入牛顿迭代法

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

有

$$x_1 = -1 + \frac{-5}{5+5} = -0.050, \quad x_2 = -0.5 + \frac{f(-0.5)}{f'(-0.5)} = -0.212$$

$$x_3 = -0.212 + \frac{f(-0.212)}{f'(-0.212)} = -0.200$$

$$x_4 = -0.200 + \frac{f(-0.200)}{f'(-0.200)} = -0.200$$

所以 $x^* \approx -0.20$ 。

2.收敛速度

定理 2-8 设 $f(x^*)=0, f'(x^*)\neq 0, f''(x^*)$ 在 x^* 邻域连续，则牛顿法在 x^* 局部收敛，且至少 2 阶收敛。并有

$$\lim_{k\to\infty}\frac{e_{k+1}}{e_k^2}=\frac{f''(x^*)}{2f'(x^*)}$$

例　设函数 $f(x)=(x^3-a)^2$，写出解 $f(x)=0$ 的牛顿迭代格式，并证明此格式的收敛阶。

解　将 $f(x)=(x^3-a)^2$， $f'(x)=6(x^3-a)x^2$，代入牛顿迭代法，有

$$x_{k+1}=x_k-\frac{(x_k^3-a)^2}{6(x_k^3-a)x_k^2}=\frac{5}{6}x_k+\frac{a}{6x_k^2}$$

迭代函数 $\varphi(x)=\dfrac{5}{6}x+\dfrac{a}{6x^2}$ ， $\varphi'(x)=\dfrac{5}{6}-\dfrac{a}{3x^3}$， 又已知 $x^*=\sqrt[3]{a}$，

$\varphi'(\sqrt[3]{a})=0.5<1\neq 0$，所以该牛顿迭代法为线性收敛。

降为线性收敛的原因在于，$f'(x)=0$.

3 重根时的修正

重根数已知时。

当 x^*是 $f(x)=0$ 的 m （其中 $m\geq 2$ ）重根时，牛顿法

$$x_{k+1}=x_k-\frac{f(x_k)}{f'(x_k)}$$

仅为线性收敛。改进牛顿法

$$x_{k+1}=x_k-m\frac{f(x_k)}{f'(x_k)}$$

是二阶收敛的方法。

例　设函数$f(x)=(x^3-a)^2$，写出解$f(x)=0$的牛顿迭代格式，并证明此格式的收敛阶。

解　将$f(x)=(x^3-a)^2$，$f'(x)=6(x^3-a)x^2$，代入牛顿迭代法，有

$$x_{k+1}=x_k-\frac{(x_k^3-a)^2}{6(x_k^3-a)x_k^2}=\frac{5}{6}x_k+\frac{a}{6x_k^2}$$

迭代函数$\varphi(x)=\dfrac{5}{6}x+\dfrac{a}{6x^2}$，$\varphi'(x)=\dfrac{5}{6}-\dfrac{a}{6x^3}$，

又已知$x^*=\sqrt[3]{a}$， $\varphi'(\sqrt[3]{a})=0.5<1\neq 0$，所以牛顿迭代法为线性收敛。

由于$x^*=\sqrt[3]{a}$是所给方程的二重根，用重根时的修正牛顿法$x_{k+1}=x_k-2\dfrac{f(x_k)}{f'(x_k)}$，将$f(x)=(x^3-a)^2$，

$f'(x)=6(x^3-a)x^2$，代入

$$x_{k+1}=x_k-2\frac{(x_k^3-a)^2}{6(x_k^3-a)x_k^2}=\frac{2}{3}x_k+\frac{a}{3x_k^2}$$

迭代函数 $\phi(x)=\dfrac{2}{3}x+\dfrac{a}{3x^2}$ ，$\phi'(x)=\dfrac{2}{3}-\dfrac{2a}{3x^3}$，又已知 $x^*=\sqrt[3]{a}$， $\varphi'(\sqrt[3]{a})=0$，所以牛顿迭代法为平方收敛。

当重根数未知时，令 $\mu(x)=\dfrac{f(x)}{f'(x)}$，若 x^* 是 $f(x)$ 的 m 重根，则 $f(x)=(x-x^*)^m q(x)$，且 $q(x^*)\neq 0$，此时

$$\mu(x)=\frac{(x-x^*)^m q(x)}{m(x-x^*)^{m-1}q(x)+(x-x^*)^m q'(x)}=\frac{(x-x^*)q(x)}{mq(x)+(x-x^*)q'(x)}$$

故 x^* 是 $\mu(x)$ 的单重零点。牛顿迭代法修正为 $x_{k+1}=x_k-\dfrac{\mu(x_k)}{\mu'(x_k)}$ 则是二阶收敛的。此时 $\varphi(x)=x-\dfrac{\mu(x)}{\mu'(x)}=x-\dfrac{f'(x)f(x)}{f(x)^2-f(x)f''(x)}$，迭代格式

$$x_{k+1}=x_k-\frac{f'(x_k)f(x_k)}{f'(x_k)^2-f(x_k)f''(x_k)},\quad k=0,1,\cdots$$

3

线性代数方程组的数值解法

线性代数方程组的数值解法

n阶线性代数方程组

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \cdots\cdots\cdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases}$$

其矩阵形式为 Ax=b ，其中

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

或

$$\sum_{j=1}^{n} a_{ij} x_j = b_i \qquad i = 1, 2, \cdots, n$$

线性代数方程组的数值解法

工程应用中大多数采用直接法与迭代法

直接法

计算步数有限
适用于低阶稠密阵求解
不适合大型问题
高斯消去法
三角分解法

迭代法

逐次逼近策略
内存要求低
需要考虑收敛性问题
适用于大型问题求解
Jacobi/ Gauss-Seidel
松弛法

3.1　高斯消去法

- 顺序高斯消去法

基本思想：

　　用逐次消去未知数的方法把原方程组化为上三角形方程组进行求解 。

　　求解过程两步走：

　　1. 消元过程：用初等行变换将原方程组的系数矩阵化为上三角阵；依从左到右、自上而下的次序将主对角元下方的元素化为零

　　2. 回代过程：先对上三角形方程组的最后一个方程求解，将所得解逐步往上一个方程代入求解。

3.1 高斯消去法

- 顺序高斯消去法

1. 预备操作：用第k行主元对k行元素和右端b值归一化，
   使该行主元变为1

2. 以第k行主元为基准对以下第i行计算乘数（k从1开始）

$$m_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}, i = k+1, \dots, n \qquad a_{kk}^{(k)} \neq 0$$

3. 对k以下各行的第j列消元

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik} a_{kj}^{(k)} \ , j = k+1, \dots, n$$

$$b_i^{(k+1)} = b_i^{(k)} - m_{ik} b_k^{(k)} \ , i = k+1, \dots, n$$

3.1 高斯消去法

- 顺序高斯消去法

经n-1次消元，得到 $A^{(n)}x = b^{(n)}$

上三角阵

$$A^{(n)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} \\ & a_{22}^{(2)} & & a_{2n}^{(2)} \\ \vdots & & \ddots & \vdots \\ & & \cdots & a_{nn}^{(n)} \end{pmatrix}$$

右端矢量

$$b^{(n)} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_n^{(n)} \end{pmatrix}$$

```
void forward_cancellation()
{
  // stage 1:
  for(auto k=0;k<n-1;k++)
  {
    const auto Akk=A[k][k];
    for(auto j=k;j<n;j++)
      A[k][j]/=Akk;
    b[k]/=Akk;
    for(auto i=k+1;i<n;i++)
    {
      auto mik=A[i][k]; //
      for(auto j=0;j<n;j++)
        A[i][j] -= mik*A[k][j];
      b[i] -= mik * b[k];
    }
  } // i loop
}
```

3.1 高斯消去法

- 顺序高斯消去法

自下向上回代，得到

$$x_n = \frac{b_n^{(n)}}{a_{nn}^{(n)}}$$

$$x_i = \left(b_i^{(i)} - \sum_{j=i+1}^{n} a_{ij}^{(i)} x_j \right) / \boxed{a_{ii}^{(i)}}$$

非零主元是关键
可进行归一化

$$i = n-1, \dots, 2,1$$

```
void back_substitution()
{
  // stage 2: back substitution
  x[n-1]= b[n-1]/A[n-1][n-1];
  for(auto i=n-2;i>=0;i--)
  {
    for(auto j=i+1;j<n;j++)
      b[i] -= A[i][j]*x[j];
    x[i] = b[i] / A[i][i];
  }

}
```

3.1 高斯消去法

- 高斯消去法的使用条件

$$m_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}, i = k+1, \dots, n$$

$$a_{kk}^{(k)} \neq 0, \qquad \mathrm{k} = 1, 2, \dots, \mathrm{n}$$

3.1 高斯消去法

- 高斯消去法的使用条件

定理3-1： 方程组系数阵顺序主子式全不为零，则高斯消去法可以求解

顺序主子式

$$D_i = \begin{vmatrix} a_{11} & \cdots & a_{1i} \\ \vdots & & \vdots \\ a_{i1} & \cdots & a_{ii} \end{vmatrix} \neq 0 \qquad i = 1,2,\ldots n$$

