

# 数值计算方法和算法

张韵华 奚梅成 陈效群 编 著

科学出版社  
营销宣传

2002

## 内 容 简 介

本书介绍各种常用的数值计算方法,简述计算方法的计算对象、计算原理和计算步骤,给出部分数值方法的算法描述,并附有一些用 C 语言编写的方法的程序和解题实例,以及符号计算语言 Mathematica 做计算方法题目的函数和实例。

本书选材适中,例题丰富,便于自学,以 \* 标记有难度的内容以便取舍,适合于不同层次的读者。本书可作为普通高校本科生和计算机专科生学习计算方法的教材,也可作为工程技术人员的参考资料。

### 图书在版编目 (CIP) 数据

数值计算方法和算法/张韵华等编著. -北京:科学出版社, 2000.1  
ISBN 7-03-007377-0

I. 数… II. 张… III. ①数值计算-计算方法②算法理论 IV. 0241

中国版本图书馆 CIP 数据核字 (1999) 第 65149 号

科学出版社  
营销宣传

科学出版社 出版

北京东黄城根北街 16 号

邮政编码: 100717

印刷

科学出版社发行 各地新华书店经销

\*

2000 年 1 月第 一 版	开本: 787×1092 1/16
2002 年 1 月第三次印刷	印张: 12 1/2
印数: 6 101—11 100	字数: 289 000

定价: 18.00 元

(如有印装质量问题, 我社负责调换〈环伟〉)

# 前 言

随着现代科学技术的发展和计算机的广泛使用，数值计算方法不仅要面对数学工作者、数值计算专家，还要更多地面对一般的工程技术人员和各行各业的设计人员。

为了顾及一般读者，本书力求通俗易懂、简洁实用。其内容按插值、数值微分和积分、曲线拟合、非线性方程求根、解线性方程组、计算特征值和特征向量、常微分方程数值解的顺序安排。第9章给出调用 Mathematica 软件直接做数值题目的部分样例。全书约需 40 学时。本书介绍的计算方法都有相对的独立性，可以根据不同的教学对象和要求选择其中的某些章、节和知识点，书中以 \* 标记略有难度的内容以供选用。

本书以能正确选择计算对象的计算方法为前提，领会计算原理和掌握计算步骤为主干线，淡化数学定理证明中的严谨性部分，强化数值方法与计算机技术的应用能力训练，为此取书名为“数值计算方法和算法”。希望读者通过本书的学习掌握数值计算中的基本思想和方法，培养自行处理常规数值计算问题的能力，为深入学习数值方法打好基础，也为部分读者调用各类程序包解决问题创造条件。

本书 1999 年 9 月出版，此次重印主要做了修订和勘误工作；在内容上增加了少量的在应用软件中的实用算法，例如：用牛顿插值构造埃尔米特插值的方法；汇集了部分上机作业题，供学生上机实习时选用。

本书是作者在中国科学技术大学多年讲授计算方法课程的基础上编写而成的，可作为一般理工科（非数学系和计算机系）以及工商科专业的计算方法教材，也可作为工程技术人员的参考用书。

本书的插图和大部分程序由中国科学技术大学数学系陈长松博士完成，部分程序和例题由窦斗硕士完成。在此表示感谢。

编者还要向使用本教材的教师和学生表示深切的谢意，感谢他们对本书提出的修订意见。最后，感谢科学出版社和本书责任编辑对出版本书所做的工作。

编 者

2001 年 11 月

# 目 录

第 0 章 绪论 .....	( 1 )
0.1 数值计算方法与算法 .....	( 1 )
0.2 误差与有效数字 .....	( 1 )
0.3 约束误差 .....	( 3 )
0.4 范数 .....	( 3 )
0.4.1 向量范数 .....	( 3 )
0.4.2 矩阵范数 .....	( 5 )
第 1 章 插值 .....	( 9 )
1.1 插值 .....	( 9 )
1.2 拉格朗日(Lagrange)插值 .....	( 9 )
1.2.1 线性插值 .....	( 10 )
1.2.2 二次插值 .....	( 11 )
1.2.3 $n$ 次拉格朗日插值多项式 .....	( 13 )
1.3 牛顿(Newton)插值 .....	( 16 )
1.3.1 差商及其计算 .....	( 17 )
1.3.2 牛顿插值 .....	( 18 )
1.4* 埃尔米特(Hermite)插值 .....	( 21 )
1.5 分段插值 .....	( 25 )
1.5.1 龙格(Runge)现象 .....	( 25 )
1.5.2 分段线性插值 .....	( 26 )
1.6 三次样条函数 .....	( 28 )
1.6.1 三次样条插值的 $M$ 关系式 .....	( 28 )
1.6.2 三次样条插值的 $m$ 关系式 .....	( 30 )
1.7 程序示例 .....	( 31 )
习题 1 .....	( 36 )
第 2 章 数值微分和数值积分 .....	( 38 )
2.1 数值微分 .....	( 38 )
2.1.1 差商与数值微分 .....	( 38 )
2.1.2 插值型数值微分 .....	( 40 )
2.1.3 样条插值数值微分 .....	( 41 )
2.2 数值积分 .....	( 42 )
2.2.1 插值型数值积分 .....	( 42 )
2.2.2 牛顿-柯特斯(Newton-Cote's)积分 .....	( 43 )
2.3 复化数值积分 .....	( 46 )
2.3.1 复化梯形积分 .....	( 46 )

2.3.2 复化辛普森积分 .....	( 48 )
2.3.3 复化积分的自动控制误差算法 .....	( 49 )
2.3.4 龙贝格(Romberg)积分 .....	( 52 )
2.4 重积分计算 .....	( 53 )
2.5* 高斯(Gauss)型积分公式介绍 .....	( 56 )
2.6 程序示例 .....	( 58 )
习题 2 .....	( 60 )
<b>第 3 章 曲线拟合的最小二乘法 .....</b>	<b>( 63 )</b>
3.1 拟合曲线 .....	( 63 )
3.2 线性拟合和二次拟合函数 .....	( 64 )
3.3 解矛盾方程组 .....	( 67 )
3.4 程序示例 .....	( 73 )
习题 3 .....	( 77 )
<b>第 4 章 非线性方程求根 .....</b>	<b>( 78 )</b>
4.1 实根的对分法 .....	( 78 )
4.2 迭代法 .....	( 79 )
4.3 牛顿迭代法 .....	( 81 )
4.4 弦截法 .....	( 84 )
4.5 非线性方程组的牛顿方法 .....	( 85 )
4.6 程序示例 .....	( 88 )
习题 4 .....	( 90 )
<b>第 5 章 解线性方程组的直接法 .....</b>	<b>( 91 )</b>
5.1 消元法 .....	( 92 )
5.1.1 三角形方程组的解 .....	( 92 )
5.1.2 高斯消元法与列主元消元法 .....	( 93 )
5.1.3 高斯-若尔当(Gauss-Jordan)消元法 .....	( 98 )
5.2 直接分解法 .....	( 99 )
5.2.1 多利特尔分解 .....	( 101 )
5.2.2 库朗分解 .....	( 105 )
5.2.3 追赶法 .....	( 107 )
5.2.4 对称矩阵的 $LDL^T$ 分解 .....	( 108 )
5.3* 矩阵的条件数 .....	( 110 )
5.4 程序示例 .....	( 111 )
习题 5 .....	( 119 )
<b>第 6 章 解线性方程组的迭代法 .....</b>	<b>( 121 )</b>
6.1 雅可比迭代 .....	( 122 )
6.1.1 雅可比迭代格式 .....	( 122 )
6.1.2 雅可比迭代收敛条件 .....	( 124 )
6.2 高斯-塞德尔(Gauss-Seidel)迭代 .....	( 125 )

