计 算 方 法

任课教师：李若泰

理学院

第二章 插值与逼近

- §2.2 *Lagrange* 插值多项式
- §2.3 *Newton* 插值多项式
- §2.6 正交多项式
- §2.7 最佳平方逼近
- §2.8 曲线拟合的最小二乘

§ 2.2 Lagrange 插值多项式

Lagrange 插值多项式形式：

$$\boldsymbol{L_n(x) = l_0(x)y_0 + l_1(x)y_1 + \cdots + l_n(x)y_n}$$

其中$l_k(x)$满足：$l_k(x_j) = \delta_k^j$，且$l_k(x) \in \mathrm{P}^{\mathrm{n}}$

插值基函数

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

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

§ 2.2 Lagrange 插值多项式

五. 截断误差:

定理3　插值多项式余项有如下表示形式

$$R_n(x) = f(x) - L_n(x)$$

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

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

Lagrange插值多项式的余项

§ 2.3 Newton插值多项式

令$\pi_k(x) = (x - x_0)(x - x_1)\cdots(x - x_{k-1})$

为牛顿插值基函数$k = 0,1,\ldots,N.$

$$I_N f(x) = \sum_{k=0}^{N} y_k l_k(x) = \sum_{k=0}^{N} a_k \pi_k(x)$$

其中$\pi_k(x) = (x - x_0)(x - x_1)\cdots(x - x_{k-1})$

$$= \pi_{k-1}(x) * (x - x_{k-1})$$

§ 2.3 Newton插值多项式(课本)

如何计算牛顿插值的系数a_k?

- 一. 利用插值条件，求解线性方程组
- 二. 利用差商表计算$Newton$插值

§ 2.3 Newton插值多项式

求$I_N f(x)=\sum_{k=0}^{N} a_k \pi_k(x)$系数

$$
\begin{aligned}
&a_0 + 0 + \dots + 0 = f(x_0)\\
&a_0 + a_1\pi_1(x_1) + 0 \dots + 0 = f(x_1)\\
&\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\\
&a_0 + a_1\pi_1(x_N) + \dots + a_N\pi_N(x_N) = f(x_N)
\end{aligned}
$$

解出$\mathrm{A}=\Pi^{-1}F$　　　　考虑$\pi_k(x_j)=0,\ j<k$

§ 2.3 Newton插值多项式(课本)

- **四. Newton 基本插值公式**

设插值多项式

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

满足插值条件 $N(x_i) = f_i$, $i = 0,1,\cdots,n$

则待定系数为 $a_0 = f_0$ $a_1 = f[x_0, x_1]$

$$a_2 = f[x_0, x_1, x_2] \quad a_n = f[x_0, x_1, \cdots, x_n]$$

牛顿公式

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

2.6 正交多项式

勒让德多项式函数

$$P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n,$$

递推关系式

$$P_0 = 1, \quad P_1 = x$$

$$(n+1)P_{n+1}(x) - (2n+1)xP_n(x) + nP_{n-1}(x) = 0, \quad n \geqslant 1$$

正交性

$$\int_{-1}^{1} P_m(x)P_n(x)dx = \begin{cases} 0, & m \neq n \\ \dfrac{2}{2n+1}, & m = n \end{cases}$$

切比雪夫多项式函数

$$T_n(x) = \cos\left(n \cos^{-1} x\right), \quad n = 0, 1, \cdots .$$

递推关系式

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), \qquad n \geqslant 1,$$
$$T_0(x) \equiv 1, \quad T_1(x) = x,$$

正交性

$$\int_{-1}^{1} T_k(x)T_j(x)(1 - x^2)^{-\frac{1}{2}} dx = \frac{c_k \pi}{2} \delta_{kj},$$ where $c_0 = 2$ and $c_k = 1$ for $k \geqslant 1$.