经变换得

$$D_i = \begin{vmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1i}^{(1)} \\ & a_{22}^{(2)} & \cdots & a_{2i}^{(2)} \\ & & \ddots & \vdots \\ & & & a_{ii}^{(i)} \end{vmatrix} = \prod_{k=1}^{i} a_{kk}^{(k)} \neq 0$$

对角占优阵满足上述性质

3.1 高斯消去法

- 高斯消去法的使用条件

定义3-1：若矩阵A满足每行对角元素绝对值均大于同行其他元素绝对值之和，则称A为严格行对角占优阵

$$|a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ij}| \qquad i = 1,2,\dots n$$

定理3-2： 方程组Ax=b，若系数阵A为严格对角占优阵，则高斯消去法求解时主元全不为零

$$a_{kk}^{(k)} \neq 0, \qquad \mathrm{k} = 1, 2, \dots, \mathrm{n}$$

3.1 高斯消去法

- 3.1.2 列主元高斯消去法

1）对非奇异阵，顺序消元也可能导致主元为零

2）主元绝对值过小会导致舍入误差过大，影响计算精度

举例

$$\begin{aligned} 10^{-5}x_1 + x_2 &= 1 \\ x_1 + x_2 &= 2 \end{aligned}$$

利用顺序消去（四位浮点十进制计算），得

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

结果严重失真

- 计算乘数所用分母过小，导致误差过大
- 大数吃小数

解决方案：先调整顺序再消元

$$\begin{aligned} x_1 + x_2 &= 2 \\ 10^{-5}x_1 + x_2 &= 1 \end{aligned}$$

解得

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

结果误差非常小

- 添加选主元步骤
- 全选主元/列选主元

3.1 高斯消去法

- 3.1.2 列主元高斯消去法

列选主元

在第k消元步，找到k列a_{kk}及以下绝对值最大元素

设主元在l行　$|a_{lk}| = max_{k \le i \le n} |a_{ik}|$

若l, k不相等，将l, k行互换，使新主元换至a_{kk}位置

再继续消元过程

列选主元计算量小，更易采用

定理3-4：若n阶线性方程组对称且严格对角占优，则其对角元素全是列主元

3.1 高斯消去法

- 3.1.2 列主元高斯消去法

列选主元例3-4

$$\begin{aligned} 12x_1 - 3x_2 + 3x_3 &= 15 \\ -18x_1 + 3x_2 - x_3 &= -15 \\ x_1 + x_2 + x_3 &= 6 \end{aligned}$$

增广矩阵

$$\begin{pmatrix} -18 & 3 & -1 & -15 \\ 0 & \frac{7}{6} & \frac{17}{18} & \frac{31}{6} \\ 0 & 0 & \frac{22}{7} & \frac{66}{7} \end{pmatrix}$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

行列式

$$|A| = (-1)^m \prod_{i=1}^{n} a_{ii}^{(i)} = (-1)^2(-18) \times \frac{7}{6} \times \frac{22}{7} = -66$$

3.1　高斯消去法

- 3.1.3 Gauss-Jordan消去法

在消元步将上三角阵对角化后，回代过程可以省略

$$D=\begin{pmatrix} a_{11}^{(1)} & \cdots & \\ \vdots & \ddots & \vdots \\ & \cdots & a_{nn}^{(n)} \end{pmatrix}$$

利用列主元将该列其他元素均消为零

试用G-J方法对如下增广矩阵求解

$$\begin{pmatrix} 2 & 2 & 3 & 3 \\ 4 & 7 & 7 & 1 \\ -2 & 4 & 5 & -7 \end{pmatrix}$$

3.1 高斯消去法

- 3.1.3 Gauss-Jordan消去法

Gauss-Jordan消去法可以与列选主元结合，并归一化

试求解
$$\begin{pmatrix} 2 & -1 & 3 & 1 \\ 4 & 2 & 5 & 4 \\ 1 & 2 & 0 & 7 \end{pmatrix}$$

一次归一化需n-k+1次除法

一次消去需(n-1)(n-k+1)次乘法

Gauss-Jordan消去法计算量 $\displaystyle\sum_{k=1}^{n} n(n-k+1) = \frac{n^2}{2}(n+1) \approx \frac{n^3}{2}$

无需回代，算法简单

3.1 高斯消去法

- 3.1.3 Gauss-Jordan消去法

Gauss-Jordan消去法可用于非奇异阵求逆

记非奇异阵A的逆为 $A^{-1} = X = \begin{pmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nn} \end{pmatrix}$

则求逆操作等价于求n阶矩阵X，使

$$AX = I$$

将矩阵X和单位阵I按列分块后，

$$X = (x_1, x_2, \dots x_n) \qquad x_j = \left(x_{1j}, x_{2j}, \dots x_{nj}\right)^T$$

$$I = (e_1, e_2, \dots e_n) \qquad e_j = (0,0, \dots 1, \dots 0)^T$$

则上述求逆操作等价于求解系数阵相同的n个方程组，使 $Ax_j = e_j$

试对以下矩阵求逆 $\begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 2 \\ 1 & 3 & 4 \end{pmatrix}$

3.2 矩阵三角分解法

- 3.2.1 Gauss消去法的矩阵描述 — LU分解

经过n-1次消元

$$\begin{cases} M_{n-1}\dots M_2 M_1 A^{(1)} & = A^{(n)} \\ M_{n-1}\dots M_2 M_1 b^{(1)} & = b^{(n)} \end{cases}$$

消元所得上三角阵记为U，则有

$$A^{(1)} = A = M_1^{-1} M_2^{-1} \dots M_{n-1}^{-1} U = LU$$

变换矩阵

$$L = M_1^{-1} M_2^{-1} \dots M_{n-1}^{-1} = \begin{pmatrix} 1 & & & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & 1 & & \\ & & l_{43} & \ddots & \\ \vdots & \vdots & \vdots & & \\ l_{n1} & l_{n2} & l_{n3} & \dots & 1 \end{pmatrix}$$

Gauss消去法本质上是将系数阵分解为LU矩阵之积

3.2 矩阵三角分解法

- 3.2.2 矩阵的直接三角分解 （直接分解无需消元步骤）

定义3-2 将矩阵A分解为下三角阵L和上三角阵U的乘积称为对A的LU分解

$$A = LU$$

- LU分解若存在，且不要求必为单位阵，则此分解不唯一
- 若L或U中一个为单位三角阵，则LU分解唯一
- Dolittle杜里特尔分解：单位下三角L乘以上三角U
- Crout克罗特分解：下三角L乘以单位上三角U

$$\begin{pmatrix} 2 & 3 \\ 4 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 2 & 3 \\ 0 & -5 \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ -2 & 3 \end{pmatrix} \begin{pmatrix} -2 & -3 \\ 0 & -5/3 \end{pmatrix}$$

3.2　矩阵三角分解法

• 3.2.2 矩阵的直接三角分解　（直接分解无需消元步骤）

$$A=\underbrace{L}_{\text{单位下三角阵}}\underbrace{U}_{\text{上三角阵}}=\begin{bmatrix}1 & & & \\ l_{21} & 1 & & \\ \vdots & & \ddots & \\ l_{n1} & l_{n2} & \cdots & 1\end{bmatrix}\begin{bmatrix}u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & & & u_{nn}\end{bmatrix}$$

$$A=\underbrace{L}_{\text{下三角阵}}\underbrace{U}_{\text{单位上三角阵}}=\begin{bmatrix}l_{11} & & & \\ l_{21} & l_{22} & & \\ \vdots & & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn}\end{bmatrix}\begin{bmatrix}1 & u_{12} & \cdots & u_{1n} \\ & 1 & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & & & 1\end{bmatrix}$$

3.2 矩阵三角分解法

- 3.2.2 矩阵的直接三角分解 （无消元步骤）

定理3-5 （Dolittle、Crout分解的可唯一分解充分条件）

矩阵A各阶主子式不为零，则可以唯一分解为单位下三角L和非奇异上三角U的乘积

<u>存在性</u>由高斯消去法的矩阵描述给出

<u>唯一性</u>由反证法得到

假设存在两种不同LU分解 $A = LU = L'U'$，由A行列式非零知L, U, L′, U′均非奇异， 有

$$(L')^{-1}L = U'U^{-1}$$

以Dolittle分解为例，该式左端为单位下三角，右端上三角阵因此也必须为n阶单位阵，由于两边相等，因此只可能是上式两端均等于标准单位阵I，则$L = L', U = U'$，唯一性得证

