

Course: 数值计算王兵团

Date: 2020.8.15

Author: KafCoppelia

Chapter 1 绪论

1.1 计算机的数系及对数的处理

1.1.1 机器数系

计算机中使用的数系成为机器数系。

计算机中的实数（浮点数）

β 进制的浮点数：

$$x = \pm \beta^c \times 0.a_1a_2 \cdots a_t, a_i \in \{0, 1, \cdots, \beta - 1\}$$

机器数系

$$F(\beta, t, L, U) = \{\pm \beta^c \times 0.a_1a_2 \cdots a_t | a_k \in \{0, 1, \cdots, \beta - 1\}, L \leq c \leq U\}$$

t ：字长，正整数； β ：进制，一般取2、8、10、16；

c ：阶码，整数， $L \leq c \leq U$ ， L 与 U 固定整数。

机器数系特点

- 有限的离散集；
- 有绝对值最大非零数（M）和最小非零数（m）；
- 数绝对值大于M，产生上溢错误；小于m，产生下溢错误，统称溢出错误。
- 计算机把尾数为0且阶数最小的数表示数零。

1.1.2 计算机接受、处理数据的方法

计算对数接收

1、 $x \in F(\beta, t, L, U)$ ，原样接收 x ；

2、 $x \notin F(\beta, t, L, U)$ ， $m \leq |x| \leq M$ ，则用 $F(\beta, t, L, U)$ 中罪属于靠近 x 的数 $fl(x)$ （机器数）表示并记录 x 。

计算机对数的运算处理

1、加减法：对阶，后与暖，再舍入；

2、乘除法：先运算，再舍入。

1.2 误差

准确值与近似值的差异就是误差。

误差来源：

- 模型误差（也称描述误差）
- 观测误差（数据误差）
- 截断误差（方法误差）
- 舍入误差（计算误差）

定义1 绝对误差（简称误差）： $e(x^*) = x^* - x$ ， x 是准确值， x^* 是 x 的一个近似值。

定义2 相对误差： $e_r(x^*) = \frac{e^*}{x} = \frac{x^* - x}{x}$ ，相对误差绝对值越小，近似程度越高。

定义3 称满足 $|e^*| = |x^* - x| \leq \varepsilon^*$ 的正数 ε^* 为近似值的误差限。

定义4 满足 $|e_r^*| = \left| \frac{x^* - x}{x} \right| \leq \varepsilon_r^*$ 的正数 ε_r^* 称为相对误差限。

实用中, $\varepsilon_r^* = \frac{\varepsilon^*}{x^*}$

微分与误差的对应

$$e(x^*) = x^* - x = dx \quad e_r(x^*) = \frac{dx}{x} = d \ln x$$

绝对误差和相对误差与微分的关系

$$dx = e(x^*) \quad d \ln x = e_r(x^*)$$

四则运算的误差

$$1、e(x^* \pm y^*) = e(x^*) \pm e(y^*)$$

$$2、e(x^* \cdot y^*) \approx y^* e(x^*) + x^* e(y^*)$$

$$3、e\left(\frac{x^*}{y^*}\right) \approx \frac{y^* e(x^*) - x^* e(y^*)}{(y^*)^2}$$

函数的误差

$$\text{对于一元函数 } y = f(x), \text{ 有: } |e(f(x^*))| \approx |f'(x^*)| \cdot |e(x^*)|$$

表示 x^* 的误差经 f 作用后被放大(或缩小) $|f'(x^*)|$ 倍,称绝对误差增长因子。

1.3 有效数字

有效数字可以将误差量化。

1.3.1 有效数字定义

定义 若近似数 x^* 的误差限是其某一位上数字的半个单位,就说近似数 x^* 准确到该位;由该位自右向左数到 x^* 的第一个非零数字若有 n 位,就称近似数 x^* 有 n 位有效数字。

有效数字的数学描述 略

如果十进制 x 经四舍五入得到近似数 x^* ,则 x^* 的有效数字位为将 x^* 写为规格化浮点数后的尾数的位数。

e.g. 0.00345四舍五入得 0.35×10^{-2} 可知近似数有2位有效数字。

1.3.2 有效数字与误差的关系

定理 设十进制近似数 $x^* = \pm 0.a_1 a_2 \cdots a_k \times 10^m, a_1 \neq 0, k \geq n$

$$1、\text{若 } x^* \text{ 有 } n \text{ 位有效数字, 则有: } |e_r(x^*)| = \left| \frac{x^* - x}{x^*} \right| \leq \frac{1}{2a_1} \times 10^{1-n};$$

$$2、(\text{要求 } x^* \text{ 已知}) \text{ 若 } x^* \text{ 的相对误差 } |e_r(x^*)| = \left| \frac{x^* - x}{x^*} \right| \leq \frac{1}{2(a_1 + 1)} \times 10^{1-n}, \text{ 则 } x^* \text{ 有 } n \text{ 位有效数字。}$$

1.4 数值计算研究内容及常用概念

1.4.1 数值计算研究内容

1、连续系统的离散化

2、离散型方程的数值求解

1.4.2 数值分析的常用概念

1、数值问题

由一组已知数据(输入数据)求出一组结果数据(输出数据),使得这两组数据之间满足余弦指定的某种关系问题。

2、数值解

由近似公式计算出的解。

3、算法

由给定的已知量,经有限次四则运算及规定的运算顺序,求出所关心未知量的数值解。

数值分析本质上是研究和构造算法的。

4、计算量

一个算法所需要的乘法和除法总次数 N 。计算量单位flop，表示完成一次浮点数乘法或除法所需时间，可以衡量算法优劣。

e.g. $[A]_{10 \times 20}$ 、 $[B]_{20 \times 50}$ 、 $[C]_{50 \times 1}$ ，给出计算 $D = ABC$ 的两种算法及其对应计算量。

算法1： $D = (AB)C$ ， $N = 10 \times 20 \times 50 + 10 \times 50 \times 1 = 10500 \text{ flop}$

算法2： $D = A(BC)$ ， $N = 10 \times 20 \times 1 + 20 \times 50 \times 1 = 1200 \text{ flop}$

5、病态问题（蝴蝶效应）

初始数据的微小变化，导致计算结果的剧烈变化的问题。

初始数据的微小变化，只引起计算结果微小变化的计算问题称为良态问题。

6、数值稳定算法

如果算法的初始数据有误差，而在计算过程中产生的误差不增长，则为数值稳定算法。

1.5 科学计算需要注意

1、避免相近二数相减， $\therefore e_r(x - y) = \frac{e(x) - e(y)}{x - y}$

2、避免用接近零的数做除数， $\therefore e(\frac{x}{y}) = \frac{x}{y}[e_r(x) - e_r(y)]$

3、控制误差的积累和传播

方法：避免大数吃小数，用数值稳定的算法。

4、简化计算过程

方法：化简公式或减少运算次数。

e.g. $y = \frac{1}{\sqrt{x+1} - \sqrt{x}}$ ， $x \gg 1$

公式变形： $y = \sqrt{x+1} + \sqrt{x}$

e.g.2 设 n 次多项式 $P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_0$ ，试构建一个计算 $P_n(x)$ 的算法，使计算量尽可能少。

$P_n(x) = ((\cdots((a_n x + a_{n-1})x + a_{n-2})x + \cdots + a_1)x + a_0)$ ，每个括号中有同样的 $ax + b$ 结构，共 n 个括号，只需要 n 个计算量。

Chapter 2 非线性方程的求根方法

2.1 基本概念

2.1.1 方程与根

1、方程

- 函数方程： $f(x) = 0$ ， $f(x)$ 是 x 的连续函数
- 非线性方程： $f(x)$ 不是线性函数的函数方程
- 代数方程（ $n > 1$ ）： $a_n x^n + a_{n-1} x^{n-1} + \cdots + a_0 = 0$
- 超越方程： $f(x)$ 不是多项式函数的函数方程

2、方程的根

(1) 满足 $f(x) = 0$ 的值，称为方程的根、解、也称函数的零点。

(2) 单根、重根

非线性方程绝大部分没有求根公式，需寻找近似根的方法。

求根问题三要素：

- 根的存在性
- 根的范围
- 根的精确化（核心）

求根方法两类：区间法、迭代法；方法共同点：构造收敛根的数列 $\{x_k\}$ 。

2.1.2 收敛阶

非线性方程求根方法的关键是怎样构造收敛根的数列。收敛快慢是指标。借助 p 阶无穷小概念给出描述收敛速度收敛阶的概念。

定义 如果某求根数列 $x_k \rightarrow x^*$ ，且

$$\lim_{k \rightarrow \infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|^p} = C \quad p, C > 0$$

则称对应方法收敛阶为 p 。

收敛阶二要素：

- 1、 $x_k \rightarrow x^*$
- 2、 $\lim_{k \rightarrow \infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|^p} = C \quad p, C > 0$

收敛阶越大，收敛越快，方法越好。

常用的收敛阶名称：

- 线性收敛： $p = 1, C < 1$
- 平方收敛： $p = 2$
- 超线性收敛： $1 < p < 2$

2.2 二分法

也称对分区间法、对分法，是最简单的求根方法，属于区间法求根类型。

基本思想：利用连续函数零点定理，将含根区间主次减半缩小的方法构造点列来逼近跟。

2.2.1 二分法公式构造

理论依据：零点定理

设连续函数 $f(x)$ 在 $[a, b]$ 只有一个根，且 $f(a)f(b) < 0$

- 1、记 $I_0 = [a, b]$ ，取区间中点 x_0
- 2、判别 $f(x_0)$ 的值
 - 若等于0，则 $x^* = x_0$ ，停止
 - 若 $f(x_0)f(a) < 0$ ，记 $I_1 = [a, x_0]$ ，否则记 $I_1 = [x_0, b]$
- 3、再取区间中点
- 4、若满足根的精度要求，则停止否则继续。

2.2.2 二分法分析

1、二分法求根数列

记第 k 次二分含根区间 $I_k = [a_k, b_k]$ ，则求根数列算式为：

$$x_k = 0.5(a_k + b_k)$$

2、二分法的收敛性：收敛

3、二分法计算控制

$\therefore |x^* - x_k| \leq \frac{1}{2^{k+1}}(b - a)$ 对给定的计算精度 $\varepsilon > 0$ ，要成立 $|x^* - x_k| < \varepsilon$ ，即二分次数为：

$$k > \frac{\ln(b-a) - \ln \varepsilon}{\ln 2} - 1.$$

4、二分法的误差估计

- 事先估计： $|x^* - x_k| \leq \frac{1}{2^{k+1}}(b-a)$
- 事后估计： $|x^* - x_k| \leq |x_k - x_{k-1}|$

事先估计的 k 往往偏大，主要用于理论估计；

事后估计的 k 往往偏小，主要用于实际控制。

2.3 简单迭代法

2.3.1 介绍及构造原理

基本思想：利用对方程 $f(x) = 0$ 做等价变换根不发生变化的特点，将方程等价变形获得迭代计算公式。

公式构造：

将 $f(x) = 0$ 等价变换为 $x = \varphi(x)$ ，则有 $f(x^*) = 0 \Leftrightarrow x^* = \varphi(x^*)$ ，说明根 x^* 可由 $\varphi(x)$ 计算出来。

构造过程：

- 1、将 $f(x) = 0$ 改写成 $x = \varphi(x)$ 等价形式（不动点方程）；
- 2、构造迭代公式 $x_{k+1} = \varphi(x_k)$ ；
- 3、取一个初值 x_0 ，由上式算出数列 x_k ；

迭代函数： $\varphi(x)$

迭代数列： $\{x_k | x_{k+1} = \varphi(x_k)\}$

2.3.2 简单迭代法分析

1、几何意义

$x = \varphi(x)$ 的根是 $y = x$ 与 $y = \varphi(x)$ 交点横坐标。

2、收敛性：若收敛能收敛到根

2.3.3 重要定理

定理1 设迭代函数 $\varphi(x)$ 满足两个条件（充分条件）

- 1、 $\forall x \in [a, b] \Rightarrow \varphi(x) \in [a, b]$
- 2、 $\exists L \in (0, 1), \forall x_1, x_2 \in [a, b] \Rightarrow |\varphi(x_1) - \varphi(x_2)| \leq L|x_1 - x_2|$

则有

- 1、 $\varphi(x)$ 在 $[a, b]$ 中有唯一不动点 x^* ；
- 2、 $\forall x_0 \in [a, b] \Rightarrow x_{k+1} = \varphi(x_k)$ 产生的迭代数列 $x_k \rightarrow x^*$

