21世纪高等院校教材

数值计算方法

魏毅强　张建国　张洪斌　等　编

科学出版社
www.sciencep.com

内 容 简 介

本书介绍数值计算方法的研究对象、内容和特点，主要内容为误差理论、方程求根、线性方程组的数值方法、矩阵的特征值与特征向量问题、代数插值、数据拟合与函数逼近、数值积分与数值微分、常微分方程数值解法、偏微分方程的数值解法和数值试验．每章都配有一定量的习题，书末附有答案．

本书可作为高等院校计算机和计算专业本科生教材，也可供相关专业的教师和科技工作者参考．

图书在版编目(CIP)数据

数值计算方法/魏毅强，张建国，张洪斌等编．—北京：科学出版社，2004
21世纪高等院校教材
ISBN 7-03-013488-5

Ⅰ．数… Ⅱ．①魏…②张…③张… Ⅲ．数值计算-计算方法-高等学校-教材 Ⅳ．O241

中国版本图书馆CIP数据核字(2004)第050681号

责任编辑：马长芳 李鹏奇 责任校对：包志虹
责任印制：安春生 封面设计：陈 敬

科学出版社 出版
北京东黄城根北街16号
邮政编码：100717
http://www.sciencep.com
源海印刷有限责任公司印刷
科学出版社发行 各地新华书店经销
*
2004年8月第 一 版 开本：B5(720×1000)
2004年8月第一次印刷 印张：18 1/2
印数：1—6 500 字数：352 000
定价：24.00元
(如有印装质量问题，我社负责调换〈路通〉)

前　言

随着科学技术的飞速发展和计算机的广泛应用，科学计算已发展成为科学研究和工程技术中不可缺少的重要方法之一.掌握数值计算方法的基本知识，熟练地运用计算机进行科学计算，已成为当代理工科大学生必备的基础与技能.本书正是为适应这一需要而编写的.

本书讲述以计算机为计算工具的数值计算方法，内容共分为10章.第1章主要介绍计算机数系、误差概念及数值计算中的若干原则；第2章介绍求解代数方程和超越方程根的基本方法，包括增值寻根法、二分法、迭代法、牛顿法、割线法；第3章主要讲述线性方程组的数值方法，包括各种消元法、矩阵分解法及迭代法；第4章讲述特征值与特征向量的计算方法，包括幂法与反幂法、雅可比方法、多项式法、QR迭代法；第5章介绍了用实验数据建立数学模型的曲线拟合与逼近理论；第6章讲述实验数据点之间的插值法，包括拉格朗日插值法、牛顿插值法、埃尔米特插值法；第7章讲述数值积分与数值微分，包括牛顿-科茨公式、龙贝格算法、高斯公式、插值型求导公式；第8章讲述常微分方程数值解法，主要包括欧拉法、龙格-库塔法、亚当斯法、方程组与高阶方程及边值问题的数值解法；第9章主要讲述了椭圆型、抛物型、双曲型三种偏微分方程中最基本形式的数值解法，包括差分法、变分法与有限元法；第10章主要给出了对应第2～9章的数值实验.第1～9章后配有适量的习题，供读者练习.

本书注重理论联系实际，在介绍基本算法的同时，结合工程实例，讨论算法的逻辑结构、计算步骤和编程技巧.本书还强调对学生实际能力的培养，强调与计算机的密切结合，书中绝大部分算法都给出了详细的流程图，据此读者容易编程上机计算.读者只有编写出满足精度和计算速度要求的实用程序才能体会到每种算法的实质，有助于读者巩固、加深、拓广所学的基本理论与方法，积累计算经验，提高理论联系实际的能力和分析问题与解决问题的能力.

本教材是编者多年来教学工作经验的总结，是在原有讲义的基础上修改补充编写而成的，其中第1章由太原理工大学魏毅强编写，第2章由北方工业大学张建国编写，第3,4,7章由太原理工大学王彩贤编写，第5,6,10章由太原理工大学王淑丽编写，第8章由北方工业大学邹杰涛编写，第9章由太原理工大学张洪斌编写.

由于编者水平所限，书中难免有错误不妥之处，恳请读者指正.