各阶主子式不为零也可以称为顺序主子阵非奇异

3.2　矩阵三角分解法

- 3.2.2 矩阵的直接三角分解　（无消元步骤）
  Dolittle分解

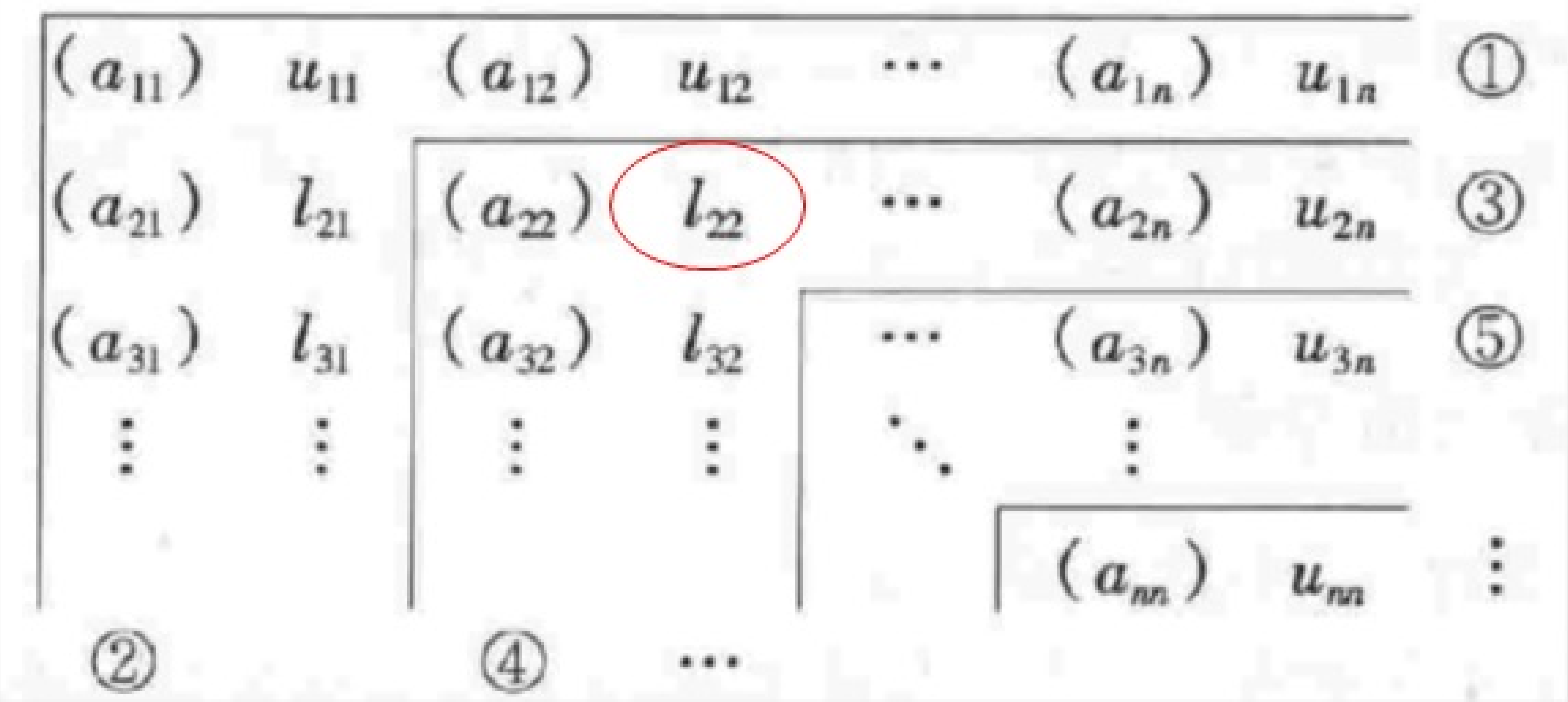
基于LU分解，矩阵行列式可由U阵主对角元素乘积获得

$$|A| = |L||U| = u_{11}u_{22}\cdots u_{nn}$$

3.2 矩阵三角分解法

- 3.2.2 矩阵的直接三角分解 （无消元步骤）

  Dolittle分解基本步骤



计算顺序：按框由外向内，先行后列

计算方法：

$$u_{ri} = a_{ri} - \sum_{k=1}^{r-1} l_{rk} u_{ki}, i = r, r+1, \cdots, n$$

$$l_{ir} = \left(a_{ir} - \sum_{k=1}^{r-1} l_{ik} u_{kr}\right) / u_{rr}, i = r+1, r+2, \cdots, n$$

3.2 矩阵三角分解法

- 3.2.2 矩阵的直接三角分解

例3-7 对矩阵A做Dolittle分解，计算行列式

$$A=\begin{pmatrix}2 & 2 & 3\\ 4 & 7 & 7\\ -2 & 4 & 5\end{pmatrix}$$

$$u_{11}=2, u_{12}=2, u_{13}=3$$

$$l_{21}=\frac{4}{2}=2, l_{31}=\frac{-2}{2}=-1$$

$$u_{22}=a_{22}-l_{21}u_{12}=7-2\times 2=3$$

$$u_{23}=a_{23}-l_{21}u_{13}=7-2\times 3=1$$

$$l_{32}=\frac{1}{u_{22}}(a_{32}-l_{31}u_{12})=\frac{1}{3}(4+1\times 2)=2$$

$$u_{33}=a_{33}-l_{31}u_{13}-l_{32}u_{23}=5-(-1)3-2\times 1=6$$

$$L=\begin{pmatrix}1 & & \\ 2 & 1 & \\ -1 & 2 & 1\end{pmatrix}, U=\begin{pmatrix}2 & 2 & 3\\ & 3 & 1\\ & & 6\end{pmatrix}$$

$$|A|=u_{11}u_{22}u_{33}=2\times 3\times 6=36$$

(2)	2	(2)	2	(3)	3
(4)	2	(7)	3	(7)	1
(−2)	−1	(4)	2	(5)	6

3.2 矩阵三角分解法

- 3.2.3 用矩阵三角分解法解线性方程组

$$Ax = LUx = b$$

转化为两步执行

$$\begin{aligned} Ly &= b \\ Ux &= y \end{aligned}$$

$$\begin{cases} y_1 = b_1 \\ y_i = b_i - \sum_{k=1}^{i-1} l_{ik}\, y_k, i = 2,3,\cdots,n \end{cases}$$

$$\begin{cases} x_n = y_n / u_{nn} \\ x_i = \left(y_i - \sum_{k=i+1}^{n} u_{ik}\, x_k\right) / u_{ii}\,, i = n-1,\cdots,2,1 \end{cases}$$

3.2 矩阵三角分解法

- 3.2.3 用矩阵三角分解法解线性方程组

例3-8

$$\begin{pmatrix} 2 & 2 & 3 \\ 4 & 7 & 7 \\ -2 & 4 & 5 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \\ -7 \end{pmatrix}$$

$$L = \begin{pmatrix} 1 & & \\ 2 & 1 & \\ -1 & 2 & 1 \end{pmatrix}, U = \begin{pmatrix} 2 & 2 & 3 \\ & 3 & 1 \\ & & 6 \end{pmatrix}$$

$$\begin{pmatrix} 1 & & \\ 2 & 1 & \\ -1 & 2 & 1 \end{pmatrix}\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \\ -7 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 2 & 3 \\ & 3 & 1 \\ & & 6 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 3 \\ -5 \\ 6 \end{pmatrix}$$

$$y_1 = 3,$$
$$y_2 = 1 - 2 \times 3 = -5,$$
$$y_3 = -7 - (-1)3 - 2(-5) = 6$$

$$x_3 = 1, x_2 = -2, x_1 = 2$$

3.2　矩阵三角分解法

- 3.2.3 用矩阵三角分解法解线性方程组

广义矩阵形式

方程组 $Ax=b$ 化为 $Ux=y$ 时，A通过LU分解得到U，b 通过LU分解得到y。则$Ax=b$化为$Ux=y$时，将b增广到A进行LU分解得到y。

(a_{11})	u_{11}	(a_{12})	u_{12}	$\cdots$	(a_{1n})	u_{1n}	(b_1)	y_1
(a_{21})	l_{21}	(a_{22})	l_{22}	$\cdots$	(a_{2n})	u_{2n}	(b_2)	y_2
(a_{31})	l_{31}	(a_{32})	l_{32}	$\cdots$	(a_{3n})	u_{3n}	(b_3)	y_3
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
				$\cdots$	(a_{nn})	u_{nn}	(b_n)	y_n

3.2 矩阵三角分解法

- 3.2.3 用矩阵三角分解法解线性方程组

$$A=\begin{pmatrix}2 & 2 & 3\\ 4 & 7 & 7\\ -2 & 4 & 5\end{pmatrix}$$

$$\begin{cases}2x_1+2x_2+3x_3=3\\ 4x_1+7x_2+7x_3=1\\ -2x_1+4x_2+5x_3=-7\end{cases}$$

(2) 2	(2) 2	(3) 3	(3) 3
(4) 2	(7) 3	(7) 1	(1)-5
(-2) -1	(4) 2	(5) 6	(-7)6

$$L=\begin{pmatrix}1 & & \\ 2 & 1 & \\ -1 & 2 & 1\end{pmatrix},U=\begin{pmatrix}2 & 2 & 3\\ & 3 & 1\\ & & 6\end{pmatrix},y=\begin{pmatrix}3\\ -5\\ 6\end{pmatrix}$$

$$\begin{cases}2x_1+2x_2+3x_3=3\\ 3x_2+x_3=-5\\ 6x_3=6\end{cases}\qquad\begin{cases}x_3=1\\ x_2=-2\\ x_1=2\end{cases}$$

3.2 矩阵三角分解法

- 3.2.3 用矩阵三角分解法解线性方程组

线性方程组系情况

$$\begin{cases} Ax = b_1 \\ Ax = b_2 \\ Ax = b_3 \\ \quad\vdots \\ Ax = b_n \end{cases}$$