证明略。

定理1的条件2不易操作，有如下推论

推论1 设迭代函数 $\varphi(x)$ 满足

- 1、 $\forall x \in [a, b] \Rightarrow \varphi(x) \in [a, b]$
- 2、 $|\varphi'(x)| \leq L < 1, x \in [a, b]$

也有定理1结论。

获得收敛迭代格式关键点：

- 1、选含有唯一根的区间 $[a, b]$ ；
- 2、选迭代函数 $\varphi(x)$ ；

- 3、检验条件1、2，要给出具体 L 值；
- 4、条件不成立时，重做1或2后再检验。

2.3.4 局部收敛定理

$|\varphi'(x)| \leq L < 1$ 在较小范围容易做到。

定理2 局部收敛定理

1、 x^* 是迭代函数 $\varphi(x)$ 的不动点： $x^* = \varphi(x^*)$

2、 $\varphi'(x)$ 在点 x^* 处连续

则有

- 1、 $|\varphi'(x^*)| < 1 \Rightarrow x_{k+1} = \varphi(x_k)$ 局部收敛于不动点 x^* ；
- 2、 $|\varphi'(x^*)| > 1$ ，且 $\varphi'(x)$ 无零点 $\Rightarrow x_{k+1} = \varphi(x_k)$ 对任何初值 $x_0 \neq x^*$ 都不收敛到 x^* 。

2.3.5 误差估计

定理3 在收敛定理的条件下，误差估计式：

1、 $|x^* - x_k| \leq \frac{L}{1-L}|x_k - x_{k-1}|$ ，事后估计

2、 $|x^* - x_k| \leq \frac{L^k}{1-L}|x_1 - x_0|$ ，事前估计

迭代计算控制：要 $|x^* - x_k| < \varepsilon$ ，只要

$$\frac{L^k}{1-L}|x_1 - x_0| < \varepsilon \Rightarrow k \geq \ln \left| \frac{(1-L)\varepsilon}{x - x_0} \right| / \ln L, \text{ 或 } \frac{L}{1-L}|x_k - x_{k-1}| < \varepsilon \Rightarrow |x_k - x_{k-1}| \leq \varepsilon.$$

2.3.6 高阶收敛定理

定理4 设 x^* 是迭代函数的不动点，有 $\varphi^{(m)}(x) \in C[x^*], m = 2, 3, \dots$ $\varphi'(x^*) = 0, \varphi^{(m-1)}(x^*) = 0, \varphi^{(m)}(x^*) \neq 0$ ，则有

1、 $x_{k+1} = \varphi(x)$ 是 m 阶收敛；

2、 $\lim_{k \rightarrow \infty} \frac{x_{k+1} - x^*}{(x_k - x^*)^m} = \frac{\varphi^{(m)}(x^*)}{m!}.$

推论

- 1、 $x^* = \varphi(x^*), \varphi'(x^*) = 0 \Rightarrow x_{k+1} = \varphi(x_k)$ 至少平方收敛；
- 2、 $x^* = \varphi(x^*), \varphi'(x^*) = 0, \varphi''(x^*) = 0 \Rightarrow x_{k+1} = \varphi(x_k)$ 至少立方收敛；
- 3、 $x^* = \varphi(x^*), \varphi^{(k)}(x^*) \neq 0 \Rightarrow x_{k+1} = \varphi(x_k)$ ，收敛阶 $\leq k$ 。

2.4 Newton迭代法

基本思想：将函数做线性化处理，把方程转化为对应的近似方程，再构造迭代函数。

2.4.1 公式构造

构造过程：

- 1、将函数 $f(x)$ 在 x_k 展开成泰勒级数；
- 2、取 $f(x)$ 关于 $x - x_k$ 的线性部分，将方程 $f(x) = 0$ 替换为近似方程 $L(x) = 0 \Rightarrow f(x_k) + f'(x_k)(x - x_k) = 0$ ；
- 3、求出近似方程的解，记为 x_{k+1} ，得迭代格式： $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}.$

几何意义：也称切线法。

2.4.2 重要定理

定理5 设 $f(x^*) = 0, f'(x^*) \neq 0, f(x) \in C^2[N(x^*)]$, 则Newton迭代格式 $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$ 至少是平方收敛。

线性化技术：将非线性函数线性化。

Newton迭代法的误差关系：
$$x^* - x_{k+1} = \left[-\frac{f''(\xi)}{2f'(x_k)} \right] (x^* - x_k)^2$$

定理6（收敛定理）设函数 $f(x)$ 在 $[a, b]$ 上有连续二阶导数，满足3个条件：

- 1、 $f(a)f(b) < 0$
- 2、 $f'(x) \neq 0, x \in [a, b]$
- 3、 $f''(x), x \in [a, b]$ 不变号 $\Rightarrow \forall x_0 \in [a, b]$, 只要 $f(x_0)f''(x_0) > 0$

则Newton迭代格式产生的数列一定收敛于 $[a, b]$ 上的唯一根。（充分条件）

2.5 Newton迭代法变形与推广

2.5.1 割线法

- 1、公式构造

用近似导数替代原公式中的导数可得：
$$x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} f(x_k)$$

割线法迭代公式是两点迭代公式，收敛阶为1.618。

- 2、几何意义：

2.5.2 推广

n 元非线性方程组。

Chapter 3 线性方程组的解法

3.1 基本概念

- 1、 n 元矩阵表示： $Ax = B$
- 2、解： x_1^*, \dots, x_n^*
- 3、解法：
 - 直接法：用计算公式直接求出
 - 迭代法：用迭代公式计算出满足精度要求近似解（逼近）

3.2 线性方程组的迭代解法

类似非线性方程求根的简单迭代法，Jacobi迭代法、Seidel迭代法及Sor法等。

基本思想：将线性方程组 $Ax = B$ 等价变形为 $x = Bx + g$ ，然后构造向量迭代公式：

$$x^{(k+1)} = Bx^{(k)} + g, k = 0, 1, 2, \dots$$

给定一个初始向量 $x^{(0)}$ ，带入迭代公式计算 $x^{(1)}, x^{(2)}, \dots$

3.2.1 迭代公式构造

- 1、Jacobi迭代公式构造

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases} \Leftrightarrow \begin{cases} x_1 = \frac{1}{a_{11}}(b_1 - a_{12}x_2 - a_{13}x_3 - \cdots - a_{1n}x_n) \\ x_2 = \frac{1}{a_{22}}(b_2 - a_{21}x_1 - a_{23}x_3 - \cdots - a_{2n}x_n) \\ \vdots \\ x_n = \frac{1}{a_{nn}}(b_n - a_{n1}x_1 - a_{n2}x_2 - \cdots - a_{nn-1}x_{n-1}) \end{cases}$$

写成Jacobi迭代格式：

$$\begin{cases} x_1^{(k+1)} = \frac{1}{a_{11}}(b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - \cdots - a_{1n}x_n^{(k)}) \\ x_2^{(k+1)} = \frac{1}{a_{22}}(b_2 - a_{21}x_1^{(k)} - a_{23}x_3^{(k)} - \cdots - a_{2n}x_n^{(k)}) \\ \cdots \\ x_n^{(k+1)} = \frac{1}{a_{nn}}(b_n - a_{n1}x_1^{(k)} - a_{n2}x_2^{(k)} - \cdots - a_{nn-1}x_{n-1}^{(k)}) \end{cases}$$

2、Seidel迭代公式构造 (Jacobi的改进，但不能取代)

$$\begin{cases} x_1^{(k+1)} = \frac{1}{a_{11}}(b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - \cdots - a_{1n}x_n^{(k)}) \\ x_2^{(k+1)} = \frac{1}{a_{22}}(b_2 - a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} - \cdots - a_{2n}x_n^{(k)}) \\ \cdots \\ x_n^{(k+1)} = \frac{1}{a_{nn}}(b_n - a_{n1}x_1^{(k+1)} - a_{n2}x_2^{(k+1)} - \cdots - a_{nn-1}x_{n-1}^{(k+1)}) \end{cases}$$

3、Sor迭代公式构造 (Seidel的改进，超松弛迭代法)

用Seidel迭代格式算出的 $x^{(k+1)}$ 记为 $\tilde{x}^{(k+1)}$ 得到增量： $\Delta x = \tilde{x}^{(k+1)} - x^{(k)}$ ，做增量加速处理 (ω 松弛因子)

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

3.2.2 向量迭代格式

1、Jacobi向量迭代格式

$$\text{记: } L = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ -a_{21} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -a_{n1} & -a_{n2} & \cdots & 0 \end{bmatrix}, U = \begin{bmatrix} 0 & -a_{12} & \cdots & -a_{1n} \\ 0 & 0 & \cdots & -a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}, D = \begin{bmatrix} a_{11} & & & \\ & a_{22} & & \\ & & \ddots & \\ & & & a_{nn} \end{bmatrix},$$

$$x^{(k)} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix}, b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

有：

$$x^{(k+1)} = D^{-1}(L + U)x^{(k)} + D^{-1}b \rightarrow x^{(k+1)} = B_J x^{(k)} + g_J, B_J \text{称为Jacobi迭代矩阵。}$$

2、Seidel向量迭代格式

$$x^{(k+1)} = D^{-1}(Lx^{(k+1)} + Ux^{(k)} + b) \Rightarrow x^{(k+1)} = (D - L)^{-1}Ux^{(k)} + (D - L)^{-1}b \rightarrow x^{(k+1)} = B_S x^{(k)} + g_S, B_S \text{称为Seidel迭代矩阵。}$$

3、Sor向量迭代格式

$$x^{(k+1)} = B_\omega x^{(k)} + g_\omega$$

$$B_\omega = (D - \omega L)^{-1}[(1 - \omega)D + \omega U], g_\omega = \omega(D - \omega L)^{-1}b$$

上面三种向量迭代格式可统一写成：

$$x^{(k+1)} = Bx^{(k)} + g, B \text{为迭代矩阵。}$$

3.2.3 范数

收敛的本质是距离趋于0，两个实数的距离是绝对值，向量的距离用范数描述。

1、范数定义

设 L 是数域 K 上的一个线性空间，如果定义在 L 上的实值函数 $P(x)$ 满足：

$$1) \forall x \in L \Rightarrow P(x) \geq 0, P(x) = 0 \Leftrightarrow x = 0;$$

$$2) \forall x \in L, \lambda \in K \Rightarrow P(\lambda x) = |\lambda|P(x);$$

$$3) \forall x, y \in L \Rightarrow P(x + y) \leq P(x) + P(y)$$

则称 P 是 L 的范数， $P(x)$ 为 x 的一个范数。记 $P(x) = \|x\|_p = \|x\|$ 则

$$1) \|x\| \geq 0 \text{ 且 } \|x\| = 0 \Leftrightarrow x = 0;$$

$$2) \|\lambda x\| = |\lambda| \cdot \|x\|;$$

$$3) \|x + y\| \leq \|x\| + \|y\|$$

2、数值分析中常用的线性空间

$$1) n\text{维向量空间 } R^n$$

$$2) \text{矩阵空间 } R^{m \times n}$$

$$3) \text{连续函数空间 } C[a, b] = \{f(x) | f(x) \text{ 在 } [a, b] \text{ 连续}\}$$

3、 $R^{n \times n}$ 中范数矩阵定义 略

3.2.4 数值分析中的范数

1、 R^n 中的向量范数

$$1) \|x\|_1 = \sum_{k=1}^n |x_k|$$

$$2) \|x\|_2 = \sqrt{\sum_{k=1}^n x_k^2}$$

$$3) \|x\|_\infty = \max_{1 \leq k \leq n} |x_k|$$

式中向量 $x = (x_1, \dots, x_n)^T$

2、矩阵算子范数

由范数相容性： $\|Ax\| \leq \|A\| \|x\|$,

矩阵 A 算子范数定义：

$$\|A\|_P = \max_{x \in R^n, x \neq 0} \frac{\|Ax\|_P}{\|x\|_P}, \text{算子范数} \Rightarrow \text{矩阵范数, 反向不可.}$$