编者

2004.5

目　录

第1章 绪 论

本章简要介绍数值计算方法的研究对象、内容和特点，讨论浮点数、误差的基本概念，并且提出在数值计算中应当普遍遵循的若干原则.

1.1 数值计算方法的研究对象和特点

随着电子技术的发展和科学研究、生产实践的需要，电子计算机的使用日益广泛.计算机作为科学计算的主要工具越来越不可缺少，因而要求研究适合计算机使用的数值计算方法.为了更具体地说明数值计算方法的研究对象，我们考察用计算机解决科学计算问题的一般过程，可以概括为

实际问题 → 数学模型 → 计算方法 → 程序设计 → 上机计算

由实际问题应用有关科学知识和数学理论建立数学模型这一过程，通常作为应用数学的任务.而根据数学模型提出求解的计算方法直到编出程序上机算出结果，进而对计算结果进行分析，这一过程则是计算数学的任务，也是数值计算方法的研究对象.因此，数值计算方法就是研究用计算机解决数学问题的数值方法及其理论.它的内容包括：误差理论、线性与非线性方程(组)的数值解、矩阵的特征值与特征向量计算、曲线拟合与函数逼近、插值方法、数值积分与数值微分、常微分方程与偏微分方程数值解等.

数值计算方法是一门与计算机使用密切结合的实用性很强的数学课程，它既有纯数学的高度抽象性与严密科学性的特点，又有应用广泛性与实际试验的高度技术性的特点.例如，考虑线性方程组的解，在“线性代数”中，只介绍解的存在唯一性及有关理论和精确解法，用这些理论和方法还不能直接在计算机上求解.我们知道，用克拉默(Cramer)法则求解一个 n 阶线性方程组，要算 $n+1$ 个 n 阶行列式，总共需要 $(n-1)(n+1)n!$ 次乘法，当 n 充分大时，计算量是相当惊人的.如一个20阶不算太大的方程组大约要做 10^{21} 次乘法，这项计算即使用每秒百亿次的计算机去做，也要连续工作数千年才能完成，当然这是完全没有实际意义的.而如果用消元法，求解一个 n 阶线性方程组大约需要 $\frac{1}{3}n^3+n^2$ 次乘法，一个20阶的方程组即使用一台小型计算器也能很快解出来.这一简单的例子告诉我们，能否正确地制定算法，是科学计算成败的关键.另外，要求解这类问题还应根据方程特点，研究适合计算机使用的满足精度要求的，计算时间省的有效算法及其相关理论.在实现这

些算法时往往还要根据计算机容量、字长、速度等指标，研究具体求解步骤和程序设计技巧.有的方法在理论上虽不够严密，但通过实际计算、对比分析等手段，证明是行之有效的方法，也应该采用，这些都是数值计算方法应有的特点.概括起来有四点：

第一，面向计算机，要根据计算机特点提供实际可行的有效算法，即算法只能包括加、减、乘、除运算和逻辑运算，是计算机能直接处理的.

第二，有可靠的理论分析，能任意逼近并达到精度要求，对近似算法要保证收敛性和数值稳定性，还要对误差进行分析，这些都建立在相应数学理论基础上.

第三，要有好的计算复杂性，时间复杂性好是指节省时间，空间复杂性好是指节省存储量，这也是建立算法要研究的问题，它关系到算法能否在计算机上实现.

第四，要有数值实验，即任何一个算法除了从理论上要满足上述三点外，还要通过数值试验证明是行之有效的.

根据“数值计算”的特点，学习时，首先要注意方法处理的技巧及其与计算机的结合，要重视误差分析、收敛性及稳定性的基本理论，其次，要通过例子，学习使用各种数值方法解决实际计算问题.

本章先对计算机数系和计算的误差做一些初步介绍.

1.2　浮　点　数

数值计算的工具是电子计算机，计算机的字长和运算方式对数值计算的结果有直接的影响.对给定的数值方法，一个注意到计算机有限字长和运算方式的程序员可以写出具有较高计算精度的程序，反之，也会得到十分粗糙甚至完全失真的计算结果.因此，了解计算机数的表示和运算方式对使用计算机十分必要.

