TEACHING MATERIALS FOR COLLEGE STUDENTS

高等学校教材

数值计算方法

第二版

李维国　同登科　主编

中国石油大学出版社

TEACHING MATERIALS FOR COLLEGE STUDENTS

高 等 学 校 教 材

数值计算方法

第二版

李维国　同登科　主编

中国石油大学出版社

图书在版编目(CIP)数据

数值计算方法/李维国主编. —2版. —东营:中国石油大学出版社,2009.1
ISBN 978-7-5636-2710-3

Ⅰ. 数… Ⅱ. 李… Ⅲ. 数值计算—计算方法 Ⅳ. O241

中国版本图书馆 CIP 数据核字(2009)第 009767 号

中国石油大学(华东)规划教材

书　　名:数值计算方法(第二版)
作　　者:李维国　同登科

责任编辑:刘　洋(电话 0546—8392860)
封面设计:九天设计

出 版 者:中国石油大学出版社(山东 东营　邮编 257061)
网　　址:http://www.uppbook.com.cn
电子信箱:shiyoujiaoyu@126.com
印 刷 者:东营市新华印刷厂
发 行 者:中国石油大学出版社(电话 0546—8392791,8392563)
开　　本:180×235　印张:19.625　字数:393 千字
版　　次:2009 年 1 月第 2 版第 1 次印刷
定　　价:28.00 元

第二版前言

本书在2004年第一版的基础上，根据几年来的使用情况，改正了部分错误，并对部分内容做了修改.新版《数值计算方法》共分九章，在原第二章中增加了抛物线法的内容，并将其中关于非线性方程组的迭代解法的内容合并到第六章中并重新编写；原第五章前移为第三章并重新编写；原第三章改为第四章，增加了正交多项式的内容；原第四章改为第五章，增加了积分方程数值解的内容；在第七章中增加了曲线多项式拟合的内容，并对部分内容重新编写；原第九章和第十章内容合并为一章，并重新编写.选学内容使用“*”标注.各章所附的数值实验也进行了部分更新，使得本书内容更加科学和流畅.

新版《数值计算方法》的编写工作由李维国和同登科主持，周生田、刘新海和聂立新负责具体实施，王际朝等参与了部分章节的编写.全书最后由李维国统稿.

我们再次感谢南京大学的沈祖和教授，也欢迎使用本书的老师、同学及广大读者对本书内容批评指正.

编　者

2008年10月

第一版前言

几年以来,我系从事计算数学与应用数学教学与科研的教师就筹划编写一本信息与计算科学专业学生使用的较适中的教材.经过我们编写组全体同仁的艰苦努力,今天终于与读者见面了.

由于电子计算机的迅速发展,近年来本专业学生接受计算机知识越来越早,且已有初步的信息处理与计算能力,对一些数值软件有一点了解,然而缺乏系统的数值计算理论和计算方法的设计思想.考虑到21世纪本专业教学改革的特点和对学生在数值计算方面的要求,根据系统性与科学性相结合的原则,结合我们多年的教学实践,并注意到我们在使用国内外同类教材时发现的一些问题,在深度和广度上做恰当的处理,力求得到一本既有理论深度和较系统的专业基础知识,又不使其过于专门化而包罗万象的教材.本教材旨在通过对一些典型问题和典型算法的剖析,使学生循序渐进地掌握本课程的基本理论和分析解决问题的基本思路与技巧.

全书分三部分,共十章.第一部分为数值分析(第一章至第四章);第二部分为数值代数部分(第五章至第八章);第三部分常微分方程数值解法(第九章至第十章).每章还附有数值实验(包括实验目的、问题提出与实验要求),它是本书的又一特色,给教授本课程的教师和学生提供一些训练的素材.全书讲授时数为80～96个学时(也可增加16个学时的实验课).

本书第一章和第八章由李维国博士编写;第二章、第三章和第四章由同登科博士编写;第五章、第六章和第七章由黄炳家副教授编写;第九章、第十章及各章的数值实验由王子亭博士编写.全书最后由李维国统稿.