6.3 松弛迭代 .....	( 128 )
6.4 逆矩阵计算 .....	( 129 )
6.5 程序示例 .....	( 130 )
习题 6 .....	( 135 )
<b>第 7 章 计算矩阵的特征值和特征向量 .....</b>	<b>( 137 )</b>
7.1 幂法 .....	( 137 )
7.1.1 幂法运算 .....	( 137 )
7.1.2 幂法的规范运算 .....	( 140 )
7.1.3* 关于幂法的初始值 .....	( 142 )
7.2 反幂法 .....	( 142 )
7.3 实对称矩阵的雅可比方法 .....	( 143 )
7.4 程序示例 .....	( 148 )
习题 7 .....	( 152 )
<b>第 8 章 常微分方程数值解 .....</b>	<b>( 153 )</b>
8.1 欧拉(Euler)公式 .....	( 153 )
8.1.1 基于差商的欧拉公式 .....	( 153 )
8.1.2* 欧拉公式的收敛性 .....	( 156 )
8.1.3 基于数值积分的差分方法 .....	( 157 )
8.2 龙格-库塔方法 .....	( 159 )
8.2.1 二阶龙格-库塔方法 .....	( 159 )
8.2.2 四阶龙格-库塔公式 .....	( 161 )
8.2.3 步长的自适应 .....	( 162 )
8.3 线性多步法 .....	( 163 )
8.4 常微分方程组的数值解法 .....	( 166 )
8.4.1 一阶常微分方程组的数值解法 .....	( 166 )
8.4.2 高阶常微分方程数值方法 .....	( 168 )
8.5* 常微分方程的稳定性 .....	( 169 )
8.6 程序示例 .....	( 171 )
习题 8 .....	( 174 )
<b>第 9 章* 在 Mathematica 中做题 .....</b>	<b>( 175 )</b>
9.1 符号计算系统 Mathematica 基本操作 .....	( 175 )
9.2 插值 .....	( 178 )
9.3 数值积分 .....	( 179 )
9.4 曲线拟合 .....	( 180 )
9.5 非线性方程 .....	( 181 )
9.6 方程组求解 .....	( 182 )
9.7 计算特征值和特征向量 .....	( 184 )
9.8 常微分方程数值解 .....	( 185 )
<b>上机作业题 .....</b>	<b>( 188 )</b>
<b>参考文献 .....</b>	<b>( 192 )</b>

# 第 1 章 插 值

## 1.1 插 值

在实际问题中,有时只能给出函数  $f(x)$  在平面上的一些离散点的值  $\{(x_i, f(x_i))\}$ ,  $i=0, 1, \dots, n$ , 而不能给出  $f(x)$  的具体解析表达式, 或者  $f(x)$  的表达式过于复杂而难于运算。这时我们需要用近似函数  $\varphi(x)$  来逼近函数  $f(x)$ , 在数学上常用的函数逼近的方法有:

- 插值。
- 一致逼近。
- 均方逼近或称最小二乘法。

本章讨论用插值逼近函数的方法。

什么是插值? 简单地讲, 用给定的未知函数  $f(x)$  的若干函数值的点构造  $f(x)$  的近似函数  $\varphi(x)$ , 要求  $\varphi(x)$  与  $f(x)$  在给定点的函数值相等, 则称函数  $\varphi(x)$  为插值函数。例如: 在服装店订做风衣时, 选择好风衣的样式后, 服装师量出并记下你的胸围、衣长和袖长等几个尺寸, 这几个尺寸就是风衣函数的插值点数值, 在衣料上画出的裁剪线就是服装师构造的插值函数  $\varphi(x)$ , 裁剪水平的差别就在于量准插值点和构造合乎身材的插值函数。

**定义:**  $f(x)$  为定义在区间  $[a, b]$  上的函数,  $x_0, x_1, \dots, x_n$  为  $[a, b]$  上  $n+1$  个互不相同的点,  $\Phi$  为给定的某一函数类。若  $\Phi$  上有函数  $\varphi(x)$ , 满足

$$\varphi(x_i) = f(x_i), i = 0, 1, \dots, n$$

则称  $\varphi(x)$  为  $f(x)$  关于节点  $x_0, x_1, \dots, x_n$  在  $\Phi$  上的插值函数; 称点  $x_0, x_1, \dots, x_n$  为插值节点; 称  $\{(x_i, f(x_i))\}, i=0, 1, \dots, n$  为插值型值点, 简称型值点或插值点;  $f(x)$  称为被插函数。

这样, 对函数  $f(x)$  在区间  $[a, b]$  上的各种计算, 就用对插值函数  $\varphi(x)$  的计算取而代之。

构造插值函数需要关心下列问题:

- 插值函数是否存在?
- 插值函数是否惟一?
- 如何表示插值函数?
- 如何估计被插函数与插值函数的误差?

## 1.2 拉格朗日(Lagrange)插值

可对插值函数  $\varphi(x)$  选择多种不同的函数类型, 由于代数多项式具有简单和一些良好的特性, 例如, 多项式是无穷光滑的, 容易计算它的导数和积分, 故常选用代数多项式作