1.2.1　定点数

设 r 为大于1的正整数，a_i 为 $0,1,\cdots,r-1$ 中的某一个，位数有限的 r 进制正数可以写成

$$x \triangleq a_{l-1}a_{l-2}\cdots a_0 \cdot a_{-1}a_{-2}\cdots a_{-m} \tag{1.2.1}$$

x 有 l 位整数，有 m 位小数.因为进位制的基数是 r，所以

$$x = a_{r-1}r^{l-1} + a_{l-2}r^{l-2} + \cdots + a_0 r^0 + a_{-1}r^{-1} + \cdots + a_{-m}r^{-m} \tag{1.2.2}$$

当 $l=4, m=4, r=10$ 时

$$109.312,\quad 0.4375,\quad 4236$$

分别表示为

$$0109.3120,\quad 0000.4375,\quad 4236.0000$$

这种把小数点永远固定在指定位置上位数有限的数称为定点数，称 $n = l + m$ 为

字长.一般地常取 $l = n, m = 0$ 或 $l = 0, m = n$.

当 $l = m = 4$ 而 $r = 10$ 时,8 位定点非零数中绝对值最小和最大的数分别为

$$0000.0001, \quad 9999.9999$$

由此可见,定点数所能表示的数的范围非常小.

值得指出的是,在定点数运算系统中,不仅要求运算操作数在它所能表示的范围内,而且还要求运算结果也在它所能表示的数的范围内,否则会产生溢出.例如,在左边定小数点($l = 0$)的定点运算系统中,$0.5 + 0.6 = 0.1$ 产生上溢出,在 16 位二进制系统的计算机上计算 $\omega = \dfrac{2^{-7} \times 2^{-9}}{2^{-10}}$,利用算法 $\omega = \dfrac{2^{-7} \times 2^{-9}}{2^{-10}} = \dfrac{0}{2^{-10}} = 0$ 产生下溢出.而利用算法 $\omega = \left(\dfrac{2^{-7}}{2^{-5}}\right) \times \left(\dfrac{2^{-9}}{2^{-5}}\right) = 2^{-2} \times 2^{-4} = 2^{-6}$ 则会得到正确的结果.因此,在编制定点运算程序时,要尽量避免运算结果的上、下溢出,计算次序的选择要十分慎重.

1.2.2 浮点数

用于数值计算的计算机多采用浮点系统.因为用浮点方式表示的数有比较大的取值范围,且浮点运算有较高的计算精度,从而为编制程序提供了方便.

设 s 是 r 进制数,p 是 r 进制正负整数或零,r 进制数 x 可以用 s 和 r^p 的乘积表示为

$$x = s \times r^p \tag{1.2.3}$$

再设 s 的整数部分等于零,即 s 满足条件

$$-1 < s < 1 \tag{1.2.4}$$

则形如(1.2.3)而满足条件(1.2.4)的 r 进制数 x 称为 r 进制浮点数.s 和 p 分别称为浮点数 x 的尾数和阶数.如果尾数的小数位数等于有限正整数 t,则把 x 称为 t 位浮点数.

此外,如果还要求尾数 s 小数点后第一位数字不等于 0,也就是要求尾数 s 满足条件

$$r^{-1} \leqslant s < 1 \tag{1.2.5}$$

则形如(1.2.3)而满足条件(1.2.5)的浮点数称为 r 进制规格化浮点数.

例如,十进制数

$$0.003012, \quad 0.3217, \quad 283.4$$

的规格化浮点数分别为

$$0.3012 \times 10^{-2}, \quad 0.3217 \times 10^{0}, \quad 0.2834 \times 10^{3}$$

二进制数

$$1001.101, \quad 0.10101, \quad 0.00101$$

的规格化浮点数分别为

$$0.1001101\times 2^4,\quad 0.10101\times 2^0,\quad 0.101\times 2^{-2}$$

显然，只要数 $x\neq 0$，则 x 一定可以表示为规格化浮点数，这样一来，一个数的数量级就一目了然.

