

关于第一次实验.

后验误差估计 $f(x) - L_n(x) \approx \frac{x-x_0}{x_0-x_{n+1}} (L_n(x) - L_n(x_0))$

\downarrow \downarrow

用 x_0, x_1, \dots, x_n 用 x_1, x_2, \dots, x_{n+1}

我写作业的时候没想清楚,有个同学其实写错了但我忘写批注了.

· 计算器求逆

<https://jingyan.baidu.com/article/380abd0aa0a5c95d90192cba.html>

若型号为 fx-991ES PLUS,

选择矩阵
形状

① mode \rightarrow 6: MATRIX \rightarrow 1: MatA \rightarrow 1: 3x3 \rightarrow 输入矩阵各个元素的值, 输入结束后按 AC 保存
(其他也可, 只是选择存放矩阵的变量而已)

\rightarrow SHIFT + 4 \rightarrow 3: MatA \rightarrow $x^{-1} + =$ (△标记的为计算器上按钮)
(4上面有MATRIX标志)

计算器其他使用技巧:

<https://zhuanlan.zhihu.com/p/78367952>

2015 年春季学期计算方法 (B) 复习提纲精简版

第 0 章 绪论

0.1 数值计算方法与算法

0.2 误差与有效数字

定义 如果精确值 x^* 与近似值 x 的误差的绝对值不超过某正数 ε , 即

$$|e| = |x^* - x| \leq \varepsilon \quad \text{称 } \varepsilon \text{ 为绝对误差限或误差限。}$$

例: 若 $x^* = 0.0123456$, 则它的误差限是: $|e| = |x^* - x| < 10^{-8} \cdot 5 = \frac{1}{2} 10^{-7}$

产生误差的因素很多, 产生误差的原因主要有

原始误差 截断误差 舍入误差

定义 当 x 的误差限为某一位的半个单位, 则这一位到第一个非零位的位数称为 x 的有效位数。

0.3 约束误差

0.4 范数

0.4.1 向量范数

对任一向量 $X \in R^n$, 按照一个规则确定一个实数与它对应, 记该实数为 $\|X\|$, 若 $\|X\|$ 满足下面三个性质:

(i) $\forall X \in R^n$, 有 $\|X\| \geq 0$, 当且仅当 $X = 0$ 时, $\|X\| = 0$ (非负性)

[编者注: 考试常有类似的判断题, 判断是否可以构成范数, 在判断的时候非负性尤其要注意,

当且仅当 $X = 0$ 时, $\|X\| = 0$, 才可以构成范数]

(ii) $\forall X \in R^n, \alpha \in R$, 有 $\|\alpha X\| = |\alpha| \|X\|$ (齐次性)

(iii) $\forall X, Y \in R^n$, 有 $\|X + Y\| \leq \|X\| + \|Y\|$ (三角不等式)

那么称该实数 $\|X\|$ 为向量 X 的范数。

向量范数 向量 $X = (x_1, x_2, \dots, x_n)^T$ 的 L_p 范数定义为

$$\|X\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{1/p}, \quad 1 \leq p \leq +\infty$$

其中, 经常使用三种向量范数是 $p = 1, 2, \infty$

$$\|X\|_1 = |x_1| + |x_2| + \dots + |x_n| = \sum_{i=1}^n |x_i|$$

$$\|X\|_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2} = \sqrt{\sum_{i=1}^n x_i^2} \quad \text{或写成} \quad \|X\|_2 = \sqrt{(X, X)}$$

$$\|X\|_\infty = \max\{|x_1|, |x_2|, \cdots, |x_n|\} = \max_{1 \leq i \leq n}\{|x_i|\}$$

0.4.2 矩阵范数

矩阵范数

矩阵范数性质

- (1) $\|A\| > 0$, 当且仅当 $A = 0$ 时, $\|A\| = 0$ (非负性)
- (2) $A \in R^{n \times n}, \forall \lambda \in R \quad \|\lambda A\| = |\lambda| \|A\|$ (齐次性)
- (3) 对于任意两个阶数相同的矩阵 A, B , 有 $\|A + B\| \leq \|A\| + \|B\|$ (三角不等式)
- (4) A, B 为同阶矩阵 $\|AB\| \leq \|A\| \|B\|$
- (5) A 为 n 阶阵, 对 $\forall x \in R^n$, 恒有 $\|AX\| \leq \|A\| \cdot \|X\|$ (相容性)

常用矩阵范数

对应于向量的三种范数, 相对应的三种矩阵范数形式为:

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}| = \max_{1 \leq j \leq n} (|a_{1,j}| + |a_{2,j}| + \cdots + |a_{n,j}|) \quad \text{[最大的行和]}$$

$$\|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}| = \max_{1 \leq i \leq n} (|a_{i,1}| + |a_{i,2}| + \cdots + |a_{i,n}|) \quad \text{[最大的列和]}$$

$$\|A\|_2 = \sqrt{\rho(A^T A)} = \sqrt{\lambda_1} \quad \text{[} A^T A \text{ 的绝对值最大的特征值的平方根]}$$

符号 $\rho(A^T A)$ 称为 $A^T A$ 的谱半径, $A^T A$ 的特征值 $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$ 。

还要介绍一种向量范数, 称为 Euclid 或 Schur 范数, 用 $\|A\|_E$ 表示

$$\text{定义为} \quad \|A\|_E = \sqrt{\sum_{i=1}^n \sum_{j=1}^n |a_{ij}|^2}$$

例: $A = \begin{pmatrix} -1 & 2 \\ 3 & 7 \end{pmatrix}$, 分别求 $\|A\|_1$, $\|A\|_\infty$, $\|A\|_2$, $\|A\|_E$

解: $\|A\|_1 = \max\{|-1|+3, 2+7\} = 9$ $\|A\|_\infty = \max\{|-1|+2, 3+7\} = 10$

$$A^T A = \begin{pmatrix} -1 & 3 \\ 2 & 7 \end{pmatrix} \begin{pmatrix} -1 & 2 \\ 3 & 7 \end{pmatrix} = \begin{pmatrix} 10 & 19 \\ 19 & 53 \end{pmatrix}$$

$A^T A$ 的特征值 $\lambda_1 = 60.19, \lambda_2 = 2.81$

$$\|A\|_2 = \sqrt{60.19} = 7.75822 \quad \|A\|_F = \sqrt{63}$$

第 1 章 插 值

1.1 插值

1.2 多项式插值的拉格朗日 (Lagrange) 型式

1.2.1 线性插值

$$L_1(x) = \frac{x-x_1}{x_0-x_1} f(x_0) + \frac{x-x_0}{x_1-x_0} f(x_1)$$

记 $l_0(x) = \frac{x-x_1}{x_0-x_1}, \quad l_1(x) = \frac{x-x_0}{x_1-x_0}$ 称 $l_0(x), l_1(x)$ 为插值基函数,

1.2.2 二次插值

给定三个插值点 $(x_i, f(x_i)), i=0,1,2$; 其中 x_i 互不相等, 构造函数 $f(x)$ 的二次插值多项式?

$$L_2(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} f(x_0) + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} f(x_1) + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} f(x_2)$$

$$= \sum_{i=0}^2 l_i(x) f(x_i)$$

1.2.3 n 次拉格朗日插值多项式

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

$$L_n(x) = \sum_{i=0}^n l_i(x) f(x_i)$$

n 次插值多项式的误差

定理 2 设 $L_n(x)$ 是 $[a, b]$ 上过 $(x_i, f(x_i)), i=0,1,\cdots,n$ 的 n 次插值多项式, $x_i \in [a, b], x_i$ 互

不相同, 当 $f \in C^{n+1}[a, b]$ 时, 则插值多项式的误差:

$$R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} (x-x_0)(x-x_1) \cdots (x-x_n) \quad \text{其中 } \xi \in [a, b]$$

证明: 可设 $R_n(x) = k(x)(x-x_0)(x-x_1) \cdots (x-x_n)$ 引入变量为 t 的函数 $\varphi(t)$

$$\varphi(t) = f(t) - L_n(t) - k(x)(t-x_0)(t-x_1)\cdots(t-x_n)$$

$\varphi(t)$ 至少有 $n+2$ 个零点, 由于 $f \in C^{n+1}[a, b]$, 由 Rolle 定理, 得到 $k(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}$

$$R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-x_0)(x-x_1)\cdots(x-x_n) \quad \xi \in [a, b] \quad *$$

$R_n(x)$ 是插值多项式 $L_n(x)$ 的截断误差, 也称插值余项。

[编者注: 写差值误差的时候通常根据定理 2 来写, 需要注意:

若给定 $x_0, f(x_0), f'(x_0); x_1, f(x_1)$, 则插值余项 $R_2(x) = \frac{f^{(3)}(\xi)}{3!}(x-x_0)^2(x-x_1)$

若给定 $x_0, f(x_0), f'(x_0); x_1, f(x_1), f'(x_1)$, 则余项 $R_3(x) = \frac{f^{(4)}(\xi)}{4!}(x-x_0)^2(x-x_1)^2]$

1.3 多项式插值的牛顿 (Newton) 型式

1.3.1 差商及其计算

$$\text{一阶差商} \quad f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

$$\text{二阶差商} \quad f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0}$$

$$\text{零阶差商} \quad f[x_0] = f(x_0)。$$

k 阶差商 设 x_0, x_1, \dots, x_k 互不相同, $f(x)$ 关于 x_0, x_1, \dots, x_k 的 k 阶差商为

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

性质 1 k 阶差商 $f[x_0, x_1, \dots, x_k]$ 是由函数值 $f(x_0), f(x_1), \dots, f(x_k)$ 的线性组合而成。

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

性质 2 若 i_0, i_1, \dots, i_k 为 $0, 1, \dots, k$ 的任一排列, 则 $f[x_0, x_1, \dots, x_k] = f[x_{i_0}, x_{i_1}, \dots, x_{i_k}]$

此即插商的值只与节点有关而与节点顺序无关。

性质 3 若 $f(x)$ 为 m 次多项式, 则 $f[x_0, x_1, \dots, x_{k-1}, x]$ 为 $m-k$ 次多项式。

差商计算

差商表

1.3.2 Newton 插值

线性插值，二次多项式插值，n 次 Newton 差值函数

$$N_1(x) = f(x_0) + f[x_0, x_1](x - x_0)$$

$$N_2(x) = f(x_0) + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1)$$

$$N(x) = f[x_0] + \sum_{k=1}^n f[x_0, x_1, \dots, x_k](x - x_0)(x - x_1) \cdots (x - x_{k-1})$$

其中 $R(x) = f[x, x_0, x_1, \dots, x_n] \prod_{i=0}^n (x - x_i)$ 为插值多项式的误差。

[编者注：该式计算不出差商的准确值]

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

至多为 n 次多项式，可以验证 $N(x_i) = f(x_i)$, $i = 1, 2, \dots, n$ ；称 $N(x)$ 是过 $n+1$ 个插值点的（至

多） n 阶 Newton 插值多项式。也有： $N(x) = f[x_0] + \sum_{k=1}^n f[x_0, x_1, \dots, x_k](x - x_0)(x - x_1) \cdots (x - x_{k-1})$

牛顿插值多项式的承袭性质表现在

$$N_k(x) = N_{k-1}(x) + t_k(x)f[x_0, x_1, \dots, x_k] \quad t_k(x) = (x - x_0)(x - x_1) \cdots (x - x_{k-1})$$