为插值函数。

### 1.2.1 线性插值

**问题 1.1:** 给定两个插值点  $(x_0, y_0), (x_1, y_1)$ , 其中  $x_0 \neq x_1$ , 怎样做通过这两点的一次插值函数?

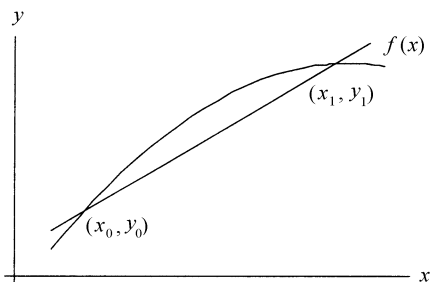


图 1.1 线性插值函数  $L_1(x)$

过两点作一条直线, 这条直线就是通过这两点的一次多项式插值函数, 简称线性插值。如图 1.1 所示。

在初等数学中, 可用两点式、点斜式或截距式构造通过两点的一条直线。

下面先用待定系数法构造插值直线。

设直线方程为  $L_1(x) = a_0 + a_1 x$ , 将  $(x_0, y_0), (x_1, y_1)$  分别代入直线方程  $L_1(x)$  得:

$$\begin{cases} a_0 + a_1 x_0 = y_0 \\ a_0 + a_1 x_1 = y_1 \end{cases}$$

当  $x_0 \neq x_1$  时, 因  $\begin{vmatrix} 1 & x_0 \\ 1 & x_1 \end{vmatrix} \neq 0$ , 所以方程组有解, 而且解是惟一的。这也表明, 平面上两个点, 有且仅有一条直线通过。用待定系数法构造插值多项式的方法简单直观, 容易看到解的存在性和惟一性, 但要解一个方程组才能得到插值函数的系数, 因工作量较大和不便向高阶推广, 故这种构造方法通常不宜采用。

当  $x_0 \neq x_1$  时, 若用两点式表示这条直线, 则有:

$$L_1(x) = \frac{x - x_1}{x_0 - x_1} y_0 + \frac{x - x_0}{x_1 - x_0} y_1 \quad (1.1)$$

这种型式称为拉格朗日插值多项式。

记  $l_0(x) = \frac{x - x_1}{x_0 - x_1}$ ,  $l_1(x) = \frac{x - x_0}{x_1 - x_0}$ ,  $l_0(x)$ ,  $l_1(x)$  称为插值基函数, 计算  $l_0(x)$ ,  $l_1(x)$  的值, 易见

$$l_i(x_j) = \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \quad (1.2)$$

在拉格朗日插值多项式中可将  $L_1(x)$  看做两条直线  $\frac{x - x_1}{x_0 - x_1} y_0, \frac{x - x_0}{x_1 - x_0} y_1$  的叠加, 并可看到两个插值点的作用和地位都是平等的。

拉格朗日插值多项式型式免了解方程组的计算, 易于向高次插值多项式型式推广。  
**线性插值误差**

**定理 1:** 记  $L_1(x)$  为以  $(x_0, y_0), (x_1, y_1)$  为插值点的插值函数,  $x_0, x_1 \in [a, b]$ ,  $x_0 \neq x_1$ 。这里  $y_0 = f(x_0)$ ,  $y_1 = f(x_1)$ , 设  $f(x)$  一阶连续可导,  $f''(x)$  在  $(a, b)$  上存在, 则对任意给定的  $x \in [a, b]$ , 至少存在一点  $\zeta \in [a, b]$ , 使

$$R(x) = f(x) - L_1(x) = \frac{f''(\zeta)}{2!} (x - x_0)(x - x_1), \zeta \in [a, b] \quad (1.3)$$



证明:令  $R(x)=f(x)-L_1(x)$ , 因  $R(x_0)=R(x_1)=0$ ,  $x_0, x_1$  是  $R(x)$  的根, 所以可设

$$R(x)=k(x)(x-x_0)(x-x_1)$$

对任何一个固定的点  $x$ , 引进辅助函数  $\Psi(t)$ :

$$\Psi(t)=f(t)-L_1(t)-k(x)(t-x_0)(t-x_1)$$

则  $\Psi(x_i)=0, i=0,1$ 。

由定义可得  $\Psi(x)=0$ , 这样  $\Psi(t)$  至少有 3 个零点, 不失一般性, 假定  $x_0 < x < x_1$ , 分别在  $[x_0, x]$  和  $[x, x_1]$  上应用洛尔定理, 可知  $\Psi'(t)$  在每个区间至少存在一个零点, 不妨记为  $\zeta_1$  和  $\zeta_2$ , 即  $\Psi'(\zeta_1)=0$  和  $\Psi'(\zeta_2)=0$ , 对  $\Psi'(t)$  在  $[\zeta_1, \zeta_2]$  上应用洛尔定理, 得到  $\Psi''(t)$  在  $[\zeta_1, \zeta_2]$  上至少有一个零点  $\zeta$ ,  $\Psi''(\zeta)=0$ 。

现在对  $\Psi(t)$  求二次导数, 其中  $L''_1(t)=0$  ( $L_1(t)$  为  $t$  的线性函数), 故有

$$\Psi''(t)=f''(t)-2!k(x)$$

代入  $\zeta$ , 得

$$f''(\zeta)-2!k(x)=0$$

所以

$$k(x)=f''(\zeta)/2!$$

即

$$R(x)=\frac{f''(\zeta)}{2!}(x-x_0)(x-x_1), \zeta \in [a, b]$$

### 1.2.2 二次插值

**问题 1.2:** 给定三个插值点  $(x_i, f(x_i)), i=0,1,2$ , 其中  $x_i$  互不相等, 怎样构造函数  $f(x)$  的二次(抛物线)插值多项式?

平面上的三个点能确定一条二次曲线, 如图 1.2 所示。

仿造线性插值的拉格朗日插值, 即用插值基函数的方法构造插值多项式。设  $L_2(x)=l_0(x)f(x_0)+l_1(x)f(x_1)+l_2(x)f(x_2)$ , 每个基函数  $l_i(x)$  是一个二次函数, 对  $l_0(x)$  来说, 要求  $x_1, x_2$  是它的零点, 因此可设

$$l_0(x)=A(x-x_1)(x-x_2)$$

同理  $l_1(x), l_2(x)$  也有相对应的形式, 得

$$L_2(x)=A(x-x_1)(x-x_2)f(x_0)+B(x-x_0)(x-x_2)f(x_1)+C(x-x_0)(x-x_1)f(x_2)$$

将  $x=x_0$  代入  $L_2(x)$ , 得

$$L_2(x_0)=A(x_0-x_1)(x_0-x_2)f(x_0)=f(x_0)$$

$$\therefore A=\frac{1}{(x_0-x_1)(x_0-x_2)}$$

$$l_0(x)=A(x-x_1)(x-x_2)=\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}$$

同理将  $x=x_1, x=x_2$  代入  $L_2(x)$  得到  $B$  和  $C$  的值, 以及  $l_1(x)$  和  $l_2(x)$  的表达式。

科学出版社  
营销宣传

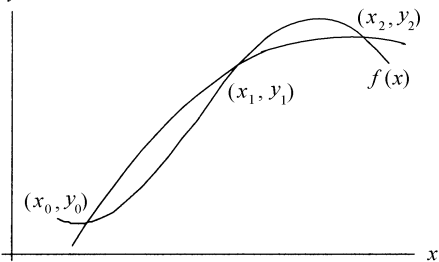


图 1.2 二次(抛物线)插值函数  $L_2(x)$

$$B = \frac{1}{(x_1 - x_0)(x_1 - x_2)}$$

$$C = \frac{1}{(x_2 - x_0)(x_2 - x_1)}$$

$$l_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)}$$

$$l_2(x) = \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}$$

$$\begin{aligned} \therefore L_2(x) &= \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)}f(x_0) + \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)}f(x_1) \\ &\quad + \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}f(x_2) \\ &= \sum_{i=0}^2 l_i(x)f(x_i) \end{aligned} \quad (1.4)$$