1.2.3　计算机数系

上面介绍的数的浮点表示方法为计算机所通用，是我们研究数值方法的基础，任一计算机只能用有限的位数来表示浮点的尾数和阶数. 设进位制为 r，阶数 p 满足条件

$$l\leqslant p\leqslant u \tag{1.2.6}$$

其中 l,u 为整数，它们主要由计算机用多少位数来表示阶数所确定. 如果尾数的小数位数为 t，则计算机数系由一切阶数满足(1.2.6)的 t 位 r 进制浮点数的集合 F 组成，F 中的浮点数具有以下形式

$$\begin{aligned} x&=\pm\left(\frac{d_1}{r}+\frac{d_2}{r^2}+\cdots+\frac{d_t}{r^t}\right)\cdot r^p\\ &\triangleq\pm 0.d_1d_2\cdots d_t\times r^p \end{aligned} \tag{1.2.7}$$

其中 $d_1,d_2,\cdots,d_t$ 为整数，满足关系

$$0\leqslant d_i\leqslant r-1,\qquad i=1,2,\cdots,t \tag{1.2.8}$$

若对 $x\neq 0$，规定(1.2.7)中 $d_1\neq 0$，则 F 为规格化的浮点数系. 不难证明，F 中共有

$$2(r-1)r^{t-1}(u-l+1)+1 \tag{1.2.9}$$

个浮点数. 例如，若 $r=2,t=3,l=-1,u=2$，则相应的浮点数系 F 中共有33个浮点数.

当 $r=10,t=4,l=-99,u=99$ 时，

$$-0.0001\times 10^{-99},\qquad 0.0001\times 10^{-99}$$

是数系 F 中绝对值最小的非零数，而

$$-0.9999\times 10^{99},\qquad 0.9999\times 10^{99}$$

是此数系中的最小数和最大数，若计算的中间结果超出了上述范围，则称为溢出.

由此可见，在计算机数系 F 中，数的个数有限，数系中的每一个数都是有理数. 从整体看，数系中的数分布很不均匀；从局部看，阶数相同的数，又以相等的距离，分布在数轴的某一段上. 所以计算机数系是由一些残缺不全，分布不均匀的数组成，如果运算结果超出了 F 的范围，则产生溢出.

在计算机中，常用尾数等于0而阶数最小的数来表示零. 例如在上述计算机数系中，用 0×10^{-99} 来表示常数零. 零不能化为规格化浮点数.

1.3 误差的基本概念

除了极个别的情况外，数值计算总是近似计算，实际计算结果与理论结果之间存在着误差. 数值分析的任务之一是将误差控制在一定的允许范围内或者至少对误差有所估计.

1.3.1 误差的来源

用计算机解决科学计算问题首先要建立数学模型，它是对被描述的实际问题进行抽象，简化而得到的，因而是近似的，我们把数学模型与实际问题之间出现的这种误差称为模型误差. 只有实际问题提法正确，建立数学模型时又抽象，简化得合理，才能得到好的结果. 由于这种误差难于用数量表示，通常都假定数学模型是合理的，这种误差可忽略不计，在数值计算方法中不予讨论.

在数学模型中往往还有一些根据观测得到的物理量，如温度、长度、电压等等，这些参量受测量工具及手段的影响，测量的结果不可能绝对正确，由此产生的误差称为观测误差. 观测误差在数值计算方法中也不予讨论.

在数学模型不能得到精确解时，通常要用数值方法求它的近似解，其近似解与精确解之间的误差称为截断误差或方法误差. 例如，函数 $f(x)$ 用泰勒多项式

$$P_n(x) = f(0) + f'(0)x + \frac{1}{2!}f''(0)x^2 + \cdots + \frac{1}{n!}f^{(n)}(0)x^n \quad (1.3.1)$$

近似代替时，有误差

$$R_n(x) = f(x) - P_n(x) = \frac{1}{(n+1)!}f^{(n+1)}(\xi)x^{n+1} \quad (1.3.2)$$

其中 ξ 在 0 与 x 之间. 这种误差就是截断误差.

有了求解数学问题的计算公式以后，用计算机做数值计算时，由于计算机的字长有限，原始数据常常不属于计算机数系，而采用计算机数系中和它们比较接近的数来表示它们，由此产生的误差以及计算过程又可能产生新的误差，这些误差称为舍入误差. 例如，用 3.14159 近似代替 π，产生的误差

$$R = \pi - 3.14159 = 0.0000026\cdots$$

就是舍入误差.

