大学数学系列丛书

数值分析简明教程

SHUZHI FENXI JIANMING JIAOCHENG

王兵团　张作泉　赵平福　编著

清华大学出版社
http://www.tup.tsinghua.edu.cn
北京交通大学出版社
http://press.bjtu.edu.cn

大学数学系列丛书

数值分析简明教程

王兵团　张作泉　赵平福　　编著

清华大学出版社
北京交通大学出版社

·北京·

内 容 简 介

本书是为非数学专业理工科大学生和研究生学习数值分析课程所编写的教材。与一般的数值分析教材不同，本书编排由浅入深，采用全新的数值分析论述方式，重点突出数值分析课程的核心和实用性，弱化其数学理论性，特别强调数值分析“立足近似、追求可用”的特点和其内涵的科学研究方法，更加适合学生自学数值分析知识和教师进行数值分析或计算方法课程的研究型教学。

本书的主要内容包括：非线性方程求根方法，线性方程组的解法，求矩阵特征值和特征向量的方法，插值与拟合方法，数值积分与数值微分和常微分方程初值数值解法。

图书在版编目（CIP）数据

数值分析简明教程/王兵团，张作泉，赵平福编著．—北京：清华大学出版社；北京交通大学出版社，2012.8
（大学数学系列丛书）
ISBN 978－7－5121－1110－3

Ⅰ. ①数… Ⅱ. ①王… ②张… ③赵… Ⅲ. ①数值分析-高等学校-教材 Ⅳ. ①O241

中国版本图书馆 CIP 数据核字（2012）第 177905 号

责任编辑：黎　丹
出版发行：清 华 大 学 出 版 社　　邮编：100084　　电话：010－62776969
　　北京交通大学出版社　　邮编：100044　　电话：010－51686414
印 刷 者：北京瑞达方舟印务有限公司
经　　销：全国新华书店
开　　本：185×230　　印张：13.25　　字数：297 千字
版　　次：2012 年 8 月第 1 版　　2012 年 8 月第 1 次印刷
书　　号：ISBN 978－7－5121－1110－3/O・108
印　　数：1～4 000 册　　定价：26.00 元

前　言

数值分析是帮助科研人员了解和进行有效科学计算的桥梁，也是现代科学计算的基础，其理论性和实用性都很强。

数值分析的核心是算法构造和误差分析。与理论数学立足“准确、追求准确结果”不同的是，数值分析是“立足近似、追求可用结果”。它面对有误差的原始数据，采用近似方法来获得满足精度要求的近似结果。这种近似处理技术是解决实际问题的重要手段之一。

此外，数值分析含有丰富的科学研究方法和创新内容。了解科学研究方法论且熟悉数值分析的学者会发现整个数值分析的内容本身就是科学研究方法的实际应用，其每种算法的构造过程都是科学方法的具体实现。

本书是为非数学专业理工科大学生和研究生学习数值分析课程所编写的教材。与一般的数值分析教材不同的是，本书编排突出数值分析课程的核心和实用性，追求数值分析概念的理解和算法构造，弱化数值分析的数学理论性，特别强调数值分析“立足近似、追求可用”的特点和其内涵的科学研究方法，更加适合学生自学数值分析知识和教师进行数值分析或计算方法课程的研究型教学。

本书每章配以大量的精选例题和习题，还有思考题、数值实验和知识扩展阅读，目的在于帮助学生在学好数值分析课程的同时，能在更高层次理解所学知识。本书每章的思考题和数值实验也是授课教师进行研究型教学的素材。此外，以此书为授课内容的教学录像已经由超星学术视频录制完成并上线播放以方便读者上网学习。读者若想看授课视频，可以在百度的网页搜索栏中键入：“王兵团，数值分析”，即可找到该视频或直接登录网址 http：//video. chaoxing. com/serie _400001778. shtml。