学习本书所需要的数学基础是微积分和线性代数,以及常微分方程的基本概念.本书还附有一定数量的习题,通过这些习题可以加深对各章

内容的理解，掌握必要的解题技巧．本书可作为理工科大学、高等师范院校信息与计算科学专业或其他相关专业的教材或教学参考用书，或该专业数值逼近与数值代数课程的相应教材，也可供从事科学与工程计算的科技人员参考．

我们特别感谢南京大学的沈祖和教授，他仔细阅读了全书并提出了宝贵意见．我系研究生鲍文娣、陈金海、王磊、阮宗利、石丽娜等也做了部分工作，在此一并表示感谢．我们希望使用本书的老师、同学及广大读者对本书提出批评指正．

编　者

2004 年 3 月

目　录

Contents

第一章　绪论 …… 1
§1.1　误差 …… 1
1.1.1　误差的来源 …… 1
1.1.2　误差分析的基本概念 …… 2
1.1.3　数值算法与算法的数值稳定性 …… 4
§1.2　误差分析的方法与原则 …… 7
§1.3　算法的软件实现与计算机的数系结构 …… 10
习题一 …… 12
数值实验一 …… 13
第二章　非线性方程的数值解法 …… 17
§2.1　二分法 …… 17
§2.2　迭代法 …… 19
2.2.1　不动点迭代法 …… 19
2.2.2　不动点迭代法的一般理论 …… 22
2.2.3　局部收敛性与收敛阶 …… 24
§2.3　迭代收敛的加速方法 …… 27
2.3.1　使用两个迭代值的组合方法 …… 27
2.3.2　斯蒂芬森迭代法 …… 29
§2.4　牛顿迭代法 …… 31
§2.5　弦割法与抛物线法 …… 35
2.5.1　弦割法 …… 35
2.5.2　抛物线法 …… 37
习题二 …… 40
数值实验二 …… 41
第三章　线性方程组的直接解法 …… 45
§3.1　三角形方程组和三角分解 …… 45

3.1.1 三角形方程组的解法 …… 45
3.1.2 高斯变换 …… 47
3.1.3 三角分解的计算 …… 48
3.1.4 其他的三角分解 …… 51
§3.2 选主元三角分解 …… 52
§3.3 平方根法 …… 57
§3.4* 分块三角分解 …… 61
§3.5 向量范数和矩阵范数 …… 63
3.5.1 向量范数 …… 63
3.5.2 矩阵范数 …… 64
§3.6 线性方程组的敏度分析与病态方程组的解法 …… 70
3.6.1 线性方程组的敏度分析 …… 70
3.6.2 病态方程组的解法 …… 73
习题三 …… 74
数值实验三 …… 76
第四章 多项式插值与函数逼近 …… 80
§4.1 插值问题 …… 80
§4.2 代数插值多项式的构造方法 …… 82
4.2.1 拉格朗日插值法 …… 82
4.2.2 牛顿插值法 …… 85
§4.3 埃尔米特插值问题 …… 90
4.3.1 埃尔米特插值多项式的构造 …… 90
4.3.2 埃尔米特插值多项式的存在唯一性以及误差估计 …… 91
4.3.3 带不完全导数的埃尔米特插值多项式举例 …… 92
§4.4 分段插值 …… 93
4.4.1 高次插值的评述 …… 93
4.4.2 分段插值 …… 95
§4.5 三次样条插值函数 …… 99
4.5.1 三次样条插值函数的力学背景 …… 99
4.5.2 三次样条插值函数 …… 99
4.5.3 三次样条插值函数的性质 …… 103
§4.6 函数逼近 …… 105
4.6.1 函数逼近问题 …… 105
4.6.2 最佳平方逼近 …… 107

4.6.3　正交多项式 …… 109
4.6.4　最佳一致逼近 …… 114
4.6.5　最佳一致逼近多项式求法的讨论 …… 119
4.6.6　离散的最佳逼近问题 …… 121
习题四 …… 122
数值实验四 …… 124
第五章　数值积分与数值微分 …… 128
§5.1　数值求积的基本问题 …… 128
5.1.1　引言 …… 128
5.1.2　求积公式的代数精度 …… 129
5.1.3　求积公式的收敛性与稳定性 …… 130
§5.2　牛顿-柯特斯公式 …… 131
5.2.1　插值型求积公式 …… 131
5.2.2　牛顿-柯特斯公式 …… 132
5.2.3　几种低阶求积公式的余项 …… 134
§5.3　复化求积公式 …… 135
5.3.1　复化梯形公式 …… 135
5.3.2　复化辛普森公式 …… 136
5.3.3　自动选取积分步长 …… 139
§5.4　龙贝格求积公式 …… 140
§5.5　高斯求积公式 …… 142
5.5.1　高斯求积问题的提出 …… 142
5.5.2　高斯求积公式 …… 144
§5.6　积分方程的数值解 …… 149
§5.7　数值微分 …… 150
5.7.1　插值型的求导公式 …… 150
5.7.2　用三次样条插值函数求数值导数 …… 152
习题五 …… 153
数值实验五 …… 155
第六章　线性与非线性方程组的迭代解法 …… 158
§6.1　Jacobi 迭代法和 Gauss-Seidel 迭代法 …… 158
6.1.1　Jacobi 迭代法 …… 158
6.1.2　Gauss-Seidel 迭代法 …… 159
§6.2　Jacobi 迭代与 G-S 迭代的收敛性分析 …… 159