观测误差和原始数据的舍入误差，就其来源说，有所不同，就其对计算结果的影响看，完全一样，数学描述和实际问题之间的模型误差，往往是计算工作者不能独立解决的，甚至是尚待研究的课题. 基于这些原因，在数值计算方法课程中所涉及的误差，一般指舍入误差(包括初始数据的误差) 和截断误差. 讨论它们在计算过程中的传播和对计算结果的影响；研究控制它们的影响以保证最终结果有足够

的精度;既希望解决数值问题的算法简便而有效,又想使最终结果准确而可靠.

1.3.2　绝对误差和相对误差

设数 x(精确值) 有一个近似值为 x^*, 记

$$e(x^*) \triangleq x^* - x \tag{1.3.3}$$

称 $e(x^*)$ 为近似值 x^* 的绝对误差,简称误差.

注意这样定义的误差 $e(x^*)$ 可正可负,当它为正时,近似值 x^* 偏大,叫做强近似值;当它为负时,近似值 x^* 偏小,叫做弱近似值.

准确值 x 一般是未知的,因而绝对误差 $e(x^*)$ 也是未知的,但往往可以估计出绝对误差的一个上界,即可以找出一个正数 η,使

$$|e(x^*)| \leqslant \eta \tag{1.3.4}$$

实践中用 $|e(x^*)|$ 尽可能小的上界 $\varepsilon(x^*)$ 估计 x^* 的误差,称 $\varepsilon(x^*)$ 为 x^* 的绝对误差限(或误差限).

例如,$\pi = 3.14159265358\cdots$,若取 $\pi^* = 3.14159$, 于是

$$|e(\pi^*)| \leqslant 0.000003$$

则 $\varepsilon(x^*) = 0.000003$ 就可以作为用 π^* 近似表示 π 的绝对误差限.

显然,误差限 $\varepsilon(x^*)$ 总是正数,且

$$|e(\pi^*)| \leqslant \varepsilon(x^*) \tag{1.3.5}$$

即

$$x^* - \varepsilon(x^*) \leqslant x \leqslant x^* + \varepsilon(x^*) \tag{1.3.6}$$

这个不等式在应用上常常采用如下写法:

$$x = x^* \pm \varepsilon(x^*) \tag{1.3.7}$$

例如,用毫米刻度的米尺测量一长度 x 时,如果该长度接近某一刻度 x^*,则 x^* 作为 x 的近似值时

$$|e(x^*)| = |x^* - x| \leqslant \frac{1}{2}(\text{毫米}) = 0.5(\text{毫米})$$

它的误差限是 $\varepsilon(x^*) = 0.5$ 毫米. 如果读出的长度为 $x^* = 765$,则有 $|765 - x| \leqslant 0.5$,从这个不等式我们仍不能知道准确的 x 值,只知道 $764.5 \leqslant x \leqslant 765.5$,即 x 在区间[764.5,765.5] 内.

绝对误差还不足以刻画近似数的精确程度,例如,有两个量 $x = 10 \pm 1, y = 1000 \pm 10$,虽然 x 的绝对误差限比 y 的绝对误差限小,但 $\dfrac{\varepsilon(y^*)}{y^*} = \dfrac{10}{1000} = 1\%$ 比 $\dfrac{\varepsilon(x^*)}{x^*} = \dfrac{1}{10} = 10\%$ 要小得多,这说明 $y^* = 1000$ 作为 y 的近似值远比 $x^* = 10$ 作为 x 的近似值的近似程度要好得多. 所以,除考虑误差的大小外,还应考虑准确

值 x 本身的大小.我们把近似值的误差 $e(x^*)$ 与准确值 x 的比值,记作

$$e_r(x^*) \triangleq \frac{e(x^*)}{x} = \frac{x^* - x}{x} \tag{1.3.8}$$

称为近似值 x^* 的相对误差.