有关误差

$$R(x) = f[x, x_0, x_1, \dots, x_n] \prod_{i=0}^n (x - x_i) \text{ 为插值多项式的误差。}$$

当 $f \in C^{n+1}[a, b]$ 时，有 $R(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{i=0}^n (x - x_i) = f[x, x_0, \dots, x_n] \prod_{i=0}^n (x - x_i)$

故有 $\frac{f^{(n+1)}(\xi)}{(n+1)!} = f[x, x_0, x_1, \dots, x_n]$

例 设 $f(x) = 10x^3 - 100x + 1$ ，计算 $f[x_0, x_1, x_2, x_3]$ ， $f[x_0, x_1, x_2, x_3, x_4]$ 。[必考题型]

$$f[x_0, x_1, x_2, x_3] = \frac{f^{(3)}(\xi)}{3!} = \frac{60}{6} = 10, \quad f[x_0, x_1, x_2, x_3, x_4] = \frac{f^{(4)}(\xi)}{4!} = \frac{0}{4!} = 0$$

*1.4 Hermite 插值

常用 Hermite 插值描述如下：对于 $f(x)$ 具有一阶连续导数，以及插值点 x_i , $i=0, 1, \dots, n$, x_i 互

不相同，若有至多为 $2n+1$ 次的多项式函数 $H_{2n+1}(x)$ 满足

$$H_{2n+1}(x_i) = f(x_i) \quad H'_{2n+1}(x_i) = f'(x_i), i = 0, 1, \dots, n$$

则称 $H_{2n+1}(x)$ 为 $f(x)$ 关于节点 $\{x_i\}_{i=0}^n$ 的 Hermite 的插值多项式。

$$h_i(x) = (1 - 2(x - x_i)l_i'(x_i))l_i^2(x),$$

$$h_i(x) = (1 - 2(x - x_i)\sum_{j \neq i} \frac{1}{x_i - x_j})l_i^2(x),$$

$$g_i(x) = (x - x_i)l_i^2(x)$$

[编者注：此时插值基函数 $l_i'(x_i) = \sum_{j \neq i} \frac{1}{x_i - x_j}$]

容易证明，当 $f \in C^{2n+2}[a, b]$ 时，误差为：

$$R(x) = f(x) - H_{2n+1}(x) = \frac{f^{(2n+2)}(\xi)}{(2n+2)!} (x - x_0)^2 (x - x_1)^2 \cdots (x - x_n)^2, \xi \in [a, b]$$

用牛顿插值构造 Hermite 插值

[编者注：考试时求 Hermite 插值建议用这种方法！]

$$H_{2n+1}(x) = f[z_0] + \sum_{k=1}^{2n+1} f[z_0, z_1, \dots, z_k](x - z_0)(x - z_1) \cdots (x - z_{k-1})$$

其中 $z_{2k} = z_{2k+1} = x_i, f[z_{2k}, z_{2k+1}] = f'(x_i), k = 0, 1, \dots, n$

[编者注：1. 在给定题目条件时，若给定了 $f^{(n)}(x_i)$ ，那么 $f^{(n-1)}(x_i), f^{(n-2)}(x_i), \dots, f(x_i)$ 也

必须给定；2. 最后的插值表达式要按照 Z 的多项式来写]

1.5 分段插值函数

1.5.1 龙格(Runge)现象

1.5.2 分段线性插值

$$p_i(x) = \frac{x - x_{i+1}}{x_i - x_{i+1}} f(x_i) + \frac{x - x_i}{x_{i+1} - x_i} f(x_{i+1}), x_i \leq x \leq x_{i+1}$$

1.6 三次样条函数

$S(x)$ 在 $[a, b]$ 上有连续的二阶导数

1.6.1 三次样条插值的 M 关系式

$$S(x) = S_i(x) = \frac{(x_{i+1} - x)^3 M_i + (x - x_i)^3 M_{i+1}}{6h_i} + \frac{(x_{i+1} - x)y_i + (x - x_i)y_{i+1}}{h_i}$$

$$-\frac{h_i}{6} [(x_{i+1} - x)M_i + (x - x_i)M_{i+1}], x \in [x_i, x_{i+1}] \quad i = 1, 2, \dots, n$$

$$\mu_i M_{i-1} + 2M_i + \lambda_i M_{i+1} = d_i \quad i = 1, 2, \dots, n-1$$

$$\lambda_i = \frac{h_i}{h_i + h_{i-1}}, \mu_i = 1 - \lambda_i, \quad d_i = \frac{6}{h_i + h_{i-1}} \left(\frac{y_{i+1} - y_i}{h_i} - \frac{y_i - y_{i-1}}{h_{i-1}} \right) = 6y[x_{i-1}, x_i, x_{i+1}]$$

得到 $n+1$ 个未知数的 $n-1$ 个方程组。现补充两个边界条件，使方程组只有唯一解。下面分三种情况讨论边界条件。

(1) 给定 M_0, M_n 的值 ($M_0 = 0, M_n = 0$ 时，称为自然边界条件)，此时 $n-1$ 阶方程组有 $n-1$ 个未知量 $M_i, i = 1, 2, \dots, n-1$

$$\begin{bmatrix} 2 & \lambda_1 & & & \\ \mu_2 & 2 & \lambda_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \lambda_{n-2} & 2 & \lambda_{n-1} \\ & & & \mu_{n-1} & 2 \end{bmatrix} \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_{n-2} \\ M_{n-1} \end{bmatrix} = \begin{bmatrix} d_1 - \mu_1 M_0 \\ d_2 \\ \vdots \\ d_{n-2} \\ d_{n-1} - \lambda_{n-1} M_n \end{bmatrix}$$

(2) 给定 $S'(x_0) = m_0, S'(x_n) = m_n$ 的值，将 $S'(x_0) = m_0, S'(x_n) = m_n$ 的值分别代入 $S'(x)$ 在 $[x_0, x_1], [x_{n-1}, x_n]$ 中的表达式，得到另外两个方程。

$$2M_0 + M_1 = \frac{6}{h_0} [f[x_0, x_1] - m_0] = d_0 \quad \text{及} \quad M_{n-1} + 2M_n = \frac{6}{h_{n-1}} [m_n - f[x_{n-1}, x_n]] = d_n$$

得到 $n+1$ 阶的方程组

$$\begin{bmatrix} 2 & 1 & & & \\ u_1 & 2 & \lambda_1 & & \\ & u_2 & 2 & \lambda_2 & \\ & & \ddots & \ddots & \ddots \\ & & & u_{n-2} & 2 & \lambda_{n-1} \\ & & & & 1 & 2 \end{bmatrix} \begin{bmatrix} M_0 \\ M_1 \\ M_2 \\ \vdots \\ M_{n-1} \\ M_n \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \end{bmatrix}$$

(3) 被插函数以 $x_n - x_0$ 为基本周期时，即 $f(x_0) = f(x_n)$,

例 给出离散数值表

x_i	1.1	1.2	1.4	1.5
y_i	0.4000	0.8000	1.6500	1.8000

取 $M_0 = M_n = 0$ ，构造三次样条插值的 M 关系式，并计算 $f(1.25)$ 。

解 由题中 (x_i, y_i) 的数值，计算得 $h_0 = 0.1, h_1 = 0.2, h_2 = 0.1$;

$$\begin{cases} \lambda_1 = 0.6667, \\ \mu_1 = 0.3333, \end{cases} \quad \begin{cases} \lambda_2 = 0.3333, \\ \mu_2 = 0.6667, \end{cases} \quad d_1 = 5, \quad d_2 = -55.$$

由 $M_0 = M_n = 0$ 的边界条件, 得

$$\begin{bmatrix} 2 & 0.6667 \\ 0.6667 & 2 \end{bmatrix} \begin{bmatrix} M_1 \\ M_2 \end{bmatrix} = \begin{bmatrix} 5 \\ -55 \end{bmatrix},$$

解得 $M_1 = 13.125$, $M_2 = -31.875$ 。

因此, 三次样条插值的分段表达为

$$S(x) = \begin{cases} 21.875x^3 - 72.1875x^2 + 83.1875x - 32.875, & x \in [1.1, 1.2], \\ -37.5x^3 + 141.5625x^2 - 173.75x + 59.725, & x \in [1.2, 1.4], \\ 53.125x^3 - 239.0625x^2 + 358.0625x - 179.05, & x \in [1.4, 1.5]. \end{cases}$$

特别地, $f(1.25) \approx S(1.25) = 1.0436$ 。

1.6.2 三次样条插值的 m 关系式

$$\begin{aligned} S(x) = & \left(1 - 2 \frac{x - x_i}{x_i - x_{i+1}}\right) \left(\frac{x - x_{i+1}}{x_i - x_{i+1}}\right)^2 y_i + (x - x_i) \left(\frac{x - x_{i+1}}{x_i - x_{i+1}}\right)^2 m_i \\ & + \left(1 - 2 \frac{x - x_{i+1}}{x_{i+1} - x_i}\right) \left(\frac{x - x_i}{x_{i+1} - x_i}\right)^2 y_{i+1} + (x - x_{i+1}) \left(\frac{x - x_i}{x_{i+1} - x_i}\right)^2 m_{i+1} \\ & \lambda_i m_{i-1} + 2m_i + \mu_i m_{i+1} = c_i \quad i = 1, 2, \dots, n-1 \end{aligned}$$

$$\lambda_i = \frac{h_i}{h_i + h_{i-1}}, \mu_i = 1 - \lambda_i, \quad c_i = 3 \left[\lambda_i y[x_{i-1}, x_i] + \mu_i y[x_i, x_{i+1}] \right]$$

再附加两个边界条件, 即可解出 m_i 的值。附加的边界条件情况同 M 关系式中类似, 不再详说。

[这里的 m 方法的推导以及第三种追赶法的推导回去要自己推导一下啊]

第 2 章 数值微分和数值积分

2.1 数值微分

2.1.1 差商与数值微分

向前差商

$$f'(x_0) \approx \frac{f(x_0 + h) - f(x_0)}{h}$$

$x_0 + h$ 的位置在 x_0 的前面, 因此称为向前差商。得向前差商的截断误差阶

$$R(x) = f'(x_0) - \frac{f(x_0 + h) - f(x_0)}{h} = -\frac{h}{2} f''(\xi) = O(h)$$

向后差商

$$f'(x_0) \approx \frac{f(x_0) - f(x_0 - h)}{h}$$

与计算向前差商的方法类似，由泰勒展开得到向后差商的截断误差阶

$$R(x) = f'(x_0) - \frac{f(x_0) - f(x_0 - h)}{h} = O(h)$$

中心差商

用差商近似导数有：

$$f'(x_0) \approx \frac{f(x_0 + h) - f(x_0 - h)}{2h}$$

$$R(x) = f'(x_0) - \frac{f(x_0 + h) - f(x_0 - h)}{2h} = -\frac{h^2}{12} [f'''(\xi_1) + f'''(\xi_2)] \quad [\text{编者注：设 } f'''(x) \text{ 连续}]$$

$$= -\frac{h^2}{6} f'''(\xi) = O(h^2) \quad x_0 - h \leq \xi \leq x_0 + h$$

2.1.2 插值型数值微分 [编者注：多用 Lagrange 插值多项式]