§ 2.7 最佳平方逼近

- **求f(x)的N次最佳平方逼近多项式p(x)：**

求$p(x) \in P^N, s.t.$

$$\int_{-1}^{1} \left(f(x) - p(x) \right)^2 \rho dx \quad \text{取到最小}$$

切比雪夫多项式，取$\rho(x) = \left(1 - x^2\right)^{-1/2}$

$$f(x) \approx p(x) = \sum_{k=0}^{N} a_k P_k(x)$$

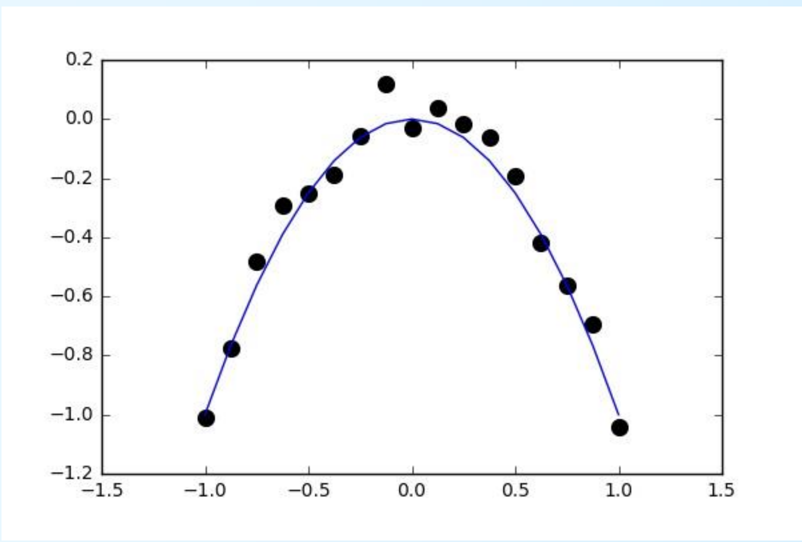
勒让德多项式，取$\rho(x) = 1$

$$\text{其中} a_k = \frac{(f(x), P_k(x))}{(P_k(x), P_k(x))} = \frac{2k+1}{2} \int_{-1}^{1} f(x) P_k(x) dx$$

§ 2.8 最小二乘法

$$(x_k, y_k)_{k=0}^{N} \Rightarrow f(x) \approx I_N f(x) = \sum_{k=0}^{N} a_k P_k(x)$$

$$\approx g(x) = \sum_{k=0}^{n} a_k P_k(x)$$



$$a_k = \frac{(I_N f(x), P_k(x))}{(P_k, P_k)}$$

$$= \frac{\sum_{j=0}^{N} f(x_j) P_k(x_j) \omega_j}{\sum_{j=0}^{N} P_k(x_j)^2 \omega_j}$$

$$k = 0, 1, \ldots, n$$

第三章 数值积分与数值微分

- §3.2 牛顿科特斯求积公式
- §3.4 高斯求积方法
- §3.5 数值微分

§ 3.1 引言

构造数值求积公式的基本方法

一般形式为：

$$\int_a^b f(x)dx \approx \sum_{i=0}^{n} A_i f(x_i)$$

$$\left.\begin{array}{l} x_i \to \text{求积节点} \\ A_i \to \text{求积系数} \end{array}\right\} \text{常数，与} f(x) \text{无关}$$

- **插值型求积分公式 (interpolation type quadrature)**

§ 3.1 引言

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

积分: $\int_{-1}^{1} f(x)dx \approx \int_{-1}^{1} I_N f(x)dx = \int_{-1}^{1} \sum_{k=0}^{n} l_k(x) f(x_k)dx$

$$= \sum_{k=0}^{n} f(x_k) \int_{-1}^{1} l_k(x)dx$$

其中: $\int_{-1}^{1} l_k(x)dx = \int_{-1}^{1} \prod_{\substack{i=0,\\ i\neq k}}^{N} \frac{x-x_i}{x_k-x_i}dx,\quad k=0,1,\ldots,N,$

§3.1 引言

三、求积公式的余项

带余项（截断误差）的形式:

$$\int_a^b f(x)dx = \sum_{i=0}^{n} A_i f(x_i) + R[f]$$

其截断误差为:

$$R[f] = \int_a^b R(x)dx = \int_a^b \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x)dx$$

其中，$\omega_{n+1}(x) = (x-x_0)(x-x_1)\ldots(x-x_n)$，$\xi \in (a,b)$。

§ 3.2 牛顿—柯特斯公式

常用的几个$Newton-Cotes$公式

1. $n=1$时， 梯形公式

$$\int_a^b f(x)dx \approx \frac{b-a}{2}[f(a)+f(b)]$$

2. $n=2$时， 辛普生($Simpson$)公式(抛物线公式)

$$\int_a^b f(x)dx \approx \frac{b-a}{6}[f(a)+4f(\frac{a+b}{2})+f(b)]$$

3. $n=4$时，五点柯特斯公式 **Boole's rule**

$$\int_a^b f(x)dx \approx \frac{b-a}{90}[7f(x_0)+32f(x_1)+12f(x_2)+32f(x_3)+7f(x_4)]$$

其中，$x_k=a+k\cdot\frac{b-a}{4}(k=0,1,2,3,4)$。

§ 3.2 复合牛顿—柯特斯公式

复合梯形公式为:

$$\int_a^b f(x)dx \approx T_n = \frac{h}{2}\left[f(a) + 2\sum_{k=1}^{n-1} f(x_k) + f(b) \right]$$

余项为:

$$\int_a^b f(x)dx - T_n = \frac{-(b-a)h^2}{12} f''(\xi), a < \xi < b$$

§ 3.2 复合牛顿—柯特斯公式

基本公式：$\int_a^b f(x)dx \approx \frac{b-a}{6}[f(a)+4f(\frac{a+b}{2})+f(b)]$

复化Simpson公式为:

$$\int_a^b f(x)dx \approx S_n$$

$$= \frac{h}{6}\left[f(a) + 2\sum_{k=1}^{n-1} f(x_{k-1}) + 4\sum_{k=1}^{n} f(x_{k-\frac{1}{2}}) + f(b) \right]$$