在实际计算中,由于真值 x 总是未知的,且由于

$$\frac{e(x^*)}{x} - \frac{e(x^*)}{x^*} = \frac{e(x^*)(x^* - x)}{xx^*} = \frac{[e(x^*)]^2}{x(x + e(x^*))} = \frac{[e_r(x^*)]^2}{1 + e_r(x^*)}$$

是 $e_r(x^*)$ 的平方项级,故当 $e_r(x^*)$ 较小时,常取

$$e_r(x^*) = \frac{e(x^*)}{x^*} = \frac{x^* - x}{x^*} \tag{1.3.9}$$

相对误差也可正可负,它的绝对值的上界称为该近似值的相对误差限,记作 $\varepsilon_r(x^*)$,即

$$|e_r(x^*)| \leqslant \frac{\varepsilon(x^*)}{|x^*|} \triangleq \varepsilon_r(x^*) \tag{1.3.10}$$

由定义可知,绝对误差与绝对误差限是有量纲的量,而相对误差和相对误差限是无量纲的量.

1.3.3 有效数字

如果近似值 x^* 的误差限是某一位的半个单位,该位到 x^* 的第一位非零数字共有 n 位,则我们称 x^* 有 n 位有效数字.

例如,$x = \pi = 3.14159265\cdots$,取 $x^* = 3.14$ 时,

$$|x^* - x| \leqslant 0.002 \leqslant 0.005$$

所以,$x^* = 3.14$ 作为 π 的近似值时,就有 3 位有效数字;而取 $x^* = 3.1416$ 时,

$$|x^* - x| \leqslant 0.000008 \leqslant 0.00005$$

所以,$x^* = 3.1416$ 作为 π 近似值时,就有 5 位有效数字.一般地,在 r 进制中,设近似值 x^* 可表示为

$$x^* = \pm (a_1 r^{-1} + a_2 r^{-2} + \cdots + a_n r^{-n}) \times r^m \tag{1.3.11}$$

$a_1 \neq 0$,且

$$|x^* - x| \leqslant \frac{1}{2} r^{m-n} \tag{1.3.12}$$

则由定义可知,x^* 有 n 位有效数字.

当 $r = 10$ 时,(1.3.11) 式中表示十进制数,而当 $r = 2$ 时,(1.3.11) 式表示二进制规格化浮点数.

例 1 按四舍五入原则,写出下列各数具有 5 位有效数字的近似数

187.9325, 0.03785551, 8.000033, 2.7182818

按定义，上述各数具有 5 位有效数字的近似数分别是

$$187.93,\quad 0.037856,\quad 8.0000,\quad 2.7183$$

注意到，$x = 8.000033$ 的 5 位有效数字是 8.0000，而不是 8，8 只有 1 位有效数字.

(1.3.11) 式说明，有效位数与小数点的位置无关，而具有 n 位有效数字的近似数 x^* 其误差限为

$$\varepsilon(x^*) = \frac{1}{2} \times r^{m-n} \tag{1.3.13}$$

在 m 相同的条件下，有效位数越多，则绝对误差限越小. 而有效数字与相对误差限有下列关系.

定理 1　用(1.3.11)式表示的近似数 x^*，若具有 n 位有效数字，则其相对误差限为

$$|e_r(x^*)| \leqslant \frac{1}{2a_1} \times r^{-(n-1)} \tag{1.3.14}$$

证明　由(1.3.11) 式知，$|x^*| \geqslant a_1 \cdot r^{m-1} > 0$，故

$$|e_r(x^*)| = \frac{|x^* - x|}{x^*} \leqslant \frac{\frac{1}{2} \times r^{m-n}}{a_1 \cdot r^{m-1}} = \frac{1}{2a_1} r^{-(n-1)}$$

定理 2　由(1.3.11)式表示的近似数 x^*，若满足

$$|e_r(x^*)| \leqslant \frac{1}{2(a_1+1)} r^{-(n-1)}$$

则 x^* 至少有 n 位有效数字.

证明　因为 $|x^* - x| = |x^*| \cdot |e_r(x^*)|$，且 $|x^*| \leqslant (a_1+1) r^{m-1}$ 故

$$|x^* - x| \leqslant (a_1+1) r^{m-1} \cdot \frac{1}{2(a_1+1)} \cdot r^{-(n-1)} = \frac{1}{2} \times r^{m-n}$$

故 x^* 至少有 n 位有效数字.

定理 1 说明，近似数 x^* 的有效位数越多，它的相对误差限越小；反之，x^* 的相对误差越小，它的有效位数越多.

例 2　要使 $\sqrt{20}$ 的近似值的相对误差限小于 0.1%，要取几位有效数字.