本书在编写中受到很多教师和学生的关注，北京交通大学研究生院给予了大力支持，还得到北京交通大学教材出版基金的资助，在此表示衷心感谢。

由于水平所限，书中难免有不妥之处，恳请读者指正。

编者

2012 年 6 月

目　录

第1章　绪　论

本章主要介绍科学计算的特点及数值分析基本知识．它们对学习数值分析、了解科学计算原理，以及进行科学计算都是很有帮助的．

1.1　学习数值分析的重要性

数值分析主要研究怎样用计算机来对各种数学问题进行科学计算的方法和理论．与其他理论数学（微积分、高等代数等）不同，数值分析能根据近似的数据，采用近似的方法去获得满足要求的近似结果．科学计算是通过计算机进行数学计算的，很多人，甚至为数不少的一些科研人员，常常认为在进行科学计算时，只要根据对应的一些数学公式，用一种计算机语言正确编程，计算机就一定能给出正确的结果！问题是这样简单吗？请看下面的例子．

【例 1-1】　将数列 $I_n=\int_0^1 \frac{x^n}{x+5}\mathrm{d}x$ 写成递推公式的形式，并计算数列 I_1，I_2，… 的值．

解：因为

$$\begin{aligned} I_n &= \int_0^1 \frac{x^n+5x^{n-1}-5x^{n-1}}{x+5}\mathrm{d}x \\ &= \int_0^1 x^{n-1}\mathrm{d}x-5\int_0^1 \frac{x^{n-1}}{x+5}\mathrm{d}x=\frac{1}{n}-5I_{n-1} \end{aligned}$$

得到计算 I_n 的递推公式

$$I_n=\frac{1}{n}-5I_{n-1},\quad n=1,2,\cdots \tag{1-1}$$

由 $I_0=\int_0^1 \frac{1}{x+5}\mathrm{d}x=\ln\frac{6}{5}$，借助递推公式（1-1）可依次算出 I_1，I_2，…

因为 $\ln\frac{6}{5}$ 是一个符号，不是数值，要参与递推公式计算，必须用数字表示才行．但 $\ln\frac{6}{5}$ 是一个无理实数，不能用有限位数字表示，为实现计算，这里取 $\ln\frac{6}{5}$ 准确到小数点后8位的近似值作为初始值 I_0．在字长为8的计算机上编程计算，出现了 $I_{12}=-0.32902110\times 10^2$ 的结果，这显然是错误的！因为数列 I_n 的被积函数在积分区间上是非负的，故总应有 $I_n\geqslant 0$ 才对！（读者可以在自己的计算机上用递推公式（1-1）编程做一个数值实验，来检验当 n 较大时，I_n 的计算结果会出现负数的现象．）

类似上面的例子还有很多，在此不再列举了．上面例子说明，并不是把科学计算中涉及的数学公式机械地编程计算就一定能得到正确的结果，这其中有很多值得认识和探讨的问题．当然这里并不是否定计算机在科学计算中的作用．事实上，正是由于计算机的出现，才使得科学计算的作用变得越发重要起来．

用计算机做科学计算时，绝大部分情况下都能得到所需要的计算结果，只在少数情况下可能会出错．为杜绝这种错误，在用计算机做科学计算时对可能出现的问题要有所警惕并注意采用合适的方法，我们就能少犯或不犯错误．怎样使计算机的计算结果可信和尽量减少出错的情况，在科学计算中是非常重要的，因为错误的计算结果会产生错误的结论或否定原来正确的数学模型，这会给科研工作带来很大的损失．

现在很多科学研究和工程问题的解决都是借助计算机进行的．通常**用计算机解决实际问题有四个步骤：**

①建立数学模型；

②选择数值方法；

③编写程序；

④上机计算。

建立数学模型是对实际问题进行分析后，根据其内在规律作出简化假设，并运用适当的数学工具将其变成一个数学问题，其表现形式可能是一个方程组、一个函数极小化、一个积分计算式、一个微分方程及它们的不同组合等．建立数学模型需要一定的专业知识．

选择数值方法是为已建立的数学模型选择合适的一个或几个数值计算方法，以用于编程和计算．这里要考虑的问题是所选择的方法能否达到要求的精度、方法的计算量是否太大、程序能否实现及方法对数据的微小扰动反应是否敏感等，这些问题正是数值分析所研究和讨论的．了解如何处理这些问题，就可以最大限度地保证计算机的求解少犯错误．