其中$h = \frac{b-a}{n}$, $x_k = a + kh$, $x_{k-\frac{1}{2}} = a + (k - \frac{1}{2})h$

余项为：

$$\int_a^b f(x)dx - S_n = \frac{-(b-a)h^4}{2880} f^{(4)}(\xi),\ a < \xi < b$$

§ 3.4 高斯求积方法

定理：点$x_0,\ldots,x_N$满足$L_{N+1}(x)=0$，$f(x)\in P^{2N+1}$

$$\int_{-1}^{1} f(x)dx=\sum_{k=0}^{N} f(x_k)\omega_k，\text{其中}\omega_k=\int_{-1}^{1} l_k(x)dx$$

x_k称为高斯勒让德点，ω_k称为高斯勒让德权重系数

$$\omega_k=\int_{-1}^{1} l_k(x)dx=\int_{-1}^{1}\prod_{\substack{i=0\\ i\neq k}}^{N}\frac{(x-x_i)}{(x_k-x_i)}dx=\frac{1}{(L_{N+1}'(x_k))^2}\frac{2}{1-x_k^{\ 2}}$$

注：对$f^{2N+1}(x)\in P^{2N+1}$精确成立

§ 3.4 高斯求积方法

x_k为切比雪夫-高斯-罗巴托点，

ω_k为切比雪夫-高斯-罗巴托权重系数

则$\int_{-1}^{1} g(x)\frac{1}{\sqrt{1-x^2}}dx \approx \sum_{k=0}^{N} g(x_k)\omega_k$，注：对$g^{2N-1}(x)\in P^{2N-1}$精确成立

$$\int_{-1}^{1} f(x)dx = \int_{-1}^{1} f(x)\sqrt{1-x^2}\,\frac{1}{\sqrt{1-x^2}}dx \approx \sum_{k=0}^{N} f(x_k)\sqrt{1-x_k^{\ 2}}\,\omega_k$$

Chebyshev-Gauss:

$$x_j = \cos\frac{(2j+1)\pi}{2N+2}, \quad \omega_j = \frac{\pi}{N+1}, \quad 0 \leqslant j \leqslant N.$$

3、勒让德插值多项式

- **掌握了高斯积分，如何解决正交多项式插值问题？计算插值系数a_k？**

对于勒让德-高斯-罗巴托点有： 注：对$f^{2N-1}(x)\in P^{2N-1}$精确成立

$$I_N f(x)=\sum_{k=0}^{N} y_k l_k(x)=\sum_{k=0}^{N} a_k P_k(x)$$

$$\text{其中} a_k=\frac{2k+1}{2}\sum_{j=0}^{N} f(x_j)P_k(x_j)\omega_j \qquad (k=0,\ldots,N-1)$$

$$a_N=\frac{N}{2}\sum_{j=0}^{N} f(x_j)P_N(x_j)\omega_j$$

对于勒让德-高斯点时， a_k=？

3、切比雪夫插值多项式

$$I_N f(x) = \sum_{k=0}^{N} a_k T_k(x),$$

$$a_k = \frac{1}{\left(T_k(x), T_k(x)\right)_\omega} \sum_{j=0}^{N} f(x_j) T_k(x_j) \omega_j,$$

$$k = 0, 1, \ldots N-1.$$

注：对$f^{2N-1}(x) \in P^{2N-1}$精确成立

$$a_N = \frac{1}{\pi} \sum_{j=0}^{N} f(x_j) T_N(x_j) \omega_j$$

一阶两点微商公式

$$\begin{cases} u'(x_0) \approx \dfrac{u(x_1)-u(x_0)}{h} \\ u'(x_1) \approx \dfrac{u(x_1)-u(x_0)}{h} \end{cases}$$

一阶三点微商公式

$$\begin{cases} u'(x_0) \approx \dfrac{1}{2h}[-3u(x_0)+4u(x_1)-u(x_2)] \\ u'(x_1) \approx \dfrac{u(x_2)-u(x_0)}{2h} \\ u'(x_2) \approx \dfrac{1}{2h}[u(x_0)-4u(x_1)+3u(x_2)] \end{cases}$$