也容易验证:

$$l_0(x_0) = 1, \quad l_0(x_1) = 0, \quad l_0(x_2) = 0$$

$$l_1(x_0) = 0, \quad l_1(x_1) = 1, \quad l_1(x_2) = 0$$

$$l_2(x_0) = 0, \quad l_2(x_1) = 0, \quad l_2(x_2) = 1$$

插值基函数仍然满足:

$$l_i(x_j) = \delta_{ij} = \begin{cases} 1, & i=j \\ 0, & i \neq j \end{cases}$$

## 二次插值函数误差

$$R_2(x) = \frac{f^{(3)}(\xi)}{3!}(x - x_0)(x - x_1)(x - x_2), \xi \in [\min\{x_0, x_1, x_2, x\}, \max\{x_0, x_1, x_2, x\}] \quad (1.5)$$

上式证明完全类似于线性插值误差的证明,故省略。

插值作为函数逼近方法,常用来作函数的近似计算。当计算点落在插值点区间之内时叫做内插,否则叫做外插。内插的效果一般优于外插。

**例 1.1:** 给定  $\sin 11^\circ = 0.190809$ ,  $\sin 12^\circ = 0.207912$ 。构造线性插值函数并用插值函数计算  $\sin 11^\circ 30'$  和  $\sin 10^\circ 30'$ 。

解:构造线性插值函数:

$$L_1(x) = \frac{(x - 12)}{11 - 12} 0.190809 + \frac{(x - 11)}{12 - 11} 0.207912$$

分别将  $x = 11.5$ ,  $x = 10.5$  代入上式,得

$$L_1(11.5) = 0.199361, \text{准确值 } \sin 11^\circ 30' = 0.199368$$

$$L_1(10.5) = 0.182258, \text{准确值 } \sin 10^\circ 30' = 0.182236$$

$$R(x) = \frac{f''(\xi)}{2!}(x - x_0)(x - x_1) = \frac{-\sin(\xi)}{2}(x - 11)(x - 12)$$

$$|R(x)| \leq \frac{1}{2} |(11.5 - 11)(11.5 - 12)| = 0.125$$

**例 1.2:** 给定  $\sin 11^\circ = 0.190809$ ,  $\sin 12^\circ = 0.207912$ ,  $\sin 13^\circ = 0.224951$ 。构造二次插值

函数并计算  $\sin 11^\circ 30'$ 。

$$\begin{aligned} \text{解: } L_2(x) &= \frac{(x-12)(x-13)}{(11-12)(11-13)} 0.190809 + \frac{(x-11)(x-13)}{(12-11)(12-13)} 0.207912 \\ &\quad + \frac{(x-11)(x-12)}{(13-11)(13-12)} 0.224951 \\ L_2(11.5) &= 0.199369, \text{准确值 } \sin 11^\circ 30' = 0.199368 \end{aligned}$$

**例 1.3:** 要制做三角函数  $\sin x$  的函数值表, 已知表值有四位小数, 要求用线性插值引起的截断误差不超过表值的舍入误差, 试决定其最大允许步长。

**解:** 设最大允许步长  $h = h_i = x_i - x_{i-1}$

$$\begin{aligned} |R(x)| &= \left| \frac{f''(\xi)}{2!} (x - x_{i-1})(x - x_i) \right| = \left| \frac{\sin \xi}{2} (x - x_{i-1})(x - x_i) \right| \\ &\leq \frac{1}{2} |(x - x_{i-1})(x - x_i)| \leq \frac{1}{2} \left| \left[ \frac{x_{i-1} + x_i}{2} - x_{i-1} \right] \left[ \frac{x_{i-1} + x_i}{2} - x_i \right] \right| \\ &= \frac{1}{8} |(x_{i-1} - x_i)(x_i - x_{i-1})| = \frac{h^2}{8} < \frac{1}{2} \cdot 10^{-4} \\ h &\leq 0.02 \end{aligned}$$

### 1.2.3 $n$ 次拉格朗日插值多项式

**问题 1.3:** 给定平面上两个互不相同的插值点  $(x_i, f(x_i))$ ,  $i=0, 1$ , 有且仅有一条通过这两点的直线; 给定平面上三个互不相同的插值点  $(x_i, f(x_i))$ ,  $i=0, 1, 2$ , 有且仅有一条通过这三个点的二次曲线; 给定平面上  $n+1$  个互不相同的插值点  $(x_i, f(x_i))$ ,  $i=0, 1, 2, \dots, n$ , 互不相同是指  $x_i$  互不相等, 是否有且仅有一条不高于  $n$  次的插值多项式曲线, 如果曲线存在, 那么如何简单地作出这条  $n$  次插值多项式曲线?

**分析:**  $n$  次多项式  $P_n(x) = a_0 + a_1 x + \dots + a_n x^n$ , 它完全由  $n+1$  个系数  $a_0, a_1, \dots, a_n$  决定。若曲线  $P_n(x)$  通过给定平面上  $n+1$  个互不相同的插值点  $(x_i, f(x_i))$ ,  $i=0, 1, 2, \dots, n$ , 则  $P_n(x)$  应满足  $P_n(x_i) = f(x_i)$ ,  $i=0, 1, 2, \dots, n$ , 事实上一个插值点就是一个插值条件。

将  $(x_i, f(x_i))$ ,  $i=0, 1, 2, \dots, n$  依次代入  $P_n(x)$  中得到线性方程组:

$$\begin{cases} a_0 + a_1 x_0 + a_2 x_0^2 + \dots + a_n x_0^n = f(x_0) \\ a_0 + a_1 x_1 + a_2 x_1^2 + \dots + a_n x_1^n = f(x_1) \\ \dots \\ a_0 + a_1 x_n + a_2 x_n^2 + \dots + a_n x_n^n = f(x_n) \end{cases} \quad (1.6)$$

方程组的系数行列式是范德蒙(Vandermonde)行列式:

$$V(x_0, x_1, \dots, x_n) = \begin{vmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n \\ 1 & x_1 & x_1^2 & \dots & x_1^n \\ \dots & \dots & \dots & \dots & \dots \\ 1 & x_n & x_n^2 & \dots & x_n^n \end{vmatrix} = \prod_{0 \leq j < i \leq n} (x_i - x_j)$$

当  $x_i$  互异时,  $\prod_{0 \leq j < i \leq n} (x_i - x_j) \neq 0$ , 所以方程组(1.6)的解存在且惟一。即问题 1.3 的解存在而且惟一。