编写程序是根据所选的数值计算方法，用计算机语言写出源程序．科学计算中常用的计算机语言是 Fortran、C 和 Basic 语言等，当然现在有一些数学软件包可以用来做科学计算，如 Matlab 和 Mathematica 等，这些数学软件由于具有编程简单灵活的特点，也常用于编程中，但对较为复杂的大型问题，由于要考虑计算效率和灵活多样性，用计算机语言编写源程序还是不可缺少的．

上机计算是将已知的原始数据输入计算机，计算机按程序指令进行计算，以得到所需的结果．若结果不满意，应检查所选数值方法是否合理或编程是否考虑不周；若还不行，应分析对应的数学模型的特点及合理性，再选择新的算法．程序调通后，通常可先用一些其他数据检验程序的质量和对错，以确保上机计算的结果可信．

实际问题的解决有定性和定量之分，通常定量比定性的更具说服力．例如，某公司投资一个项目能否赚钱是定性问题，而能赚多少钱是定量问题．显然，在目前竞争激烈的环境中，公司应更关心定量问题．定量的结果一般是要经过科学计算得出的，在数学模型正确的前提下，计算结果的正确性取决于合适数值方法的选择，这再次说明科学计算方法的重要性．

1.2 计算机中的数系与运算特点

1.2.1 计算机的数系

数学计算是建立在数据的体系之中的．要在计算机上进行数学计算，就要了解计算机的数系和计算机是如何进行数字运算的，这有助于帮助构造和分析各种适用于计算机有效计算的数值方法．在科学计算中，数字计算主要是实数之间的运算．数学理论告诉我们：实数集是稠密的无限集，其中任何一个非零实数可表示为

$$x = \pm 10^c \times 0.a_1a_2a_3\cdots \tag{1-2}$$

其中，$a_i \in \{0,1,2,\cdots,9\}$，$c$ 为整数．式（1-2）表示的数 x 称为十进制浮点数．类似地，数学上可以方便地定义 β 进制的浮点数

$$x = \pm \beta^c \times 0.a_1a_2a_3\cdots \quad , a_i \in \{0,1,2,\cdots,\beta-1\}$$

在计算机中，由于机器本身的限制，数学中的实数被表示为

$$x = \pm \beta^c \times 0.a_1a_2a_3\cdots a_t \quad , a_i \in \{0,1,2,\cdots,\beta-1\} \tag{1-3}$$

其中，t 是正整数，表示计算机的**字长**；c 是整数，满足 $L \leqslant c \leqslant U$，$L$ 和 U 为固定整数．对不同的计算机，t、L 和 U 是不同的，而 β 一般取为 2，8，10 和 16．式（1-3）表示的数 x 称为 t 位 β 进制浮点数，其中 c 称为**阶码**，$0.a_1a_2a_3\cdots a_t$ 称为**尾数**，这样一些数的全体

$$f(\beta,t,L,U) = \{\pm \beta^c \times 0.a_1a_2a_3\cdots a_t \mid a_i \in \{0,1,\cdots,\beta-1\}, L \leqslant c \leqslant U\}$$

称为**机器数系**，它是计算机进行实数运算所使用的数系．

显然，**机器数系是有限的离散集**．该数集从总体看，在实数轴上分布不是均匀的，但从局部看，阶数相同的数又等距地分布在实数轴的某一段上．机器数系中有绝对值最大和最小的非零数 M 和 m，例如在 4 位十进制浮点数系 $F(10,4,-99,99)$ 中，绝对值最大的非零数 $M = \pm 10^{99} \times 0.9999$，绝对值最小的非零数 $m = \pm 10^{-99} \times 0.0001$．

若一个非零实数的绝对值大于 M，则计算机产生**上溢错误**，若其绝对值小于 m，则计算机产生**下溢错误**．上溢时，计算机中断程序处理；下溢时，计算机将此数用零表示并继续执行程序．无论是上溢，还是下溢，都称为**溢出错误**．