$A \in R^{n\times n}, x, b_1, b_2, b_3, \cdots \in R^n$

当右端项为单位阵I各列向量时

$$AX = I \qquad A、X、I \in R^{n\times n}$$

可求逆矩阵

$$X = A^{-1}$$

3.2 矩阵三角分解法

• 3.2.3 用矩阵三角分解法解线性方程组

例3-10 线性方程组系方法求逆矩阵 $A=\begin{pmatrix}1&1&-1\\1&2&-2\\-2&1&1\end{pmatrix}$

利用Dolittle分解

(1)	1	(1)	1	(−1)	−1	(1)	1	(0)	0	(0)	0
(1)	1	(2)	1	(−2)	−1	(0)	−1	(1)	1	(0)	0
(−2)	−2	(1)	3	(1)	2	(0)	5	(0)	−3	(1)	1

$$\begin{pmatrix}1&1&-1\\&1&-1\\&&2\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\end{pmatrix}=\begin{pmatrix}1\\-1\\5\end{pmatrix}\quad a_1=x_1=\begin{cases}2\\1.5\\2.5\end{cases}$$

$$\begin{pmatrix}1&1&-1\\&1&-1\\&&2\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\end{pmatrix}=\begin{pmatrix}0\\1\\-3\end{pmatrix}\quad a_2=x_2=\begin{cases}-1\\-0.5\\-1.5\end{cases}$$

$$\begin{pmatrix}1&1&-1\\&1&-1\\&&2\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\end{pmatrix}=\begin{pmatrix}0\\0\\1\end{pmatrix}\quad a_3=x_3=\begin{cases}0\\0.5\\0.5\end{cases}$$

$$A^{-1}=\begin{pmatrix}2&-1&0\\1.5&-0.5&0.5\\2.5&-1.5&0.5\end{pmatrix}$$

3.2 矩阵三角分解法

- 3.2.3 用矩阵三角分解法解线性方程组

克罗特分解情况类似

$$\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix} = \begin{pmatrix} l_{11} & & & \\ l_{21} & l_{22} & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{pmatrix} \begin{pmatrix} 1 & u_{12} & u_{13} & \cdots & u_{1n} \\ & 1 & u_{23} & \cdots & u_{2n} \\ & & \ddots & & \vdots \\ & & & & 1 \end{pmatrix}$$

$$l_{ir} = a_{ir} - \sum_{k=1}^{r-1} l_{ik} u_{kr}, i = r, r+1, \cdots, n$$

$$u_{ri} = \left(a_{ri} - \sum_{k=1}^{r-1} l_{rk} u_{ki} \right) / l_{rr}, i = r+1, r+2, \cdots, n$$

方程组分为两步求解

$$\begin{cases} y_1 = b_1 / l_{11} \\ y_i = \left(b_i - \sum_{k=1}^{i-1} l_{ik} y_k \right) / l_{ii}, i = 2,3, \cdots, n \end{cases}$$

$$\begin{cases} x_n = y_n \\ x_i = y_i - \sum_{k=i+1}^{n} u_{ik} x_k, \quad i = n-1, \cdots, 2,1 \end{cases}$$

3.2 矩阵三角分解法

- 3.2.3 用矩阵三角分解法解线性方程组

例3-11

$$\begin{pmatrix} 2 & -1 & 1 \\ 4 & 2 & 1 \\ 2 & 1 & 2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 5 \\ 4 \\ 5 \end{pmatrix}$$

$$\begin{pmatrix} 2 & & \\ 4 & 4 & \\ 2 & 2 & \frac{3}{2} \end{pmatrix}\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 5 \\ 4 \\ 5 \end{pmatrix}$$

$$y_1 = \frac{5}{2}, \quad y_2 = -\frac{3}{2}, \quad y_3 = 2$$

$$\begin{pmatrix} 1 & -\frac{1}{2} & \frac{1}{2} \\ & 1 & -\frac{1}{4} \\ & & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} \frac{5}{2} \\ -\frac{3}{2} \\ 2 \end{pmatrix}$$

$$x_1 = 2, x_2 = -1, x_1 = 1$$

(2)	2	(−1)	$-\frac{1}{2}$	(1)	$\frac{1}{2}$	(5)	$\frac{5}{2}$
(4)	4	(2)	4	(1)	$-\frac{1}{4}$	(4)	$-\frac{3}{2}$
(2)	2	(1)	2	(2)	$\frac{3}{2}$	(5)	2

3.6.1 迭代原理

- 定义3-9 对线性方程组 $Ax=b$ 用等价方程 $\boldsymbol{x}=\boldsymbol{B}\boldsymbol{x}+f$ 建立迭代格式

$$x^{(k+1)}=Bx^{(k)}+f, k=0,1,\cdots$$

并逐步求解的方法叫迭代法。

若 $\lim\limits_{k\to+\infty} x^{(k)}=x^{*}$ ，称迭代法收敛

否则称迭代法发散

3.6.2 Jacobi迭代

对一般线性方程组 $\sum_{j=1}^{n} a_{ij} x_j = b_i, i = 1,2,\cdots,n$

设系数阵对角元素非零，按照Jacobi迭代格式得

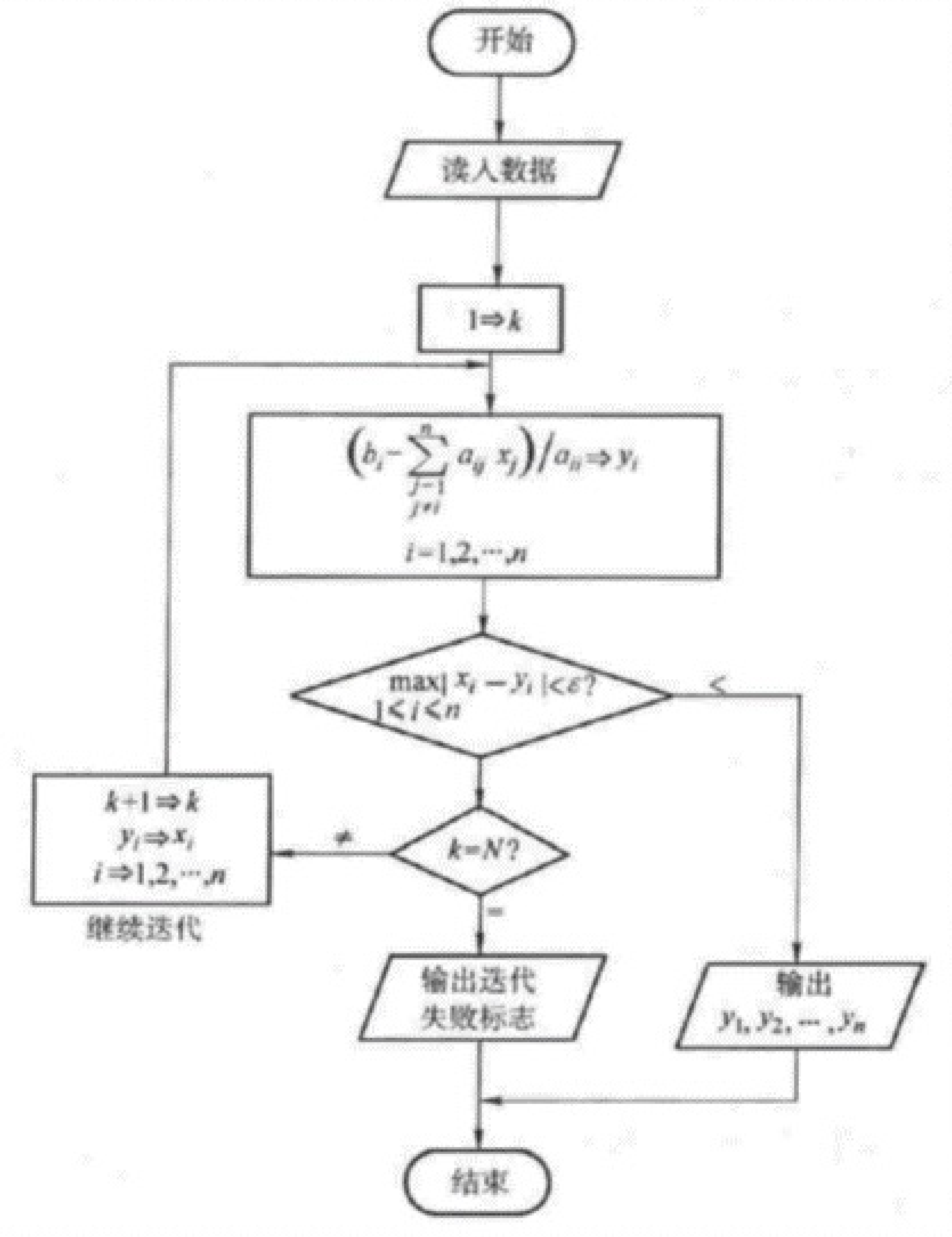
$$x_i = \frac{1}{a_{ii}}\left(b_i - \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij} x_j\right), i = 1,2,\cdots,n$$

可以据此进行如下迭代

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij} x_j^{(k)}\right), i = 1,2,\cdots,n$$