对于给定的 $f(x)$ 的函数表，建立插值函数 $L(x)$ ，用插值函数 $L(x)$ 的导数近似函数 $f(x)$ 的导数。

$$f(x) \approx L_n(x) = \sum_{i=0}^n \ell_i(x) f(x_i), \quad f'(x) \approx L'_n(x) = \sum_{i=0}^n \ell'_i(x) f(x_i)$$

$$\text{当 } x = x_j \text{ 时, } f'(x_j) = \sum_{i=0}^n \ell'_i(x_j) f(x_i) \quad j = 0, 1, \dots, n$$

$$\text{误差项} \quad R(x) = \frac{d}{dx} \left(\frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{i=0}^n (x - x_i) \right) \quad R(x_j) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{\substack{i=0 \\ i \neq j}}^n (x_j - x_i)$$

2.2 数值积分

$$\text{记} \quad I(f) = \int_a^b f(x) dx, \quad I_n(f) = \sum_{i=0}^n \alpha_i f(x_i)$$

在本章中，用 $I(f)$ 表示精确积分值，用 $I_n(f)$ 表示近似积分值， $\{x_i\}$ 称为求积节点， α_i 称为

求积系数，确定 $I_n(f)$ 中积分系数 α_i 的过程就是构造数值积分公式的过程。

代数精度

记 $[a, b]$ 上以 $x_i, i = 0, 1, \dots, n$ 为积分节点的数值积分公式为 $I_n(f) = \sum_{i=0}^n \alpha_i f(x_i)$

若 $I_n(f)$ 满足 $E_n(x^k) = I(x^k) - I_n(x^k) = 0, \quad k = 0, 1, \dots, m$ 而 $E_n(x^{m+1}) \neq 0$, 则称 $I_n(f)$ 具有 m 阶代数精度。由此可知当 $I_n(f)$ 具有 m 阶代数精度时, 对任意的不高于 m 次多项式 $f(x)$ 都有 $I(f) = I_n(f)$ 。

[编者注: 在求数值积分的余项时, 要先计算其代数精度, 常用方法是: 分别带入 $1, x, x^2$, 对比等式两边, 看是否精确相等。进而确定泰勒展开的阶数: 泰勒展开阶数=代数精度数]

2.2.1 插值型数值积分

$$\int_a^b f(x) dx \approx \int_a^b L_n(x) dx = \int_a^b \sum_{i=0}^n \ell_i(x) f(x_i) dx = \sum_{i=0}^n \left[\int_a^b \ell_i(x) dx \right] f(x_i)$$

$$\text{记 } \alpha_i = \int_a^b \ell_i(x) dx, \text{ 则有 } I_n(f) = \int_a^b L_n(x) dx = \sum_{i=0}^n \alpha_i f(x_i) \quad [\text{积分系数是对基函数的积分}]$$

数值积分误差, 也就是对插值误差的积分值

$$E_n(f) = \int_a^b R_n(x) dx = \frac{1}{(n+1)!} \int_a^b f^{(n+1)}(\xi(x)) \prod_{i=0}^n (x - x_i) dx$$

对一般的函数 $E_n(f) \neq 0$, 但若 $f(x)$ 是一个不高于 n 次的多项式, 由于 $f^{(n+1)}(x) = 0$, 而有 $E_n(f) = 0$ 。因此, n 阶插值多项式型式的数值积分公式至少有 n 阶代数精度。

2.2.2 牛顿-柯特斯 (Newton-Cote's) 积分 [等距积分]

(牛顿-柯特斯积分系数和积分节点以及积分区间无直接关系, 系数固定而易于计算。)

梯形积分 [n=1 的等距分布]

$$\int_a^b f(x) dx \approx \frac{b-a}{2} [f(a) + f(b)], \text{ 记 } T(f) = \frac{b-a}{2} [f(a) + f(b)], \text{ 称 } T(f) \text{ 为梯型积分公式。}$$

梯形求积公式具有一阶代数精度。

$$E_1(x) = \frac{f''(\eta)}{2!} \int_a^b (x-a)(x-b) dx = -\frac{f''(\eta)}{12} (b-a)^3, \quad a \leq \eta \leq b$$

辛普森 (Simpson) 积分 [n=2 的等距分布]

$$\int_a^b f(x) dx \approx I_2(f) = S(f) = \frac{b-a}{6} \left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right]$$

称 $S(f)$ 为辛普森 (Simpson) 或抛物线积分公式。 $S(f)$ 具有三阶代数精度。

关于牛顿-柯特斯积分误差, 这里不加证明给出如下结果:

(1). 若 n 为奇数, $f \in C^{n+1}[a, b]$, 积分公式有 n 阶代数精度。

(2).若 n 为偶数, $f \in C^{n+2}[a, b]$, 积分公式具有 $n+1$ 阶代数精度。

Newton-Cotes 系数表

n	$c_0^{(n)}$	$c_1^{(n)}$	$c_2^{(n)}$	$c_3^{(n)}$	$c_4^{(n)}$	$c_5^{(n)}$	$c_6^{(n)}$
1	$\frac{1}{2}$	$\frac{1}{2}$					
2	$\frac{1}{6}$	$\frac{4}{6}$	$\frac{1}{6}$				
3	$\frac{1}{8}$	$\frac{3}{8}$	$\frac{3}{8}$	$\frac{1}{8}$			
4	$\frac{7}{90}$	$\frac{16}{45}$	$\frac{2}{15}$	$\frac{16}{45}$	$\frac{7}{90}$		
5	$\frac{19}{288}$	$\frac{25}{96}$	$\frac{25}{144}$	$\frac{25}{144}$	$\frac{25}{96}$	$\frac{19}{288}$	
6	$\frac{41}{840}$	$\frac{9}{35}$	$\frac{9}{280}$	$\frac{34}{105}$	$\frac{9}{280}$	$\frac{9}{35}$	$\frac{41}{840}$

$$\sum_{i=0}^n C_i = 1$$

2.3 复化数值积分

2.3.1 复化梯形积分

计算公式

$$T(h) = T_n(f) = h \left[\frac{1}{2} f(a) + \sum_{i=1}^{n-1} f(a + ih) + \frac{1}{2} f(b) \right]$$

截断误差

$$E_n(f) = -\frac{nh^3}{12} f''(\xi) = -\frac{h^2}{12} (b-a) f''(\xi) = -\frac{(b-a)^3}{12n^2} f''(\xi), \quad a \leq \xi \leq b$$

2.3.2 复化 Simpson 积分

把积分区间分成偶数 $2m$ 等分, 记 $n = 2m$, 其中 $n+1$ 是节点总数, m 是积分子区间的总数。

记 $h = \frac{b-a}{n}$, $x_i = a + ih$, $i = 0, 1, \dots, n$, 在每个子区间 $[x_{2i}, x_{2i+2}]$ 上用 Simpson 数值积分公式计

算, 则得到复化 Simpson 公式, 记为 $S_n(f)$ 。

计算公式

$$I(f) = \int_a^b f(x) dx = \sum_{i=0}^{m-1} \int_{x_{2i}}^{x_{2i+2}} f(x) dx$$

注意区间的分割方法! 区间端点、区间长度应与公式匹配。例如对于这里给出的分割方法, 积分区间长度就是 $2h$

$$\text{而 } \int_{x_{2i}}^{x_{2i+2}} f(x)dx = \frac{2h}{6} [f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})] - \frac{(2h)^5}{2880} f^{(4)}(\xi_i)$$

$$S_n(f) = \sum_{i=0}^{m-1} \frac{2h}{6} [f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})] = \frac{h}{3} \left[f(a) + 4 \sum_{i=0}^{m-1} f(x_{2i+1}) + 2 \sum_{i=1}^{m-1} f(x_{2i}) + f(b) \right]$$

截断误差

$$\begin{aligned} I(f) - S_n(f) &= -\frac{(2h)^5}{2880} \sum_{i=0}^{m-1} f^{(4)}(\xi_i) = -\frac{(2h)^5 m}{2880} f^{(4)}(\xi) \\ &= \frac{-(b-a)^5}{2880m^4} f^{(4)}(\xi) = \frac{-(b-a)^5}{180n^4} f^{(4)}(\xi) \\ &= -\frac{(b-a)f^{(4)}(\xi)}{180} h^4 = O(h^4), \quad a \leq \xi \leq b \end{aligned}$$

与复化梯形公式类似，误差的截断误差按照 h^4 或说 $\frac{1}{n^4}$ 的下降速度下降，

2.3.3 复化积分的自动控制误差算法

2.3.4 龙贝格 (Romberg) 积分

龙贝格计算公式:
$$R_{k,j} = R_{k,j-1} + \frac{R_{k,j-1} - R_{k-1,j-1}}{4^{j-1} - 1}, k = 2, 3, \dots$$

对每一个 k, j 从 2 做到 k ，一直做到 $|R_{k,k} - R_{k-1,k-1}|$ 小于给定控制精度时停止计算。

$$\begin{array}{llll} R_{1,1} = T_n & & & \\ R_{2,1} = T_{2n} & R_{2,2} = S_n = T_{2n} + \frac{1}{3}(T_{2n} - T_n) & & \\ R_{3,1} = T_{4n} & R_{3,2} = S_{2n} = T_{4n} + \frac{1}{3}(T_{4n} - T_{2n}) & R_{3,3} = C_n = S_{2n} + \frac{1}{15}(S_{2n} - S_n) & \\ R_{4,1} = T_{8n} & R_{4,2} = S_{4n} = T_{8n} + \frac{1}{3}(T_{8n} - T_{4n}) & R_{4,3} = C_{2n} = S_{4n} + \frac{1}{15}(S_{4n} - S_{2n}) & R_{4,4} = C_{2n} + \frac{1}{63}(C_{2n} - C_n) \\ \vdots & \vdots & \vdots & \vdots \\ O(h^2) & O(h^4) & O(h^6) & O(h^8) \end{array}$$

2.4 重积分计算

$$= hk \sum_{i=0}^m \sum_{j=0}^n c_{i,j} f(x_i, y_j)$$

积分区域的 4 个角点的系数是 1/4，4 个边界的系数是 1/2，内部节点的系数是 1。

*2.5 高斯 (Gauss) 型积分

$$I(f) = \int_a^b f(x)dx \text{ 关于积分节点 } x_0, x_1, \dots, x_n \text{ 的数值积分公式为 } I_n(f) = \sum_{i=0}^n \alpha_i f(x_i), \text{ 它至少有}$$

n 阶代数精度，

定理 2.1 $I(f) = \int_a^b f(x)dx$ 关于积分节点 x_1, x_2, \dots, x_n 的数值积分公式 $I_n(f) = \sum_{i=1}^n \alpha_i f(x_i)$ 的

代数精度不超过 $2n-1$ 阶。[起始积分节点与精度表达式要匹配！]

定理 2.2 对 $I(f) = \int_{-1}^1 f(x)dx$ ，若选取正交多项式 $L_n(x)$ 的 n 个零点 $x_1^{(n)}, x_2^{(n)}, \dots, x_n^{(n)}$ 为数值积

分节点，则其数值积分公式 $I_n(f) = \sum_{i=1}^n \alpha_i^{(n)} f(x_i^{(n)})$ 具有 $2n-1$ 阶代数精度。

2.5.1 勒让德(Legendre)多项式

n 次多项式 $L_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n$ 称为 Legendre 多项式， $\{L_n(x)\}$ 为 $[-1, 1]$ 上的正交