通过求解(1.6)得到插值多项式  $P_n(x)$ ,因其计算量太大而不可取,仿照线性以及二次插值多项式的拉格朗日型式,我们可构造  $n$  次拉格朗日插值多项式。

对于  $n+1$  个互不相同的插值节点  $x_i, i=0,1,2,\cdots, n$ ,由  $n$  次插值多项式的惟一性,可对每个插值节点  $x_i$  作出相应的  $n$  次插值基函数  $l_i(x), i=0,1,2,\cdots, n$ 。

要求  $x_0, x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_n$  是  $l_i(x)$  的零点,因此可设

$$l_i(x)=a_i(x-x_0)(x-x_1)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)$$

由  $l_i(x_i)=1$ ,将  $x=x_i$  代入  $l_i(x)$ ,得到

$$\begin{aligned} l_i(x_i) &= a_i(x_i-x_0)(x_i-x_1)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)=1 \\ l_i(x) &= \frac{(x-x_0)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)}{(x_i-x_0)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)} = \prod_{\substack{0\leqslant j\leqslant n \\ j\neq i}} \frac{x-x_j}{x_i-x_j} \end{aligned} \tag{1.7}$$

作其组合:

$$L_n(x)=\sum_{i=0}^n l_i(x)f(x_i) \tag{1.8}$$

那么  $L_n(x)$  不高于  $n$  次且满足  $L_n(x_i)=f(x_i), i=0,1,2,\cdots, n$ ,故  $L_n(x)$  就是关于插值点  $x_0, x_1, \cdots, x_n$  的插值多项式,这种插值型式称为拉格朗日插值多项式。 $\{l_i(x)\}$  称为关于节点  $\{x_i\}$  的拉格朗日基函数。

**例 1.4:**给出下列插值节点数据,做三次拉格朗日插值多项式,并计算  $f(0.6)$ 。

$x_i$	-2.00	0.00	1.00	2.00
$f(x_i)$	17.00	1.00	2.00	17.00

解:基函数为:

$$\begin{aligned} l_0(x) &= \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} = \frac{(x-0)(x-1.00)(x-2.00)}{(-2.00-0)(-2.00-1.00)(-2.00-2.00)} \\ &= -\frac{1}{24}x(x-1)(x-2) \\ l_1(x) &= \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} = \frac{1}{4}(x+2)(x-1)(x-2) \\ l_2(x) &= \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} = -\frac{1}{3}(x+2)x(x-2) \\ l_3(x) &= \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)} = \frac{1}{8}(x+2)x(x-1) \end{aligned}$$

三次拉格朗日插值多项式:

$$\begin{aligned} L_3(x) &= -\frac{17}{24}x(x-1)(x-2) + \frac{1}{4}(x+2)(x-1)(x-2) - \frac{2}{3}(x+2)x(x-2) \\ &\quad + \frac{17}{8}(x+2)x(x-1) \end{aligned}$$

$$f(0.6)\approx L_3(x)=-0.472$$

**$n$  次插值多项式的误差**

**定理 2:**设  $L_n(x)$  是  $[a, b]$  上过  $(x_i, f(x_i)), x_i\in[a, b], i=0,1,\cdots, n$  的  $n$  次插值多项式,  $x_i$  互不相等,当  $f\in C^{n+1}[a, b]$  时,则插值多项式的误差:

$$R_n(x) = \frac{f^{(n+1)}(\zeta)}{(n+1)!} (x-x_0)(x-x_1)\cdots(x-x_n), \text{ 其中 } \zeta \in [a, b] \quad (1.9)$$

证明\*: 记  $R_n(x) = f(x) - L_n(x)$ 。由于  $L_n(x_i) = f(x_i)$ ,  $i=0, 1, \dots, n$ , 因而  $x_0, x_1, \dots, x_n$  是  $R_n(x)$  的根, 于是可设

$$R_n(x) = K(x)(x-x_0)(x-x_1)\cdots(x-x_n)$$

下面的目标是算出  $K(x)$ , 为此引入变量为  $t$  的函数  $\Psi(t)$ :

$$\Psi(t) = f(t) - L_n(t) - K(x)(t-x_0)(t-x_1)\cdots(t-x_n) \quad (1.10)$$

令  $t = x_i$ , 得  $\Psi(x_i) = 0$ ,  $i=0, 1, \dots, n$

令  $t = x$ , 由定义  $\Psi(x) = f(x) - L_n(x) - K(x)(x-x_0)(x-x_1)\cdots(x-x_n) = 0$   
即  $\Psi(t)$  至少有  $n+2$  个零点, 由于  $f \in C^{n+1}[a, b]$ , 由洛尔定理,  $\Psi'(t)$  在  $\Psi(t)$  相邻的两个零点之间至少有一个零点, 即  $\Psi'(t)$  至少有  $n+1$  个零点。同理再对  $\Psi''(t)$  应用洛尔定理, 即  $\Psi''(t)$  至少有  $n$  个零点, 反复应用洛尔定理得到  $\Psi^{(n+1)}(t)$  至少有一个零点  $\zeta$ 。

另一方面, 对  $\Psi(t)$  求  $n+1$  阶导数, 有

$$\Psi^{(n+1)}(t) = f^{(n+1)}(t) - K(x)(n+1)!$$

令  $t = \zeta$ , 有

$$0 = \Psi^{(n+1)}(\zeta) = f^{(n+1)}(\zeta) - K(x)(n+1)!$$

$$\text{得到 } K(x) = \frac{f^{(n+1)}(\zeta)}{(n+1)!}$$

$$R_n(x) = \frac{f^{(n+1)}(\zeta)}{(n+1)!} (x-x_0)(x-x_1)\cdots(x-x_n), \zeta \in [a, b] \quad (1.11)$$

由于  $\Psi^{(n+1)}(t)$  的零点  $\zeta$  与  $\Psi(t)$  的零点  $x, x_0, \dots, x_n$  有关, 因而  $\zeta$  为  $x$  的函数。

若  $|f^{(n+1)}(x)| \leq M, x \in [a, b]$ , 则  $R_n(x)$  可表示为

$$|R_n(x)| \leq \frac{M}{(n+1)!} \prod_{i=0}^n |x - x_i| \quad (1.12)$$

由(1.11)式可以看到, 当  $f(x)$  是不高于  $n$  次的多项式时,  $R_n(x) = 0$ , 即  $L_n(x) = f(x)$ 。

对于函数  $f(x) = x^k, k=0, 1, \dots, n$ , 关于节点  $x_0, x_1, \dots, x_n$  的拉格朗日插值多项式就是其本身, 故拉格朗日基函数  $\{l_i(x)\}$  满足

$$L_n(x) = \sum_{i=0}^n l_i(x) x_i^k = x^k, k=0, 1, \dots, n$$

$$\text{令 } k=0, \text{ 得到 } \sum_{i=0}^n l_i(x) \equiv 1.$$

定理 2 给出了当被插函数充分光滑时的插值误差或称插值余项表达式, 但是, 在实际计算中, 并不知道  $f(x)$  的具体表示, 难以得到  $f^{(n+1)}(x)$  的形式或较精确的界限  $M$ , 因此也难以得到界  $|R_n(x)|$ 。在实际计算中, 可对误差运用下面的事后估计方法。