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}\right), \quad i = 1,2,\cdots,n$$

迭代精度取决于偏差 $e = \max_{1\le i\le n} |x_i - y_i|$



3.6.2 Jacobi迭代

例3-27 求解方程组 $\begin{cases} 10x_1 - 2x_2 - x_3 = 3 \\ -2x_1 + 10x_2 - x_3 = 15 \\ -x_1 - 2x_2 + 5x_3 = 10 \end{cases}$

按对角项分离未知量 $\begin{cases} x_1 = 0.2x_2 + 0.1x_3 + 0.3 \\ x_2 = 0.2x_1 + 0.1x_3 + 1.5 \\ x_3 = 0.2x_1 + 0.4x_2 + 2 \end{cases}$

每次Jacobi迭代只用到上一次迭代值

其迭代公式为

$$\begin{cases} x_1^{(k+1)} = 0.2x_2^{(k)} + 0.1x_3^{(k)} + 0.3 \\ x_2^{(k+1)} = 0.2x_1^{(k)} + 0.1x_3^{(k)} + 1.5 \\ x_3^{(k+1)} = 0.2x_1^{(k)} + 0.4x_2^{(k)} + 2 \end{cases}$$

迭代法将线性方程组的联立求解转化为彼此独立的线性表达式的求值过程，使问题大大简化

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
						k	0	1	2	3	4	5	6	7	8	9
10	-2	-1		3		x1	0	0.3	0.8	0.918	0.9716	0.9894	0.9962	0.9986	0.9995	0.9998
-2	10	-1		15		x2	0	1.5	1.76	1.926	1.97	1.9897	1.9961	1.9986	1.9995	1.9998
-1	-2	5		10		x3	0	2	2.66	2.864	2.954	2.9823	2.9938	2.9977	2.9992	2.9997

3.6.3 Gauss-Seidel迭代

对一般线性方程组 $\sum_{j=1}^{n} a_{ij} x_j = b_i, i = 1,2,\cdots,n$

设系数阵对角元素非零，按照Gauss-Seidel迭代格式得

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}\right), i = 1,2,\cdots,n$$

每次G-S迭代利用最新迭代值进行推进，可能会有助于加速收敛

编程时，可以将新值$x_j^{(k+1)}$存放在老值$x_j^{(k)}$所占用的单元内，公式变为

$$\left(b_i - \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij} x_j\right)/a_{ii} \to x_i, i = 1,2,\cdots,n$$

3.6.3 Gauss-Seidel迭代

例3-27 Gauss-Seidel迭代

$$
\begin{cases}
x_1^{(k+1)} = 0.2x_2^{(k)} + 0.1x_3^{(k)} + 0.3 \\
x_2^{(k+1)} = 0.2x_1^{(k+1)} + 0.1x_3^{(k)} + 1.5 \\
x_3^{(k+1)} = 0.2x_1^{(k+1)} + 0.4x_2^{(k+1)} + 2
\end{cases}
$$

A	B	C	D	E	F	G	H	I	J	K	L	M	N
						k	0	1	2	3	4	5	6
10	-2	-1		3		x1	0	0.3	0.88	0.9843	0.9978	0.9997	1
-2	10	-1		15		x2	0	1.56	1.944	1.9922	1.9989	1.9999	2
-1	-2	5		10		x3	0	2.684	2.954	2.9938	2.9991	2.9999	3

3.6.4 松弛法

迭代方式、问题性状都会影响迭代速度

迭代法的实际计算量难于估计

借助松弛方法可以提高G-S迭代速度

$$\tilde{x}_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}\, x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}\, x_j^{(k)}\right)$$

在前一步结果与G-S最新结果之间通过适当加权平均

可以获得更好的近似值

$$x_i^{(k+1)} = \omega \tilde{x}_i^{(k+1)} + (1-\omega) x_i^{(k)}$$

$$x_i^{(k+1)} = (1-\omega) x_i^{(k)} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}\, x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}\, x_j^{(k)}\right)$$

松弛因子

$0 < \omega < 1$ 亚松弛

$1 < \omega < 2$ 超松弛

3.6.4 松弛法

例3-28 用SOR法解方程组

$$\begin{pmatrix} 4 & -2 & -4 \\ -2 & 17 & 10 \\ 4 & 10 & 9 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 10 \\ 3 \\ -7 \end{pmatrix}$$

迭代公式为

$$\begin{cases} x_1^{(k+1)} = (1-\omega)x_1^{(k)} + \dfrac{\omega}{4}\left(10 + 2x_2^{(k)} + 4x_3^{(k)}\right) \\ x_2^{(k+1)} = (1-\omega)x_2^{(k)} + \dfrac{\omega}{17}\left(3 + 2x_1^{(k+1)} - 10x_3^{(k)}\right) \quad k = 0,1,2,\cdots \\ x_3^{(k+1)} = (1-\omega)x_3^{(k)} + \dfrac{\omega}{9}\left(-7 - 4x_1^{(k+1)} - 10x_2^{(k+1)}\right) \end{cases}$$

3.6.5 迭代公式的矩阵表示

线性方程组 $Ax=b$ 的系数矩阵 $A=\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \vdots & \\ a_{n1} & \cdots & a_{nn} \end{pmatrix}$

可以分解为

$$A=-L+D-U$$
$$=\begin{pmatrix} 0 & & & & \\ a_{21} & 0 & & 0 & \\ a_{31} & a_{32} & 0 & & \\ \vdots & \vdots & & \ddots & \\ a_{n1} & a_{n2} & \cdots & a_{n-1} & 0 \end{pmatrix}+\begin{pmatrix} a_{11} & & & \\ & a_{22} & & 0 \\ & & \ddots & \\ & 0 & & a_{nn} \end{pmatrix}+\begin{pmatrix} 0 & a_{12} & a_{13} & \cdots & a_{1n} \\ & & a_{23} & \cdots & a_{2n} \\ & & \ddots & & \vdots \\ & 0 & & & a_{(n-1)a} \\ & & & & 0 \end{pmatrix}$$

严格下三角阵　　对角阵　　严格上三角阵

三种迭代格式总结

表达式：$x^{(k+1)} = Bx^{(k)} + f \qquad k = 0,1,\cdots,n$

其中 B 是迭代矩阵，f 为常数项。

雅可比迭代： $J = B = I - D^{-1}A$， $f = D^{-1}b$ 。

高斯-赛德尔迭代法：$G = (D-L)^{-1}U$ ， $f = (D-L)^{-1}b$ 。

松 弛 法 ： $L_\omega = (D-\omega L)^{-1}((1-\omega)D + \omega U)$ ，$f = \omega(D-\omega L)^{-1}b$ 。

4

插值法

插值定义

插值：已知某些(有限)点的函数值求其余点的函数值。

定义　函数 $y=f(x)$ 在区间$[a,b]$上有函数值 $y_i = f(x_i)$，$i = 0,1,\cdots,n$

即

$$\begin{array}{ccccc} x_0 & x_1 & x_2 & \ldots\ldots & x_n \\ y_0 & y_1 & y_2 & \ldots\ldots & y_n \end{array}$$

其中$x_0,x_1,x_2,\ldots,x_n$是区间$[a,b]$上的互异点，要构造一个简单的函数$\varphi(x)$作为 $f(x)$ 的近似表达式，使满足

$$\boldsymbol{\varphi(x_i) = y_i, i = 0,1,\cdots,n} \quad \text{(插值条件)}$$

这类问题称为插值问题。

$f(x)$ -----被插值函数,　　　　**$\varphi(x)$-----$f(x)$的插值函数,**

$x_0,x_1,x_2,\ldots,x_n$ -----插值节点，$[a,b]$称为插值区间

求插值函数的方法称为插值法。

取$x\in[a,b]$，可计算$f(x)$ 的近似值$\varphi(x)$, x 称为插值点。

4.1.1 代数插值

代数多项式插值:当选择代数多项式作为插值函数时，称为代数多项式插值。

定义(代数多项式插值) 设函数$y=f(x)$在$[a,b]$上已知 $n+1$个点

$a \le x_0 < x_1 < \ldots\ldots < x_n \le b$上的函数值

$$y_0, y_1, \ldots\ldots, y_n$$

求一个次数不高于n的代数多项式

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

使满足插值条件 $P(x_i) = y_i, \quad i = 0, 1, \cdots, n$

称$P(x)$为 $f(x)$ 的n次插值多项式。

代数插值的特点：满足在n+1个节点上插值多项式$P(x)$ 的次数不超过n次。

4.1.2 代数多项式插值的唯一性

定理 $n+1$个互异节点处满足插值条件$P(x_i)=y_i \quad i=0,1,\cdots,n$ 的n次代数多项式是唯一的。

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

唯一性说明不论用哪种方法构造的插值多项式只要满足相同的插值条件，其结果都是互相恒等的。

推论 当$f(x)$是次数不超过n的多项式时，其n次插值多项式就是$f(x)$本身。

例 在直线上取两个点进行插值，插值多项式就是这条直线。在二次抛物线上取三个点进行插值，插值多项式就是这条二次抛物线。

在直线上取三个点进行插值，插值多项式还是这条直线。在二次抛物线上取四个点进行插值，插值多项式也是这条二次抛物线。

4.2.2 拉格朗日插值多项式

定理　若

$$l_k(x)=\prod_{\substack{i=0\\ i\neq k}}^{n}\frac{x-x_i}{x_k-x_i}$$

$$=\frac{(x-x_0)(x-x_1)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_0)(x_k-x_1)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}\qquad k=0,1,\cdots,n$$