多项式系，即 $(L_n(x), L_m(x)) = \int_{-1}^1 L_n(x) L_m(x) dx = 0, m \neq n$

关于 $L_n(x)$ ，它具有如下性质：

(1) $L_n(x)$ 在 $(-1, 1)$ 上有 n 个相异的实根 $x_1^{(n)}, x_2^{(n)}, \dots, x_n^{(n)}$ 。

(2) $L_n(x)$ 在 $[-1, 1]$ 上正交于任何一个不高于 $n-1$ 次的多项式，即若 $P(x)$ 为一个不高于 $n-1$ 次的

多项式，则 $(L_n(x), P(x)) = \int_{-1}^1 L_n(x) P(x) dx = 0$

n	$x_i^{(n)}$	$\alpha_i^{(n)}$	n	$x_i^{(n)}$	$\alpha_i^{(n)}$
1	0	2		± 0.6612093865	0.3607615730
2	± 0.5773502692	1		± 0.2386191861	0.4679139346
3	± 0.7745966692	0.5555555556	7	± 0.9491079123	0.1294849662
	0	0.8888888889		± 0.7415311856	0.2797053915
4	± 0.8611363116	0.3478548451		± 0.4058451514	0.3818300505
	± 0.3399810436	0.6521451549		0	0.4179591837
5	± 0.9061798459	0.2369268851	8	± 0.9602898565	0.1012285363
	± 0.5384693101	0.4786286705		± 0.7966664774	0.2223810345
	0	0.5688888889		± 0.5255324099	0.3137066459
6	± 0.9324695142	0.1713244924		± 0.1834346425	0.3626837834

第 3 章 曲线拟合的最小二乘法

3.1 拟合曲线

$$\text{均方误差 } R_2^2 = \sum_{i=1}^m |\varphi(x_i) - y_i|^2$$

3.2 线性拟合和二次函数拟合

线性拟合

给定一组数据 $(x_i, y_i), i = 1, 2, \dots, m$ 做拟合直线 $p(x) = a + b(x)$, 根据均方误差最小可以

求得拟合曲线法方程为
$$\begin{pmatrix} m & \sum_{i=1}^m x_i \\ \sum_{i=1}^m x_i & \sum_{i=1}^m x_i^2 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^m y_i \\ \sum_{i=1}^m x_i y_i \end{pmatrix},$$
 求出 a, b 即可。

二次函数拟合

给定一组数据 $(x_i, y_i), i = 1, 2, \dots, m$ 做拟合直线 $p(x) = a_0 + a_1 x + a_2 x^2$, 根据均方误差最

小可求得拟合曲线法方程
$$\begin{pmatrix} m & \sum_{i=1}^m x_i & \sum_{i=1}^m x_i^2 \\ \sum_{i=1}^m x_i & \sum_{i=1}^m x_i^2 & \sum_{i=1}^m x_i^3 \\ \sum_{i=1}^m x_i^2 & \sum_{i=1}^m x_i^3 & \sum_{i=1}^m x_i^4 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^m y_i \\ \sum_{i=1}^m x_i y_i \\ \sum_{i=1}^m x_i^2 y_i \end{pmatrix},$$
 求出 a_0, a_1, a_2 即可。

形如 $y = ae^{bx}$ 的曲线拟合

法 1: 进行线性变换后再进行线性拟合。其他类似的函数也可以通过线性变换后进行拟合。

法 2: 解矛盾方程组方法求解。 $A^T A \alpha = A^T Y$

$$\begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_m \end{pmatrix} \begin{pmatrix} 1 \\ x_1 \\ \vdots \\ x_m \end{pmatrix} \begin{pmatrix} \ln a \\ b \end{pmatrix} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_m \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

3.3 解矛盾方程组

具体描述见教材。

定理 3.1 (1) A 为 m 行 n 列的矩阵, b 为列向量。 $A^T A X = A^T b$ 称为矛盾方程组 $AX=b$ 的法方程, 法方程恒有解。

(2) X 是 $\min \|AX - b\|_2^2$ 的解, 当且仅当 X 满足 $A^T A X = A^T b$, 即 X 是法方程的解。

具体证明与例题见教材。

第 4 章 非线性方程求根

4.1 实根的对分法

4.2 迭代法

定理 1: 若 $\varphi(x)$ 定义在 $[a, b]$ 上, 如果 $\varphi(x)$ 满足 1) 当 $x \in [a, b]$ 时有 $a \leq \varphi(x) \leq b$;

2) $\varphi(x)$ 在 $[a, b]$ 上可导, 并且存在 $\boxed{\text{正数 } L < 1}$, 使对任意的 $x \in [a, b]$, 有 $|\varphi'(x)| \leq L$; 则在 $[a, b]$ 上有唯一的点 x^* 满足 $x^* = \varphi(x^*)$, 称 x^* 为 $\varphi(x)$ 的不动点。而且迭代格式 $x_{k+1} = \varphi(x_k)$ 对任意的

初值 $x_0 \in [a, b]$ 均收敛于 $\varphi(x)$ 的不动点 x^* , 并有误差估计式 $|x^* - x_k| \leq \frac{L^k}{1-L} |x_1 - x_0|$

[编者注: 如果 $|\varphi'(x)| > 1$, 不能保证不收敛]

其一, 等价形式 $x = \varphi(x)$ 应满足 $\boxed{|\varphi'(x^*)| < 1}$; 其二, 初始必须取自 x^* 的充分小邻域, 这个邻域大小决定于函数 $f(x)$, 以及做出的等价形式 $x = \varphi(x)$ 。

4.3 牛顿迭代法

迭代格式

迭代格式:
$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \quad k = 1, 2, \dots$$

牛顿迭代格式对应于 $f(x) = 0$ 的等价方程是 $\varphi(x) = x - \frac{f(x)}{f'(x)}$, $\varphi'(x) = \frac{f(x)f''(x)}{(f'(x))^2}$

若 α 是 $f(x)$ 的单根时, $f(\alpha) = 0, f'(\alpha) \neq 0$, 则有 $|\varphi'(\alpha)| = 0$,

2. 收敛的阶

若存在 $M > 0$, $\lim_{k \rightarrow \infty} \frac{|e_{k+1}|}{|e_k|^n} = \lim_{k \rightarrow \infty} \frac{|x_{k+1} - \alpha|}{|x_k - \alpha|^n} = M$, 称收敛的阶为 n 牛顿迭代是二阶迭代方法。

例 用牛顿迭代法求方程 $x^3 - 7.7x^2 + 19.2x - 15.3 = 0$, 在 $x_0 = 1$ 附近的根。

解 $f(x) = x^3 - 7.7x^2 + 19.2x - 15.3$,

$$x_{k+1} = x_k - \frac{x_k^3 - 7.7x_k^2 + 19.2x_k - 15.3}{3x_k^2 - 15.4x_k + 19.2} = x_k - \frac{((x_k - 7.7)x_k + 19.2)x_k - 15.3}{(3x_k - 15.4)x_k + 19.2}$$

k	x_k	$f(x)$
0	1.00	-2.8
1	1.41176	-0.727071
2	1.62424	-0.145493
3	1.6923	-0.0131682
4	1.69991	-0.0001515
5	1.7	0

4.4 弦截法

1. 弦截法迭代格式

$$x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}, \quad k = 1, 2, \dots$$

弦截法收敛的阶 弦截法为 1.618 阶迭代方法。

例 用弦截法求方程 $x^3 - 7.7x^2 + 19.2x - 15.3 = 0$ 的根, 取 $x_0 = 1.5, x_1 = 4.0$ 。

解 $f(x) = x^3 - 7.7x^2 + 19.2x - 15.3, \quad x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}$

4.5 非线性方程组的牛顿方法

每一次迭代都是解一个方程组

$$J(x_k, y_k) \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} -f_1(x_k, y_k) \\ -f_2(x_k, y_k) \end{pmatrix}$$

$x_{k+1} = x_k + \Delta x, y_{k+1} = y_k + \Delta y$ 直到 $\max(|\Delta x|, |\Delta y|) < \varepsilon$ 为止。

例 求解非线性方程组 $\begin{cases} f_1(x, y) = 4 - x^2 - y^2 = 0 \\ f_2(x, y) = 1 - e^x - y = 0 \end{cases}$ 取初始值 $\begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \begin{pmatrix} 1 \\ -1.7 \end{pmatrix}$

解 $J(x, y) = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{bmatrix} = \begin{pmatrix} -2x & -2y \\ -e^x & -1 \end{pmatrix} \quad J(x_0, y_0) = \begin{pmatrix} -2 & 3.4 \\ -2.71828 & -1 \end{pmatrix}$

$$\begin{pmatrix} f_1(x_0, y_0) \\ f_2(x_0, y_0) \end{pmatrix} = \begin{pmatrix} 0.11 \\ -0.01828 \end{pmatrix} \quad \begin{cases} -2\Delta x + 3.4\Delta y = -0.11 \\ -2.71828\Delta x - \Delta y = 0.01828 \end{cases}$$

解方程得 $\begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} 0.004256 \\ -0.029849 \end{pmatrix}$

$$\therefore w_1 = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} 1 \\ -1.7 \end{pmatrix} + \begin{pmatrix} 0.004256 \\ -0.029849 \end{pmatrix} = \begin{pmatrix} 1.004256 \\ -1.729849 \end{pmatrix}$$

继续做下去, 直到 $\max(|\Delta x|, |\Delta y|) < 10^{-5}$ 时停止。

第 5 章 解线性方程组的直接法

5.1 消元法

5.1.1 三角形方程组的解

对角形方程组

求解 n 阶对角方程的运算量 (时间复杂度) $T(n) = O(n)$ 。

下三角方程组

$$x_i = (b_i - \sum_{j=1}^{i-1} \ell_{ij} x_j) / \ell_{ii}, \quad i = 1, 2, \dots, n$$

运算量（时间复杂度） $T(n) = \sum_{i=1}^n i = \frac{n(n+1)}{2} = O(n^2)$

上三角方程组

通式 $x_i = (b_i - \sum_{j=i+1}^n u_{ij} x_j) / u_{ii}, i = n, n-1, \dots, 2, 1$

时间复杂度: $T(n) = \sum_{i=n}^1 (n-i+1) = \sum_{j=1}^n j = \frac{n(n+1)}{2} = O(n^2)$

5.1.2 高斯（Gauss）消元法与列主元消元法

高斯（Gauss）消元法的运算量

整个消元过程的乘法和除法的运算量为 $\sum_{k=1}^{n-1} (n-k)(n-k+1) + \sum_{k=1}^{n-1} (n-k)$,

常数运算量

回代过程的乘除运算量为 $\frac{n(n+1)}{2}$,

系数运算量

故 Gauss 消元法的运算量为 $\frac{n^3}{3} + n^2 - \frac{1}{3}n = O(n^3)$

高斯（Gauss）消元法的可行性

在上面的消元法中，未知量是按照在方程组中的自然顺序消去的，也叫顺序消元法。

在消元过程中假定对角元素 $a_{kk}^{(k-1)} \neq 0$ ，消元步骤才能顺利进行，由于顺序消元不改变 A 的主子式值，故高斯消元法可行的充分必要条件为 **A 的各阶顺序主子式不为零**。但是，实际上只要 $\det A \neq 0$ ，方程组 $Ax = b$ 就有解。故高斯消元法本身具有局限性。

调换方程组的次序是依照在运算中做分母量的绝对值尽量大，以减少舍入误差的影响。如果在一系列中选取按模最大的元素，将其调到主方程位置再做消元，则称为列主元消元法。

5.1.3 Gauss-Jordan 消元法

5.2 直接分解法 [必考!]