通常，计算机把尾数为 0 且阶数最小的数表示数零．

1.2.2 计算机对数的接收与计算处理

实数集合是稠密无限的，而计算机使用的数系是有限的．为建立数学中的实数集和计算机中机器数系的对应，计算机采用下面方式进行这种对应．

设非零实数 x 是计算机接收的实数，则计算机对其的处理为

①若 $x \in F(\beta,t,L,U)$，则原样接收 x；

②若 $x \notin F(\beta,t,L,U)$　但 $m \leqslant |x| \leqslant M$，则用 $f(\beta,t,L,U)$ 中最接近 x 的数 $fl(x)$ 表示

并记录 x，以便后面处理．

计算机是怎样做数学运算的呢？首先要说明的是计算机本质上只能做加减乘除运算．两个数在计算机中参与运算的方式为

①加减法：先对阶，后运算，再舍入；

②乘除法：先运算，再舍入．

计算机的运算一般是在计算机中央处理器的运算器中进行，而运算器中一般是多倍字长存储的，故计算机参加运算的数据允许有超出原机器数系字长的数出现．

例如，某计算机的数系 $F(10,4,-99,99)$ 的两个数 $x_1=0.2337\times10^{-1}$ 和 $x_2=0.3364\times10^{2}$，则运算过程如下

$$\begin{aligned} fl(x_1+x_2) &= fl(0.2337\times10^{-1}+0.3364\times10^{2}) \\ &\overset{\text{对阶}}{=} fl(0.0002337\times10^{2}+0.3364\times10^{2}) \\ &\overset{\text{运算}}{=} fl(0.3366337\times10^{2}) \\ &\overset{\text{舍入}}{=} 0.3366\times10^{2} \end{aligned}$$

$$\begin{aligned} fl(x_1\cdot x_2) &= fl(0.2337\times10^{-1}\times0.3364\times10^{2}) \\ &\overset{\text{运算}}{=} fl(0.7861668\times10^{0}) \\ &\overset{\text{舍入}}{=} 0.7862\times10^{0}=0.7862 \end{aligned}$$

由于上述计算机接收和运算数据的特点，常使得数学上很准确完美的公式在计算机上编程计算时，却得不到正确的结果！与理论数学计算特点不同，数值分析的误差分析方法可以构造出专门适用于计算机这种计算特点的有效科学计算技术．

在计算机的数系 $F(\beta,t,L,U)$ 中，把尾数的第一位 a_1 不为零的数称为**规格化的浮点数**．规格化浮点数表示一个实数，具有表示形式唯一和精度高的特点，但并不是 $F(\beta,t,L,U)$ 中每个数都可以用规格化浮点数表示，例如数 $0.00\cdots1\times\beta^{L}$，就不能表示为规格化浮点数．

1.3 误　　差

科学计算的实质是用近似的数据，通过近似有效地计算去获得可用的结果．其强调的是计算结果的可用性，而不是准确性，因为科学计算是以解决实际问题为目的的．科学计算保证计算结果可用性的依据是对计算误差控制．误差是科学计算经常要考虑的问题之一，很多经计算机运算得出的不正确结果，多半是由误差引起的．

准确值与近似值的差异就是误差．误差无处不在，但并不可怕．在解决实际问题中，只要能对误差进行合适的处理和控制，就可以有效地解决实际问题．

1.3.1 误差的来源

误差可以来自很多场合，但其来源主要有 4 个方面：**模型误差**（也称描述误差）、**观测**

误差（也称数据误差）、**截断误差**（也称方法误差）、**舍入误差**（也称计算误差）.