则

$$\begin{cases} l_k(x_i)=1\ \ (k=i) & , \\ l_k(x_i)=0\ \ (k\neq i) & , \quad i、k=0,1,\cdots,n \end{cases}$$

$l_k(x)$称为n次拉格朗日插值基函数。

基函数的特点

1. 基函数的个数等于节点数。

2. $n+1$个节点的基函数是n次代数多项式。

3. 基函数和每一个节点都有关。节点确定，基函数就唯一的确定。

4. 基函数和被插值函数无关。

5. 基函数在节点取值为0或1，即第i个节点x_i基函数$l_i(x_i)$为1，其余节点$l_k(x_i)$均为0。

定理　n次拉格朗日插值多项式

$$L(x)=\sum_{k=0}^{n} y_k l_k(x)=y_0 l_0(x)+y_1 l_1(x)+\cdots+y_n l_n(x)$$

其中 $l_k(x)=\prod_{\substack{i=0\\ i\neq k}}^{n}\frac{x-x_i}{x_k-x_i}$

拉格朗日三次多项式

$$L(x)=\sum_{k=0}^{3} y_k l_k(x)$$

$$=\frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)}y_0+\frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}y_1$$

$$+\frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)}y_2+\frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}y_3$$

4.2.3 插值余项和误差估计

余项(截断误差)　$R(x) = f(x) - L(x)$

定理　设函数$f(x)$在包含节点$x_0, x_1, \dots, x_n$的区间$[a,b]$上有$n+1$阶导数，则

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

其中　$\omega(x) = \prod_{i=0}^{n}(x - x_i) = (x - x_0)(x - x_1)\cdots(x - x_n)$ ，$\xi \in (a,b)$

带余项的拉格朗日插值多项式

$$f(x) = L(x) + R(x) = L(x) + \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega(x)$$

拉格朗日插值特点

1、当插值点$x \in (a,b)$时称为内插，否则称为外插。内插的精度高于外插的精度。

2、余项公式含有n+1阶导数，一般要求$f(x)$足够光滑。

3、插值多项式$P(x)$只与数据点$(x_i, f(x_i))$有关，与节点排列顺序无关，与$f(x)$无关，但余项$R(x)$与$f(x)$有关。

4、若$f(x)$是次数不超过n的多项式，取n+1个节点插值时，插值多项式就是其自身。

5、基函数之和为1，即 $l_0(x)+l_1(x)+\cdots+l_n(x)=1$

6、次数≤n的多项式，其n次拉格朗日插值多项式就是其本身。

例 设 $f(x)=x^4$，用拉格朗日余项定理写出以 $-1,0,1,3$ 为节点的三次插值多项式。

解

$$L(x)=f(x)-\frac{f^{(n+1)}(\xi)}{(n+1)!}\omega(x)$$

$$=x^4-\frac{4!}{4!}(x+1)(x-0)(x-1)(x-3)$$

$$L(x)=x^4-x(x+1)(x-1)(x-3)=x^4-(x^3-x)(x-3)$$

三次插值多项式 $L(x)=3x^3+x^2-3x$

线性插值多项式的截断误差

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

ξ是在包含x, x_0, x_1的区间内某数。

推论 设函数$f(x)$在区间$[x_0, x_1]$上二阶导数连续，并记 $M_2 = \max\limits_{x_0 \leqslant x \leqslant x_1} |f''(x)|$，则$f(x)$过点$(x_0, f(x_0))$和$(x_1, f(x_1))$的线性插值余项$R(x)$有上届估计式

$$|R(x)| \leqslant \frac{M_2}{8}(x_1 - x_0)^2, \quad x \in [x_0, x_1]$$

4.4 牛顿插值

牛顿插值解决拉格朗日插值为提高精度增加插值节点时，要重新计算全部基函数，整个插值多项式的结构都会改变的问题。

逐次线性插值虽然有“承袭性”，但其算式是递推型的，不便于进行理论上的分析，牛顿插值是一种具有“承袭性”的插值，而且表达式也很简单，便于进行理论分析。

差商及其性质,牛顿插值多项式。

4.4.1 差商（均差）及其性质

1 差商的定义 差商是函数增量与其自变量的增量的比(商)。

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

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

函数f的n阶差商

高阶差商是由比它低一阶的两个差商的差商组成。

$$f[x_k,x_{k+1},\cdots,x_{k+n-1},x_{k+n}]=\frac{f[x_{k+1},\cdots,x_{k+n-1},x_{k+n}]-f[x_k,x_{k+1},\cdots,x_{k+n-1}]}{x_{k+n}-x_k}$$

共n+1个节点

差商表

x_k	零阶差商	一阶差商	二阶差商	三阶差商
x_0	$f(x_0)$			
x_1	$f(x_1)$	$f[x_0,x_1]$		
x_2	$f(x_2)$	$f[x_1,x_2]$	$f[x_0,x_1,x_2]$	
x_3	$f(x_3)$	$f[x_2,x_3]$	$f[x_1,x_2,x_3]$	$f[x_0,x_1,x_2,x_3]$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

4.4.2 差商的性质

（1） n 阶差商 $f[x_0, x_1, \cdots, x_n]$ 是函数值 $f(x_0), f(x_1), \cdots, f(x_n)$ 的线性组合，即

$$f[x_0,x_1,\cdots,x_n]=\sum_{k=0}^{n}\frac{f(x_k)}{(x_k-x_0)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}$$

$$f[x_0, x_1, \cdots, x_n] = \sum_{k=0}^{n} \frac{f(x_k)}{\omega'(x_k)}$$

（2） 差商具有对称性：任意改变节点的次序差商值不变。例如 $f[0,2,4]=f[2,0,4]=f[4,2,0]$等。

（3） 若 $f[x,x_0,\cdots,x_k]$ 是x的m次多项式，则 $f[x,x_0,\cdots,x_k,x_{k+1}]$ 是x的m-1次多项式。

（4） 若$f(x)$是n次多项式，则 $f[x,x_0,x_1,\cdots,x_n]$ 恒为0 。

4.4.2 牛顿插值公式

1、牛顿插值公式的建立

x_k	零阶差商	一阶差商	二阶差商	乘积因子
x_0	$f(x_0)$			1
x_1	$f(x_1)$	$f[x_0,x_1]$		$(x-x_0)$
x_2	$f(x_2)$	$f[x_1,x_2]$	$f[x_0,x_1,x_2]$	$(x-x_0)(x-x_1)$
x_3	$f(x_3)$	$f[x_2,x_3]$	$f[x_1,x_2,x_3]$	……
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

2、牛顿插值的特点

(1) $N(x)$次数不超过n次，项数不超过n+1项。各项系数是差商表上对角线的各阶差商值。

(2) $N(x)$满足插值条件，在节点上$f(x_i)=N(x_i)$。

(3) 增加一个节点，只需增加一项。

$$n=1,\ N(x)=f(x_0)+f[x_0,x](x-x_0)$$

$$n=2,\ N(x)=f(x_0)+f[x_0,x_1](x-x_0)+f[x_0,x_1,x](x-x_0)(x-x_1)$$

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

$$R(x)=f[x_0,x_1,\cdots,x_n,x]\ (x-x_0)(x-x_1)\cdots(x-x_n)$$

$$f(x)=N(x)+R(x)$$

例 4-13 已知函数 $f(x)$ 在节点 $x=0$，1，2，4 处的函数值 $f(x)$ 分别是 3，6，11，51，求二次和三次牛顿插值多项式并计算 $f(0.5)$ 的近似值。

解 根据给定的函数值构造差商表

x_i	$f(x_i)$	1 阶	2 阶	3 阶	因子
0	3				1
1	6	3			x
2	11	5	1		$x(x-1)$
4	51	20	5	1	$x(x-1)(x-2)$