5.2.1 Doolittle 分解

Doolittle 分解步骤

若 A 各阶主子式不为零，可分解为 $A=LU$ ，其中 L 为 **单位下三角阵**，U 为 **上三角阵**，即

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & a_{n3} & a_{nn} \end{pmatrix} = \begin{pmatrix} 1 & & & \\ l_{21} & 1 & & \\ \cdots & \cdots & \cdots & \\ l_{n1} & l_{n2} & \cdots & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \cdots & \cdots \\ & & & u_{nn} \end{pmatrix}$$

计算 U 的第一行元素 $u_{11}, u_{12}, \dots, u_{1n}$ 计算 L 的第一列元素 $l_{21}, l_{31}, \dots, l_{n1}$

计算 U 的第二行元素 计算 L 的第二列元素

.....

计算 U 的第 k 行元素 $u_{kk}, u_{k,k+1}, \dots, u_{k,n}$ 计算 L 的第 k 列元素 $l_{k+1,k}, l_{k+2,k}, \dots, l_{n,k}$

用 LU 直接分解方法求解方程组所需要的计算量仍为 $\frac{1}{3}n^3 + O(n^2)$, 用直接分解法解方程

$Ax = b$ 。首先作出分解 $A = LU$, 令 $UX=Y$, 解方程组 $LY = b$, 由于 L 是单位下三角阵, 容易得到

$$y_i = b_i - \sum_{j=1}^{i-1} l_{ij} x_j, \quad i = 1, 2, \dots, n$$

再解方程组 $UX=Y$, 其中

$$x_i = (y_i - \sum_{j=i+1}^n u_{ij} x_j) / u_{ii}, \quad i = n, n-1, \dots, 2, 1$$

5.2.2 Courant 分解

Courant 分解步骤

在矩阵 $A = LU$ 的 Courant 分解形式中, L 是下三角矩阵, U 是单位上三角矩阵,

$$\text{记 } L = \begin{pmatrix} l_{11} & & & \\ l_{21} & l_{22} & & \\ \dots & \dots & \dots & \\ l_{n1} & l_{n2} & \dots & l_{nn} \end{pmatrix}, \quad U = \begin{pmatrix} 1 & u_{12} & \dots & u_{1n} \\ & 1 & \dots & u_{2n} \\ & & \dots & \dots \\ & & & 1 \end{pmatrix}$$

矩阵 Courant 分解的次序与 Doolittle 分解的次序不同, 设 A 的各阶主子式不为零, 按计算 L 的第一列, U 的第一行, \dots , L 的第 k 列, U 的第 k 行的顺序进行计算,

如果 A 的各阶顺序主子式不为零, $l_{kk} \neq 0$, Courant 分解才能顺利进行。

5.2.3 追赶法

时间复杂度 $O(n)$

5.2.4 对称正定矩阵的 LDL^T 分解

对于对称正定矩阵, 常用的是 LDL^T 分解。

$$A = \begin{pmatrix} 1 & & & \\ \ell_{21} & 1 & & \\ & \dots & \dots & \\ \ell_{n1} & \ell_{n2} & \dots & 1 \end{pmatrix} \begin{pmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_n \end{pmatrix} \begin{pmatrix} 1 & l_{12} & \dots & l_{1n} \\ & 1 & \dots & l_{2n} \\ & & \ddots & \vdots \\ & & & 1 \end{pmatrix}$$

L 是对角元素为 1 的单位下三角阵。

对矩阵 A 做 Doolittle 或 Courant 分解, 共计计算 n^2 个矩阵元素; 对称矩阵的 LDL^T 分解,

只需计算 $\frac{n(n+1)}{2}$ 元素, 减少了近一半的工作量。

$$A = L(DL^T) = L\tilde{U} = \begin{pmatrix} 1 & & & \\ l_{21} & 1 & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & 1 \end{pmatrix} \begin{pmatrix} d_1 & d_1 l_{12} & \cdots & d_1 l_{1n} \\ & d_2 & \cdots & d_2 l_{2n} \\ & & \ddots & \vdots \\ & & & d_n \end{pmatrix}$$

例 用 LDL^T 分解求解方程组 $\begin{pmatrix} 1 & -1 & 1 \\ -1 & 3 & -2 \\ 1 & -2 & 4.5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 4 \\ -8 \\ 12 \end{pmatrix}$

解 k=1: $d_1 = a_{11} = 1, \quad \begin{matrix} l_{21} = a_{21} / d_1 = -1 \\ l_{31} = a_{31} / d_1 = 1 \end{matrix}$

k=2: $d_2 = a_{22} - l_{21}^2 d_1 = 2, \quad l_{32} = (a_{32} - l_{31} l_{21} d_1) / d_2 = -0.5$

k=3: $d_3 = a_{33} - l_{31}^2 d_1 - l_{32}^2 d_2 = 3$

$$L = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 1 & -0.5 & 1 \end{pmatrix}, \quad D = \begin{pmatrix} 1 & & \\ & 2 & \\ & & 3 \end{pmatrix}$$

由 $LDL^T X = b$, $LZ = b, \quad Z = (4, -4, 6)^T$

$$DY = Z, \quad Y = (4, -2, 2)^T \quad L^T X = Y, \quad X = \begin{pmatrix} 1 \\ -1 \\ 2 \end{pmatrix}$$

* 5.3 矩阵的条件数

第 6 章 解线性方程组的迭代法

迭代矩阵的谱半径 $\rho(M) = \max_{1 \leq i \leq n} |\lambda_i| < 1$ 是迭代收敛的充分必要条件, 其中 λ_i 是矩阵 M 的特征根。 [注意与第四章迭代收敛条件相区别!]

称谱半径小于 1 的矩阵为收敛矩阵。若 $\|A\|_p$ 为矩阵 A 的范数, 则总有 $\rho(A) \leq \|A\|_p$ 。

因此, 若 $\|A\|_p < 1$, 则 A 必为收敛矩阵。要特别注意是, 当 $\|M\|_1 > 1$ 或 $\|M\|_\infty > 1$ 时, 不能判

断迭代序列发散。范数小于 1 只是判断迭代矩阵收敛的充分条件, 当迭代矩阵的一种范数使 $\|B\| > 1$, 并不能确定迭代矩阵是否为收敛矩阵。

6.1 简单 (Jacobi) 迭代

6.1.1 简单 (Jacobi) 迭代格式

记 $b_{ij} = -a_{ij} / a_{ii}, g_i = y_i / a_{ii}$, 构造迭代形式

$$\begin{cases} x_1^{(k+1)} = b_{12}x_2^{(k)} + b_{13}x_3^{(k)} + \cdots + b_{1n}x_n^{(k)} + g_1 \\ x_2^{(k+1)} = b_{21}x_1^{(k)} + b_{23}x_3^{(k)} + \cdots + b_{2n}x_n^{(k)} + g_2 \\ \dots \\ x_n^{(k+1)} = b_{n1}x_1^{(k)} + b_{n2}x_2^{(k)} + \cdots + b_{nn}x_n^{(k)} + g_n \end{cases}$$

Jacobi 迭代矩阵

令 $D = \begin{pmatrix} a_{11} & & \\ & a_{22} & \\ & & \ddots \\ & & & a_{nn} \end{pmatrix}$, $AX = (D + A - D)X = y$, 得到等价方程组

$$DX = (D - A)X + y \quad X^{(k+1)} = D^{-1}(D - A)X^{(k)} + D^{-1}y \quad X^{(k+1)} = BX^{(k)} + g$$

记 $B = I - D^{-1}A$ $g = D^{-1}y$ 迭代矩阵

例 用 Jacobi 方法解下列方程组

$$\begin{cases} 2x_1 - x_2 - x_3 = -5 \\ x_1 + 5x_2 - x_3 = 8 \\ x_1 + x_2 + 10x_3 = 11 \end{cases}$$

解 方程组的迭代格式:

$$\begin{cases} x_1^{(k+1)} = 0.5x_2^{(k)} + 0.5x_3^{(k)} - 2.5 \\ x_2^{(k+1)} = -0.2x_1^{(k)} + 0.2x_3^{(k)} + 1.6 \\ x_3^{(k+1)} = -0.1x_1^{(k)} - 0.1x_2^{(k)} + 1.1 \end{cases}$$

或

$$\begin{pmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \end{pmatrix} = \begin{pmatrix} 0 & 0.5 & 0.5 \\ -0.2 & 0 & 0.2 \\ -0.1 & -0.1 & 0 \end{pmatrix} \begin{pmatrix} x_1^{(k)} \\ x_2^{(k)} \\ x_3^{(k)} \end{pmatrix} + \begin{pmatrix} -2.5 \\ 1.6 \\ 1.1 \end{pmatrix}$$

常数项不要忘记除!
上标 k 不要忘记加括号!

$$\therefore \|B\|_1 = 0.7 \quad \therefore \text{Jacobi 迭代收敛。}$$

取初始值 $X^{(0)} = (0, 0, 0)^T$,

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$\ X^{(k)} - X^{(k-1)}\ _\infty$
0	0	0	0	
1	-2.5	1.6	1.1	2.5
2	-1.15	2.32	1.19	1.35
3	-0.745	2.068	0.983	0.405
4	-0.9745	1.9456	0.9677	0.2295
5	-1.04335	1.98844	1.00289	0.06885

$$\begin{array}{c|c|c|c|c} 6 & -1.00433 & 2.00925 & 1.00549 & 0.039015 \\ \hline \end{array}$$

方程组的准确解是 $\{-1, 2, 1\}$ 。

6.1.2 Jacobi 迭代收敛条件

对于方程组 $AX = y$, 构造 Jacobi 迭代格式, $X^{(k+1)} = BX^{(k)} + g$, 其中 $B = I - D^{-1}A$, $g = D^{-1}y$,

从迭代矩阵考虑:

1) 迭代矩阵的谱半径 $\rho = \max_{1 \leq i \leq n} |\lambda_i| < 1$ 时, 迭代收敛, 这是收敛的充分必要条件。

2) 迭代矩阵的某范数 $\|B\| < 1$ 时, 是迭代收敛的充分条件。

从系数矩阵考虑:

A 为行对角优阵或列对角优阵 Jacobi 迭代收敛。

定理 6.1: 若方程组 $AX = y$ 的系数矩阵 A, 满足下列条件之一, 则其 Jacobi 迭代收敛。

(1) A 为严格行对角优阵, 即 $|a_{ii}| > \sum_{j \neq i} |a_{ij}|$, $i=1, 2, \dots, n$

(2) A 为严格列对角优阵, 即 $|a_{jj}| > \sum_{i \neq j} |a_{ij}|$, $j=1, 2, \dots, n$

6.2 高斯-赛德尔 (Gauss-Seidel) 迭代

6.2.1 高斯-赛德尔迭代公式

在 Jacobi 迭代中, 用 $(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)})^T$ 的值计算 $x_i^{(k+1)}$, $i=1, 2, \dots, n$ 的值, $x_i^{(k+1)}$ 的计算公式是

$$(X^{(k+1)} = BX^{(k)} + g) \quad x_i^{(k+1)} = \sum_{j=1}^n b_{ij} x_j^{(k)} + g_i \quad x_i^{(k+1)} = \sum_{j=1}^{i-1} b_{ij} x_j^{(k+1)} + \sum_{j=i+1}^n b_{ij} x_j^{(k)} + g_i$$

事实上, 在计算 $x_i^{(k+1)}$ 前, 已经得到 $x_1^{(k+1)}, \dots, x_{i-1}^{(k+1)}$ 的值, 不妨将已算出的分量直接代入迭代式