解　由于 $4 < \sqrt{20} < 5$，所以 $a_1 = 4$，由定理有

$$\frac{1}{2a_1} \times 10^{-n+1} \leqslant 0.1\%$$

即 $10^{n-4} \geqslant \frac{1}{8}$，得 $n \geqslant 4$. 故只要对 $\sqrt{20}$ 的近似数取 4 位有效数字，其相对误差就可小于 0.1 %，因此，可取 $\sqrt{20} \approx 4.472$.

1.4 误差传播

1.4.1 误差分析的重要性

在数值计算方法中,除了研究数学问题的算法外,还要研究计算结果的误差是否满足精度要求,这就是误差估计问题.下面举例说明误差分析的重要性.

例 1 计算 $I_n = \int_0^1 \frac{x^n}{x+10}\mathrm{d}x$,并估计误差.

解 因为

$$I_n + 10I_{n-1} = \int_0^1 \frac{x^n + 10x^{n-1}}{x+10}\mathrm{d}x = \int_0^1 x^{n-1}\mathrm{d}x = \frac{1}{n}$$

可得递推关系

$$I_n = \frac{1}{n} - 10I_{n-1}, \qquad n = 1,2,\cdots$$

其中

$$I_0 = \int_0^1 \frac{1}{x+10}\mathrm{d}x = \ln 11 - \ln 10 = \ln 1.1$$

如果取 $\tilde{I}_0 = 0.095310$ 作为 I_0 的近似,则其误差为 $|e(\tilde{I}_0)| = |\tilde{I}_0 - I_0| \leqslant 0.0000002$,并由递推公式

$$(\mathrm{A}) \begin{cases} \tilde{I}_0 = \dfrac{1}{n} - 10\tilde{I}_{n-1}, & n = 1,2,\cdots \\ \tilde{I}_0 = 0.095310 \end{cases}$$

计算结果见表 1-1.

表 1-1

n	$\tilde{I}_n$	I_0^*
0	0.095310	0.095310
1	0.046900	0.046898
2	0.031000	0.031018
3	0.023333	0.023154
4	0.016667	0.018464
5	0.033333	0.015357
6	−0.166667	0.013093
7	1.809524	0.011932

从表中看出，$\tilde{I}_5 > \tilde{I}_4$，且 $\tilde{I}_6$ 出现负值，这与一切 $I_{n-1} > I_n > 0$ 相矛盾. 因此，当 n 较大时，用 $\tilde{I}_n$ 近似 I_n 显然是不正确的. 这里计算公式与每步计算都是正确的，那么什么原因使计算结果出现错误呢？主要就是初值 $\tilde{I}_0$ 有误差 $e(\tilde{I}_0)$，由此引起以后各步计算的误差 $e(\tilde{I}_n)$，它满足关系

$$e(\tilde{I}_n) = -10e(\tilde{I}_{n-1}), \qquad n = 1,2,\cdots$$

从而

$$e(\tilde{I}_n) = (-10)^n e(\tilde{I}_0)$$

这说明 $\tilde{I}_0$ 有误差 $e(\tilde{I}_0)$，则 $\tilde{I}_n$ 就有 $e(\tilde{I}_0)$ 的 $(-10)^n$ 倍误差.

我们下面换一种计算方法. 由于 $0 < x < 1$ 时，

$$\frac{1}{11}x^n \leqslant \frac{x^n}{10+x} \leqslant \frac{1}{10}x^n$$

所以

$$\frac{1}{11(n+1)} \leqslant I_n \leqslant \frac{1}{10(n+1)}$$

我们粗略地取

$$I_7^* = \frac{1}{2}\left(\frac{1}{11\times 8} + \frac{1}{10\times 8}\right) = 0.011932$$

然后将递推公式倒过来使用，即由公式

$$\text{(B)} \quad \begin{cases} I_{n-1}^* = \dfrac{1}{10}\left(\dfrac{1}{n} - I_n^*\right), & n = 7,6,\cdots,1 \\ I_7^* = 0.011932 \end{cases}$$

计算结果见表 1-1 的 I_n^* 列. 尽管 I_7^* 是粗略地取的，有很大误差 $e(I_7^*)$，但因误差随传播逐步缩小，$e(I_0^*)$ 比 $e(I_n^*)$ 缩小了 $(-10)^n$ 倍. 故计算的数值可靠，可用 I_n^* 近似 I_n.