给出  $n+2$  个插值节点  $x_0, x_1, \dots, x_{n+1}$ , 任选其中的  $n+1$  个插值节点, 不妨取  $x_i, i=0, 1, \dots, n$ , 构造一个  $n$  次插值多项式, 记为  $L_n(x)$ 。在  $n+2$  个插值节点中另选  $n+1$  个插值点, 不妨取  $x_i, i=1, 2, \dots, n+1$ , 构造一个  $n$  次插值多项式, 记为  $\widetilde{L}_n(x)$ 。由定理

2,可得到

$$f(x) - L_n(x) = \frac{f^{(n+1)}(\xi_1)}{(n+1)!} (x - x_0)(x - x_1) \cdots (x - x_n) \quad (1.13)$$

$$f(x) - \widetilde{L}_n(x) = \frac{f^{(n+1)}(\xi_2)}{(n+1)!} (x - x_1)(x - x_2) \cdots (x - x_{n+1}) \quad (1.14)$$

设  $f^{(n+1)}(x)$  在插值区间内连续而且变化不大,有  $f^{(n+1)}(\xi_1) \approx f^{(n+1)}(\xi_2)$ , 则

$$\frac{f(x) - L_n(x)}{\widetilde{L}_n(x)} \approx \frac{x - x_0}{x - x_{n+1}}$$

从而可得到

$$f(x) \approx \frac{x - x_{n+1}}{x_0 - x_{n+1}} L_n(x) + \frac{x - x_0}{x_{n+1} - x_0} \widetilde{L}_n(x) \quad (1.15)$$

$$f(x) - L_n(x) \approx \frac{x - x_0}{x_0 - x_{n+1}} (L_n(x) - \widetilde{L}_n(x)) \quad (1.16)$$

## 拉格朗日插值多项式的算法

下面用伪码描述拉格朗日插值多项式的算法。

1: 输入: 插值节点控制数  $n$ , 插值点序列  $(x_i, y_i), i=0, 1, \cdots, n$ , 要计算的函数点  $x$ 。

2: FOR  $i := 0, 1, \cdots, n$  !  $i$  控制拉格朗日基函数序列

{tmp:=1;

2.1 FOR  $j := 0, 1, \cdots, i-1, i+1, \cdots, n$

{ ! 对于给定  $x$ , 计算拉格朗日基函数  $l_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x - x_j}{x_i - x_j}$

tmp:=tmp \*  $(x - x_j) / (x_i - x_j)$ ;

} ! tmp 表示拉格朗日基函数  $l_i(x)$

2.2  $fx := fx + tmp * y_i$

}

3: 输出  $L_n(x)$  的计算结果  $fx$ 。

伪码又称过程设计语言,主要用于描述算法。它是某种高级语言和自然语言的混杂语言,它取某种高级语言中的一些关键字,用于描述算法的结构化构造和数据说明等。伪码的语句中嵌有自然语言的叙述,伪码易于理解和修改,也易于转化为程序代码,是一种使用频率较高的写算法的语言。

在伪码中,惊叹号“!”表示注释语句。

## 1.3 牛顿(Newton)插值

拉格朗日插值多项式的优点是格式整齐和规范,它的缺点是计算量大且没有承袭性质,当需要增加插值节点时,不得不重新计算所有插值基函数  $l_i(x)$ 。本节给出具有承袭性质的牛顿插值多项式,并首先介绍在牛顿插值中需要用到的差商计算。

### 1.3.1 差商及其计算

#### 一阶差商

称函数值的差  $f(x_1) - f(x_0)$  与自变量的差  $x_1 - x_0$  之比为  $f(x)$  关于点  $x_0, x_1$  的一阶差商, 并记为  $f[x_0, x_1]$ , 即

$$f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

而称  $f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0}$  为  $f(x)$  关于点  $x_0, x_1, x_2$  的二阶差商。

函数关于  $x_0$  的零阶差商即为函数在  $x_0$  的函数值,  $f[x_0] = f(x_0)$ 。

#### $k$ 阶差商

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

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

差商有很多性质, 我们仅列举其中的两条。

**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。

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

用归纳法可以证明这一性质。

例如:

$$\begin{aligned} f[x_0, x_1, x_2] &= \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} \\ &= \frac{f(x_0)}{(x_0 - x_1)(x_0 - x_2)} + \frac{f(x_1)}{(x_1 - x_0)(x_1 - x_2)} + \frac{f(x_2)}{(x_2 - x_0)(x_2 - x_1)} \end{aligned}$$

**性质 2:** 若  $i_0, i_1, \dots, i_k$  为  $0, 1, \dots, k$  的任一排列, 则

$$f[x_0, x_1, \dots, x_k] = f[x_{i_0}, x_{i_1}, \dots, x_{i_k}]$$

该性质表明差商的值只与节点有关而与节点的顺序无关, 即差商对节点具有对称性, 这一性质由性质 1 可直接推出。

#### 差商的计算

按照差商定义, 用两个  $k-1$  阶差商的值计算  $k$  阶差商, 通常用差商表的形式计算和存放(见表 1.1)。

由于差商对节点具有对称性, 可以任意选择两个  $k-1$  阶差商的值计算  $k$  阶差商。例如:

$$f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} = \frac{f[x_0, x_2] - f[x_0, x_1]}{x_2 - x_1}$$

表 1.1 差商表

$i$	$x_i$	$f(x_i)$	一阶差商	二阶差商	三阶差商	$n$ 阶差商
0	$x_0$	$f(x_0)$				
1	$x_1$	$f(x_1)$	$f[x_0, x_1]$			
2	$x_2$	$f(x_2)$	$f[x_1, x_2]$	$f[x_0, x_1, x_2]$		
3	$x_3$	$f(x_3)$	$f[x_2, x_3]$	$f[x_1, x_2, x_3]$	$f[x_0, x_1, x_2, x_3]$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
$n$	$x_n$	$f(x_n)$	$f[x_{n-1}, x_n]$	$f[x_{n-2}, x_{n-1}, x_n]$	$f[x_{n-3}, \dots, x_n]$	$f[x_0, x_1, \dots, x_n]$

例 1.5: 计算  $(-2, 17)$ ,  $(0, 1)$ ,  $(1, 2)$ ,  $(2, 19)$  的一至三阶差商。

$i$	$x_i$	$f(x_i)$	$f[x_{i-1}, x_i]$	$f[x_{i-2}, x_{i-1}, x_i]$	$f[x_{i-3}, x_{i-2}, x_{i-1}, x_i]$
0	-2	17			
1	0	1	-8		
2	1	2	1	3	
3	2	19	17	8	5/4