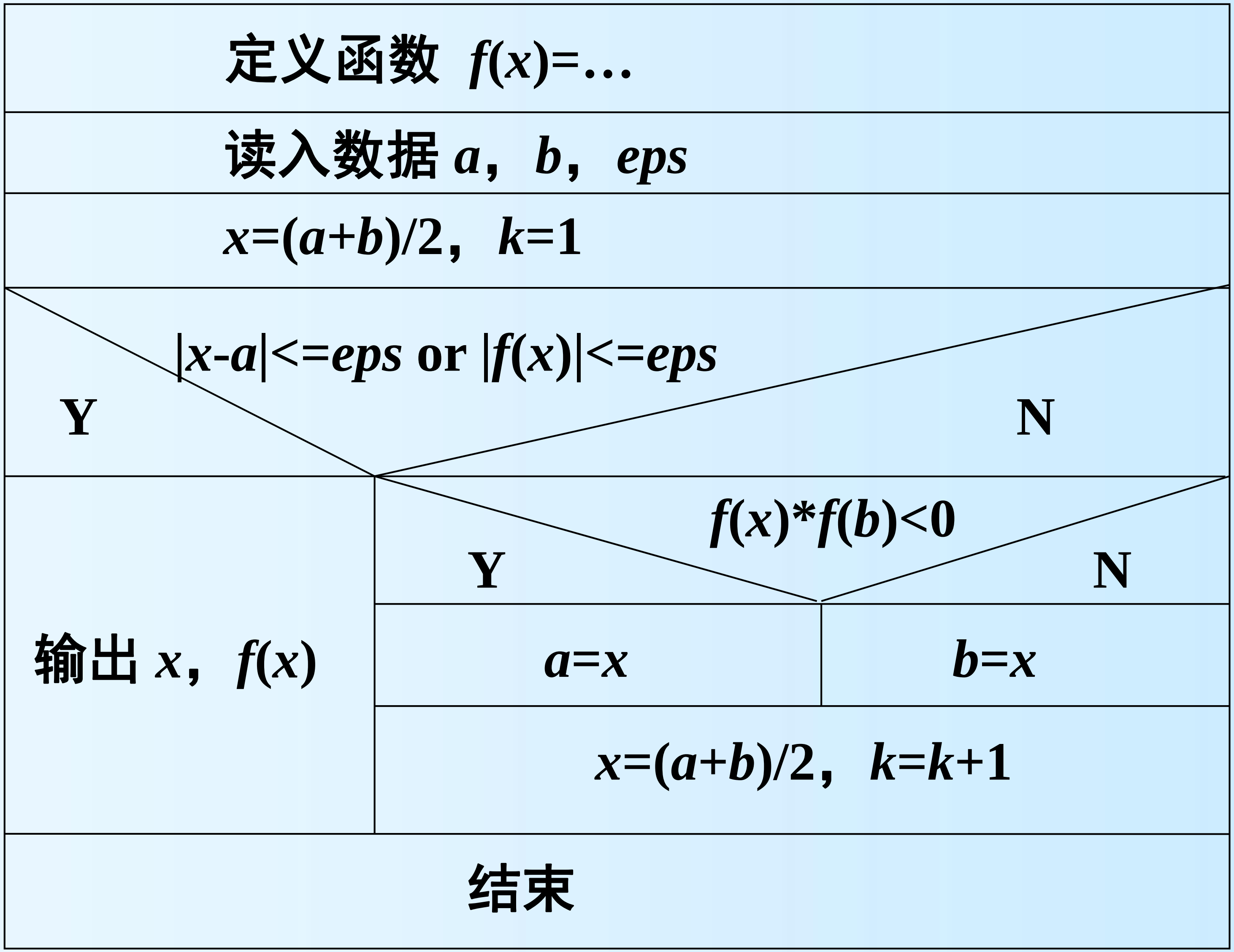
二阶三点微商公式

$$u''(x_1) \approx \frac{u(x_0)-2u(x_1)+u(x_2)}{h^2}$$

第四章 非线性方程的数值解法

- §4.2 二分法
- §4.3 迭代法
- §4.4 牛顿法
- §4.5 正割法

■ 二分法的N - S图 或 二分法的实现算法



§ 4.3 迭代法

一. 迭代格式的构造

改写方程： $f(x)=0 \Leftrightarrow x=g(x)$ 且 g 连续。

建立迭代格式：$x_{n+1}=g(x_n)$，得到序列 $\{x_n\}$

考虑方程$x=g(x), g(x)\in C[a,b]$, 若

(1)当$x\in[a,b]$时, $g(x)\in[a,b]$;