二次牛顿插值多项式选最接近 0.5 的三个节点 0，1，2 组成，即

$$N_2(x) = 3 + 3x + x(x-1) = x^2 + 2x + 3$$

由此，有

$$f(0.5) \approx N_2(0.5) = 4.25$$

三次牛顿插值多项式

$$N_3(x) = N_2(x) + x(x-1)(x-2) = x^3 - 2x^2 + 4x + 3$$

$$f(0.5) \approx N_3(0.5) = 4.625$$

4.4.3 差商和导数

若 $f(x)$ 在$[a,b]$ 上存在n 阶导数，由余项表达式可得，n 阶差商与导数有如下关系

$$f[x_0,x_1,\cdots,x_n]=\frac{f^{(n)}(\xi)}{n!}\qquad \xi\in[x_0,x_n]$$

例　若函数 $f(x)=x^7+x^4+3x+5$, 求 $f[2^0,2^1,\cdots,2^7]$和 $f[2^0,2^1,\cdots,2^8]$

解　因 $f[x_0,x_1,\ldots\ldots x_n]=\dfrac{f^{(n)}(\xi)}{n!}$，有

$$f[2^0,2^1,\ldots\ldots 2^7]=\frac{f^{(7)}(\xi)}{7!}=\frac{7!}{7!}=1$$

$$f[2^0,2^1,\ldots\ldots 2^8]=\frac{f^{(8)}(\xi)}{8!}=\frac{0}{8!}=0$$

重节点时差商和函数导数的关系、该关系解决了含有给定函数值和导数值组合的差商的计算

设已知数据表，其中0是二重节点，1是三重节点

x	0	1
$f(x)$	3	5
$f'(x)$	4	6
$f''(x)$		7

建立差商表

x	$f(x)$	一阶差商	二阶差商	三阶差商	四阶差商
0	3				
0	3	$f[0,0]=f'(0)=4$			
1	5	$f(0,1)=\frac{5-3}{1-0}=2$	$f(0,0,1)=\frac{2-4}{1-0}=-2$		
1	5	$f(1,1)=f'(1)=6$	$f(0,1,1)=\frac{6-2}{1-0}=4$	6	
1	5	$f(1,1)=f'(1)=6$	$f(1,1,1)=\frac{1}{2}f''(1)=\frac{7}{2}$	$-\frac{1}{2}$	$-\frac{13}{2}$

$$N_4(x)=3+4(x-0)-2(x-0)(x-0)+6(x-0)(x-0)(x-1)-\frac{13}{2}(x-0)(x-0)(x-1)(x-1)$$

4.4.4 差分

当插值节点等距分布时，被插值函数的平均变化率与自变量的区间无关，差商就可用差分来表示，这时牛顿插值公式的形式更简单。

向前差分

设相邻两节点的距离 h 是常数，并称之为步长，则有 $x_k=x_0+kh$，$k=0, 1, \cdots, n$，被插值函数 $y=f(x)$ 在插值节点上的值 $f(x_k)=y_k$，$k=0, 1, \cdots, n$，将函数 $y=f(x)$ 在区间 $[x_k, x_{k+1}]$ 上的增量 $y_{k+1}-y_k$ 叫作函数在节点 x_k 的一阶差分，记作 $\Delta y_k=y_{k+1}-y_k$，这样 $f(x)$ 在各节点的一阶差分

$$\Delta y_0 = y_1 - y_0,\ \Delta y_1 = y_2 - y_1,\ \cdots,\ \Delta y_{n-1} = y_n - y_{n-1} \tag{4-26}$$

一阶差分的差分叫作函数的二阶差分

$$\Delta^2 y_0 = \Delta y_1 - \Delta y_0 = y_2 - 2y_1 + y_0$$

$$\Delta^2 y_1 = \Delta y_2 - \Delta y_1 = y_3 - 2y_2 + y_1$$

$$\cdots$$

$$\Delta^2 y_{n-2} = \Delta y_{n-1} - \Delta y_{n-2} = y_n - 2y_{n-1} + y_{n-2}$$

写成一般形式

$$\Delta^2 y_k = \Delta y_{k+1} - \Delta y_k = y_{k+2} - 2y_{k+1} + y_k,\quad k = 0,\ 1,\ \cdots \tag{4-27}$$

一般地，n 阶差分定义为

$$\Delta^n y_k = \Delta^{n-1} y_{k+1} - \Delta^{n-1} y_k \tag{4-28}$$

写成差分表的形式，用以方便地计算各阶差分

表 4-5

x_k	y_k	Δy_k	$\Delta^2 y_k$	$\Delta^3 y_k$	$\Delta^4 y_k$	…
x_0	y_0					
x_1	y_1	Δy_0				
x_2	y_2	Δy_1	$\Delta^2 y_0$			
x_3	y_3	Δy_2	$\Delta^2 y_1$	$\Delta^3 y_0$		
x_4	y_4	Δy_3	$\Delta^2 y_2$	$\Delta^3 y_1$	$\Delta^4 y_0$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	

$$N(x_0 + th) = y_0 + \frac{t}{1!}\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_0 + \cdots + \frac{t(t-1)\cdots[t-(n-1)]}{n!}\Delta^n y_0$$

$$= \sum_{i=0}^{n} C_t^i \Delta^i y_0$$

$$R(x) = C_t^{n+1} h^{n+1} f^{(n+1)}(\xi) = \frac{t(t-1)\cdots(t-n)}{(n+1)!} h^{n+1} f^{(n+1)}(\xi)$$

差分与差商的关系

$$\Delta^n y_0 = n!\ h^n f[x_0, x_1, \cdots, x_n]$$

利用差商和导数的关系，可推出差分和导数的关系

$$\Delta^n y_0 = h^n f^{(n)}(\xi),\ \xi \in [x_0,\ x_n]$$

向前差分 $\Delta y_0 = y_1 - y_0$, $\Delta^n y_k = \Delta^{n-1} y_{k+1} - \Delta^{n-1} y_k$

向后差分 $\nabla y_0 = y_0 - y_{-1}$; $\nabla^k y_i = \nabla^{k-1} y_i - \nabla^{k-1} y_{i-1}$

中心差分 $\delta y_0 = y_{\frac{1}{2}} - y_{-\frac{1}{2}}$ $\delta^k y_i = \delta^{k-1} y_{i+\frac{1}{2}} - \delta^{k-1} y_{i-\frac{1}{2}}$

向后差分和中心差分都可以转化为向前差分

$$\nabla^n y_k = \Delta^n y_{k-n} \qquad \delta^n y_k = \nabla^n y_{k-\frac{n}{2}}$$

例 已知$f(x)=\sin(x)$的数值如下，用向前牛顿插值公式计算sin(0.578 91)的近似值。

x	0.4	0.5	0.6	0.7
$\sin x$	0.389 42	0.479 43	0.564 64	0.644 22

解 构造差分表，如表4-8所示。

表 4-8

x	$\sin x$	Δ	Δ^2	Δ^3
0.4	0.389 42			
0.5	0.479 43	0.090 01		
0.6	0.564 64	0.085 21	0.004 80	
0.7	0.644 22	0.079 58	−0.005 63	−0.000 83

使用向前插值公式，取$x_0=0.5$，$x_1=0.6$，$x_2=0.7$，$x=x_0+th$，$h=0.1$，$t=\dfrac{x-x_0}{h}=0.789\ 1$，于是

$$N(0.578\ 91)=y_0+t\Delta y_0+t(t-1)\frac{\Delta^2 y_0}{2}$$

$$=0.479\ 43+0.789\ 1\times0.085\ 21+0.789\ 1\times(0.789\ 1-1)\times\frac{(-0.005\ 63)}{2}$$

$$=0.547\ 14$$

4.5 反插值

已知$(x_i, f(x_i)), i = 0,1,\cdots,n$，求某一 x 处 $f(x)$ 的近似值，这是插值问题。

已知$(x_i, f(x_i)), i = 0,1,\cdots,n$，若要求 $f(x)$ 等于某一值时 x 的近似值，这与插值问题恰好相反，这便是反插值问题。

用反插值时要求函数 $y=f(x)$的反函数存在，即要求 $y=f(x)$单调或节点x_i的函数值$f(x_i)$严格单调排列。

当取 y=0,求 Q(y) 的值 Q(0) 时，即是求$f(x)$零点的问题。

例　连续函数 $f(x)$ 在 $x=-3,-1,0,4$ 时的值分别是 $-1,0,2,10$ 。

（1）用拉格朗日二次插值计算 $f(3)$ 。

（2）用拉格朗日三次插值确定 $f(x)=1$ 时， x 的近似值。

解　（1）根据已知条件列表

x	-3	-1	0	4	3
$f(x)$	-1	0	2	10	?

取靠近 3 的三个节点-1，0，4，作拉格朗日二次插值

$$p(x)=\frac{(x-0)(x-4)}{(-1-0)(-1-4)}\times 0+\frac{(x+1)(x-4)}{(0+1)(0-4)}\times 2+\frac{(x+1)(x-0)}{(4+1)(4-0)}\times 10$$ 将

$x=3$，代入

$$p(3)=\frac{(3-0)(3-4)}{(-1-0)(-1-4)}\times 0+\frac{(3+1)(3-4)}{(0+1)(0-4)}\times 2+\frac{(3+1)(3-0)}{(4+1)(4-0)}\times 10=8$$

（2）由于 $f(x)$ 是单调连续函数，用反插值，将函数表转换成反函数表

$y=f(x)$	-1	0	2	10	1
$x=f^{-1}(y)$	-3	-1	0	4	?