3、 $R^{n \times n}$ 中的矩阵范数

$$1) \text{列范数: } \|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$$

$$2) \text{行范数: } \|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|$$

$$3) F\text{范数: } \|A\|_F = \sqrt{\sum_{i,j=1}^n a_{ij}^2}, \text{不是算子范数}$$

$$4) 2\text{范数: } \|A\|_2 = \sqrt{\lambda_{\max}}, \lambda_{\max} \text{ 是 } A^T A \text{ 的最大特征值}$$

4、范数等价与向量极限

$$1) \text{范数等价: } m\|x\|_p \leq \|x\|_q \leq M\|x\|_p, \forall x \in L$$

R^n 上所有范数等价。

$$2) \lim_{k \rightarrow \infty} x^{(k)} = x^* \Leftrightarrow \lim_{k \rightarrow \infty} \|x^{(k)} - x^*\| = 0$$

3.2.5 谱半径

1、定义： $A \in R^{n \times n}$, λ_k 是 A 的特征值， $k = 1, 2, \dots, n$ ，称 $\rho(A) = \max_{1 \leq k \leq n} |\lambda_k|$ 为矩阵 A 的谱半径。

2、谱半径与矩阵范数的关系

定理1 $\rho(A) \leq \|A\|$ ， $\|\cdot\|$ 是任意的矩阵范数。

3、一些关于谱半径的结论

$$1) \rho(A^k) = [\rho(A)]^k$$

$$2) \forall \varepsilon > 0, \text{ 存在一种矩阵范数使得 } \|A\| \leq \rho(A) + \varepsilon$$

$$3) \text{ 当矩阵 } A \text{ 对称时, } \rho(A) = \|A\|_2$$

$$4) \text{ 当矩阵 } A \text{ 为方阵时有 } \lim_{k \rightarrow \infty} A^k = 0 \Leftrightarrow \rho(A) < 1$$

3.2.6 迭代收敛定理

1、迭代收敛定理

定理2 $x^{(k+1)} = Bx^{(k)} + g$ 对任意 $x^{(0)}$ 都收敛 $\Leftrightarrow \rho(B) < 1$

2、迭代收敛判别条件

判别条件1（充分条件）若存在迭代矩阵 B 的某种矩阵范数满足 $\|B\| < 1$ ，则 $\forall x^{(0)} \in R^n, x^{(k+1)} = Bx^{(k)} + g$ 产生的迭代序列都收敛到其不动点 x^* 。

判别条件2 若 A 为严格对角占优矩阵，则线性方程组 $Ax = B$ 的Jacobi和Seidel迭代对任何初值 $x^{(0)}$ 都收敛。

$$\text{严格行对角占优矩阵 } A = (a_{ij})_{n \times n} : |a_{kk}| > \sum_{j=1, j \neq k}^n |a_{kj}|, \forall k$$

$$\text{严格列对角占优矩阵 } A = (a_{ij})_{n \times n} : |a_{kk}| > \sum_{i=1, i \neq k}^n |a_{ik}|, \forall k$$

定理：严格对角占优矩阵是非奇异的。

判别条件3 若 A 是正定矩阵，则 $Ax = B$ 的Seidel迭代对任何初值 $x^{(0)}$ 都收敛。

正定矩阵： A 的 k 阶主子式 $|A_k| > 0, k = 1, \dots, n$

3.2.7 收敛误差估计

定理4 设矩阵 B 的某种矩阵范数 $\|B\| < 1$ ，则有：

$$1、\|x^{(k)} - x^*\| \leq \frac{\|B\|}{1 - \|B\|} \|x^{(k)} - x^{(k-1)}\|$$

$$2、\|x^{(k)} - x^*\| \leq \frac{\|B\|^k}{1 - \|B\|} \|x^{(1)} - x^{(0)}\|$$

[比较非线性方程求根误差定理](#)

3.3 线性方程组的直接解法

有Gauss消元法（基础）、LU分解法等。

三角方程组可以回代求解，一般方程组可用消元的方式化为三角方程组求解。

基本思想 先将线性方程组通过消元方法化为同解的上三角方程组，然后从该三角方程组中按第 n 个方程、 $n-1$ 个方程、.....逐步回代求出解。

3.3.1 Gauss 消元法

1、Gauss消元法公式构造

$$\text{记原方程组为: } \begin{cases} a_{11}^{(0)}x_1 + a_{12}^{(0)}x_2 + \cdots + a_{1n}^{(0)}x_n = b_1^{(0)} \\ a_{21}^{(0)}x_1 + a_{22}^{(0)}x_2 + \cdots + a_{2n}^{(0)}x_n = b_2^{(0)} \\ \vdots \\ a_{n1}^{(0)}x_1 + a_{n2}^{(0)}x_2 + \cdots + a_{nn}^{(0)}x_n = b_n^{(0)} \end{cases}$$

(1) 消元

顺序做了n-1次消元后，变为上三角方程组：

$$\begin{cases} a_{11}^{(0)}x_1 + a_{12}^{(0)}x_2 + \cdots + a_{1n}^{(0)}x_n = b_1^{(0)} \\ a_{22}^{(1)}x_2 + \cdots + a_{2n}^{(1)}x_n = b_2^{(1)} \\ \vdots \\ a_{nn}^{(n-1)}x_n = b_n^{(n-1)} \end{cases}$$

(2) Gauss消元公式 略

3.3.2 Gauss 消元法分析

1、Gauss消元法计算量

消元步	1	2	...	n-1
乘法次数	$(n-1)n$	$(n-1)(n-2)$...	1×2
除法次数	$n-1$	$n-2$...	1

$$N_{\text{消}} = \sum_{k=1}^{n-1} k(k+1) + \sum_{k=1}^{n-1} k = \frac{n}{3}(n^2-1) + \frac{1}{2}n(n-1), N_{\text{回}} = \frac{1}{2}n(n+1)$$

$$\text{计算量: } N = \frac{n^3}{3} + n^2 - \frac{n}{3} \approx \frac{n^3}{3}.$$

2、Gauss消元法矩阵解释

是对 (A, b) 做初等行变换，每次对一个方程做一次消元时，对应的初等矩阵为：

$$M_{ik} = \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & \vdots & \ddots & & \\ & m_{ik} & \cdots & 1 & & \\ & & & & \ddots & \\ & & & & & 1 \end{bmatrix}$$

3、Gauss消元法可使用的条件

定理1 $a_{kk}^{(k-1)} \neq 0, \forall k \Leftrightarrow$ 矩阵 A 的所有顺序主子式 $\neq 0$

定理2 矩阵 A 的所有顺序主子式 $\neq 0$ ，则Gauss消元法可使用

4、Gauss消元法缺点

缺点1 要求 $a_{kk}^{(k-1)} \neq 0 (k = 1, 2, \dots, n)$

缺点2 在使用Gauss消元法进行计算机求解时，有时求出的解是错的。

3.3.3 主元消元法

列主元消元法、全主元消元法

比较：

Gauss消元法：计算时间最短，精度最差；

全主元法：计算时间最长，精度最高；

实践表明：列主元法具有良好的数值稳定，且计算量远低于全主元法。

3.3.4 LU 分解法

1、常用的矩阵三角分解

LU分解也成为了矩阵三角分解。常用的分解有三种：

1) Doolittle分解

$$A = LU \quad L = \begin{bmatrix} 1 & & & \\ l_{21} & 1 & & \\ \vdots & & \ddots & \\ l_{n1} & l_{n2} & \cdots & 1 \end{bmatrix}, U = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & & & u_{nn} \end{bmatrix}$$

2) Grout分解 略

3) LDU分解 略

$$A = LDU$$

2、基本思想

把一般线性方程组求解问题化为几个三角方程组来解。

$$Ax = b \leftrightarrow LUx = b$$

$$Ax = b \leftrightarrow LDUx = b$$

3、Doolittle分解法公式构造 略

4、Dlitle分解算法

$$1) \quad u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik}u_{kj}, i \leq j \quad l_{ij} = (a_{ij} - \sum_{k=1}^{j-1} l_{ik}u_{kj})/u_{jj}, i > j, \text{计算出 } L、U \text{ 矩阵}$$

2) 回代求解 $Ly = b$ ，解得 y^*

3) 回代求解 $Ux = y^*$ ，得 $Ax = b$ 的解 x

5、 A 可以进行Doolittle分解的条件

定理1 非奇异矩阵 A 的Doolittle分解是唯一的。

定理2 若 A 各阶顺序主子式不为0，则 A 有唯一的Doolittle分解。

能进行Gauss消元法就能做Doolittle分解。

6、Doolittle分解的紧凑格式 略

3.3.5 追赶法

是求三对角线性方程组专用的方法。

$$1、\text{三对角方程组} : \begin{cases} b_1x_1 + c_1x_2 & = d_1 \\ a_ix_{i-1} + b_ix_i + c_{i+1}x_{i+1} & = d_i \quad (i = 2, 3, \cdots, n-1) \\ a_nx_{n-1} + b_nx_n & = d_n \end{cases}$$

2、追赶法公式推导

3、追赶法求解公式

$$q_1 = b_1, y_1 = d_1$$

$$\begin{cases} p_k = a_k/q_{k-1} \\ q_k = b_k - p_k c_{k-1} \quad (k = 2, 3, \dots, n) \\ y_k = d_k - p_k y_{k-1} \end{cases}$$

$$x_n = y_n/q_n$$

$$x_k = (y_k - c_k x_{k+1})/q_k \quad (k = n-1, \dots, 1)$$

用追赶法来求解三对角线性方程组，计算量只有 $5n-4$ 。

3.3.6 特殊解法

特殊线性方程组：指系数矩阵有很多0元素或有特殊结构的方程组。

用LU分解法解该类方程组会产生较多无效的零元计算，损失计算效率。因此应构造能去掉其中无效、重复计算部分的内容的专用算法。

3.4 线性方程组解对系数的敏感性

概念描述：指由于系数的变化（扰动），导致所求解的变化情况。

3.4.1 系数敏感分析

设方程组 $Ax = b$ 的解为 x^* ，系数扰动后对应的线性方程组为

$$(A + \delta A)x = b + \delta b$$

又设扰动方程组的准确解为 $x^* + \delta x$ 有

$$Ax^* = b; \quad (A + \delta A)(x^* + \delta x) = b + \delta b$$

系数敏感对解的相对误差用 $\frac{\|\delta x\|}{\|x^*\|}$ 的大小来描述。

1) 考虑 $\delta A = 0$ ，即 $Ax = b + \delta b$

$$\text{有 } Ax^* = b, A(x^* + \delta x) = b + \delta b$$

(过程略)

$$\frac{\|\delta x\|}{\|x^*\|} \leq \|A\| \cdot \|A^{-1}\| \cdot \frac{\|\delta b\|}{\|b\|}$$

2) 考虑 $\delta b = 0$ ，即 $(A + \delta A)x = b$

$$\text{可得: } \frac{\|\delta x\|}{\|x^*\|} \leq \frac{\|A\| \cdot \|A^{-1}\| \cdot \frac{\|\delta A\|}{\|A\|}}{1 - \|A\| \cdot \|A^{-1}\| \cdot \frac{\|\delta A\|}{\|A\|}}$$

3) 一般情况 $(A + \delta A)x = b + \delta b$

$$\text{可得: } \frac{\|\delta x\|}{\|x^*\|} \leq \frac{\|A\| \cdot \|A^{-1}\|}{1 - \|A\| \cdot \|A^{-1}\| \cdot \frac{\|\delta A\|}{\|A\|}} \left(\frac{\|\delta A\|}{\|A\|} + \frac{\|\delta b\|}{\|b\|} \right)$$

矩阵 A 的条件数： $Cond_p(A) = \|A\|_p \cdot \|A^{-1}\|_p, p = 1, 2, \infty$

注：

- $Cond_p(A)$ 的值与矩阵范数有关，但相对大小一致。
- 条件数值越大，解对系数越敏感，方程组越病态。

条件数性质：

$$1、A \text{可逆} \Rightarrow Cond_p(A) \geq 1$$

$$2、A \text{可逆} \Rightarrow Cond_p(\alpha A) = Cond_p(A), \forall \alpha \in R, \alpha \neq 0$$

$$3、A \text{正交} \Rightarrow Cond_2(A) = 1$$

病态矩阵特点：

- 1、行列式某些行、列近似相关；
- 2、元素间相差大数量级，且无规则；
- 3、主元消去过程中出现小主元；
- 4、特征值相差大数量级。

设 \bar{x} 是线性方程组 $Ax = b$ 的近似解，称 $r = A\bar{x} - b$ 为方程的残向量（或残量）

当矩阵 A 非病态时，残量可刻画近似解的准确程度。

$$\frac{\|\bar{x} - x^*\|}{\|x^*\|} \leq \text{Cond}(A) \frac{\|r\|}{\|b\|}$$

Chapter 4 求矩阵特征值和特征向量的方法

4.1 问题的描述与基本概念

实用中在求解代数特征值问题时一般不用线性代数的方法，而采用迭代变换的计算机求解方法。

4.1.1 基本概念

- 1、矩阵 A 的特征多项式： $f_A(\lambda) = \det(A - \lambda I)$
- 2、矩阵 A 的特征多项式方程： $f_A(\lambda) = \det(A - \lambda I) = 0$
- 3、矩阵 A 的特征值： $f_A(\lambda) = 0$ 的根
- 4、特征值与特征向量的关系： $A\vec{x} = \lambda\vec{x}, \vec{x} \neq 0$
- 5、矩阵的特征值和特征向量知识