6.2.1　收敛的充分必要条件与误差估计 …… 159
6.2.2　收敛速度 …… 168
§6.3　超松弛迭代法 …… 169
6.3.1　超松弛迭代法 …… 169
6.3.2　SOR 迭代法的收敛性 …… 171
6.3.3　最佳松弛因子与迭代法的比较 …… 173
6.3.4　块超松弛迭代法 …… 174
§6.4　共轭梯度法 …… 175
6.4.1　最速下降法 …… 175
6.4.2　共轭梯度法及其基本性质 …… 178
6.4.3　实用共轭梯度法及其收敛性 …… 183
6.4.4*　预处理方法与 Krylov 子空间方法简介 …… 185
§6.5　非线性方程组的迭代解法 …… 188
6.5.1　非线性 Jacobi 迭代、Gauss-Seidel 迭代和 SOR 迭代 …… 189
6.5.2　Newton 迭代法及其改进算法 …… 190
6.5.3*　大范围算法简介 …… 192
习题六 …… 193
数值实验六 …… 195
第七章　曲线拟合与线性最小二乘问题 …… 199
§7.1　线性最小二乘问题 …… 199
7.1.1　问题的引入 …… 199
7.1.2　最小二乘多项式拟合 …… 200
7.1.3　解的存在性、唯一性 …… 201
§7.2　广义逆矩阵与最小二乘解 …… 205
7.2.1　定义与表示 …… 205
7.2.2　基本性质 …… 208
§7.3　正交化方法 …… 209
7.3.1　Gram-Schmidt 正交化方法 …… 209
7.3.2　正交分解和线性方程组的最小二乘解 …… 213
7.3.3　Householder 变换与 Givens 变换 …… 217
§7.4*　奇异值分解 …… 223
习题七 …… 225
数值实验七 …… 226
第八章　特征值问题的计算方法 …… 231

§8.1　基本概念与性质 …… 231
§8.2　幂法与反幂法 …… 233
§8.3　Jacobi 方法 …… 238
8.3.1　经典 Jacobi 方法 …… 238
8.3.2　循环 Jacobi 方法及其变形 …… 241
§8.4　***QR*** 方法 …… 242
8.4.1　基本迭代与收敛性 …… 243
8.4.2　实 Schur 标准形 …… 244
8.4.3　上 Hessenberg 化 …… 245
8.4.4　三对角化 …… 248
8.4.5　隐式对称 ***QR*** 迭代 …… 249
8.4.6　隐式对称 ***QR*** 算法 …… 250
§8.5*　二分法 …… 251
习题八 …… 256
数值实验八 …… 258
第九章　常微分方程数值解法 …… 262
§9.1　引言 …… 262
§9.2　Euler 方法 …… 263
9.2.1　Euler 方法及其稳定性 …… 263
9.2.2　局部误差和方法的阶 …… 266
9.2.3　Euler 方法的误差分析 …… 267
§9.3　Runge-Kutta 方法 …… 269
9.3.1　Runge-Kutta 方法的基本思想 …… 269
9.3.2　显式 Runge-Kutta 方法及其稳定性 …… 270
9.3.3　隐式 Runge-Kutta 方法 …… 277
§9.4　线性多步法与预估-校正格式 …… 279
§9.5　理论分析 …… 282
9.5.1　单步法的收敛性 …… 282
9.5.2　稳定性 …… 283
9.5.3　收敛性 …… 284
§9.6　方程组与高阶方程的数值方法 …… 285
§9.7　刚性方程组 …… 286
§9.8　边值问题 …… 289
9.8.1　问题提出 …… 289

9.8.2 打靶法 …… 290
习题九 …… 292
数值实验九 …… 293
参考文献 …… 296
名词索引 …… 298

第一章　绪　论

数值计算方法(numerical computational method)又称数值分析(numerical analysis),主要研究适合计算机求解的各种数学问题的近似解法及其理论,其内容包括数值逼近、数值代数、常微分与偏微分方程的数值解等.