模型误差是在建立数学模型时，由于忽略了一些次要因素而产生的误差．它是数学建模阶段要考虑的误差，不是数值分析可以解决的．

观测误差是在采集原始数据时，由仪器的精度或其他客观因素产生的误差．它也不是数值分析能解决的问题．

截断误差是对参与计算的数学公式作简化处理后所产生的误差，例如要计算函数 e^x 在某点的值，由于 e^x 的展开式 $e^x=1+x+\frac{x^2}{2!}+\cdots+\frac{x^n}{n!}+\cdots$ 为无穷项相加，而计算机不能做无限运算，因此要用 e^x 的展开式计算 e^x 在某点的值是不行的，但可用其展开式的前 $n+1$ 项代替无穷项来近似计算 e^x 在某点的值，即取近似公式 $e^x\approx 1+x+\frac{x^2}{2!}+\cdots+\frac{x^n}{n!}$ 去计算 e^x 在某点的值，这样产生的误差就是截断误差．由于科学计算经常要把一些数学函数变成计算机易于处理的形式，这样总会产生截断误差，因此截断误差是数值分析主要研究的误差，很多好的数值计算方法都是巧妙地处理截断误差得出的．

舍入误差是计算机因数系有限，在接收和运算数据时引起的误差．它也是数值分析关注的内容．本章例 1－1 的错误就是舍入误差造成的．

1.3.2 误差的定义

定义 1－1 设 x 是准确值，x^* 是 x 的一个近似值，称差 x^*-x 为近似值 x^* 的**绝对误差**，简称**误差**，记为 e^* 或 $e(x^*)$，即 $e(x^*)=x^*-x$.

由于准确值 x 通常是未知的，故误差 e^* 一般是计算不出来的，为此引入误差限来对误差进行估计．

定义 1－2 称满足 $|e^*|=|x^*-x|\leqslant\varepsilon^*$ 的正数 ε^* 为近似值 x^* 的**误差限**.

误差限一般是可以求出的，例如用具有毫米刻度的皮尺去测量某个物件的长度，测得的数据与物件的实际长度不会超过半个毫米．这半个毫米就是该物件长度的误差限．误差限给出了准确值 x 所在的范围为 $x^*-\varepsilon^*\leqslant x\leqslant x^*+\varepsilon^*$，该范围常用 $x=x^*\pm\varepsilon^*$ 表示．

误差限是误差绝对值的上界，显然误差限是不唯一的．由于合适的最小的误差限一般求不出来，实用中可以根据问题的特点选择一个合适的误差限即可．通常误差限一般取为获得该近似数仪器精度的半个单位．显然误差限 ε^* 越小，近似值越精确．

绝对误差不能反映近似值 x^* 的近似程度．例如某个量的准确值 $x_1=9$，其近似值 $x_1^*=10$；另一个量的准确值 $x_2=999$，其近似值 $x_2^*=1000$，这两个量的绝对误差都是 1，但显然 x_2^* 的近似程度比 x_1^* 好．为描述近似数的近似程度，需引入如下相对误差概念．

定义 1-3 设 x 是准确值，x^* 是 x 的一个近似值，称 $\dfrac{e^*}{x}=\dfrac{x^*-x}{x}$ 为近似值 x^* 的**相对误差**，记为 e_r^* 或 $e_r(x^*)$，即

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

相对误差的绝对值越小，近似程度越高．例如，前面例子近似值 x_1^* 和 x_2^* 的相对误差分别为 $e_r(x_1^*)=1/9$ 和 $e_r(x_2^*)=1/999$，因为 $|e_r(x_1^*)|\geqslant|e_r(x_2^*)|$，所以 x_2^* 比 x_1^* 逼近程度好．

同样，由于准确数 x 通常是未知的，导致相对误差不可计算，为此引入估计相对误差的相对误差限概念．

定义 1-4 称满足 $|e_r^*|=\left|\dfrac{x^*-x}{x}\right|\leqslant\varepsilon_r^*$ 的正数 ε_r^* 为近似数 x^* 的**相对误差限**．

相对误差限不如绝对误差限容易得到，实用中常借助绝对误差限来估计．为方便估计相对误差限，实用中常用 $\dfrac{e^*}{x^*}=\dfrac{x^*-x}{x^*}$ 代替 $e_r(x^*)=\dfrac{e^*}{x}=\dfrac{x^*-x}{x}$ 进行误差限的计算，这样就得到实用中容易计算的相对误差限 $\varepsilon_r^*=\dfrac{\varepsilon^*}{|x^*|}$．显然，该值越小，近似程度越高．