(2)$\exists 0\le L<1$使得$|g'(x)|\le L<1$对$\forall x\in[a,b]$成立。

则任取$x_0\in[a,b]$, 由$x_{k+1}=g(x_k)$得到的序列$\{x_k\}_{k=0}^{\infty}$收敛

§ 4.4 牛顿法

□ **2. 方法**

——Taylor展开线性化（重要思想）

设x_k是$f(x)=0$的一个近似根，将$f(x)$在x_k处做一阶$Taylor$(泰勒)展开：

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

$$f(x)\approx f(x_k)+f'(x_k)(x-x_k)$$

$f(x)=0$近似于$f(x_k)+f'(x_k)(x-x_k)=0$

设$f'(x)\neq 0$，解出x记为x_{k+1}，则

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

五. 简化牛顿法

牛顿法需计算$f'(x_k)$，若用一个给定常数值c代$f'(x)$，

则 $x_{k+1}=x_k-f(x_k)/c$

要使该式收敛，则

$$g(x) = x-f(x)/c$$

$$g'(x) = 1-f'(x)/c$$

$$|g'(x)| = |1-f'(x)/c| < 1$$

即$0<f'(x)/c<2$

取c与$f'(x)$同号，且$f'(x)/c<2$即可，这是切线方程的固定斜率。

§ 4.4 牛顿—雷扶生法

将牛顿迭代法与下山法结合起来使用，即在下山法保证函数值下降的前提下，用牛顿迭代法加快收敛速度。把这一算法称为牛顿下山法。即

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

其中λ（$0 < \lambda < 1$）为下山因子

满足 $\left|f(x_{k+1})\right| < \left|f(x_k)\right|$

§ 4.5 正割法（弦截法）

▪ 二. 正割法（弦截法）基本思想

为避免计算函数的导数 $f'(x_k)$，使用差商

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

代替牛顿公式中的导数 $f'(x_k)$，即得到迭代公式

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

$$(k=1,2,\cdots)$$

称为正割法迭代公式，相应的迭代法称为正割法。

§ 迭代法的收敛阶

- **二. 常见迭代过程的收敛阶**
  - □ 一般迭代法：$p=1$，线性收敛
  - □ 牛顿法：$p=2$，二阶收敛
  - □ 正割法：$p=1.618$，超线性收敛

$e_k = x - x_k$ 当 $k \to \infty$ 时成立

$$\frac{e_{k+1}}{{e_k}^p} \to C \quad（C \neq 0，常数）$$

则称迭代过程是 p 阶收敛的。

第五章 线性方程组的数值解法

§ 5.2 高斯消去法

- 三. 求解线性方程组(1)的顺序Gauss消去法

记 $\mathrm{A}^{(1)}=\mathrm{A}, \mathrm{b}^{(1)}=\mathrm{b}, a_{ij}^{(1)}=a_{ij}, b_i^{(1)}=b_i$

则,线性方程组(1)的增广矩阵为

$$(\mathrm{A}^{(1)}, \mathrm{b}^{(1)})=\begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2n}^{(1)} & b_2^{(1)} \\ a_{31}^{(1)} & a_{32}^{(1)} & a_{33}^{(1)} & \cdots & a_{3n}^{(1)} & b_3^{(1)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{n1}^{(1)} & a_{n2}^{(1)} & a_{n3}^{(1)} & \cdots & a_{nn}^{(1)} & b_n^{(1)} \end{pmatrix}$$

§ 5.2 高斯消去法

- 如此继续消元下去，第n-1步结束后得到矩阵：

$$(\mathrm{A}^{(n)},\mathrm{b}^{(n)})=\begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \dots & a_{2n}^{(2)} & b_2^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \dots & a_{3n}^{(3)} & b_3^{(3)} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & a_{nn}^{(n)} & b_n^{(n)} \end{pmatrix}$$

这就完成了消元过程。

§ 5.2 高斯消去法

- 对应的方程组变成：

$$\begin{cases} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \ldots + a_{1n}^{(1)}x_n = b_1^{(1)} \\ \qquad\qquad a_{22}^{(2)}x_2 + \ldots + a_{2n}^{(2)}x_n = b_2^{(2)} \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \qquad\qquad\qquad\qquad a_{nn}^{(n)}x_n = b_n^{(n)} \end{cases} \qquad \text{（式2）}$$

对此方程组进行回代，就可求出方程组的解。

$$\begin{cases} x_n = \dfrac{b_n^{(n)}}{a_{nn}^{(n)}} \\ x_i = \dfrac{\left(b_i^{(i)} - \sum\limits_{j=i+1}^{n} a_{ij}^{(i)}x_j\right)}{a_{ii}^{(i)}}, i = n-1, n-2, \cdots, 1 \end{cases} \qquad \text{（式3）}$$

元素$a_{kk}^{(k)}$称为约化的主元素。

§ 5.4 矩阵的三角分解

若$a_{11}^{(1)} \neq 0$，令$m_{i1} = a_{i1}^{(1)} \div a_{11}^{(1)}, i = 2,3,\cdots,n$, 记

$$L_1 = \begin{pmatrix} 1 & & & & \\ -m_{21} & 1 & & & \\ -m_{31} & & 1 & & \\ \vdots & & & \ddots & \\ -m_{n1} & & & & 1 \end{pmatrix}$$

初等下三角阵

则有 $$A^{(2)} = L_1 A^{(1)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} \end{pmatrix}$$

§ 5.4 矩阵的三角分解

若$a_{22}^{(2)} \neq 0$，令$m_{i2} = a_{i2}^{(2)} \div a_{22}^{(2)}, i = 3,4,\cdots,n$,记

$$L_2 = \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & -m_{32} & 1 & & \\ & \vdots & & \ddots & \\ & -m_{n2} & & & 1 \end{pmatrix}$$