5

曲线拟合的最小二乘法

5.1.1 最小二乘原理

- 拟合函数类的选择

该函数空间由一组线性无关的给定函数 $\varphi_0, \varphi_1, \cdots, \varphi_n$ 张成

$$\varphi(x) \in \Phi = Span\{\varphi_0, \varphi_1, \cdots, \varphi_n\}$$

其线性拟合模型表示为

$$\varphi(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \cdots + a_n\varphi_n(x)$$

最小二乘的拟合目标是：

找到拟合参数序列$\{a_i\}, i = 0,1, \ldots, n$，使得

拟合函数

$$\varphi^*(x) = a_0^*\varphi_0(x) + a_1^*\varphi_1(x) + \cdots + a_n^*\varphi_n(x) = \sum_{i=0}^{n} a_i^* \varphi_i(x)$$

满足

$$\|\delta\|_2^2 = \sum_{i=1}^{m} [\varphi^*(x_i) - f(x_i)]^2 = \min_{\varphi(x)\in\Phi} \sum_{i=1}^{m} [\varphi(x_i) - f(x_i)]^2$$

最小二乘法曲线拟合要求误差平方和最小

5.1.1 最小二乘原理

- 最小二乘一般解法

引入记号

$$\begin{cases}(\varphi_k,\varphi_j)=\sum_{i=1}^{m}\varphi_k(x_i)\varphi_j(x_i)\\ (\varphi_k,f)=\sum_{i=1}^{m}\varphi_k(x_i)f(x_i)\end{cases}$$

方程组可改写为

$$a_0(\varphi_k,\varphi_0)+a_1(\varphi_k,\varphi_1)+\cdots+a_n(\varphi_k,\varphi_n)=(\varphi_k,f)\quad k=0,1,\cdots,n$$

或矩阵形式

$$\begin{pmatrix}(\varphi_0,\varphi_0) & (\varphi_0,\varphi_1) & \cdots & (\varphi_0,\varphi_n)\\ (\varphi_1,\varphi_0) & (\varphi_1,\varphi_1) & \cdots & (\varphi_1,\varphi_n)\\ \vdots & \vdots & & \vdots\\ (\varphi_n,\varphi_0) & (\varphi_n,\varphi_1) & \cdots & (\varphi_n,\varphi_n)\end{pmatrix}\begin{pmatrix}a_0\\ a_1\\ \vdots\\ a_n\end{pmatrix}=\begin{pmatrix}(\varphi_0,f)\\ (\varphi_1,f)\\ \vdots\\ (\varphi_n,f)\end{pmatrix}$$

- 方程个数取决于基函数个数
- 系数矩阵对称
- 基函数线性无关时上述方程组有唯一系数矢量解（对应最小二乘解）

5.1.1 最小二乘原理

- 最小二乘的加权解法

$$\sum_{i=1}^{m}\omega_i\,\varepsilon_i^2=\sum_{i=1}^{m}\omega_i\,[\varphi(x_i)-f(x_i)]^2$$

- 权重 $\{\omega_\mathrm{i}\}$ 恒为正值
- 矩阵系数改写为

$$\begin{cases}(\varphi_k,\varphi_j)=\sum_{i=1}^{m}\omega_i\,\varphi_k(x_i)\varphi_j(x_i)\\ (\varphi_k,f)=\sum_{i=1}^{m}\omega_i\,\varphi_k(x_i)f(x_i)\end{cases}$$

5.1.2 直线拟合

- 已知数据点序列 $(x_i, y_i), i = 1,2,\cdots,m$ 大致为直线分布，利用最小二乘原理，构造直线y=a+bx，使残差平方和最小

$$min\sum_{i=1}^{m}[y_i-(a+bx_i)]^2$$

- 直线参数确定
  - 令 $\varphi_0 = 1,\quad \varphi_1 = x$
- 方程组变为

$$\begin{pmatrix} \sum_{i=1}^{m} 1 & \sum_{i=1}^{m} x_i \\ \sum_{i=1}^{m} x_i & \sum_{i=1}^{m} x_i^2 \end{pmatrix} \binom{a}{b} = \begin{pmatrix} \sum_{i=1}^{m} y_i \\ \sum_{i=1}^{m} x_i\, y_i \end{pmatrix}$$

5.1.3 超定方程组的最小二乘解

- 超定方程组内方程个数m多于未知量个数n，可按照最小二乘法求解

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m \end{cases}$$

$$\sum_{j=1}^{n} a_{ij}\, x_j = b_i, \quad i = 1,2,\cdots,m$$

矩阵形式为 $\mathrm{Ax} = \mathrm{B}$

定义残差及二次函数 $\delta_i = \sum_{j=1}^{n} a_{ij}\, x_j - b_i, \quad i = 1,2,\cdots,m$

$$J = \sum_{i=1}^{m} \delta_i^2 = \sum_{i=1}^{m} \left[\sum_{j=1}^{n} a_{ij}\, x_j - b_i \right]^2$$

5.1.3 超定方程组的最小二乘解

- 由极值条件

$$\frac{\partial J}{\partial x_k}=0 \quad k=0,1,\cdots,n$$

$$\sum_{j=1}^{n}\left(\sum_{i=1}^{m} a_{ij}\,a_{ik}\right)x_j=\sum_{i=1}^{\mathrm{m}} a_{ik}\,b_i$$

对应矩阵形式

$$A=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix},\quad x=\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix},\quad b=\begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$$

$$\delta=b-Ax$$

$$J=\delta^T\delta=(b-Ax)^T(b-Ax)=b^Tb-x^TA^Tb-b^TAx+x^TA^TAx$$

$$\frac{\partial J}{\partial x}=-2A^Tb+2A^TAx=0$$

$$A^TAx=A^Tb$$

5.1.4 可线化模型的最小二乘拟合

非线性过程

$$\frac{1}{y}=a+b\frac{1}{x}$$

变量代换

$$\hat{y}=\frac{1}{y},\hat{x}=\frac{1}{x}$$

可线化模型

$$\hat{y}=a+b\hat{x}$$

线性拟合

最小二乘
线性化模型

5.1.4 可线化模型的最小二乘拟合

- 给定观测数据

x	1.00	1.25	1.50	1.75	2.00
y	5.10	5.79	6.53	7.45	8.46

试用指数曲线 $y = ae^{bx}$ 拟合

变量代换 $\hat{y} = ln\, y = ln\, a + bx, \hat{a} = ln\, a \;\; \hat{y} = \hat{a} + bx$

x	1.00	1.25	1.50	1.75	2.00
$\hat{y}$	1.629	1.756	1.876	2.008	2.135

最小二乘

$$\begin{cases} 5\hat{a} + 7.5b = 9.404 \\ 7.5\hat{a} + 11.875b = 14.422 \end{cases}$$

$$\hat{a} = 1.122, b = 0.505, a = e^{\hat{a}} = 3.071$$

$$y = 3.071e^{0.505x}$$

5.1.5 多变量的数据拟合

- 矩阵形式下，多变量线性模型

$$y^* = a_0 + a_1x_1 + a_2x_2 + \cdots + a_nx_n$$

$$\begin{cases} y_1^* = a_0 + a_1x_{11} + a_2x_{21} + \cdots + a_nx_{n1} \\ y_2^* = a_0 + a_1x_{12} + a_2x_{22} + \cdots + a_nx_{n2} \\ \vdots \\ y_m^* = a_0 + a_1x_{1m} + a_2x_{2m} + \cdots + a_nx_{nm} \end{cases}$$

有如下回归方程组形式 $y = A\alpha$

其中

$$y^* = \begin{pmatrix} y_1^* \\ y_2^* \\ \vdots \\ y_m^* \end{pmatrix}, A = \begin{pmatrix} 1 & x_{11} & x_{21} & \cdots & x_{n1} \\ 1 & x_{12} & x_{22} & \cdots & x_{n2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{1m} & x_{2m} & \cdots & x_{nm} \end{pmatrix}, \alpha = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix}$$

对于m（m>n）次观测，可利用最小二乘原理确定其自变量系数

5.1.5 多变量的数据拟合

- 定义残差矢量及性能指标

$$\delta = (\delta_1, \delta_2, \cdots, \delta_m)^T$$

$$\delta = y - A\alpha$$

$$y = (y_1, y_2, \cdots, y_m)^T$$

$$J = \delta^T\delta$$

$$J = \delta^T\delta = (y - A\alpha)^T(y - A\alpha)$$
$$= y^Ty - \alpha^TA^Ty - y^TA\alpha + \alpha^TA^TA\alpha$$

构造正则方程组

$$A^TA\alpha = A^Ty$$

解得

$$\alpha = (A^TA)^{-1}A^Ty$$

5.1.6 多项式拟合

- 矩阵形式推导
  - 将m组数据代入n次多项式 $y = a_0 + a_1 x + \cdots + a_n x^n$
  - 得 $A\alpha = y$
  - 其中

$$A = \begin{pmatrix} 1 & x_1 & x_1^2 \cdots & x_1^n \\ 1 & x_2 & x_2^2 \cdots & x_2^n \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_m & x_m^2 \cdots & x_m^n \end{pmatrix}, \quad \alpha = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix}, \quad y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}$$

对应正则方程组

$$A^T A\alpha = A^T y$$

$$A^T A = \begin{pmatrix} m & \sum x_i & \sum x_i^2 & \cdots & \sum x_i^n \\ \sum x_i & \sum x_i^2 & \sum x_i^3 & \cdots & \sum x_i^{n+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \sum x_i^n & \sum x_i^{n+1} & \sum x_i^{n+2} & \cdots & \sum x_i^{2n} \end{pmatrix}, \quad A^T y = \begin{pmatrix} \sum y_i \\ \sum x_i y_i \\ \vdots \\ \sum x_i^n y_i \end{pmatrix}$$

其系数阵行列式非零，则有唯一解

THE
END