解:  $f[-2, 0] = (1 - 17) / (0 - (-2)) = -8$   
 $f[0, 1] = (2 - 1) / (1 - 0) = 1$   
 $f[1, 2] = (19 - 2) / (2 - 1) = 17$   
 $f[-2, 0, 1] = (f[0, 1] - f[-2, 0]) / (1 - (-2)) = 3$   
 $f[0, 1, 2] = (f[1, 2] - f[0, 1]) / (2 - 0) = 8$   
 $f[-2, 0, 1, 2] = (f[0, 1, 2] - f[-2, 0, 1]) / (2 - (-2)) = 5/4$

### 1.3.2 牛顿插值

#### 线性插值

问题 1.4: 给定两个插值点  $(x_0, f(x_0))$ ,  $(x_1, f(x_1))$ ,  $x_0 \neq x_1$ , 怎样构造线性牛顿插值函数?

用点斜式构造线性插值函数, 设  $N_1(x) = a_0 + a_1(x - x_0)$ , 将  $x = x_0, x = x_1$  代入得

$$N_1(x_0) = a_0 = f(x_0)$$

$$N_1(x_1) = f(x_0) + a_1(x_1 - x_0) = f(x_1)$$

所以 
$$a_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = f[x_0, x_1]$$

得到线性牛顿插值公式:

$$N_1(x) = f(x_0) + f[x_0, x_1](x - x_0) \quad (1.17)$$

由插值惟一性, 线性牛顿插值函数  $N_1(x)$  与拉格朗日插值函数  $L_1(x)$  为同一个多项式, 仅是表达型式不同而已。



## 二次插值多项式

**问题 1.5:** 给定  $(x_i, f(x_i))$ ,  $i=0, 1, 2$ ,  $x_i$  互不相同, 怎样构造二次牛顿插值多项式? 将二次多项式按幂次逐步递增的方式设置:

$$N_2(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1)$$

由  $N_2(x_0) = f(x_0)$ ,  $N_2(x_1) = f(x_1)$ , 知  $a_0 + a_1(x - x_0)$  就是  $f(x)$  关于  $x_0, x_1$  的线性牛顿插值  $N_1(x)$ , 即

$$N_2(x) = N_1(x) + a_2(x - x_0)(x - x_1)$$

在构造  $N_1(x)$  过程中已经计算出  $a_0, a_1$ 。现将  $x = x_2$  代入上式得

$$N_2(x_2) = f(x_0) + f[x_0, x_1](x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) = f(x_2)$$

整理得

$$\begin{aligned} a_2 &= \left[ \frac{f(x_2) - f(x_0)}{x_2 - x_0} - \frac{f(x_1) - f(x_0)}{x_1 - x_0} \right] / (x_2 - x_1) \\ &= \frac{f[x_0, x_2] - f[x_0, x_1]}{x_2 - x_1} = f[x_0, x_1, x_2] \end{aligned}$$

故二次牛顿插值公式为:

$$N_2(x) = f(x_0) + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) \quad (1.18)$$

## $n$ 次牛顿插值函数

**问题 1.6:** 给定  $(x_i, f(x_i))$ ,  $i=0, 1, \dots, n$ , 其中  $x_i$  互不相同, 怎样构造  $n$  次牛顿插值多项式?

由一阶差商的定义  $f[x, x_0] = \frac{f(x) - f(x_0)}{x - x_0}$ , 得到  $f(x) = f(x_0) + (x - x_0)f[x, x_0]$ ; 类似地, 由二阶差商至  $n$  阶差商的定义得到下列方程:

$$\left\{ \begin{aligned} f(x) &= f(x_0) + (x - x_0)f[x, x_0] \end{aligned} \right. \quad (1)$$

$$\left\{ \begin{aligned} f[x, x_0] &= f[x_0, x_1] + (x - x_1)f[x, x_0, x_1] \end{aligned} \right. \quad (2)$$

$$\left\{ \begin{aligned} f[x, x_0, x_1] &= f[x_0, x_1, x_2] + (x - x_2)f[x, x_0, x_1, x_2] \end{aligned} \right. \quad (3)$$

.....

$$\left\{ \begin{aligned} f[x, x_0, x_1, \dots, x_{n-1}] &= f[x_0, x_1, \dots, x_n] + (x - x_n)f[x, x_0, x_1, \dots, x_n](n+1) \end{aligned} \right.$$

用  $(x - x_0)$  乘(2)式, 用  $(x - x_0)(x - x_1)$  乘(3)式,  $\dots$ ,  $(x - x_0)\dots(x - x_{n-1})$  乘  $(n+1)$  式, 所有等式相加, 得到

$$\begin{aligned} f(x) &= f(x_0) + (x - x_0)f[x_0, x_1] + (x - x_0)(x - x_1)f[x_0, x_1, x_2] + \dots \\ &\quad + (x - x_0)(x - x_1)\dots(x - x_{n-1})f[x_0, x_1, \dots, x_n] \\ &\quad + (x - x_0)(x - x_1)\dots(x - x_n)f[x, x_0, x_1, \dots, x_n] \\ &= N(x) + R(x) \end{aligned}$$

其中

$$\begin{aligned} N(x) &= f(x_0) + (x - x_0)f[x_0, x_1] + \dots \\ &\quad + (x - x_0)(x - x_1)\dots(x - x_{n-1})f[x_0, x_1, \dots, x_n] \end{aligned}$$

为不高于  $n$  次的多项式, 可以验证  $N(x_i) = f(x_i)$ , 称  $N(x)$  是过  $n+1$  个插值点的  $n$  阶

牛顿插值多项式。

$$R(x) = f[x, x_0, x_1, \dots, x_n] \prod_{i=0}^n (x - x_i)$$

为插值多项式的误差。由插值多项式的惟一性得到拉格朗日插值多项式  $L(x)$  与牛顿插值多项式  $N(x)$  是完全相同的, 它们仅是同一插值多项式的不同表达型式而已, 因此得到拉格朗日插值多项式的误差与牛顿插值多项式的误差也完全相等。故当  $f \in C^{n+1}[a, b]$  时, 有

$$\begin{aligned} R(x) &= \frac{f^{(n+1)}(\zeta)}{(n+1)!} \prod_{i=0}^n (x - x_i) = f[x, x_0, \dots, x_n] \prod_{i=0}^n (x - x_i) \\ \frac{f^{(n+1)}(\zeta)}{(n+1)!} &= f[x, x_0, x_1, \dots, x_n] \end{aligned} \tag{1.19}$$