A 与 B 相似：若有可逆矩阵 P ，使得 $B = P^{-1}AP, A, B \in R^{n \times n}$

结论 若矩阵 A 相似矩阵 $B, B = P^{-1}AP$ ，则

- (1) A 与 B 的特征值完全相同；
- (2) 若 x 是 B 的特征向量，则 Px 为 A 的特征向量；
- (3) 相似矩阵具有相同的秩、迹、行列式和特征值。

6、正交矩阵知识

- 1) 正交矩阵：若 $A^T = A^{-1}$ ，即 $A^T A = I$ ，则 A 为正交矩阵。
- 2) 正交相似变换：若存在正交矩阵 P 满足 $P^T A P = B$ ，则称 B 与 A 正交相似。
- 3) A 是实对称矩阵， λ_n 是 A 的 n 个特征值，则存在正交矩阵 P 满足 $P^T A P = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ ， P 的各列为相应特征向量。

4.2 幂法

基本思想

利用矩阵的特征值与特征向量的关系构造迭代向量序列来求矩阵按模最大的特征值及其相应特征向量。

幂法作用

求矩阵按模最大的特征值及其相应特征向量。

4.2.1 幂法公式构造

幂法计算公式

设 n 阶矩阵 A 的 n 个特征值，满足 $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$ （即 λ_1 是按模最大的特征值），

$$\forall V^{(0)} \neq 0, V^{(k+1)} = AV^{(k)} \Rightarrow \lim_{k \rightarrow \infty} \frac{V_i^{(k+1)}}{V_i^{(k)}} = \lambda_1, \text{ 相应的特征向量为 } V^{(k)}.$$

说明

1、幂法的收敛速度取决于比值 $\frac{\lambda_2}{\lambda_1}$ ，比值越小，收敛越快。

2、 $|\lambda_1| > 1 \Rightarrow \lambda_1^k \rightarrow \infty \Rightarrow V^{(k)}$ 出现上溢错误，幂法失败。

4.2.2 规范化幂法

定理 设 n 阶矩阵 A 的 n 个特征值满足 $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$ ， λ_1 对应的特征向量为 $x^{(1)}$ ， $\forall V^{(0)} \neq 0, V^{(0)} \in R^n$ ，做计算格式：

$$u^{(0)} = V^{(0)}, \begin{cases} V^{(k)} = Au^{(k-1)} \\ m_k = \max V^{(k)} \quad k = 1, 2, \dots \Rightarrow \lim_{k \rightarrow \infty} u^{(k)} = \frac{x^{(1)}}{\max x^{(1)}} \\ u^{(k)} = V^{(k)} / m_k \quad \lim_{k \rightarrow \infty} m_k = \lambda_1 \end{cases}$$

式中 $\max \{V^{(k)}\}$ 表示 $V^{(k)}$ 绝对值最大的分量， $\{u^{(k)}\}$ 是规范化向量。

由此可得当 n 较大时， m_k 是矩阵 A 按模最大的近似特征值， $u^{(k)}$ 是对应的近似特征向量。

规范化幂法算法

1) 输入矩阵 A ， $V^{(0)}$ 和精度 ε ，使用中取 $V^{(0)} = \{1, 1, \dots, 1\}$

2) $k \leftarrow 1$

3) $V^{(k)} \leftarrow Au^{(k-1)}$

4) $m_k \leftarrow \max(V^{(k)})$, $m_{k-1} \leftarrow \max(V^{(k-1)})$

5) $u^{(k)} \leftarrow V^{(k)} / m_k$

6) 如果 $|m_k - m_{k-1}| < \varepsilon$ ，则输出 $u^{(k)}$, m_k ，停止

7) $k \leftarrow k + 1$, goto 3)

注意事项

1、当 λ_1 是 m 重根时，即

$\lambda_1 = \lambda_2 = \dots = \lambda_m$, $|\lambda_1| = \dots = |\lambda_m| > |\lambda_{m+1}| \geq \dots \geq |\lambda_n|$ ，结论也成立；

2、用幂法求出的特征值可能不是按模最大的特征值。

4.2.3 反幂法

反幂法是幂法的一个应用。他通过可逆矩阵与特征向量的关系借助幂法来求矩阵按模最小的特征值。

假设 $|\lambda_1| \geq \dots \geq |\lambda_2| > |\lambda_n| > 0 \Rightarrow A^{-1}$ 可逆，

由 $Ax^{(k)} = \lambda_k x^{(k)} \Rightarrow A^{-1}x^{(k)} = \lambda_k^{-1} x^{(k)}$

λ_k^{-1} 是 A^{-1} 的特征值， $x^{(k)}$ 是其对应的特征向量。

$|\lambda_1| \geq \dots \geq |\lambda_2| > |\lambda_n| > 0 \Rightarrow |\lambda_n^{-1}| > |\lambda_{n-1}^{-1}| \geq \dots \geq |\lambda_1^{-1}|$

对 A^{-1} 用幂法，可以求出 λ_n^{-1} ，继而求出按模最小的特征值 λ_n 。

反幂法算法

1) 输入矩阵 A ， $V^{(0)}$ 和精度 ε ，使用中取 $V^{(0)} = \{1, 1, \dots, 1\}$

2) $k \leftarrow 1$

3) $V^{(k)} \leftarrow A^{-1}u^{(k-1)}$ ； \rightarrow 求解方程组 $AV^{(k)} = u^{(k-1)}$ ，得出 $V^{(k)}$

4) $m_k \leftarrow \max(V^{(k)})$, $m_{k-1} \leftarrow \max(V^{(k-1)})$

5) $u^{(k)} \leftarrow V^{(k)} / m_k$

6) 如果 $|m_k - m_{k-1}| < \varepsilon$ ，则输出 $u^{(k)}$, m_k ，停止

7) $k \leftarrow k + 1$, goto 3)

4.3 Jacobi 方法和QR 方法

4.3.1 Jacobi 方法

求实对称矩阵的全部特征值和特征向量。

旋转矩阵：2阶旋转变换矩阵

$$\begin{pmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{pmatrix}$$

n 阶旋转变换矩阵 $J(i, j, \varphi)$ 的特点

1、是正交矩阵

2、可选 φ 对 A 做相似变换，使： $J^T(i, j, \varphi)AJ(i, j, \varphi) = A^{(1)} = (a_{ij}^{(1)})$

Jacobi 方法的理论依据

定理 设实对称矩阵 $A^{(0)} = A \in R^{n \times n}$, $\lambda_1, \lambda_2, \dots, \lambda_n$ 是对应的特征值； $J(i_k, j_k, \varphi_k) = J_k, k = 1, 2, \dots$ 是 n 解旋转矩阵序列，记

$$A^{(k)} = J_k^T A^{(k-1)} J_k, E(A^{(k)}) = \sum_{i \neq j} (a_{ij}^{(k)})^2$$

则有

1、 $E(A^{(k+1)}) \leq E(A^{(k)}), \lim_{k \rightarrow \infty} E(A^{(k)}) = 0$

2、 $\lim_{k \rightarrow \infty} A^{(k)} = \text{diag}(\lambda_1, \dots, \lambda_n)$

Jacobi精度较高，但是计算量大，对原矩阵是稀疏矩阵，旋转变换后不能保持其稀疏的性质。一般适用于阶数不高的矩阵。

4.3.2 QR 法

求中小型矩阵的全部特征值。

QR分解：将矩阵 A 分解为一个正交矩阵 Q 和一个上三角矩阵 R 相乘的形式。

QR 方法的构造

1、记 $A^{(1)} = A$

2、对 $A^{(1)}$ 做QR分解： $A^{(1)} = Q_1 R_1$ ，逆序相乘 $A^{(2)} = R_1 Q_1$

3、如果 $A^{(2)}$ 是对角线为 1×1 或 2×2 的上三角块矩阵，求出其特征值，停止

4、用 $A^{(2)}$ 代替 $A^{(1)}$ 转步骤2

逆序相乘的本质 $A^{(1)} = QR \Rightarrow R = Q^{-1}A^{(1)} = Q^T A^{(1)} \Rightarrow A^{(2)} = RQ = Q^T A^{(1)} Q$

说明矩阵序列是相似变换序列，有相同的特征值。

QR方法依据 略

Chapter 5 插值与拟合方法

5.1 问题的描述与基本概念

5.1.1 插值

1、插值函数定义

如果函数 $P(x)$ 满足： $P(x_k) = f(x_k), k = 0, 1, \dots, n$ 称 $P(x)$ 为 $f(x)$ 的一个插值函数， $f(x)$ 称为被插函数。

插值条件： $P(x_k) = f(x_k), k = 0, 1, \dots, n$

插值余项： $R(x) = f(x) - P(x)$

插值节点： $x_k, k = 0, 1, \dots, n$ ，互异

2、代数插值

如果插值函数 $P(x)$ 是多项式，称为代数插值。

$$P_m(x) = \sum_{k=0}^m a_k x^k (a_k \in R)$$

代数插值又称多项式插值，是最简单的插值函数。

如果 $P(x)$ 满足插值条件： $P(x_k) = y_k, k = 0, 1, \dots, n$

$$\begin{cases} a_0 + a_1 x_0 + \dots + a_n x_0^n = y_0 \\ a_0 + a_1 x_1 + \dots + a_n x_1^n = y_1 \\ \vdots \\ a_0 + a_1 x_n + \dots + a_n x_n^n = y_n \end{cases} \quad D = \begin{vmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n \\ 1 & x_1 & x_1^2 & \dots & x_1^n \\ & & \vdots & & \\ 1 & x_n & x_n^2 & \dots & x_n^n \end{vmatrix} = \prod_{0 \leq j < i \leq n} (x_i - x_j)$$

因为插值节点互异，有 $D \neq 0$ ，线性方程组有唯一解。

定理1 对 $n+1$ 个互异插值节点，存在一个满足插值条件的 n 次插值多项式。（但是，方程组是病态的）

定理2 对 $n+1$ 个互异插值节点，满足插值条件的 n 次插值多项式是唯一的。

3、插值名词

线性插值： $n = 1$ 的插值多项式

抛物线插值或Simpson插值： $n = 2$

内插：用 $P_n(x)$ 计算在 $[a, b]$ 内的函数值；

外插（推）：用 $P_n(x)$ 计算在 $[a, b]$ 外的函数值。

内插一般比外插精确。

5.1.2 拟合

拟合函数定义

如果函数 $\varphi(x)$ 满足：

$$\|\delta\| = \min, \delta = (\delta_0, \delta_1, \dots, \delta_n)^T \quad \delta_k = \varphi(x_k) - f(x_k) \text{ 称 } \varphi(x) \text{ 为 } f(x) \text{ 的一个拟合函数。}$$

$$\text{残差: } \delta_k = \varphi(x_k) - f(x_k)$$

$$\text{残差向量: } \delta = (\delta_0, \delta_1, \dots, \delta_n)^T$$

拟合点（可以不互异）： $x_k, k = 0, 1, \dots, n$

5.2 插值法

条件：已知： $y_k = f(x_k), k = 0, 1, \dots, n$

目的：构造一个 n 次插值多项式 $P_n(x)$

5.2.1 Lagrange 插值

基本思想

将待求的 n 次多项式插值函数改写成用已知函数值为系数的 $n+1$ 个待定 n 次多项式的线性组合形式，再利用插值条件和函数分解技术确定 $n+1$ 个待定 n 次多项式来求出插值多项式。