自 1946 年第一台电子计算机问世以来,经过半个多世纪的发展,计算机对科学技术的冲击极其深远,其主要原因当然是计算机已经并将继续极大地扩展问题的可解范围,这也使得科学与工程计算成为 20 世纪最重要的科学进展之一.大型科学与工程计算是现代科学、工程和技术发展的重要组成部分,特别是国防、能源、航天和气象等技术型密集行业更是如此.当今高度复杂的科学与工程问题的求解只有很小一部分能够用解析的方法解决,其他大部分都要通过物理实验、数值计算来揭示其内在的规律.数值计算包括了从适当的计算机结构设计出发,对相应的数学模型进行合理的算法设计,然后进行充分的数值实验、分析直至研制出相应的数值计算软件.另外,由于现代科学与工程计算的复杂性,数值实验还能代替某些物理实验无法做到的事情.如今数值计算已与理论研究及物理实验三足鼎立,并列成为当今世界科学活动的三种主要方式(图 1.0.1).为众多的科学与工程问题提供计算方法,提高计算的可靠性、有效性和精确性,便是数值计算方法这门课程的主要研究内容.

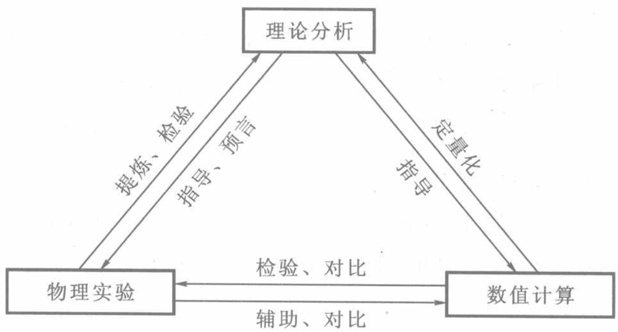

图 1.0.1

§1.1　误　差

1.1.1　误差的来源

用计算机解决科学与工程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. 数学模型与实际问题之间出现的这种误差称为模型误差. 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，这些参量显然也包含误差，这种由观测产生的误差称为观测误差. 以上两种误差不属于本书的讨论范围.

当数学模型得不到精确解时，通常要用数值方法求它的近似解，或者说用数值算法模拟数学模型，此时产生的误差称为方法误差或截断误差. 例如，求 $I=\int_0^1 \frac{\sin x}{x}\mathrm{d}x$ 的解，采用所谓梯形公式 $\hat{I}=(\sin 1+1)/2$ 来近似，这时 $\hat{I}$ 与 I 之间的误差称为梯形公式求解定积分 I 的截断误差.

有了求解数学模型或数学问题的算法以后，用计算机进行数值计算时，由于计算机的字长有限，原始数据在计算机上表示时就会产生误差，计算过程又可能会产生新的误差，这种用计算机模拟或实现算法的误差称为舍入误差. 例如，用 5 位有效数字计算 $\hat{I}$ 得到

$$\hat{I}\approx(0.841\ 47+1.000\ 0)/2.000\ 0\approx 0.920\ 74=\tilde{\hat{I}},$$

$\hat{I}$ 与 $\tilde{\hat{I}}$ 之间的误差就是舍入误差.

截断误差与舍入误差是用数值方法求解数学问题产生的误差，是数值计算中主要讨论的误差.

1.1.2 误差分析的基本概念

本书除了研究数学问题的算法外，还要研究计算解(数值解)与真解(精确解)相差多少，这就是计算的精度问题，需要对误差做出估计和分析.

• **定义 1.1.1** 设 x 为真值(精确值)，x^* 为 x 的一个近似值. 称 $e=x^*-x$ 为近似值 x^* 的绝对误差，简称误差.

显然，误差 e 可正可负，且常常是无限位的，有时不能也没有必要求得它的一个精确结果，而只需知道它的绝对值的一个上界 ε，这个上界称为绝对误差限，简称误差限. 如取 $\pi^*=3.141\ 59$，则

$$|\pi^*-\pi|\leqslant\frac{1}{2}\times 0.000\ 01,$$

即 $\varepsilon=0.000\ 005$，或 $3.141\ 585\leqslant\pi\leqslant 3.141\ 595$，有时记做 $\pi=3.141\ 59\pm 0.000\ 005$.

误差的大小还不能完全表示近似值的好坏. 若 x 的近似值 x^* 是由 x 按“四舍五入”规则得到的，则

$$|x^*-x|\leqslant\frac{1}{2}\alpha,$$

这里 α 是 x^* 的最后一位的一个单位.