则有 $$A^{(3)} = L_2 A^{(2)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3n}^{(3)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & a_{n3}^{(3)} & \cdots & a_{nn}^{(3)} \end{pmatrix}$$

§5.4 矩阵的三角分解

如此进行下去，第$n-1$步得到：

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

其中 $$L_{n-1} = \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & -m_{n,n-1} & 1 \end{pmatrix}$$

§ 5.4 矩阵的三角分解

也就是:

$$A^{(n)}=L_{n-1}A^{(n-1)}=L_{n-1}L_{n-2}A^{(n-2)}=\ldots=L_{n-1}L_{n-2}\ldots L_2L_1A^{(1)}$$

其中:

$$\mathrm{L}_k=\begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & -m_{k+1k} & 1 & & \\ & & \vdots & & \ddots & \\ & & -m_{nk} & & & 1 \end{pmatrix} \begin{array}{l} \\ \\ \\ \leftarrow 第k行 \\ \\ ,k=1,2,\cdots,n-1 \end{array}$$

所以有: $A=A^{(1)}=L_1^{-1}L_2^{-1}\ldots L_{n-1}^{-1}A^{(n)}=LU$

其中: $L=L_1^{-1}L_2^{-1}\ldots L_{n-1}^{-1}$, $U=A^{(n)}$

§ 5.4 矩阵的三角分解

而且有

$$
\mathbf{L}_k^{-1}=\begin{pmatrix}
1 & & & & & \\
 & \ddots & & & & \\
 & & 1 & & & \\
 & & m_{k+1k} & 1 & & \\
 & & \vdots & & \ddots & \\
 & & m_{nk} & & & 1
\end{pmatrix},\quad
\mathbf{L}=\begin{pmatrix}
1 & & & & \\
m_{21} & 1 & & & \\
m_{31} & m_{32} & 1 & & \\
\vdots & \vdots & \vdots & \ddots & \\
m_{n1} & m_{n2} & m_{n3} & \cdots & 1
\end{pmatrix}
$$

*L*称为单位下三角矩阵;

*U*是上三角矩阵.

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

式 $A=LU$称为矩阵A的三角分解。

§ 5.4 矩阵的三角分解

下面介绍矩阵三角分解的*Doolittle*分解方法，设

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ a_{31} & a_{32} & & a_{3n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} =$$

$$\begin{pmatrix} 1 & & & & \\ m_{21} & 1 & & & \\ m_{31} & m_{32} & 1 & & \\ \vdots & \vdots & \vdots & \ddots & \\ m_{n1} & m_{n2} & m_{n3} & \cdots & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ & u_{22} & u_{23} & \cdots & u_{2n} \\ & & u_{33} & \cdots & u_{3n} \\ & & & \ddots & \vdots \\ & & & & u_{nn} \end{pmatrix}$$

则得： $u_{1j} = a_{1j} \quad j = 1,2,\cdots n$

K=1

$m_{i1} = a_{i1} \div u_{11} \quad i = 2,3,\cdots,n$

§ 5.4 矩阵的三角分解

$$\begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1k-1} & u_{1k} & \cdots & u_{1n} \\ m_{21} & u_{22} & \cdots & u_{2k-1} & u_{2k} & \cdots & u_{2n} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ m_{k-11} & m_{k-12} & \cdots & u_{k-1k-1} & u_{k-1k} & \cdots & u_{k-1n} \\ m_{k1} & m_{k2} & \cdots & m_{kk-1} & u_{kk} & \cdots & u_{kn} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ m_{n1} & m_{n2} & \cdots & m_{nk-1} & m_{nk} & \cdots & u_{nn} \end{pmatrix}$$

$$\begin{cases} u_{1j} = a_{1j} \qquad j = 1,2,\cdots,n \\ m_{i1} = a_{i1} \div u_{11} \qquad i = 2,3,\cdots,n \\ \text{对} k = 2,3,\cdots,n, \text{计算} \\ u_{ij} = a_{ij} - \sum_{k=1}^{j-1} m_{ik} u_{kj} \quad j = i, i+1, \cdots, n \\ m_{ij} = \dfrac{1}{u_{jj}} (a_{ij} - \sum_{k=1}^{j-1} m_{ik} u_{kj}) \quad , i = j+1, \cdots, n \end{cases}$$

§ 5.4 矩阵的三角分解

由

$$\begin{pmatrix} 1 & & & \\ m_{21} & 1 & & \\ \vdots & \vdots & \ddots & \\ m_{n1} & m_{n2} & \cdots & 1 \end{pmatrix}\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}, \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & & & u_{nn} \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

可得

$$y_1 = b_1 \qquad y_i = b_i - \sum_{j=1}^{i-1} m_{ij} y_j \quad ,i = 2,3,\cdots,n$$

$$x_n = y_n \div u_{nn} \qquad x_i = \frac{1}{u_{ii}}(y_i - \sum_{j=i+1}^{n} u_{ij} x_j), i = n-1,\cdots,2,1$$