1、Lagrange 插值函数构造

$$\text{令 } n \text{ 次插值多项式为: } L_n(x) = y_0 l_{0n}(x) + \dots + y_n l_{nn}(x) = \sum_{i=0}^n y_i l_{in}(x)$$

式中 $\{l_{in}(x)\}$ 是与 $y_i = f(x_i)$ 无关的一些 n 次多项式。

n 次多项式函数的基： $l_{0n}(x), \dots, l_{nn}(x)$

为获得 $n+1$ 个 n 次多项式函数： $l_{0n}(x), \dots, l_{nn}(x)$ ，由插值条件，有：

$$L_n(x_k) = y_k l_{kn}(x_k) + \sum_{i=0, i \neq k}^n y_i l_{in}(x_k) = y_k, \forall k$$

由于 y_k 与 $l_{in}(x)$ 无关, 可得: $l_{in}(x_k) = \delta_{ik} = \begin{cases} 1 & i = k \\ 0 & i \neq k \end{cases} (i, k = 0, 1, \dots, n)$

$$l_{in}(x) = \prod_{k=0, k \neq i}^n \left(\frac{x - x_k}{x_i - x_k} \right), i = 0, 1, \dots, n, \text{ Lagrange插值基函数}$$

代入开始设定的函数

$$L_n(x) = \sum_{i=0}^n y_i \prod_{k=0, k \neq i}^n \left(\frac{x - x_k}{x_i - x_k} \right)$$

2、Lagrange 插值基函数

1) 有几个插值节点就有几个基函数;

2) $l_{in}(x)$ 有 n 个乘积因子 $\left(\frac{x - x_k}{x_i - x_k} \right), i = 0, 1, \dots, n \quad i \neq k$

3) $l_{in}(x)$ 的每个因子项的分母为对应的节点 x_i 减去其余所有节点, 分子结构同分母, 但要把被减数 x_i 换为 x 。

5.2.2 插值余项定理

1、插值余项定理

设函数 $f(x)$ 在 $[a, b]$ 上有 $n + 1$ 阶导数, 则有

$$R_n(x) = f(x) - P_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \omega_{n+1}(x), \forall x \in [a, b]$$

$$\xi \in [a, b], \quad \omega_{n+1}(x) = (x - x_0)(x - x_1) \cdots (x - x_n) = \prod_{k=0}^n (x - x_k)$$

x_k 是 (a, b) 上的插值节点, $P_n(x)$ 是 $f(x)$ 的 n 次插值多项式

2、Lagrange 插值余项分析

$$1) |R_n(x)| \leq \frac{M_{n+1}}{(n+1)!} |\omega_{n+1}(x)|, x \in [a, b] \quad M_{n+1} = \max_{a \leq x \leq b} |f^{(n+1)}(x)|$$

$$2) |R_n(x)| \leq \frac{M_{n+1}}{(n+1)!} T_{n+1}, x \in [a, b] \quad T_{n+1} = \max_{a \leq x \leq b} |\omega_{n+1}(x)|$$

3)

$$f(x) = \sum_{k=0}^n f(x_k) l_{kn}(x) + \frac{f^{(n+1)}(\xi)}{(n+1)!} \omega_{n+1}(x) \quad \because L_n(x) = P_n(x)$$

只要 $f(x)$ 有 $n + 1$ 阶导数。

5.2.3 Newton 插值

基本思想: 将待求的 n 次多项式插值函数用新的基函数表示, 再利用插值条件求出 $n+1$ 个待定多项式系数来求出 n 次插值多项式。

1、Newton 插值函数构造

差商 记 $f[x] = f(x)$, $f[x]$ 称为零阶差商。

$$\text{一阶差商: } f[x_i, x_j] = \frac{f(x_i) - f(x_j)}{x_i - x_j} = \frac{f[x_i] - f[x_j]}{x_i - x_j}$$

$$\text{二阶差商: } f[\overline{x_i, x_j, x_k}] = \frac{f[x_j, x_k] - f[x_i, x_j]}{x_k - x_i}$$

$$k\text{阶差商: } f[\overline{x_{i0}, x_{i1}, \dots, x_{ik-1}, x_{ik}}] = \frac{f[\overline{x_{i1}, x_{i2}, \dots, x_{ik}}] - f[\overline{x_{i0}, x_{i1}, \dots, x_{ik-1}}]}{x_{ik} - x_{i0}}$$

借助差商有: $a_k = f[x_0, x_1, \dots, x_k], k = 0, 1, \dots, n$

n 次Newton插值多项式: (承袭性)

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

5.2.4 Newton 插值余项

1、定理 满足插值条件的 n 次Newton插值多项式的余项为：

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

利用插值的唯一性，有

$$N_n(x) = L_n(x) \Rightarrow f(x) - N_n(x) = f(x) - L_n(x)$$

$$\text{可得到差商与微商的关系：} f[x_0, \cdots, x_n] = \frac{f^{(n)}(\xi)}{n!}$$

2、Newton插值余项分析

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

$$1) |R_n(x)| \leq \frac{M_{n+1}}{(n+1)!} |\omega_{n+1}(x)|, x \in [a, b] \quad M_{n+1} = \max_{a \leq x \leq b} |f[x_0, \cdots, x_n, x]|$$

$$2) f(x) = f(x_0) + f[x_0, x](x - x_0) \quad \because f(x_1) = f(x_0) + f[x_0, x_1](x_1 - x_0), x \leftrightarrow x_1$$

5.2.5 Hermite 插值

基本思想：仿照Lagrange插值函数的构造方法，用设定函数形式，在利用插值条件求出插值函数。（Lagrange的推广）

Hermite 插值函数构造

$$P(x) \text{ 满足在已知函数信息处相等：} P(x_k) = y_k; P'(x_k) = y'_k, k = 0, 1, \cdots, n$$

$$\text{构造 } 2n + 1 \text{ 次Hermite插值多项式函数：} H_{2n+1}(x) = \sum_{i=0}^n y_i l_{in}(x) + \sum_{i=0}^n y'_i h_{in}(x)$$

5.2.6 分段插值

基本思想：将被插函数的插值节点由小到大排序，以每对相邻的两个节点为端点获得若干个小区间，然后在每个小区间上都用 m 次插值多项式作为近似函数。

分段插值是分段函数。

分段插值提法：

设 $[a, b]$ 上的 $n+1$ 个点 $x_k: a = x_0 < x_1 < \cdots < x_n = b$ 和 $f(x_k)$ ，若有函数 $\varphi(x)$ 满足：

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

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

$$3) \varphi(x) \text{ 在每个小区间 } [x_k, x_{k+1}] \text{ 上是 } m \text{ 次多项式}$$

则称 $\varphi(x)$ 是 $[a, b]$ 上的分段 m 次插值多项式。

特别地，当 $m = 1$ 时称为分段线性插值。

1、分段插值函数的构造 略

2、分段插值的误差

定理 假设出现在如下不等式中的函数的高阶导数存在，则有：

1、分段线性插值误差估计为

$$|R_1(x)| = |f(x) - \varphi(x)| \leq \frac{h^2}{8} \max_{a \leq x \leq b} |f''(x)| \quad x \in [a, b]$$

2、分段三次Hermite插值函数误差估计为

$$|R_3(x)| = |f(x) - \varphi(x)| \leq \frac{h^4}{4! \times 2^4} \max_{a \leq x \leq b} |f^{(4)}(x)|, x \in [a, b] \quad h = \max_{0 \leq k \leq n-1} (x_{k+1} - x_k)$$

5.2.7 三次样条插值简介

1、三次样条插值函数定义

设 $S(x)$ 你故意在 $[a, b]$ 上, $[a, b]$ 的一个划分 Δ 为 $a = x_0 < x_1 < \cdots < x_n = b$, 若函数 $S(x)$ 满足

- 1) $S(x) \in C^2[a, b]$
- 2) $S(x)$ 每个小区间 $[x_k, x_{k+1}]$ 上是3次多项式

则称 $S(x)$ 是 $[a, b]$ 上的一个三次样条函数。

若同时还满足: 3) $S(x_k) = f(x_k)$

则个 $S(x)$ 是 $[a, b]$ 上的三次样条插值函数。

2、基本思想

利用三次样条插值函数定义的条件及在内节点的函数和导数的连续性质, 建立关系式来确定样条插值函数。(是分段插值函数)

3、三次样条插值函数需要额外两个边界条件

除需要 $n+1$ 个插值数据外, 还需要额外两个插值数据, 这是因为:

- 1) 三次样条插值函数在每个小区间都是三次多项式, 故在每个小区间上有4个待定系数;
- 2) 共有 n 个小区间, 有 $4n$ 个待定系数要确定;
- 3) 在 $n-1$ 个内节点有直到二阶的连续导数, 可得 $3(n-1)$ 个条件;
- 4) 由插值条件可有 $n+1$ 个条件;

故共有 $4n-2$ 个条件, 要想唯一确定三次样条插值函数, 应另加两个条件。

4、常用的确定三次样条插值函数的边界条件

1) 已知 $f''(x_0), f''(x_n)$, 令 $S''(x_0) = f''(x_0), S''(x_n) = f''(x_n)$

$f''(x_0) = f''(x_n) = 0$ 称为自然边界条件。

2) 已知 $f'(x_0), f'(x_n)$, 令 $S'(x_0) = f'(x_0), S'(x_n) = f'(x_n)$

3) 已知 $S(x_0) = S(x_n), S'(x_0) = S'(x_n), S''(x_0) = S''(x_n)$

第3类边界条件称为周期性边界条件, 周期函数的样条插值。

5、三次样条插值函数的结论

1) $\|f - S\| \leq \frac{5h^4}{384} \|f^{(4)}\|$

2) $\|f' - S'\| \leq \frac{h^3}{24} \|f^{(4)}\|$

3) $\|f'' - S''\| \leq \frac{h^2}{8} \|f^{(4)}\|, h = \max_{0 \leq k \leq n-1} h_k$

5.3 曲线拟合法

5.3.1 曲线拟合概念

1、曲线拟合函数 $\varphi(x)$ 形式

选形如 $\varphi(x) = \sum_{k=0}^m a_k \varphi_k(x), a_k \in R, \varphi_k(x)$ 已知, 满足:

$\|\delta\| = \min, \delta_k = \varphi(x_k) - f(x_k)$, 曲线拟合通常指向量范数为2的情况。

2、概念

拟合函数类: $M = \text{Span}\{\varphi_0, \varphi_1, \cdots, \varphi_m\} = \{\varphi(x) | \varphi(x) = \sum_{k=0}^m a_k \varphi_k(x), a_k \in R\}$

基函数： $\varphi_0(x), \dots, \varphi_m(x)$ ，都是已知函数。

m 次多项式函数类： $\varphi_k(x) = x^k \quad (k = 0, 1, \dots, m)$

函数线性无关与相关： $k_1\varphi_1(x) + k_2\varphi_2(x) + \dots + k_s\varphi_s(x) \equiv 0$

3、基本思想

根据给定的数据点集，选择合适的拟合函数类M，对范数为2的情况，利用最小二乘法确定具体的拟合函数。

由离散数据按 $\|\delta\|_2 = \min$ 获得的曲线拟合函数称为经验公式。

5.3.2 曲线拟合函数构造

设拟合函数： $\varphi(x) \in M = \text{Span}\{\varphi_0, \varphi_1, \dots, \varphi_m\} \Rightarrow \varphi(x) = \sum_{k=0}^m a_k \varphi_k(x), a_k \in R$

为确定系数 a_0, a_1, \dots, a_m

$$\|\delta\|_2 = \min \Rightarrow \|\delta\|_2^2 = \sum_{k=0}^n (f(x_k) - \varphi(x_k))^2$$

考虑加权残差和：

$$S(a_0, \dots, a_m) = \sum_{k=0}^n \omega_k (f(x_k) - \varphi(x_k))^2, \quad \omega_k (k = 0, 1, \dots, n) \text{已知，使其最小。}$$

$$\text{可得} \sum_{i=0}^m a_i \sum_{k=0}^n \omega_k \varphi_i(x_k) \varphi_j(x_k) = \sum_{k=0}^n \omega_k f(x_k) \varphi_j(x_k)$$