此例说明，在数值计算中如不注意误差分析，用了类似(A)的计算公式，就会出现“差之毫厘，失之千里”的错误结果. 尽管数值计算中估计误差比较困难，我们仍应重视计算过程中的误差分析.

1.4.2　四则运算的误差传播

设 x_1, x_2 的近似值分别为 x_1^*, x_2^*，有误差

$$e(x_1^*) = x_1^* - x_1, \qquad e(x_2^*) = x_2^* - x_2$$

如果以 $x_1^* + x_2^*, x_1^* - x_2^*$ 分别作为 $x_1 + x_2, x_1 - x_2$ 的近似值，则有

$$e(x_1^* \pm x_2^*) = e(x_1^*) \pm e(x_2^*) \tag{1.4.1}$$

即和的误差是误差之和，差的误差是误差之差，进一步有

$$|e(x_1^* \pm x_2^*)| \leqslant |e(x_1^*)| + |e(x_2^*)|$$

即

$$\varepsilon(x_1^* \pm x_2^*) = \varepsilon(x_1^*) + \varepsilon(x_2^*) \tag{1.4.2}$$

所以误差限之和是和或差的误差限.以上的结果适用于任意多个近似数的和或差.而相对误差有

$$e_r(x_1^* + x_2^*) = \frac{x_1}{x_1 + x_2} e_r(x_1^*) + \frac{x_2}{x_1 + x_2} e_r(x_2^*) \tag{1.4.3}$$

即和的相对误差等于各项相对误差的加权平均.

若 x_1 与 x_2 同号,则(1.4.3)式右端 $e_r(x_1^*)$ 与 $e_r(x_2^*)$ 的系数满足

$$0 < \frac{x_1}{x_1 + x_2}, \frac{x_2}{x_1 + x_2} < 1$$

且

$$\frac{x_1}{x_1 + x_2} + \frac{x_2}{x_1 + x_2} = 1 \tag{1.4.4}$$

此时,由(1.4.3)式可得

$$|e_r(x_1^* + x_2^*)| \leqslant \max\{|e_r(x_1^*)|, |e_r(x_2^*)|\}$$

即

$$\varepsilon_r(x_1^* + x_2^*) \leqslant \max\{\varepsilon_r(x_1^*), \varepsilon_r(x_2^*)\} \tag{1.4.5}$$

和的相对误差限不超过各项相对误差限中的最大者.

若 x_1 与 x_2 异号,则(1.4.3)式中两个系数的绝对值至少有一个大于1,如果这时 x_1 与 $-x_2$ 相当接近,则(1.4.3)式中的两个系数的绝对值都可能很大,从而使 $e_r(x_1^* + x_2^*)$ 很大,在这种情况下,原始数据的误差会对计算结果产生相当大的影响.

如果以 $x_1^* \cdot x_2^*$ 与 $\frac{x_1^*}{x_2^*}$ 分别作为 $x_1 \cdot x_2$ 与 $\frac{x_1}{x_2}$ 的近似值,则有

$$e(x_1^* \cdot x_2^*) \approx x_2^* \cdot e(x_1^*) + x_1^* \cdot e(x_2^*) \tag{1.4.6}$$

$$e\left(\frac{x_1^*}{x_2^*}\right) \approx \frac{x_2^* e(x_1^*) - x_1^* e(x_2^*)}{(x_2^*)^2} \tag{1.4.7}$$

于是

$$\varepsilon(x_1^* \cdot x_2^*) \approx |x_2^*| \cdot \varepsilon(x_1^*) + |x_1^*| \cdot \varepsilon(x_2^*) \tag{1.4.8}$$

$$\varepsilon\left(\frac{x_1^*}{x_2^*}\right) \approx \frac{|x_2^*| \cdot \varepsilon(x_1^*) + |x_1^*| \cdot \varepsilon(x_2^*)}{|x_2^*|^2} \tag{1.4.9}$$

例 2 求解二次方程 $x^2 - 26x + 1 = 0$,并估计误差.

解 利用二次方程的求根公式得