譬如测量一段路程,其长为 1 000 km,有 20 m 误差;另外测量一条 400 m 的跑道,也有 20 m 误差,容易想到后者的精度不如前者,这就是相对误差的概念.

• **定义 1.1.2**　近似值 x^* 的误差 e 与真值 x 的比值

$$\frac{e}{x}=\frac{x^*-x}{x}$$

称为近似值 x^* 的相对误差,记做 e_r. 相对误差可正可负,它的绝对值的上界称为相对误差限,记做 ε_r,$\varepsilon_r=\varepsilon/x$. 在实际计算时,由于真值常常是未知的,通常取

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

作为 x^* 的相对误差,条件是 $e_r=\frac{e}{x^*}$ 较小,此时

$$\frac{e}{x}-\frac{e}{x^*}=\frac{e(x^*-x)}{x^*x}=\frac{e^2}{x^*(x^*-e)}=\frac{(e/x^*)^2}{1-e/x^*}. \tag{1.1.1}$$

测量 1 000 km 的路程有 20 m 误差,其相对误差为 $20/10^6=2\times10^{-5}$;而测量 400 m 的跑道有 20 m 误差,其相对误差为 $20/400=5\%$,由此可见前者的相对误差比后者小得多.

在误差分析中,还常常用到有效数字的概念.

• **定义 1.1.3**　若近似值 x^* 与真值的误差不超过某一位的半个单位,该位到 x^* 的第一位非零数字共有 n 位,则称 x^* 有 n 位有效数字.

例 1.1.1

① 取 $x^*=3.14$ 作为 π 的近似值,x^* 就有 3 位有效数字.

② 取 $x^*=3.141\,592$ 作为 π 的近似值,x^* 就有 6 位有效数字,而 2 不是有效数字.

③ 取 $x^*=3.141\,593$ 作为 π 的近似值,x^* 就有 7 位有效数字.

④ 取 $x^*=0.030\,141\,60$ 作为 $x=0.030\,141\,594$ 的近似值,x^* 就有 7 位有效数字.

用数学的语言描述,即若 x 的近似值写做

$$x^*=\pm10^m\times(a_1\times10^{-1}+a_2\times10^{-2}+\cdots+a_n\times10^{-n}+\cdots+a_k\times10^{-k}+\cdots), \tag{1.1.2}$$

其中 m 是整数,$a_1\neq0$,$a_1,a_2,\cdots,a_k$ 是 0 到 9 中的一个数字,如果 $|x^*-x|\leqslant\frac{1}{2}\times10^{m-n}$,则 x^* 至少具有 n 位有效数字,即 $a_1,a_2,\cdots,a_n$ 为有效数字,而 $a_{n+1},\cdots,a_k,\cdots$ 不一定是有效数字.

若 x^* 的每一位都是有效数字,那么称 x^* 为有效数. 显然,若 x^* 由 x 经"四舍五入"而来,则 x^* 是有效数. 有效数字与相对误差之间有密切关系,如定理 1.1.1 所示,证明过程参见参考文献[1].

定理 1.1.1 将 x 的近似值 x^* 表示成式(1.1.2),若 a_k 是有效数字,那么相对误差不超过$\frac{1}{2a_1}\times 10^{-(k-1)}$;反之,如果已知相对误差 r,且有 $|r|\leqslant\frac{1}{2(a_1+1)}\times 10^{-(k-1)}$,那么 a_k 必为有效数字.

1.1.3 数值算法与算法的数值稳定性

数值问题是指输入数据(即问题中的自变量与原始数据)与输出数据(结果)之间函数关系的一个确定而无歧义的描述.输入、输出数据可用有限维向量表示.根据这种定义,“数学问题”不一定是“数值问题”,但它往往可用“数值问题”来逼近.例如,解常微分方程$\frac{\mathrm{d}y}{\mathrm{d}x}=x^2+y^2$,$y(0)=1$,它不是数值问题,因为输出不是数据而是连续函数 $y=y(x)$,但只要规定输出数据是 $y(x)$在 $x=h,2h,\cdots,nh$ 处的近似值,这就是一个数值问题,可用 Euler 折线法或其他数值方法求解,这些数值方法就是算法.