这就是求解方程组Ax=b的Doolittle三角分解方法。

§ 5.6 向量和矩阵的范数

向量范数例：

$$\|x\|_2 = \sqrt{x_1^{\ 2} + \cdots + x_n^{\ 2}} = \left(\sum_{i=1}^{n} x_i^{\ 2}\right)^{\frac{1}{2}}$$

$$\|x\|_\infty = \mathbf{max}\left\{ |x_1|, \cdots, |x_n| \right\} = \mathbf{max}_{1\le i\le n}\left\{ |x_i| \right\}$$

$$\|x\|_1 = |x_1| + \cdots + |x_n| = \sum_{i=1}^{n} |x_i|$$

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^{p} \right)^{1/p}$$

矩阵范数例

与前述三种向量范数相容的三种矩阵范数：

$\|A\|_2 = \max\limits_{\|x\|_2=1} \|Ax\|_2 = \sqrt{\lambda_1}$，其中$\lambda_1$是$A^T A$的最大特征值。又称为谱范数。设$A=(a_{ij})$为$n$阶方阵。

$$\|A\|_1 = \max_{\|x\|_1=1} \|Ax\|_1 = \max_{1\le j\le n} \sum_{i=1}^{n} |a_{ij}|,$$ 为矩阵的列向量的1－范数的最大值称为矩阵的列范数。

$$\|A\|_\infty = \max_{\|x\|_\infty=1} \|Ax\|_\infty = \max_{1\le i\le n} \sum_{j=1}^{n} |a_{ij}|,$$ 为矩阵的行向量的1－范数的最大值称为矩阵的行范数。

如果将矩阵范数看作R^{n^2}空间上的向量范数，则由向量范数的等价性可得矩阵范数的等价性。

§5.7 解线性方程组的迭代法

于是有迭代公式

$$\begin{cases} x_1^{(k+1)} = \dfrac{1}{a_{11}}[b_1 - 0x_1^{(k)} - a_{12}x_2^{(k)} - \cdots\cdots - a_{1n}x_n^{(k)}] \\ x_2^{(k+1)} = \dfrac{1}{a_{22}}[b_2 - a_{21}x_1^{(k)} - 0x_2^{(k)} - \cdots\cdots - a_{2n}x_n^{(k)}] \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ x_n^{(k+1)} = \dfrac{1}{a_{nn}}[b_n - a_{n1}x_1^{(k)} - a_{n2}x_2^{(k)} - \cdots\cdots - 0x_n^{(k)}] \end{cases} \quad \text{式4}$$

$$(k = 0,1,2,\cdots)$$

————————*Jacobi*迭代法

§ 5.7 解线性方程组的迭代法

■ Jacobi迭代法

方程组 $Ax=b$, 令 $A=D-L-U$

$$Dx=(L+U)x+b$$

$$x = D^{-1}(L+U)x + D^{-1}b$$

令 $B = D^{-1}(L+U)$， $g = D^{-1}b$

Jacobi迭代为 $x^{k+1} = Bx^k + g$

§ 5.7 解线性方程组的迭代法

□3. 迭代法2

迭代方法

$$\begin{cases} x_1^{(k+1)} = \dfrac{1}{a_{11}}\left[b_1 - 0x_1^{(k)} - a_{12}x_2^{(k)} - \cdots - a_{1n}x_n^{(k)}\right] \\ x_2^{(k+1)} = \dfrac{1}{a_{22}}\left[b_2 - a_{21}x_1^{(k+1)} - 0x_2^{(k)} - \cdots - a_{2n}x_n^{(k)}\right] \\ \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \\ x_n^{(k+1)} = \dfrac{1}{a_{nn}}\left[b_n - a_{n1}x_1^{(k+1)} - a_{n2}x_2^{(k+1)} - \cdots - 0x_n^{(k)}\right] \end{cases} \qquad \text{式7}$$

称为与*Jacobi*迭代法(式4)对应的*Seidel*方法。

§ 5.7 解线性方程组的迭代法

- **三.塞德尔(*Seidel*)迭代法**

方程组 Ax=b, 令 A=D-L-U

$$(D-L-U)x=b,$$

$$(D-L)x=Ux+b,$$

$$x=(D-L)^{-1}Ux+(D-L)^{-1}b,$$

令$B=(D-L)^{-1}U, g=(D-L)^{-1}b,$

高斯-赛德尔迭代法：$x^{(k+1)}=Bx^{(k)}+g$

§ 5.7 解线性方程组的迭代法

■ 四. 逐次超松弛(*SOR*)迭代法

□ 1. 迭代法

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

$$(i = 1, 2, \cdots, n)$$

(式8)

ω—松弛因子，

$\omega = 1$ 即*Seidel*方法 (式7)