中, 及时使用最新计算出的分量值。此时: $x_i^{(k+1)} = \sum_{j=1}^{i-1} b_{ij} x_j^{(k+1)} + \sum_{j=i+1}^n b_{ij} x_j^{(k)} + g_i$ 即用向量

$(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)})^T$ 计算出 $x_1^{(k+1)}$ 的值, 用向量 $(x_1^{(k+1)}, x_2^{(k)}, \dots, x_n^{(k)})^T$ 计算 $x_2^{(k+1)}$ 的值, \dots , 用向量

$(x_1^{(k+1)}, \dots, x_{i-1}^{(k+1)}, x_i^{(k)}, \dots, x_n^{(k)})^T$ 计算出 $x_i^{(k+1)}$ 的值, 这种迭代格式称为 Gauss-Seidel 迭代。

Gauss-Seidel 迭代格式

$$\begin{cases} x_1^{(k+1)} = b_{12}x_2^{(k)} + b_{13}x_3^{(k)} + \dots + b_{1n}x_n^{(k)} + g_1 \\ x_2^{(k+1)} = b_{21}x_1^{(k+1)} + b_{23}x_3^{(k)} + \dots + b_{2n}x_n^{(k)} + g_2 \\ \dots \\ x_i^{(k+1)} = b_{i1}x_1^{(k+1)} + \dots + b_{i,i-1}x_{i-1}^{(k+1)} + 0 + b_{i,i+1}x_{i+1}^{(k)} \dots + b_{i,n}x_n^{(k)} + g_i \\ \dots \\ x_n^{(k+1)} = b_{n1}x_1^{(k+1)} + b_{n2}x_2^{(k+1)} + \dots + b_{n,n-1}x_{n-1}^{(k+1)} + g_n \end{cases}$$

例 用 Gauss-Seidel 方法解方程组

$$\begin{cases} 2x_1 - x_2 - x_3 = -5 \\ x_1 + 5x_2 - x_3 = 8 \\ x_1 + x_2 + 10x_3 = 11 \end{cases}$$

解 方程的迭代格式:

$$\begin{cases} x_1^{(k+1)} = 0.5x_2^{(k)} + 0.5x_3^{(k)} - 2.5 \\ x_2^{(k+1)} = -0.2x_1^{(k+1)} + 0.2x_3^{(k)} + 1.6 \\ x_3^{(k+1)} = -0.1x_1^{(k+1)} - 0.1x_2^{(k+1)} + 1.1 \end{cases}$$

取初始值 $X^{(0)} = (0, 0, 0)$,

$$k=1 \quad x_1^{(1)} = 0.5 \cdot 0 + 0.5 \cdot 0 - 2.5 = -2.5$$

$$x_2^{(1)} = -0.2 \cdot (-2.5) + 0.2 \cdot 0 + 1.6 = 2.1 \quad x_3^{(1)} = -0.1 \cdot (-2.5) - 0.1 \cdot 2.1 + 1.1 = 1.14$$

k=2

$$x_1^{(2)} = 0.5 \cdot 2.1 + 0.5 \cdot 1.14 - 2.5 = -0.88 \quad x_2^{(2)} = -0.2 \cdot (-0.88) + 0.2 \cdot 2.1 + 1.6 = 2.004$$

高斯-赛德尔迭代矩阵

设 $A = D + L + U$ 则迭代矩阵 $S = -(D + L)^{-1}U$, $f = (D + L)^{-1}y$

高斯-赛德尔迭代收敛条件

(从迭代矩阵, 从系数矩阵)

1) 当 $\rho(S) < 1$ 或 S 的某种范数 $\|S\| < 1$ 时, 迭代收敛; 另一方面, 直接根据方程组系数矩阵的特点作出判断。

2) 定理 6.2: 若方程组系数矩阵为列或行对角优时, 则 Gauss-Seidel 迭代收敛。

3) 定理 6.3: 若方程组系数矩阵 A 为对称正定阵, 则 Gauss-Seidel 迭代收敛。

6.3 松弛迭代

松弛迭代计算公式

有 $X^{(k+1)} = X^{(k)} + \Delta X^{(k)}$, 迭代格式为

$$X^{(k+1)} = X^{(k)} + \omega \Delta X^{(k)}$$

$$X^{(k+1)} = X^{(k)} + \omega (X^{(k+1)} - X^{(k)})$$

与等式左边的 $X^{(k+1)}$ 不同, 而要用 G-S 迭代结果带入

$$X^{(k+1)} = X^{(k)} + \omega (\tilde{L}X^{(k+1)} + \tilde{U}X^{(k)} + g - X^{(k)})$$

整理得

$$X^{(k+1)} = (1 - \omega)X^{(k)} + \omega (\tilde{L}X^{(k+1)} + \tilde{U}X^{(k)} + g)$$

ω 对于不同问题有不同取值, 且 $0 < \omega < 2$

这里 ω 称为松弛因子, 上式称为松弛迭代。

迭代的分量形式为

$$\begin{cases} x_1^{(k+1)} = (1 - \omega)x_1^{(k)} + \omega(b_{12}x_2^{(k)} + b_{13}x_3^{(k)} + \cdots + b_{1n}x_n^{(k)} + g_1) \\ x_2^{(k+1)} = (1 - \omega)x_2^{(k)} + \omega(b_{21}x_1^{(k+1)} + b_{23}x_3^{(k)} + \cdots + b_{2n}x_n^{(k)} + g_2) \\ \dots \\ x_n^{(k+1)} = (1 - \omega)x_n^{(k)} + \omega(b_{n1}x_1^{(k+1)} + b_{n2}x_2^{(k+1)} + \cdots + b_{nn}x_n^{(k+1)} + g_n) \end{cases}$$

G-S 迭代的分量形式

松弛迭代收敛条件

定理 6.4: 松弛迭代收敛的必要条件 $0 < \omega < 2$ 。

定理 6.5: 若 A 为正定矩阵, 则当 $0 < \omega < 2$ 时, 松弛迭代恒收敛。

通常, 把 $0 < \omega < 1$ 的迭代称为亚松弛迭代, 把 $\omega = 1$ 的迭代称为 Gauss-Seidel 迭代, 而把 $1 < \omega < 2$ 的迭代称为超松弛迭代。

6.4 逆矩阵计算

第 7 章 计算矩阵的特征值和特征向量

7.1 幂法

7.1.1 幂法计算 计算 A 的最大特征值。

下面讨论幂法中两种比较简单的情况。

按模最大的特征值只有一个, 且是单实根

得到按模最大的特征值 $\lambda_1 \approx x_i^{(k+1)} / x_i^{(k)}, i = 1, 2, \dots, n$, 相应的特征向量近似地为 $X^{(k)}$ 。

$\{x_i^{(k+1)} / x_i^{(k)}\}$ 收敛于 λ_1 的速度取决于比值 $\left| \frac{\lambda_2}{\lambda_1} \right|$ 的大小。

按模最大的特征值是互为反号的实根

设 $\lambda_1 > 0$, 且 $\lambda_1 = -\lambda_2$ 即 $|\lambda_1| = |\lambda_2| > |\lambda_3| \geq \dots \geq |\lambda_n|$, 这时有

$$\therefore \lambda_1 = \sqrt{x_i^{(k+2)} / x_i^{(k)}}$$

得到相应的特征向量

$$\begin{cases} V_1 = X^{(k+1)} + \lambda_1 X^{(k)} \\ V_2 = X^{(k+1)} - \lambda_1 X^{(k)} \end{cases}$$

7.1.2 幂法的规范运算

在实际计算中, 通常采用规范运算, 即对 $X^{(k)}$ 的每个元素除以 $X^{(k)}$ 按模最大的分量

$$\|X^{(k)}\|_{\infty} = \max_{1 \leq i \leq n} |x_i^{(k)}|。$$

规范运算可按下面公式进行 $\begin{cases} Y^{(k)} = X^{(k)} / \|X^{(k)}\|_{\infty}, & k = 0, 1, \dots, \\ X^{(k+1)} = AY^{(k)} \end{cases}$

规范化运算保证了 $\|Y^{(k)}\| = 1$, 即 $Y^{(k)}$ 按模最大分量的值保持为 1 或 -1。下面给出在规范运算中迭代序列的几种情况:

(1) 如果 $\{X^{(k)}\}$ 收敛, 则 A 的特征值按模最大分量的值仅有一个, 且 $\lambda_1 > 0$, 对充分大的 k , 按模最大分量 $x_j^{(k)}$ 不变号, 对应的 $|y_j^{(k)}| = 1$, $\lambda_1 \approx |x_j^{(k+1)}|$, 即 $\lambda_1 \approx \max_{1 \leq i \leq n} |x_i^{(k+1)}| = |x_j^{(k+1)}|$, 相应的特征向量是 $v_1 \approx Y^{(k)}$ 。

(2) 如果 $\{X^{(2k)}\}, \{X^{(2k+1)}\}$ 分别收敛于互为反号的向量, 则按模最大的特征值也仅有一个单实根,

且 $\lambda_1 < 0$ ，即对充分大的 k ，若 $x_j^{(k)}$ 的符号交替变号，则 λ_1 为负值。 $\lambda_1 \approx -\max_{1 \leq i \leq n} |x_i^{(k)}| = -|x_j^{(k)}|$ ，相应的特征向量是 $v_1 \approx Y^{(k)}$ 。

(3) 如果 $\{X^{(2k)}\}, \{X^{(2k+1)}\}$ 分别收敛于两个不同的向量 (与(2)不同)，则按模最大的特征值有两个，是互为反号的一对实根。这时，对充分大的 k ，再作一次非规范运算 $X^{(k+1)} = AX^{(k)}$

则 $\lambda_1 \approx \sqrt{x_i^{(k+1)} / y_i^{(k-1)}}$, $\lambda_2 = -\lambda_1$ 而仍有 $\begin{cases} V_1 = X^{(k+1)} + \lambda_1 X^{(k)} \\ V_2 = X^{(k+1)} - \lambda_1 X^{(k)} \end{cases}$ [注：当 k 很大时， $X^{(k+1)} = AY^{(k)}$, $X^{(k+2)} = AX^{(k+1)} = A^2 Y^{(k)} = \lambda_1^2 Y^{(k)}$]

不规范化！

(4) 如果 $\{X^{(k)}\}$ 的趋势无一定的规律，这时 A 的按模最大的特征值的情况更为复杂，需另行处理。

例 用规范运算计算矩阵 A 的按模最大的特征值和它的特征向量。 $A = \begin{pmatrix} 9 & -3 \\ 4 & 1 \end{pmatrix}$

解 用表给出计算结果

k	$Y^{(k)}$		$X^{(k+1)}$	
0	1	1	6	5
1	1.	0.833333	6.5	4.83333
2	1.	0.74359	6.76923	4.74359
3	1.	0.700758	6.89773	4.70076
4	1.	0.681493	6.95552	4.18149
5.	1.	0.673061	6.98081	4.67306
6	1.	0.669415	6.99176	4.66942
7	1.	0.66782	6.99654	4.66782
8	1.	0.667161	6.99852	4.66716
9	1.	0.666879	6.99936	4.66688
10	1.	0.666883	6.99935	4.66688

得到按模最大的特征值 $\lambda_1 \approx 6.99936$ ，特征向量 $v_1 = \begin{pmatrix} 1.0 \\ 0.666879 \end{pmatrix}$

例 用规范运算计算矩阵 A 的按模最大的特征值和它的特征向量 $A = \begin{pmatrix} 4 & -1 & 1 \\ 16 & -2 & -2 \\ 16 & -3 & -1 \end{pmatrix}$

解 计算结果列在表中。

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$y_1^{(k)}$	$y_2^{(k)}$	$y_3^{(k)}$
0	0.5	0.5	1	0.5	0.5	1
1	2.5	5	5.5	0.454545	0.909091	1

2	1.909089	3.454538	3.553628	0.537222	0.972116	1
3	2.176772	4.65132	4.679104	0.465201	0.994091	1
4	2.176772	3.455134	3.461124	0.539392	0.998269	1
5	2.159299	4.633734	4.635485	0.465721	0.999627	1
6	1.862661	3.452282	3.452655	0.539487	0.999892	1
7	2.158056	4.632000	4.632116	0.465890	0.999975	1
8	1.863585	3.454290	3.454315	0.534950	0.999993	1
9	2.157985	4.631926	4.631926	0.465893	0.999999	1
10	1.863573	3.454290	3.454291	0.539495	1	1
11	2.157980	4.631920	4.631920	0.465893	1	1
12	1.863572	3.454288	3.454288			
13	7.454288	16.	16.			