计算的基本单位称为算法元,它由算子、输入元和输出元组成.算子可以是简单操作,如算术运算(+,-,×,÷)或逻辑运算,也可以是宏操作,如向量运算、数据传输、函数求值等.输入元和输出元可分别视为若干变量或向量.由一个或多个算法元组成一个进程,它是算法元的有限序列.一个数值问题的算法是指按规定顺序执行一个或多个完整的进程.通过它们将输入元变成一个输出元.面向计算机的算法可分为串行算法与并行算法两类.只有一个进程的算法适用于串行计算机,称为串行算法;两个与两个以上进程的算法适用于并行计算机,称为并行算法.对于一个给定的数值问题可以有许多不同的算法,它们都可以给出近似答案,但所需计算量和得到的精度可能相差很大.一个面向计算机、计算复杂性好、有可靠理论分析的算法就是一个好算法.计算复杂性包含时间复杂性和空间复杂性两方面,在同一精度下,计算时间少的为时间复杂性好,而占用内存空间少的为空间复杂性好.

例 1.1.2 计算下列多项式的值:

$$p(x)=a_0x^n+\cdots+a_{n-1}x+a_n.$$

解 这是一个数值问题,输入数据为 $a_0,\cdots,a_n$ 及 x,输出数据为 $p(x)$.若直接由 x 算出 $x^2,\cdots,x^n$,再乘以相应的系数 $a_{n-1},a_{n-2},\cdots,a_0$ 并相加,则要做$\frac{n(n+1)}{2}$次乘法和 n 次加法,占用$2n+1$个存储单元.若将 $p(x)$改写为

$$p(x)=[\cdots(a_0x+a_1)x\cdots+a_{n-1}]x+a_n,$$

用递推公式表示为

$$b_0=a_0,\quad b_i=a_i+b_{i-1}x,\quad i=1,2,\cdots,n,\quad b_n=p_n(x),$$

它只用 n 次乘法和 n 次加法,占用 $n+2$ 个存储单元,故这是一个好的串行算法,称为

秦九韶方法，也称为 Horner 算法.

对于大型计算问题，计算复杂性的差别就更大. 例如解线性方程组，当 $n=20$ 时，用 Cramer 法则和行列式定义计算，仅乘除法的运算次数就需约 9.7×10^{20} 次，用每秒运算 1 万亿次的计算机也要算 30 多年，而用 Gauss 消去法只需 2 660 次乘除运算，并且 n 愈大运算次数相差就愈大. 这个例子既表明算法研究的重要性，又说明只提高计算机速度而不改进和选用好的算法也是不行的. 人类的计算能力是计算工具的性能和计算效率的总和，因此，计算能力的提高有赖于这两方面的提高. 例如，1955 年至 1975 年的 20 年间，计算机速度提高了数千倍，而同一时间解决一定规模的椭圆型偏微分方程的计算方法效率提高了约 100 万倍，这说明对于提高计算速度，研究和选择好的算法在某种意义上说比提高计算机速度更重要，因为算法研究所需代价要小得多. 当然，选择好的算法的前提是保证计算结果的可靠性，这就要求有可靠的理论分析，使计算结果满足精度要求. 一个算法是否可靠与舍入误差是否增长密切相关.

• **定义 1.1.4**　一个算法如果输入数据有扰动(即误差)，而计算过程中舍入误差不增长，则称此算法是数值稳定的，否则就称此算法是数值不稳定的.

例 1.1.3　对 $n=0,1,\cdots,8$，计算积分 $\int_0^1\frac{x^n}{x+5}\mathrm{d}x$.

解　由于

$$y_n+5y_{n-1}=\int_0^1\frac{x^n+5x^{n-1}}{x+5}\mathrm{d}x=\frac{1}{n},$$

于是可得到计算积分 y_n 的递推公式

$$y_n=\frac{1}{n}-5y_{n-1},\quad n=1,2,\cdots,8,\tag{1.1.3}$$

其中

$$y_0=\int_0^1\frac{1}{x+5}\mathrm{d}x=\ln(x+5)\Big|_0^1=\ln\frac{6}{5}\approx0.182=\tilde{y}_0.$$

利用式(1.1.3)计算 y_n，计算取到小数点后 3 位，由于初值 y_0 用 $\tilde{y}_0$ 近似，实际计算结果为

$$\tilde{y}_n=\frac{1}{n}-5\,\tilde{y}_{n-1},\quad n=1,2,\cdots,8.\tag{1.1.4}$$

若用其他方法求 y_n 的精确解(精确到小数点后 3 位)，计算结果见表 1.1.1.

表 1.1.1

n	y_n	$\tilde{y}_n$	$\bar{y}_n$
0	0.182 322	0.182	0.182
1	0.088 392	0.090	0.088
2	0.058 039	0.050	0.058