当 $\varphi_0(x), \dots, \varphi_m(x)$ 线性无关时，法方程组正定有唯一解。可解出 a_0^*, \dots, a_m^*

所求的曲线拟合函数为：

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

求曲线拟合的算法

- 1) 画出 $(x_k, f(x_k))$ 的散点图
- 2) 选择合适的拟合函数类M
- 3) 构造对应的法方程组，求解得拟合函数。

5.3.3 曲线拟合函数分析

1、当 $\varphi_k(x) = x^k$ 时 $\Rightarrow \varphi(x) = \sum_{k=0}^m a_k x^k$

当 $\varphi(x)$ 是 m 次多项式时，称为 m 次拟合曲线。

线性拟合： $m = 1$ 的法方程组

二次拟合： $m = 2$ 的法方程组

2、可求解的非线性拟合类型

1) 指数曲线拟合

$$\varphi(x) = ae^{bx}, \text{取对数} \ln \varphi(x) = \ln a + bx$$

$$\text{令} P(x) = \ln \varphi(x), A = \ln a \Rightarrow P(x) = A + bx.$$

2) 分式曲线拟合

$$\varphi(x) = \frac{x}{ax+b}, \text{取倒数} \frac{1}{\varphi(x)} = a + b\frac{1}{x}$$

$$\text{令} P(x) = \frac{1}{\varphi(x)} \Rightarrow P(x) = a + b\frac{1}{x}$$

5.3.4 最佳平方逼近

是曲线拟合法的推广。

$$\sum_{k=0}^n \omega_k (f(x_k) - \varphi(x_k))^2 = \min \quad \int_a^b \rho(x) (f(x) - \varphi(x))^2 dx = \min$$

解 $\varphi(x)$ 称为 $f(x)$ 的最佳平方逼近函数。

Chapter 6 数值积分与数值微分方法

6.1 基本概念

6.1.1 求积公式

1、定积分离散化过程

1) 去掉极限号

2) 将其中的 ξ_i 取为 x_i

3) 减少离散化的误差, 做 $\Delta x_i \rightarrow A_i$ (待定系数)

得到计算机求定积分的近似公式: $\int_a^b f(x) dx \approx \sum_{i=1}^n A_i f(x_i)$

2、求积公式

若存在一组实数 x_1, x_2, \dots, x_n (求积节点); A_1, A_2, \dots, A_n (求积系数) 成立

$$\int_a^b f(x) dx \approx \sum_{i=1}^n A_i f(x_i), \forall f(x) \in C[a, b]$$

称如上公式为一个数值求积公式。

$$\text{求积余项: } R(f) = \int_a^b f(x) dx - \sum_{i=1}^n A_i f(x_i)$$

6.1.2 代数精度

1、代数精度定义

设 $P_m(x)$ 是 m 次多项式, 如果 $\forall P_m(x) \Rightarrow R(P_m) = 0$, 但存在一个 $P_{m+1}(x) \Rightarrow R(P_{m+1}) \neq 0$

则称该求积公式的代数精度为 m 。

结论 一个求积公式的代数精度越大, 则求积公式越好。

2、求代数精度的方法

对求积公式依次取 $f(x) = 1, x, x^1, \dots, x^k$ 带入余项公式, 若首次出现 $R(x^k) \neq 0$ 的 k 为 m , 则对应的代数精度为 $m - 1$ 。

6.2 插值型求积公式

基本思想: 利用被积函数的插值函数代替被积函数来构造求积公式。

6.2.1 插值型求积公式构造

$$\int_a^b f(x) dx \approx \sum_{i=1}^n A_i f(x_i) \quad A_i = \int_a^b l_{in-1}(x) dx$$

$$\text{关键: } A_i = \int_a^b l_{in-1}(x) dx \quad l_{in-1}(x) = \prod_{k=1, k \neq i}^n \left(\frac{x - x_k}{x_i - x_k} \right)$$

$$\text{插值型求积公式的求积余项: } R(f) = \int_a^b \frac{f^{(n)}(\xi)}{n!} \omega_n(x) dx$$

由求积余项可得: 插值型求积公式的代数精度至少为求积节点个数减1。

定理1 插值型求积公式的代数精度至少为 $n - 1$ 。

推论：插值型求积公式求积系数之和为 $b - a$ ，即： $\sum_{k=1}^n A_k = b - a$ 。

6.2.2 Newton-Cotes 求积公式

在插值型公式中，把求积节点取为等距节点，令：

$$x_i = a + (i - 1)h, h = \frac{b - a}{n - 1}, i = 1, 2, \dots, n$$

$$\text{则有：} A_i = \frac{b - a}{n - 1} \int_0^{n-1} \left(\prod_{k=1, k \neq i}^n \frac{t - k + 1}{i - k} \right) dt \Rightarrow A_i = (b - a) C_i^{(n)} \quad (\text{Cotes系数})$$

n 点的Newton-Cotes公式（等距节点求积公式）

$$\int_a^b f(x) dx \approx (b - a) \sum_{i=1}^n C_i^{(n)} f(x_i), \quad C_i^{(n)} = \frac{1}{n - 1} \int_0^{n-1} \left(\prod_{k=1, k \neq i}^n \frac{t - k + 1}{i - k} \right) dt \quad 1、\text{常用的N-C公式}$$

$$\text{梯形公式 } (n = 2) : \int_a^b f(x) dx \approx \frac{b - a}{2} (f(a) + f(b))$$

$$\text{Simpson公式或抛物线公式 } (n = 3) : \int_a^b f(x) dx \approx \frac{b - a}{6} \left(f(a) + 4f\left(\frac{a + b}{2}\right) + f(b) \right)$$

2、 n 点N-C公式的代数精度

定理 代数精度至少为 $n - 1$ 。当节点个数 n 为奇数时，对应的N-C求积公式的代数精度至少为 n 。

6.2.3 N-C 求积公式余项及稳定性

1、广义积分中值定理

$$f(x), g(x) \in C[a, b]; g(x) \text{在} [a, b] \text{上不变号}, \int_a^b f(x)g(x)dx = f(\xi) \int_a^b g(x)dx, \xi \in [a, b]$$

2、梯形公式余项

$$R_1(f) = \int_a^b \frac{f''(\xi)}{2!} (x - a)(x - b)dx = -\frac{(b - a)^3}{12} f''(\eta), \eta \in [a, b]$$

3、Simpson公式的求积余项（不能使用广义积分中值定理化简）

$$R_2(f) = \int_a^b \frac{f^{(4)}(\xi)}{4!} (x - a)(x - \frac{a + b}{2})(x - b)dx = -\frac{f^{(4)}(\eta)}{2880} (b - a)^5, \eta \in [a, b]$$

4、N-C公式的稳定性

记 $\varepsilon_k = f^*(x_k) - f(x_k)$ ，有

$$\eta_n = (b - a) \sum_{i=1}^n C_i^{(n)} \varepsilon_i$$

$$|\eta_n| \leq (b - a) \sum_{i=1}^n |C_i^{(n)}| |\varepsilon_i| \leq (b - a) \varepsilon \sum_{i=1}^n |C_i^{(n)}|, \varepsilon = \max_{1 \leq i \leq n} |\varepsilon_i|$$

$$\text{由Cotes系数表：} n \leq 8 \Rightarrow \sum_{i=1}^n |C_i^{(n)}| = \sum_{i=1}^n C_i^{(n)} = 1 \quad n > 8 \Rightarrow \sum_{i=1}^n |C_i^{(n)}| > 1$$

$n \leq 8$ 时，数值稳定； $n > 8$ 时，数值不稳定。

6.2.4 求积公式的重要定理

代数精度最大能是多少？

基本思想：在求积公式中让求积节点和求积系数都设定为待定参数，然后构造具有最高代数精度的求积公式。

定理 求积公式的代数精度最大为 $2n - 1$ 。

定义 如果 n 次多项式 $\omega_n(x)$ 满足 $\int_a^b \rho(x) P_m(x) \omega_n(x) dx = 0, m = 0, 1, \dots, n - 1$ ， $P_m(x)$ 是 m 次多项式，则称 $\omega_n(x)$ 为 $[a, b]$ 上关于权 $\rho(x)$ 的 n 次正交多项式。

正交多项式理论有 n 次正交多项式是存在唯一的，且它的 n 次根都在 $[a, b]$ 内。于是选择 n 次正交多项式的 n 个零点作为求积节点构造的插值型求积公式就具有 $2n - 1$ 次代数精度。

推论 n 点插值型求积公式的代数精度至少是 $n - 1$ ，至多为 $2n - 1$ 。

求 n 次正交多项式的零点方法：

$$\omega_n(x) = (x - x_1)(x - x_2) \cdots (x - x_n)$$

$$\int_a^b \rho(x) x^m \omega_n(x) dx = 0, m = 0, 1, \cdots, n - 1 \Rightarrow x_1, x_2, \cdots, x_n$$

6.2.5 Gauss 求积公式

1、定义

若 n 点的求积公式具有 $2n - 1$ 次代数精度，则称该求积公式为**Gauss**求积公式，对应的求积节点和求积系数分别称为**Gauss**点和**Gauss**系数。

Gauss求积公式的余项用如下公式获得：

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

2、Gauss求积公式的数值稳定性

设 $l_{in-1}(x)$ 是Gauss求积公式的插值基函数，取 $f(x) = l_{in-1}^2(x)$ ，

$$\because \int_a^b \rho(x) f(x) dx \approx \sum_{k=1}^n A_k f(x_k) \text{ 代数精度是 } 2n - 1, 0 < \int_a^b \rho(x) l_{in-1}^2(x) dx = \sum_{k=1}^n l_{in-1}^2(x_k) = A_i, \forall i$$

结论 Gauss系数都是大于0的数。

$$f(x) \equiv 1 \Rightarrow \sum_{k=1}^n A_k = \int_a^b \rho(x) dx$$

类似N-C稳定性处理方法，有舍入误差

$$|\eta_n| = \left| \sum_{k=1}^n A_k \varepsilon_k \right| \leq \sum_{k=1}^n |A_k| |\varepsilon_k| = \sum_{k=1}^n A_k |\varepsilon_k| \leq \varepsilon \sum_{k=1}^n A_k = \varepsilon \int_a^b \rho(x) dx$$

所以Gauss型求积公式是稳定的。

Gauss公式中，不同的权函数和不同积分区间，对应不同形式的Gauss公式。

3、常用Gauss公式

1) Gauss-Legendre 求积公式

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

$$\int_a^b f(x) dx = \frac{b-a}{2} \int_{-1}^1 f\left(\frac{a+b}{2} + \frac{b-a}{2}t\right) dt$$

$$\text{Legendre正交多项式: } L_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} [(x^2 - 1)^n]$$

2) Gauss-Chebyshev 求积公式

$$\int_{-1}^1 \frac{f(x)}{\sqrt{1-x^2}} dx \approx \sum_{k=1}^n A_k f(x_k)$$

$$\text{Chebyshev正交多项式: } T_n(x) = \cos(n \arccos x) \quad x_k = \cos \frac{(2k-1)\pi}{2n}, A_k = \frac{\pi}{n}, k = 1, 2, \cdots, n$$

3) Gauss-Laguerre求积公式

$$\int_0^{+\infty} e^{-x} f(x) dx \approx \sum_{k=1}^n A_k f(x_k)$$

$$\text{Laguerre正交多项式: } \bar{L}_n(x) = e^x \frac{d^n}{dx^n} (x^n e^{-x})$$

4) Gauss-Hermite求积公式

$$\int_{-\infty}^{+\infty} e^{-x^2} f(x) dx \approx \sum_{k=1}^n A_k f(x_k)$$

$$\text{Hermite正交多项式: } H_n(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n} (e^{-x^2})$$

6.3 复化求积公式

基本思想 将求积区间 $[a, b]$ 分成若干小区间，然后在每个小区间上采用数值稳定的N-C公式求小区间上的定积分，最后把所有小区间上的计算结果相加起来作为原积分的近似值。

6.3.1 复化梯形公式

1、复化梯形公式的构造

$$\text{在}[a, b]\text{行取等距节点 } x_i = a + ih, h = \frac{b-a}{n}, i = 0, 1, \dots, n$$

$$\therefore \int_a^b f(x) dx = \int_{x_0}^{x_1} f(x) dx + \dots + \int_{x_{n-1}}^{x_n} f(x) dx = \sum_{k=0}^{n-1} \int_{x_k}^{x_{k+1}} f(x) dx$$