不规范化！

$$\lambda_1 = \sqrt{x_2^{(13)} / y_2^{(11)}} = 4, \lambda_2 = -4 \quad v_1 = X^{(13)} + \lambda_1 X^{(12)} = (14.908576, 29.817152, 29.817152)$$

$$v_2 = X^{(13)} - \lambda_1 X^{(12)} = (0, 2.182848, 2.182848)$$

有时题目会通过给定初值或k的前n组值，反推求解方法并完成求解

7.2 反幂法

反幂法是计算矩阵按模最小的特征值以及相应的特征向量的数值方法。

设矩阵A可逆， λ 和 v 分别为A的特征值以及相应的特征向量，并设 $\lambda_i \neq 0, i = 1, 2, \dots, n$ 。对

$Av = \lambda v$ 两边同乘 A^{-1} ，得 $A^{-1}v = \frac{1}{\lambda}v$ ，可见A和 A^{-1} 的特征值互为倒数，而且 v 也是 A^{-1} 的特征值 $\frac{1}{\lambda}$ 的特征向量。 A^{-1} 的按模最大的特征值正是A的按模最小的特征值的倒数，用幂法计算 A^{-1} 按模最大的特征值而得到A的按模最小的特征值的方法，称为反幂法。

用幂法计算 A^{-1} 按模最大的特征值仍可用规范方法：

$$\text{任取 } X^{(0)}, \text{ 规范迭代 } \begin{cases} Y^{(k)} = X^{(k)} / \|X^{(k)}\|_{\infty} \\ X^{(k+1)} = A^{-1}Y^{(k)} \end{cases}, \quad k = 0, 1, \dots$$

在实际计算中，不是先算出 A^{-1} ，再做乘积 $A^{-1}Y^{(k)}$ ，而是解方程 $AX^{(k+1)} = Y^{(k)}$ ，求得 $X^{(k+1)}$ ，求解过程一般采用直接分解法。将A分解为 $A = LU$ ，由 $LZ^{(k+1)} = Y^{(k)}$ 解出 $Z^{(k+1)}$ ，再由 $UX^{(k+1)} = Z^{(k+1)}$ 解出 $X^{(k+1)}$ 。

规范迭代计算公式：

$$\begin{cases} Y^{(k)} = X^{(k)} / \|X^{(k)}\|_{\infty} \\ AX^{(k+1)} = Y^{(k)} \end{cases}, \quad k = 0, 1, \dots$$

$X^{(0)}$ 可能恰好为 $\lambda = 1$ 的特征向量，因此要验证一下

7.3 实对称矩阵的 Jacobi 方法

Jacobi 方法就是实现以上思想的一种数值方法，它通过一系列平面旋转变换（也是正交变换），逐渐减弱非对角线元素。计算对称矩阵A全部特征值的Jacobi方法是构造一系列的正交矩阵 Q_k 的过程。

$$\text{记 } Q(p, q, \theta) = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & \cos \theta & & \sin \theta & \\ & & & 1 & & \\ & & & & \ddots & \\ & & -\sin \theta & & & 1 \\ & & & & \cos \theta & \\ & & & & & \ddots \\ & & & & & & 1 \end{pmatrix} \begin{matrix} \rightarrow p \text{ 行} \\ \\ \\ \rightarrow q \text{ 行} \end{matrix}$$

p 列 q 列

$Q(p, q, \theta)$ 是一个正交矩阵，称为 Givens 旋转变换。下面分析 Givens 变换作用到对称矩阵后正交相似的变换效果。

$$\text{记 } s = \frac{a_{qq} - a_{pp}}{2a_{pq}}, t = \tan \theta$$

$$t = \begin{cases} t^2 + 2st - 1 = 0 & \text{的按模较小根, } s \neq 0 \\ 1, & s = 0 \end{cases}$$

$$\text{可使 } b_{pq} = 0, \quad b_{qp} = 0,$$

$$\text{由 } t = \tan \theta, \text{ 可得 } \begin{cases} \cos \theta = \frac{1}{\sqrt{1+t^2}} \\ \sin \theta = \frac{t}{\sqrt{1+t^2}} \end{cases} \quad \text{记 } \begin{cases} c = \cos \theta \\ d = \sin \theta \end{cases}, \text{ 则当 } t \text{ 按上面描述值时,}$$

$$\begin{cases} b_{ip} = b_{pi} = c a_{pi} - d a_{qi}, & i \neq p, q \\ b_{iq} = b_{qi} = d a_{pi} + c a_{qi}, & i \neq p, q \\ b_{pp} = a_{pp} - t a_{pq} \\ b_{qq} = a_{pp} + t a_{pq} \\ b_{pq} = b_{qp} = 0 \\ b_{ij} = a_{ij} \quad i \neq p, q; \quad j \neq p, q \end{cases}$$

如果选取 p, q 使 $|a_{pq}| = \max_{i \neq j} |a_{ij}|$ ，那么实施以上变换的效率将更高。Jacobi 方法用于一般阶不很高的“满矩阵”情形。

例 用 Jacobi 方法计算矩阵 $A = \begin{pmatrix} 3 & 1 & 2 \\ 1 & 3 & 4 \\ 2 & 4 & 6 \end{pmatrix}$ 的全部特征值

解 取 $p = 1, q = 2$; $s = \frac{a_{22} - a_{11}}{2a_{12}} = \frac{3-3}{2} = 0$; $t = 1$; $c = \cos \theta = \sqrt{2}/2$, $d = \sin \theta = \sqrt{2}/2$

$$\begin{cases} a_{13}^{(1)} = a_{31}^{(1)} = a_{13}c - a_{23}d = -1.4142 \\ a_{23}^{(1)} = a_{32}^{(1)} = a_{23}c + a_{13}d = 4.2426 \\ a_{11}^{(1)} = a_{11} - ta_{12} = 2 \\ a_{22}^{(1)} = a_{22} + ta_{12} = 4 \\ a_{33}^{(1)} = a_{33} = 6 \\ a_{12}^{(1)} = a_{21}^{(1)} = 0 \end{cases}$$

$$\text{即 } A^{(1)} = Q^T A Q = \begin{pmatrix} c & -d & 0 \\ d & c & 0 \\ 0 & 0 & 1 \end{pmatrix} A \begin{pmatrix} c & d & 0 \\ -d & c & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 0 & -1.4142 \\ 0 & 4 & 4.2426 \\ -1.4142 & 4.2426 & 6 \end{pmatrix}$$

$$\text{取 } p=2, q=3; \quad s = \frac{a_{33} - a_{22}}{2a_{23}} = \frac{6-4}{2 \cdot 4.2426} = 0.2357;$$

$$t = -s + \sqrt{s^2 + 1} = 0.7917$$

$$c = 1/\sqrt{1+t^2} = 0.7840, d = t/\sqrt{1+t^2} = 0.6207$$

$$\text{即 } A^{(2)} = Q^T A^{(1)} Q = \begin{pmatrix} c & -d & 0 \\ d & c & 0 \\ 0 & 0 & 1 \end{pmatrix} A^{(1)} \begin{pmatrix} c & d & 0 \\ -d & c & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 0.8778 & -1.1088 \\ 0.8778 & 0.6411 & 0 \\ -1.1088 & 0 & 9.3589 \end{pmatrix}$$

第一步将 $p=1, q=2$ 位置的元素化为零，第二步将 $p=2, q=3$ 位置的元素化为零后 $p=1, q=2$ 位置的元素 $a_{12}^{(2)}$ 又变为非零元素了，但是 $|a_{12}^{(2)}|$ 比 $|a_{12}^{(0)}|$ 的数值小。继续做下去可求出 A 的特征值为：9.52，2.29，0.183。

7.4 QR 方法简介

第 8 章 常微分方程数值解

本章主要讨论常微分方程的初值问题

$$\begin{cases} y'(x) = f(x, y) \\ y(a) = y_0 \end{cases}, (a \leq x \leq b) \quad \text{或} \quad \begin{cases} \frac{dy}{dx} = f(x, y) \\ y(a) = y_0 \end{cases}, (a \leq x \leq b)$$

常微分方程初值问题的数值解是求 $y(x)$ 在求解区间 $[a, b]$ 上剖分点列 $x_n, n=1, 2, \dots, m$ 的数

值解 y_n 。在计算中约定 $y(x_n)$ 表示常微分方程准确解的值， y_n 表示 $y(x_n)$ 的近似值。

8.1 欧拉(Euler)公式

8.1.1 基于数值微商的欧拉公式

用向前差商近似 $y'(x)$

$y_{n+1} = y_n + hf(x_n, y_n)$ 向前欧拉公式是单步的显式格式。

用向后差商近似 $y'(x)$

向后欧拉公式 $y_{n+1} = y_n + hf(x_{n+1}, y_{n+1})$ 向后欧拉公式是单步的隐式格式。隐式欧拉公

式，需要通过迭代法求得 y_{n+1} 。其中，初始值 $y_{n+1}^{(0)}$ 由向前欧拉公式提供。

下列最简单的皮卡（Picard）迭代格式：

$$\begin{cases} y_{n+1}^{(0)} = y_n + hf(x_n, y_n) \\ y_{n+1}^{(k+1)} = y_n + hf(x_{n+1}, y_{n+1}^{(k)}) \end{cases},$$

用中心差商近似 $y'(x)$

由于中心差商不是稳定格式（后叙），因此，不予采用。

*8.1.2 欧拉公式的收敛性

局部截断误差

对 $y(x_{n+1})$ 在 x_n 做泰勒展开：

$$y(x_{n+1}) = y(x_n + h) = y(x_n) + hy'(x_n) + \frac{h^2}{2!} y''(\xi_n), \quad x_n \leq \xi_n \leq x_{n+1}$$

欧拉公式是由以上展开式中截断 $\frac{h^2}{2!} y''(\xi_n)$ 而得，故 $T_{n+1} = \frac{h^2}{2!} y''(\xi_n)$ 称为欧拉公式的截断误差或称局部截断误差。若 $y(x)$ 为线性函数，则 $y''(x) \equiv 0$ ，这时局部截断误差为 0，由欧拉公式