$0 < \omega < 1$为低松弛方法

$1 < \omega < 2$为超松弛方法

(式8)一种加权平均。

§ 5.7 解线性方程组的迭代法

■ 四. 逐次超松弛(*SOR*)迭代法

$$(D-L-U)x=b，\quad Dx=(b+Lx+Ux),$$

$$x=D^{-1}(b+Lx+Ux),\quad \omega x=\omega D^{-1}(b+Lx+Ux),$$

$$x=(1-\omega)x+\omega D^{-1}(b+Lx+Ux),$$

$$Dx=D(1-\omega)x+\omega Lx+\omega Ux+\omega b,$$

$$(D-\omega L)x=\big((1-\omega)D+\omega U\big)x+\omega b,$$

$$x=(D-\omega L)^{-1}\big((1-\omega)D+\omega U\big)x+(D-\omega L)^{-1}\omega b,$$

第六章 常微分方程初值问题的数值方法

- **§6.2 欧拉方法**
- **§6.3 龙格—库塔方法**
- **§6.4 阿达姆斯方法**

§ 6.2 欧拉方法

➢ **欧拉公式：** /* Euler's Method */ explicit Euler

向前差商近似导数 → $y'(x_0) \approx \dfrac{y(x_1) - y(x_0)}{h}$

$$y(x_1) \approx y(x_0) + hy'(x_0) = y_0 + h\, f(x_0, y_0) \xlongequal{\text{记为}} y_1$$

$$\boxed{y_{i+1} = y_i + h\, f(x_i, y_i) \quad (i = 0, \dots, n-1)}$$

➢ **隐式欧拉法** /* implicit Euler method */

向后差商近似导数 → $y'(x_1) \approx \dfrac{y(x_1) - y(x_0)}{h}$

$$\longrightarrow\ y(x_1) \approx y_0 + h\, f(x_1, y(x_1))$$

$$\boxed{y_{i+1} = y_i + h\, f(x_{i+1}, y_{i+1}) \quad (i = 0, \dots, n-1)}$$

§ 6.2 欧拉方法

改进欧拉法 /* modified Euler's method */

Step 1: 先用**显式**欧拉公式作**预测**，算出 $\bar{y}_{i+1} = y_i + h\, f(x_i, y_i)$

Step 2: 再将 $\bar{y}_{i+1}$ 代入**隐式**梯形公式的右边作**校正**，得到

$$y_{i+1} = y_i + \frac{h}{2}[\, f(x_i, y_i) + f(x_{i+1}, \bar{y}_{i+1})]$$

$$y_{i+1} = y_i + \frac{h}{2}\left[f(x_i, y_i) + f\big(x_{i+1}, y_i + h\, f(x_i, y_i)\big)\right] \quad (i = 0, \ldots, n-1)$$

§ 6.3 龙格 —库塔方法

2阶龙格-库塔法 (改进的Euler方法)

考察改进的欧拉法，

$$\begin{cases} \bar{y}_k = y_{k-1} + hf(x_{k-1}, y_{k-1}) \\ y_k = y_{k-1} + \dfrac{h}{2}[f(x_{k-1}, y_{k-1}) + f(x_k, \bar{y}_k)] \end{cases}$$

可将其改写为：

$$\begin{cases} y_{i+1} = y_i + h\left[\dfrac{1}{2}K_1 + \dfrac{1}{2}K_2\right] \\ K_1 = f(x_i, y_i) \\ K_2 = f(x_i + h, y_i + hK_1) \\ y_0 = y(x_0) \end{cases}$$

§6.3 龙格—库塔方法

三阶Runge-Kutta方法

$$
\begin{cases}
y_{i+1} = y_i + \dfrac{h}{6}(K_1 + 4K_2 + K_3) \\
K_1 = f(x_i, y_i) \\
K_2 = f(x_i + \dfrac{h}{2}, y_i + \dfrac{h}{2}K_1) \\
K_3 = f(x_i + h, y_i + h(2K_2 - K_1)) \\
y_0 = y(x_0)
\end{cases}
$$

§6.3 龙格—库塔方法

四阶(经典)*Runge-Kutta*方法

$$
\begin{cases}
y_{i+1} = y_i + \dfrac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4) \\
K_1 = f(x_i, y_i) \\
K_2 = f(x_i + \dfrac{h}{2}, y_i + \dfrac{h}{2}K_1) \\
K_3 = f(x_i + \dfrac{h}{2}, y_i + \dfrac{h}{2}K_2) \\
K_4 = f(x_i + h, y_i + hK_3) \\
y_0 = y(x_0)
\end{cases}
$$

§ 6.4 阿达姆斯方法

4阶亚当姆斯显式公式

$$y_{i+1} = y_i + \frac{h}{24}(55 f_i - 59 f_{i-1} + 37 f_{i-2} - 9 f_{i-3})$$

4阶亚当姆斯隐式公式

$$y_{i+1} = y_i + \frac{h}{24}(9 f_{i+1} + 19 f_i - 5 f_{i-1} + f_{i-2})$$

较同阶显式稳定