在每个小区间上用梯形公式，有

$$\sum_{k=0}^{n-1} \int_{x_k}^{x_{k+1}} f(x) dx \approx \sum_{k=0}^{n-1} \frac{h}{2} [f(x_k) + f(x_{k+1})] = \frac{h}{2} \left[f(a) + f(b) + 2 \sum_{k=1}^{n-1} f(x_k) \right]$$

$$\int_a^b f(x) dx \approx \frac{b-a}{2n} \left[f(a) + f(b) + 2 \sum_{k=1}^{n-1} f(x_k) \right] \quad (\text{复化梯形公式})$$

2、复化梯形公式的余项

$$\text{记 } T_n = \frac{b-a}{2n} \left[f(a) + f(b) + 2 \sum_{k=1}^{n-1} f(x_k) \right]$$

$$\Rightarrow \int_a^b f(x) dx - T_n = \sum_{k=0}^{n-1} -\frac{h^3}{12} f''(\eta_k) = -\frac{b-a}{12} h^2 f''(\eta), \eta \in [a, b]$$

复化梯形余项公式：

$$R(f, T_n) = -\frac{b-a}{12} h^2 f''(\eta), \eta \in [a, b]$$

3、复化梯形公式误差估计

给定计算精度 ε ，并设 $|f''(x)| \leq M_2, x \in [a, b]$

$$|R(f, T_n)| \leq \left| -\frac{b-a}{12} h^2 f''(\eta) \right| \leq \frac{b-a}{12} h^2 M_2 \leq \varepsilon \Rightarrow h \leq \sqrt{\frac{12\varepsilon}{(b-a)M_2}}, n = \frac{b-a}{h}$$

给定 ε ，上述公式可得满足精度要求的复化梯形公式的 n 。

6.3.2 复化Simpson公式

1、复化Simpson公式构造

$$\text{在}[a, b]\text{上取等距节点 } x_i = a + ih, h = \frac{b-a}{n}, i = 0, 1, \dots, n$$

$$\therefore \int_a^b f(x) dx = \int_{x_0}^{x_1} f(x) dx + \dots + \int_{x_{n-1}}^{x_n} f(x) dx = \sum_{k=0}^{n-1} \int_{x_k}^{x_{k+1}} f(x) dx$$

在每个小区间上用抛物线公式，有

$$\sum_{k=0}^{n-1} \int_{x_k}^{x_{k+1}} f(x) dx \approx \sum_{k=0}^{n-1} \frac{h}{6} \left[f(x_k) + 4f\left(\frac{x_k + x_{k+1}}{2}\right) + f(x_{k+1}) \right] = \frac{h}{6} \left[f(a) + f(b) + 4 \sum_{k=0}^{n-1} f\left(x_{k+\frac{1}{2}}\right) + 2 \sum_{k=1}^{n-1} f(x_k) \right]$$

$$\int_a^b f(x) dx \approx \frac{b-a}{6n} \left[f(a) + f(b) + 4 \sum_{k=0}^{n-1} f\left(x_{k+\frac{1}{2}}\right) + 2 \sum_{k=1}^{n-1} f(x_k) \right] \quad (\text{复化Simpson公式})$$

2、复化Simpson公式的余项

$$\text{记 } S_n = \frac{b-a}{6n} \left[f(a) + f(b) + 4 \sum_{k=0}^{n-1} f\left(x_{k+\frac{1}{2}}\right) + 2 \sum_{k=1}^{n-1} f(x_k) \right], h = \frac{b-a}{n}$$

$$R(f, S_n) = -\frac{b-a}{180} \left(\frac{h}{2}\right)^4 f^{(4)}(\eta), \eta \in [a, b]$$

3、复化Simpson公式误差估计

给定计算精度 ε ，并设 $|f^{(4)}(x)| \leq M_4, x \in [a, b]$

$$|R(f, S_n)| \leq \left| -\frac{b-a}{180} \left(\frac{h}{2}\right)^4 f^{(4)}(\eta) \right| \leq \frac{b-a}{180} \left(\frac{h}{2}\right)^4 M_4 \leq \varepsilon$$

$$\Rightarrow h \leq 2 \times \sqrt[4]{\frac{180\varepsilon}{(b-a)M_4}}, n = \frac{b-a}{h}$$

给定 ε ，上述公式可得满足精度要求的复化Simpson公式的 n 。

6.4 Romberg 求积方法

Romberg求积方法是对复化梯形公式用加速技术得到的一种求积方法，它也成为逐次分半加速收敛法。

基本思想

将Richardson外推算法应用于复化梯形公式中，用产生的加速数列来求定积分值。

6.4.1 Richardson 外推定理

定理 设 $F_0(h) \rightarrow F^*, (h \rightarrow 0)$ 且有 $F^* - F_0(h) = \alpha_1 h^{p_1} + \alpha_2 h^{p_2} + \dots$

式中， $0 < p_1 < p_2 < \dots; p_k, \alpha_k$ 是与 h 无关的非零常数。

$$\text{若取 } F_1(h) = \frac{F_0(qh) - q^{p_1} F_0(h)}{1 - q^{p_1}}, 0 < q < 1$$

$$\text{则有 } F^* - F_1(h) = \alpha_2^{(1)} h^{p_2} + \alpha_3^{(1)} h^{p_3} + \dots$$

$$F_1(h) \rightarrow F^*, \text{ 式中 } \alpha_k^{(1)} \text{ 是与 } h \text{ 无关的非零常数。}$$

称用如上定理做加速的方法为**Richardson**外推法。

显然这种外推可以不断做下去以获得逼近更快的函数，一般有：

$$F^* - F_m(h) = \alpha_{m+1}^{(m)} h^{p_{m+1}} + \alpha_{m+2}^{(m)} h^{p_{m+2}} + \dots \Rightarrow F_m(h) = O(h^{p_{m+1}})$$

$$\text{则 } F_{m+1}(h) = \frac{F_m(qh) - q^{p_m} F_m(h)}{1 - q^{p_m}} \Rightarrow F_{m+1}(h) = O(h^{p_{m+2}}), m = 0, 1, \dots$$

6.4.2 Romberg 求积方法

$$\text{记 } I = \int_a^b f(x) dx, \quad T_0(h) = T_n \quad T_n = \frac{b-a}{2n} \left[f(a) + f(b) + 2 \sum_{k=1}^{n-1} f(x_k) \right]$$

$$\because I - T_0(h) = \alpha_1 h^2 + \alpha_2 h^4 + \dots, T_0(h) = O(h^2)$$

利用Richardson外推法做加速

$$T_1(h) = \frac{T_0(qh) - q^2 T_0(h)}{1 - q^2}, 0 < q < 1$$

$$\text{取 } q = \frac{1}{2} \Rightarrow T_1(h) = \frac{4T_0(h/2) - T_0(h)}{4 - 1}, T_1(h) = O(h^4)$$

一般经Richardson加速求定积分的序列为

$$T_0(h) = T_n, \quad T_0(h) = O(h^2)$$

$$T_m(h) = \frac{4^m T_{m-1}(h/2) - T_{m-1}(h)}{4^m - 1}, m = 1, 2, \dots \quad T_m(h) = O(h^{2m+2})$$

$$\text{注意到 } T_1(h) = S_n = \frac{4T_{2n} - T_n}{4 - 1}, \text{ 类似可得}$$

$$T_2(h) = C_n = \frac{4^2 S_{2n} - S_n}{4 - 1}, \quad T_3(h) = R_n = \frac{4^3 C_{2n} - C_n}{4 - 1}, \dots$$

梯形序列：\$\{T_1, T_2, T_{2^2}, \dots, T_{2^k}, \dots\}\$

Simpson序列：\$\{S_1, S_2, S_{2^2}, \dots, S_{2^k}, \dots\}\$

Cotes序列：\$\{C_1, C_2, C_{2^2}, \dots, C_{2^k}, \dots\}\$

Romberg序列：\$\{R_1, R_2, R_{2^2}, \dots, R_{2^k}, \dots\}\$

T_{2n} 的计算公式：

$$T_{2n} = \frac{T_n}{2} + \frac{b-a}{2n} \sum_{k=1}^n f\left(a + (2k-1)\frac{b-a}{2n}\right)$$

Romberg求积方法的计算过程：

6.5 数值微分

根据函数在若干点处的函数值去求该函数的导数近似值称为数值微分，所求导数的近似值常称为数值导数。

基本思想

用来自数据的插值函数获得该函数的导数。

6.5.1 用多项式插值函数估计导数

$$\because R_m(x) = f^{(m)}(x) - P_n^{(m)}(x) = \frac{d^m}{dx^m} \left[\frac{f^{(n+1)}(\xi)}{(n+1)!} \omega_{n+1}(x) \right] \quad (m = 1, 2, \dots)$$

式中 $P_n(x)$ 是 $f(x)$ 的 n 次插值多项式。

特别有一阶数值导数的余项关系

$$R_1(x) = \frac{f^{(n+2)}(\xi)}{(n+2)!} \cdot \frac{d\xi}{dx} \cdot \omega_{n+1}(x) + \frac{f^{(n+1)}(\xi)}{(n+1)!} \cdot \omega'_{n+1}(x)$$

在节点处有

$$R_1(x_k) = f'(x_k) - P'_n(x_k) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \cdot \omega'_{n+1}(x_k) \quad k = 0, 1, \dots, n$$

插值型求导公式多用于求在节点处的数值导数。

带余项的两点数值微分公式：

$$\text{前差公式：} f'(x_0) = \frac{1}{h} [f(x_1) - f(x_0)] - \frac{h}{2} f''(\xi)$$

$$\text{后差公式：} f'(x_1) = \frac{1}{h} [f(x_1) - f(x_0)] + \frac{h}{2} f''(\xi)$$

给定3个点 $x_k, f(x_k) (k = 0, 1, 2)$ $x_k = x_0 + kh, h > 0$ ，类似的有带余项的三点数值微分公式：

$$\text{前差公式：} f'(x_0) = \frac{1}{2h} [-3f(x_0) + 4f(x_1) - f(x_2)] + \frac{h^2}{3} f'''(\xi)$$

$$\text{中心公式：} f'(x_1) = \frac{1}{2h} [-f(x_0) + f(x_2)] - \frac{h^2}{6} f'''(\xi), \xi \in (x_0, x_2)$$

$$\text{后差公式：} f'(x_2) = \frac{1}{2h} [f(x_0) - 4f(x_1) + 3f(x_2)] + \frac{h^2}{3} f'''(\xi)$$

6.5.2 用样条插值函数估计导数

三次样条插值函数的结论：

三次样条插值函数 $S(x)$ 与被插函数 $f(x)$ 有如下逼近关系：

$$\text{a) } \|f - S\| \leq \frac{5h^4}{384} \|f^{(4)}\|$$

$$b) \|f' - S'\| \leq \frac{h^3}{24} \|f^{(4)}\|$$

$$c) \|f'' - S''\| \leq \frac{h^2}{8} \|f^{(4)}\|$$

$$h = \max_{0 \leq k \leq n-1} h_k, \quad h_k = x_{k+1} - x_k$$

Chapter 7 常微分方程初值问题数值解法

7.1 基本概念

7.1.1 微分方程数值解有关的概念

1、常微分方程初值问题一般形式

$$\begin{cases} y' = f(x, y) & a < x < b \\ y(a) = y_0 \end{cases}$$

2、数值方法

求 $y = y(x)$ 在离散点 $\{x_k\}$ 处近似值 $\{y_k\}$ 的方法。

3、数值解 $y(x_k) \approx y_k, k = 1, 2, \dots, n$

设区间 $[a, b]$ 上的一组节点为 $a = x_0 < x_1 < \dots < x_n = b$

步长： $h_k = x_{k+1} - x_k$

求数值解一般是从初值开始按递推方式求解，即：

$$y_0 = y(a) \rightarrow y_1 \rightarrow y_2 \rightarrow \dots \rightarrow y_n = y(b)$$

初值问题的解法有单步法和多步法：

单步法： $y_k \rightarrow y_{k+1}$

多步法： $\{y_{k-l}, y_{k-l+1}, \dots, y_{k-1}, y_k\} \rightarrow y_{k+1}$

数值解法还有显格式和隐格式之分。

7.1.2 初值问题化为差分方程的方法

基本思想

用数值微分法、数值积分法、Taylor展开法等离散化方法将初值问题化为差分方程后再求解。

初值问题化为差分方程的方法

- 1、用离散方法去掉微分方程中的导数得到近似离散化方程；
- 2、在近似离散化方程中用 y_k 替换 $y(x_k)$ ；
- 3、在近似离散化方程中将“ \approx ”换为“ $=$ ”。