得到的解为精确解，故欧拉公式是一阶方法。如果给定方法的局部截断误差是： $T_{n+1} = O(h^{k+1})$ ，

则称方法是 k 阶的，或称具有 k 阶精度。

8.1.3 基于数值积分的近似公式

用梯形近似积分公式计算 $\int_{x_n}^{x_{n+1}} y'(t) dt$

$$\int_{x_n}^{x_{n+1}} y'(t) dt \approx \frac{1}{2} (x_{n+1} - x_n) (y'(x_{n+1}) + y'(x_n)) = \frac{h}{2} (f(x_n, y(x_n)) + f(x_{n+1}, y(x_{n+1})))$$

$$\text{得到梯形公式：} \quad y_{n+1} = y_n + \frac{h}{2} (f(x_n, y_n) + f(x_{n+1}, y_{n+1}))$$

$$y_{n+1}^{(k+1)} = y_n + \frac{h}{2} (f(x_n, y_n) + f(x_{n+1}, y_{n+1}^{(k)}))$$

梯形公式也是隐式格式，可用皮卡迭代或牛顿迭代格式计算 y_{n+1} 。

预估-校正公式：

$$\begin{cases} \bar{y}_{n+1} = y_n + hf(x_n, y_n) \\ y_{n+1} = y_n + \frac{h}{2} [f(x_n, y_n) + f(x_{n+1}, \bar{y}_{n+1})] \end{cases}$$

也称为改进的欧拉公式

8.2 龙格-库塔方法

8.2.1 二阶龙格-库塔方法

$$\begin{cases} y_{n+1} = y_n + hk_2 \\ k_1 = f(x_n, y_n) \\ k_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_1) \end{cases}$$

二阶龙格-库塔公式的局部截断误差仍为 $O(h^3)$ 。是二阶精度的计算公式。类似地想法，可建立高阶的龙格-库塔公式，同理可知四阶龙格-库塔公式的局部截断误差界为 $O(h^5)$ 的四阶精度计算公式。

8.2.2 四阶龙格-库塔格式

四阶龙格-库塔公式

$$\begin{cases} y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ k_1 = f(x_n, y_n) \\ k_2 = f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_1) \\ k_3 = f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_2) \\ k_4 = f(x_n + h, y_n + hk_3) \end{cases}$$

例 用四阶龙格-库塔公式 解初值问题 $\begin{cases} \frac{dy}{dx} = y^2 \cos x & 0.1 \leq x \leq 0.8 \\ y(0) = 1 \end{cases}$

解 取步长 $h = 0.2$ ，计算公式为

$$\begin{cases} y_{n+1} = y_n + \frac{0.2}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ k_1 = y_n^2 \cos x_n \\ k_2 = (y_n + 0.1k_1)^2 \cos(x_n + 0.1) \\ k_3 = (y_n + 0.1k_2)^2 \cos(x_n + 0.1) \\ k_4 = (y_n + 0.2k_3)^2 \cos(x_n + 0.2) \end{cases}$$

计算结果列表中。

n	x_n	y_n	$y(x_n)$	$ y_n - y(x_n) $
1	0.2	1.24789	1.24792	0.00003
2	0.4	1.63762	1.63778	0.00016
3	0.6	2.29618	2.29696	0.00078
4	0.8	3.53389	3.53802	0.00413

8.2.3 步长的自适应

8.3 线性多步法

线性多步法的三个基本要素：

(1)积分区间 $[x_{n-p}, x_{n+1}]$; (2)构造 $y'(t)$ 的 q 次插值多项式; (3) $y'(t)$ 是显式还是隐式 (若计算 y_{n+1} 时只需要 y_n , 则是显式; 若计算 y_{n+1} 还包括 y_{n+1} , 则是隐式)

$$\text{常微分方程初值问题} \begin{cases} \frac{dy}{dx} = f(x, y) \\ y(a) = y_0 \end{cases}, \quad (a \leq x \leq b)$$

$$y(x_{n+1}) = y(x_{n-p}) + \int_{x_{n-p}}^{x_{n+1}} y'(t) dt$$

若用积分节点 $x_n, x_{n-1}, \dots, x_{n-q}$ 构造插值多项式近似 $y'(t)$, 在区间 $[x_{n-p}, x_{n+1}]$ 上计算数值积分 $\int_{x_{n-p}}^{x_{n+1}} y'(t) dt$, 则称构造计算 y_{n+1} 的方法为线性多步法。这是显式公式。

[q 次显式公式需要的积分节点 $(x_n, y'_n)(x_{n-1}, y'_{n-1}) \dots (x_{n-q}, y'_{n-q})$]

若以 $x_{n+1}, x_n, x_{n-1}, \dots, x_{n+1-q}$ 积分节点构造插值多项式近似 $y'(x)$, 在区间 $[x_{n-p}, x_{n+1}]$ 上计算数值积分 $\int_{x_{n-p}}^{x_{n+1}} y'(t) dt$, 则为隐式公式。

[q 次隐式公式需要的积分节点 $(x_{n+1}, y'_{n+1})(x_n, y'_n) \dots (x_{n-q+1}, y'_{n-q+1})$]

与通常数值积分不同的是, 在线性多步法公式中, 有两个控制量 p 和 q , p 控制积分区间,

q 控制插值节点。

特别取 $p = 0$, 有 $y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} y'(x) dx$

若积分 $\int_{x_n}^{x_{n+1}} y'(x) dx$ 用关于积分节点 $x_n, x_{n-1}, \dots, x_{n-q}$ 的数值积分近似, 就可得到显式的阿达姆斯公式。在计算中阿达姆斯的稳定性较好。

例 构造 $p = 0, q = 1$, 的显式格式 $y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} [l_0(x)y'(x_n) + l_1(x)y'(x_{n-1}) + R(x)] dx$

$$y_{n+1} = y_n + \int_{x_n}^{x_{n+1}} [l_0(x)y'(x_n) + l_1(x)y'(x_{n-1})] dx = y_n + h(\alpha_0 f(x_n, y_n) + \alpha_1 f(x_{n-1}, y_{n-1}))$$

$$\alpha_0 h = \int_{x_n}^{x_{n+1}} l_0(x) dx = \int_{x_n}^{x_{n+1}} \frac{x - x_{n-1}}{x_n - x_{n-1}} dx = \frac{3}{2} h \quad \alpha_1 h = \int_{x_n}^{x_{n+1}} l_1(x) dx = \int_{x_n}^{x_{n+1}} \frac{x - x_n}{x_{n-1} - x_n} dx = -\frac{1}{2} h$$

$$T_{n+1} = \int_{x_n}^{x_{n+1}} R(x) dx = \int_{x_n}^{x_{n+1}} \frac{y^{(3)}(\eta)}{2} (x - x_n)(x - x_{n-1}) dx = \frac{5}{12} h^3 y^{(3)}(\xi)$$

即
$$y(x_{n+1}) = y(x_n) + \frac{h}{2}[3f(x_n, y(x_n)) - f(x_{n-1}, y(x_{n-1}))] + T_{n+1}$$

截断 T_{n+1} 可得到 $y(x_{n+1})$ 近似值 y_{n+1} 的计算公式: $y_{n+1} = y_n + \frac{h}{2}[3f(x_n, y_n) - f(x_{n-1}, y_{n-1})]$

上式称为二阶显式阿达姆斯公式。

例 建立 $p=1, q=2$ 显式公式

解
$$y_{n+1} = y_{n-1} + \int_{x_{n-1}}^{x_{n+1}} [l_0(x)f(x_n, y_n) + l_1(x)f(x_{n-1}, y_{n-1}) + l_2(x)f(x_{n-2}, y_{n-2})] dx$$

$$y_{n+1} = y_{n-1} + h[a_0f(x_n, y_n) + a_1f(x_{n-1}, y_{n-1}) + a_2f(x_{n-2}, y_{n-2})]$$

$$a_0h = \int_{x_{n-1}}^{x_{n+1}} \frac{(x-x_{n-1})(x-x_{n-2})}{(x_n-x_{n-1})(x_n-x_{n-2})} dx = \frac{7}{3}h$$

$$a_1h = \int_{x_{n-1}}^{x_{n+1}} \frac{(x-x_n)(x-x_{n-2})}{(x_{n-1}-x_n)(x_{n-1}-x_{n-2})} dx = -\frac{2}{3}h$$

$$a_2h = \int_{x_{n-1}}^{x_{n+1}} \frac{(x-x_n)(x-x_{n-1})}{(x_{n-2}-x_n)(x_{n-2}-x_{n-1})} dx = \frac{1}{3}h$$

计算格式:

$$y_{n+1} = y_{n-1} + \frac{h}{3}[7f(x_n, y_n) - 2f(x_{n-1}, y_{n-1}) + f(x_{n-2}, y_{n-2})]$$

其中: $n=2, 3, \dots, m-1$, $h = \frac{b-a}{m}$, 用 龙格-库塔公式计算 y_1, y_2 。

截断误差

$$T_{n+1} = \int_{x_{n-1}}^{x_{n+1}} \frac{f^{(3)}(\xi)}{3!} (x-x_{n-2})(x-x_{n-1})(x-x_n) dx$$

$$= \frac{1}{6} \int_{x_{n-1}}^{x_{n+1}} f^{(3)}(\xi) (x-x_{n-2})(x-x_{n-1})(x-x_n) dx = \frac{1}{3} h^4 y^{(4)}(\eta)$$

$f^{(3)}(x, y) = y^{(4)}(x)$

三步三阶的
显式格式

例 构造 $P=2, q=2$ 的隐式格式

解 取 $[x_{n-2}, x_{n+1}]$ 为积分区间;

以 $(x_{n+1}, f(x_{n+1}))$, $(x_n, f(x_n))$, $(x_{n-1}, f(x_{n-1}))$ 构造拉格拉日插值多项式。

$$y_{n+1} = y_{n-2} + h[\beta_0f(x_{n+1}, y_{n+1}) + \beta_1f(x_n, y_n) + \beta_2f(x_{n-1}, y_{n-1})]$$

则
$$\beta_0h = \int_{x_{n-2}}^{x_{n+1}} \frac{(x-x_n)(x-x_{n-1})}{(x_{n+1}-x_n)(x_{n+1}-x_{n-1})} dx = \frac{3}{4}h$$

$$\beta_1 h = \int_{x_{n-2}}^{x_{n+1}} \frac{(x - x_{n+1})(x - x_{n-1})}{(x_n - x_{n+1})(x_n - x_{n-1})} dx = 0$$

$$\beta_2 h = \int_{x_{n-2}}^{x_{n+1}} \frac{(x - x_{n+1})(x - x_n)}{(x_{n-1} - x_{n+1})(x_{n+1} - x_n)} dx = \frac{9}{4} h$$

计算格式: $y_{n+1} = y_{n-2} + \frac{h}{4} [3f(x_{n+1}, y_{n+1}) + 9f(x_{n-1}, y_{n-1})]$

截断误差: $T_{n+1} = \frac{1}{6} \int_{x_{n-2}}^{x_{n+1}} y^{(4)}(\xi)(x - x_{n+1})(x - x_n)(x - x_{n-1}) dx = -\frac{3}{8} h^4 y^{(4)}(\eta)$

为了避免迭代, 可用预估-校正公式

$$\begin{cases} \bar{y}_{n+1} = y_{n-2} + \frac{h}{3} [7f(x_n, y_n) - 2f(x_{n-1}, y_{n-1}) + f(x_{n-2}, y_{n-2})] \\ y_{n+1} = y_{n-2} + \frac{h}{4} [3f(x_{n+1}, \bar{y}_{n+1}) + 9f(x_{n-1}, y_{n-1})] \end{cases}$$

带入显式格
式求 \bar{y}_{n+1}

带入显式格
式求 y_{n+1}

替换上面的计算公式。

8.4 常微分方程组的数值解法

*8.5 常微分方程的稳定性