相对误差在科学计算中的重要性比绝对误差大．

1.3.3 数值计算的误差

计算机中的数值运算本质上是加、减、乘、除四则运算，带有误差的数经过四则运算后误差的变化有如下估计关系．

定理 1-1 假设 x^* 和 y^* 分别是准确值 x 和 y 的一个近似值，则有

(1) 四则运算的绝对误差估计：

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

$$e(x^*y^*)\approx y^*e(x^*)+x^*e(y^*)$$

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

(2) 四则运算的相对误差估计：

$$e_r(x^*\pm y^*)\approx\frac{x^*e_r(x^*)\pm y^*e_r(y^*)}{x^*\pm y^*}$$

$$e_r(x^*y^*)\approx e_r(x^*)+e_r(y^*)$$

$$e_r\left(\frac{x^*}{y^*}\right)\approx e_r(x^*)-e_r(y^*)$$

证明 只证估计式 $e(x^*y^*)\approx y^*e(x^*)+x^*e(y^*)$，其余类似可证之.

由定义有

$$\begin{aligned}e(x^*y^*)&=x^*y^*-xy=x^*y^*-xy^*+xy^*-xy\\&=y^*(x^*-x)+x(y^*-y)=y^*e(x^*)+xe(y^*)\\&\approx y^*e(x^*)+x^*e(y^*)\qquad(\text{因为 } x\approx x^*)\end{aligned}$$

注意到微分的实质是变量的增量，而误差实际上就是一种增量，因此可以借助微分概念和理论来描述有关误差的结果. 实际上，注意到准确数 x 与其近似数 x^* 通常很接近，其差可认为是较小的增量，这样就可以把它们的差看作微分，即有 $e(x^*)=x^*-x=\mathrm{d}x$，对于相对误差，由微分理论可得 $e_r(x^*)=\dfrac{x^*-x}{x}=\dfrac{\mathrm{d}x}{x}=\mathrm{d}\ln x$. 这样得出近似数 x 的绝对误差和相对误差与微分的关系

$$\mathrm{d}x=e(x^*)$$
$$\mathrm{d}\ln x=e_r(x^*)$$

利用这两个关系式和微分理论，可以方便地得到很多关于近似数四则运算的绝对误差和相对误差的估计式.

虽然用微分关系描述四则运算的误差是粗略的，但所得出的结论基本上与严格用定义推导相同. 例如：

由 $\mathrm{d}(x\pm y)=\mathrm{d}x\pm\mathrm{d}y$，可得 $e(x^*\pm y^*)=e(x^*)\pm e(y^*)$；

由 $\mathrm{d}\ln(xy)=\mathrm{d}\ln x+\mathrm{d}\ln y$，可得 $e_r(x^*\cdot y^*)=e_r(x^*)+e_r(y^*)$.

【例 1-2】 考查函数 $y=x^n$ 的相对误差与自变量 x 的相对误差关系.

解 因为 $\ln y=n\ln x$，所以 $\mathrm{d}\ln y=n\mathrm{d}\ln x$，得

$$e_r\left(\left(x^*\right)^n\right)=ne_r(x^*)$$

由此可知函数 x^n 的相对误差为自变量 x 的相对误差的 n 倍，即 n 个数相乘后其结果是自变量的相对误差扩大 n 倍.

科学计算中，经常涉及函数计算，当自变量产生误差时，对应的函数值一定也产生误差，而一般的函数很多不是自变量的四则运算表示的. 处理一般函数计算的误差问题常用 Taylor 展式进行估计.

定理 1-2 设多元函数 $u=f(x_1,x_2,\cdots,x_n)$，向量自变量 $(x_1,x_2,\cdots,x_n)$ 的近似值为 $(x_1^*,x_2^*,\cdots,x_n^*)$，则有多元函数 $f(x_1,x_2,\cdots,x_n)$ 的误差估计

$$(1)\ e(f(x_1^*,x_2^*,\cdots,x_n^*))\approx\sum_{i=1}^{n}\frac{\partial f(x_1^*,x_2^*,\cdots,x_n^*)}{\partial x_i}e(x_i^*)$$