1、用数值微分法构造

$$\because y' = f(x, y) \Rightarrow y'(x) = f(x, y(x)) \Rightarrow y'(x_k) = f(x_k, y(x_k))$$

用数值微分的2点前差公式代替导数，有

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

$$\text{得近似离散化方程：} \frac{y(x_{k+1}) - y(x_k)}{x_{k+1} - x_k} \approx f(x_k, y(x_k))$$

用初值问题化差分方程方法 $y_k \rightarrow y(x_k), \approx \rightarrow =$

$$\frac{y(x_{k+1}) - y(x_k)}{x_{k+1} - x_k} \approx f(x_k, y(x_k)) \Rightarrow \frac{y_{k+1} - y_k}{h} = f(x_k, y_k), h = x_{k+1} - x_k$$

整理得

$$y_{k+1} = y_k + hf(x_k, y_k) \quad k = 0, 1, \dots, n-1 \quad (\text{Euler公式, 是显式单步法})$$

2、用数值积分法构造

对 $y' = f(x, y)$ 两边积分可得

$$y(x_{k+1}) - y(x_k) = \int_{x_k}^{x_{k+1}} y' dx = \int_{x_k}^{x_{k+1}} f(x, y(x)) dx$$

采用梯形公式 $\int_a^b f(x) dx \approx \frac{b-a}{2} (f(a) + f(b))$ 得近似离散化方程

$$y(x_{k+1}) - y(x_k) \approx \frac{h}{2} [f(x_k, y(x_k)) + f(x_{k+1}, y(x_{k+1}))]$$

$$\Rightarrow y_{k+1} = y_k + \frac{h}{2} [f(x_k, y_k) + f(x_{k+1}, y_{k+1})] \quad \text{是隐式单步法}$$

3、用Taylor展开法构造

函数 $y(x)$ 的Taylor展开式为

$$y(x_{k+1}) = y(x_k) + hy'(x_k) + \frac{h^2}{2!} y''(x_k) + \dots = y(x_k) + hf(x_k, y(x_k)) + \dots$$

取上式右端前两项，得近似离散化方程：

$$y(x_{k+1}) \approx y(x_k) + hf(x_k, y(x_k))$$

$$\text{做 } y_k \rightarrow y(x_k), \approx \Rightarrow y_{k+1} = y_k + hf(x_k, y_k)$$

7.2 数值解法的误差、阶与绝对稳定性

怎样衡量求数值解公式的好坏？

7.2.1 $y(x_k), y_k, \bar{y}_k$ 的区别

$y(x_k)$ ：解 $y(x)$ 在 x_k 的准确值，没有误差；

y_k ：解 $y(x_k)$ 的近似解，是数值解，有截断误差；

\bar{y}_k ：计算 y_k 给出的计算解，有舍入误差。

7.2.2 数值解法的误差

1、单步法数学描述

$$\text{显式：} y_{k+1} = y_k + h\varphi(x_k, y_k, h)$$

$$\text{隐式：} y_{k+1} = y_k + h\varphi(x_k, y_k, y_{k+1}, h)$$

增量函数： $\varphi()$ 与 $f(x, y)$ 有关。

$$y_{k+1} = y_k + hf(x_k, y_k) \Rightarrow \varphi() = f(x_k, y_k)$$

2、数值解法的误差

$$\text{整体截断误差：} e_{k+1} = y(x_{k+1}) - \bar{y}_{k+1}$$

$$\text{局部截断误差：} T_{k+1} = y(x_{k+1}) - y_{k+1}$$

$$T_{k+1} = y(x_{k+1}) - y(x_k) - h\varphi(x_k, y(x_k), h)$$

7.2.3 数值解法的阶、稳定性

1、数值解法的阶

$$P\text{阶方法 (精度)：} T_{k+1} = O(h^{P+1})$$

方法的阶 P 越大，方法越好。

若 $T_{k+1} = O(h^{P+1}) = g(x_k, y(x_k))h^{P+1} + O(h^{P+2})$

主局部截断误差： $g(x_k, y(x_k))h^{P+1}$

2、数值解法的稳定性

数值方法是绝对稳定的： $|\varepsilon_{k+1}| \leq |\varepsilon_k|, \varepsilon_k = y_k^* - y_k$

试验方程： $y' = \lambda y, \lambda$ 为复数。

绝对稳定域： $\{\mu = \lambda h \mid |\varepsilon_{k+1}| \leq |\varepsilon_k|, y' = \lambda y\}$

绝对稳定区间：绝对稳定域与复平面实轴的交。

绝对稳定域越大，方法的绝对稳定性越好。

7.3 Euler方法的有关问题

7.3.1 Euler方法的几何意义及误差

1、Euler方法的几何意义（折线法）

2、Euler方法的误差

Euler方法的局部截断误差： $T_k = O(h^2)$

Euler方法的总体截断误差： $|e_k| = O(h)$

$\because |e_k| = |y(x_k) - \bar{y}_k| = O(h) \rightarrow 0, h \rightarrow 0 \Rightarrow \bar{y}_k \rightarrow y(x_k)$

说明Euler方法计算所得数值解可以逼近准确解，从而Euler方法是收敛的。

7.3.2 Euler方法的稳定域及改进

1、Euler方法的稳定域

将Euler公式 $y_{k+1} = y_k + hf(x_k, y_k)$ 用于试验方程： $y' = \lambda y$ 得到

$$y_{k+1} = y_k + h\lambda y_k = (1 + \lambda h)y_k$$

设计算 y_k 时有误差： $\varepsilon_k, k = 0, 1, \dots$

$$\Rightarrow y_{k+1} + \varepsilon_{k+1} = (1 + \lambda h)(y_k + \varepsilon_k)$$

$$\Rightarrow \varepsilon_{k+1} = (1 + \lambda h)\varepsilon_k \quad |\varepsilon_{k+1}| \leq |\varepsilon_k| \Rightarrow |1 + \lambda h| \leq 1$$

Euler方法绝对稳定域为： $|1 + \lambda h| \leq 1$

Euler方法绝对稳定区间为： $-2 \leq \operatorname{Re}(\lambda h) < 0$

若 λ 是负实数，则有当步长 h 满足： $0 < h \leq -\frac{2}{\lambda}$ 可保证Euler方法的计算绝对稳定。

2、改进的Euler方法

预测： $\bar{y}_{k+1} = y_k + hf(x_k, y_k)$ (Euler方法)

校正： $y_{k+1} = y_k + \frac{h}{2}[f(x_k, y_k) + f(x_{k+1}, \bar{y}_{k+1})]$ (梯形方法)

是预测—校正公式。上面组合公式称为改进的Euler方法。

易证改进的Euler方法是二阶方法。

7.4 Runge-Kutta方法

理论上，由Taylor展开法可以构造出解初值问题的高阶求数值解方法，但这设计到要计算 $f(x, y)$ 的高阶导数，很不方便。

基本思想

将微分方程初值问题转化为积分方程问题，再对积分方程中的定积分使用待定的 m 点插值型求积公式构造高阶的函数展开模式以获得高阶数值解方法。

7.4.1 Runge-Kutta方法的一般公式

对 $y' = f(x, y)$ 两边积分，有 $y(x_{k+1}) - y(x_k) = \int_{x_k}^{x_{k+1}} y' dx = \int_{x_k}^{x_{k+1}} f(x, y(x)) dx$

用 m 个点插值型求积公式，得近似离散化方程

$$y(x_{k+1}) \approx y(x_k) + h \sum_{i=1}^m c_i f(\xi_i, y(\xi_i)) \quad \xi \in [x_k, x_{k+1}], h = x_{k+1} - x_k$$

取 $\xi_i = x_k + a_i h$ ，并对 $f(\xi_i, y(\xi_i))$ 取不同Taylor公式，得：

$$\begin{cases} y_{k+1} = y_k + h \sum_{i=1}^m c_i K_i \\ K_1 = f(x_k, y_k) \\ K_r = f\left(x_k + a_r h, y_k + h \sum_{j=1}^{r-1} b_{rj} K_j\right) (r = 2, 3, \dots, m) \end{cases}$$

利用Taylor公式对参数 $\{a_r, b_{rj}, c_i\}$ 适当选择就可以构造高阶方法。

$$\text{R-K方法的增量函数: } \varphi(x, y, h) = \sum_{i=1}^m c_i K_i$$

二元Taylor展开公式

$$f(x+h, y+k) = f(x, y) + f'_x h + f'_y k + \frac{1}{2!} (f''_{xx} h^2 + 2hk f''_{xy} + f''_{yy} k^2) + \dots$$

7.4.2 Runge-Kutta方法的构造

只讲授 $m = 2$ 的R-K方法的构造过程：

$$y_{k+1} = y_k + h(c_1 K_1 + c_2 K_2)$$

$$K_1 = f(x_k, y_k), K_2 = f(x_k + a_2 h, y_k + h b_{21} K_1)$$

其增量函数为：

$$\varphi(x, y(x), h) = c_1 f(x, y(x)) + c_2 f(x + a_2 h, y(x) + h b_{21} f(x, y(x)))$$

有4个待定参数 $\{a_1, b_{21}, c_1, c_2\}$

$$\because T_k = y(x_k + h) - y(x_k) - h\varphi(x_k, y(x_k), h)$$

为展开方便，记 $T = y(x + h) - y(x) - h\varphi(x, y(x), h)$

对 $y(x + h)$ 增量函数 $\varphi(x, y(x), h)$ 做Taylor展开，有

$$\Rightarrow T = (1 - c_1 - c_2)fh + \left[\left(\frac{1}{2} - c_2 a_2\right)f'_x + \left(\frac{1}{2} - c_2 b_{21}\right)f'_y f\right]h^2 + O(h^3)$$

$$\text{取 } c_1 + c_2 = 1, \quad \frac{1}{2} - c_2 a_2 = 0, \quad \frac{1}{2} - c_2 b_{21} = 0 \Rightarrow T = O(h^3)$$

从方程组解出 c_1, c_2, b_{21}, a_2 ，即得到一组二阶R-K公式。因方程组有3个方程4个参数，有无穷多解。

R-K方法的阶与级的关系

计算 $f(x, y)$ 的次数 m ：1 2 3 4 5 6 7

对应方法的最高阶：1 2 3 4 4 5 6

7.5 线性多步法

基本思想

用未知函数 $y(x)$ 及其导数 $y' = f(x, y)$ 在若干节点处函数值的线性组合来构造求数值解的公式。

$$\text{线性 } n \text{ 步法公式: } y_{k+1} = \sum_{i=0}^{n-1} \alpha_i y_{k-i} + h \sum_{i=0}^n \beta_i f_{k-i+1}, k = 0, 1, \dots$$

$$\alpha_i, \beta_i \text{ 为常数, } f_k = f(x_k, y_k), x_{k-i} = x_k - ih, \alpha_{n-1}^2 + \beta_{n-1}^2 \neq 0$$

$$\{y_{k-n+1}, y_{k-n+2}, \dots, y_{k-1}, y_k\} \rightarrow y_{k+1}$$

7.5.1 用数值积分法构造线性多步公式

对 $y' = f(x, y)$ 两边积分，得

$$y(x_{k+m}) = y(x_{k-m+1}) + \int_{x_{k-m+1}}^{x_{k+m}} f(x, y(x)) dx$$

对右端的定积分选择 $f(x, y(x))$ 的不同插值函数代替被积函数 $f(x, y(x))$ 、对公式中的 m 取不同的正整数即可得到不同的线性多步法计算公式。

Adams方法

$$m = 1 \Rightarrow y_{k+1} = y_k + h \sum_{i=0}^{n-1} \beta_i f_{k-i} \quad n\text{步Adams显式（外推）公式}$$

$$y_{k+1} = y_k + h \sum_{i=0}^{n-1} \beta_i f_{k+1-i} \quad n\text{步Adams隐式（内插）公式}$$

n 步Adams显式和隐式公式的局部截断误差都是 $T_{k+1} = O(h^{n+1})$

3阶Adams外推公式：

$$y_{k+1} = y_k + \frac{h}{12} (23f_k - 16f_{k-1} + 5f_{k-2})$$

$$\text{其局部截断误差：} T = \frac{3h^4}{8} y^{(4)}(\eta)$$

7.5.2 用Taylor展开法构造线性多步公式

基于Taylor展开的构造方法做法：

先给出线性多步法的计算公式模式，再对局部截断误差表达式处作Taylor展开确定公式的系数。