$$(2)\ \varepsilon(f(x_1^*,x_2^*,\cdots,x_n^*))\approx\sum_{i=1}^{n}\left|\frac{\partial f(x_1^*,x_2^*,\cdots,x_n^*)}{\partial x_i}\right|\varepsilon(x_i^*)$$

$$(3)\ \varepsilon_r(f(x_1^*,x_2^*,\cdots,x_n^*))\approx\sum_{i=1}^{n}\left|\frac{\partial f(x_1^*,x_2^*,\cdots,x_n^*)}{\partial x_i}\right|\frac{\varepsilon(x_i^*)}{|f(x_1^*,x_2^*,\cdots,x_n^*)|}$$

证明 利用 Taylor 展式有

$$e(u) = f(x_1^*, x_2^*, \cdots, x_n^*) - f(x_1, x_2, \cdots, x_n)$$

$$\approx \sum_{i=1}^{n} \frac{\partial f(x_1^*, x_2^*, \cdots, x_n^*)}{\partial x_i}(x_i^* - x_i) = \sum_{i=1}^{n} \frac{\partial f(x_1^*, x_2^*, \cdots, x_n^*)}{\partial x_i} e(x_i^*)$$

然后利用绝对误差、绝对误差限和相对误差限定义可得后面的结果.

【例 1-3】 设有一长方体水池，测得其长、宽、深分别为 50m±0.01m，25m±0.01m，20m±0.01m，试按所给数据求出该水池的容积，并给出绝对误差限和相对误差限.

解 令 L，W，H 分别代表长方体水池的长、宽、深；V 代表长方体水池的容积，有

$$V = V(L, W, H) = LWH$$

由题意有长方体水池的长、宽、深的近似值为

$$L^* = 50\text{m}, W^* = 25\text{m}, H^* = 20\text{m}, \varepsilon(L^*) = \varepsilon(W^*) = \varepsilon(H^*) = 0.01\text{m}$$

按所给数据求出该水池的容积为

$$V^* = V(L^*, W^*, H^*) = L^* W^* H^* = 50 \times 25 \times 20 = 2500(\text{m}^3)$$

因为

$$e(V^*) \approx \frac{\partial V^*}{\partial L} e(L^*) + \frac{\partial V^*}{\partial W} e(W^*) + \frac{\partial V^*}{\partial H} e(H^*)$$

$$\approx W^* H^* e(L^*) + L^* H^* e(W^*) + L^* W^* e(H^*)$$

所以

$$\varepsilon(V^*) \approx \frac{\partial V^*}{\partial L} \varepsilon(L^*) + \frac{\partial V^*}{\partial W} \varepsilon(W^*) + \frac{\partial V^*}{\partial H} \varepsilon(H^*)$$

$$\approx W^* H^* \varepsilon(L^*) + L^* H^* \varepsilon(W^*) + L^* W^* \varepsilon(H^*)$$

$$= 25 \times 20 \times 0.01 + 50 \times 20 \times 0.01 + 50 \times 20 \times 0.01$$

$$= 27.50(\text{m}^3)$$

所以

$$\varepsilon_r(V^*) = \frac{\varepsilon(V^*)}{V^*} \approx \frac{27.50}{2500} = 0.11\%$$

故本题绝对误差限和相对误差限为 27.50 m^3 和 0.11%.

1.3.4 计算机的舍入误差

设计算机的数系为 $F(\beta, t, L, U)$，m 及 M 是其中绝对值最小及最大的正数，某数 $x = \pm\beta^c \times 0.a_1a_2\cdots$，$a_1 \neq 0$ 满足 $m < |x| < M$，$x \notin F(\beta, t, L, U)$，则计算机经舍入处理后以数 $fl(x)$ 接收，即 $fl(x) = \pm\beta^c \times \bar{a}$，

$$\bar{a} = \begin{cases} 0.a_1a_2\cdots a_t, & 0 \leqslant a_{t+1} < \beta/2 \\ 0.a_1a_2\cdots a_t + \beta^{-t}, & a_{t+1} \geqslant \beta/2 \end{cases}$$

因此计算机对 x 的舍入绝对误差和舍入相对误差有如下估计