**例 1.6:** 设  $f(x) = 10x^3 - 100x + 1$ , 计算  $f[x_0, x_1, x_2, x_3], f[x_0, x_1, x_2, x_3, x_4]$ 。

解:  $f[x_0, x_1, x_2, x_3] = \frac{f^{(3)}(\zeta)}{3!} = \frac{60}{6} = 10$

$$f[x_0, x_1, x_2, x_3, x_4] = \frac{f^{(4)}(\zeta)}{4!} = \frac{0}{4!} = 0$$

记  $N_n(x) = \sum_{i=0}^n t_i(x) f[x_0, \dots, x_i]$

其中  $t_0(x) \equiv 1$

$$t_i(x) = (x - x_{i-1})(x - x_0)(x - x_1) \cdots (x - x_{i-1}), i = 1, 2, \dots, n$$

也有  $\begin{cases} t_i(x_j) = 0 & j < i \\ t_i(x_j) \neq 0 & j \geq i \end{cases}$

可以看到拉格朗日插值多项式是基函数  $\{l_i(x)\}$  的线性组合, 牛顿插值多项式是基函数  $\{t_i(x)\}$ , 即  $1, x - x_0, (x - x_0)(x - x_1), \dots, (x - x_0) \cdots (x - x_{i-1})$  的线性组合。

牛顿插值多项式的承袭性质表现在

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

对  $k-1$  阶牛顿插值多项式  $N_{k-1}(x)$ , 只需增加一项  $t_k(x) f[x_0, x_1, \dots, x_k]$ , 即可得到  $k$  阶牛顿插值多项式  $N_k(x)$ 。

**例 1.7:** 给定下列插值节点的值, 构造牛顿插值函数, 计算  $N_2(0.9), N_3(0.9)$ 。

$x_i$	-2	0	1	2
$f(x_i)$	17	1	2	19

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

取  $x_0 = -2, x_1 = 0, x_2 = 1$ ; 用例 1.5 中算出的差商代入上式得到二阶牛顿插值多项式:

$$\begin{aligned} N_2(x) &= 17 - 8(x + 2) + 3(x + 2)x \\ \therefore N_2(0.9) &= 17.0 - 8(0.9 + 2) + 3(0.9 + 2) \cdot 0.9 = 1.63 \end{aligned}$$

$$N_3(x) = N_2(x) + f[x_0, x_1, x_2, x_3](x - x_0)(x - x_1)(x - x_2)$$

取  $x_0 = -2, x_1 = 0, x_2 = 1, x_3 = 2$ ; 得到三阶牛顿插值多项式:

$$N_3(x) = 17 - 8(x+2) + 3x(x+2) + 1.25x(x+2)(x-1)$$

$$\therefore N_3(0.9) = N_2(0.9) + 1.25 \cdot 0.9 \cdot (0.9+2)(0.9-1.0) = 1.30375$$

## 牛顿插值的算法

在计算机上实现牛顿算法时,用递推形式易于实现牛顿插值多项式的承袭性质。下面为牛顿插值的算法。

1. 输入: 插值节点数  $n$ , 插值点序列  $\{x_i, f(x_i)\}, i=0, 1, \dots, n$ , 要计算的插值点  $u$ 。

2. 形成差商表  $g[k], k=1, 2, \dots, n$  !  $g[k]$  表示  $f[x_0, x_1, \dots, x_k]$

3. 置初始值  $t=1; \text{newton} = f(x_0)$

4. 对  $i=1, 2, \dots, n$

$t = (u - x_{i-1}) \cdot t$  ! 由  $t_k = (u - x_{k-1}) t_{k-1}$  形成  $(u - x_0) \cdots (u - x_{i-1})$

$\text{newton} = \text{newton} + t \cdot g[i]$

5. 输出  $f(u)$  的插值数值  $N_n(u) = \text{newton}$ 。

从表 1.1 的差商计算中可以看到, 给定  $n+1$  个插值点, 存放所有差商值需要  $n^2$  个存储单元, 而在牛顿插值表达式中只用到  $f[x_0, x_1], f[x_0, x_1, x_2], \dots, f[x_0, x_1, \dots, x_n]$  这些差商值, 在计算中可用一维数组  $g[i], i=0, 1, 2, \dots, n$  存放各阶差商值。

下面对算法的第 2 步“形成差商表  $g[k], k=1, 2, \dots, n$ ”作进一步展开。

(1) 对  $g[i], i=0, 1, 2, \dots, n$  初始化, 即  $g[j] = f(x_j), j=0, 1, 2, \dots, n$ ;

(2) 除了  $g[0](f(x_0))$  以外, 其余函数值  $g[j], j=n, n-1, \dots, 1$  在计算一阶差商后不再使用, 因此可将一阶差商  $f[x_{j-1}, x_j]$  放在  $g[j], j=n, n-1, \dots, 1$  中, 这时仍有  $g[0] = f(x_0)$ ;

(3) 计算二阶差商后也不再调用  $f[x_{j-2}, x_{j-1}]$  即  $g[j-1]$ , 可将  $f[x_{j-2}, x_{j-1}, x_j]$ ,  $j=n, n-1, \dots, 2$  放在  $g[j]$  中, 这时有  $g[j] = f[x_0, x_1, \dots, x_j]$ ;

.....

一直做到  $n$  阶差商  $f[x_0, x_1, \dots, x_n]$ , 这时仍有  $g[n] = f[x_0, x_1, \dots, x_n]$ 。

计算差商算法:

for  $i=0$  to  $n$

$g[i] = f(x_i)$

for  $k=1$  to  $n$  ! 计算  $k$  阶差商

{for  $j=n$  to  $k+1$

$g[j] = (g[j] - g[j-1]) / (x_j - x_{j-k})$

}

详细的 C 语言程序和算例请看 1.7 节。

## 1.4\* 埃尔米特(Hermite)插值

在构造插值时, 如果不仅要求插值多项式节点的函数值与被插函数的函数值相同, 还要求在节点处的插值函数与被插函数的一阶导数的值也相同, 这样的插值称为埃尔米特插值或称密切插值。

常用埃尔米特插值描述如下: 设  $f(x)$  具有一阶连续导数, 以及插值点  $x_i, i=0, 1,$

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

$$H_{2n+1}(x_i) = f(x_i)$$

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

则称  $H_{2n+1}(x)$  为  $f(x)$  关于节点  $\{x_i\}_{i=0}^n$  的埃尔米特插值多项式。

**问题 1.7:** 给定  $f(x_0) = y_0, f(x_1) = y_1, f'(x_0) = m_0, f'(x_1) = m_1, x_0 \neq x_1$ ; 怎样构造给定两个节点的函数值和一阶导数值的埃尔米特插值多项式?

分析: 用 4 个条件, 至多可确定三次多项式。设满足插值条件的三次埃尔米特插值多项式为:

$$H_3(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$$

将插值条件代入  $H_3(x)$  得到线性方程组:

$$\begin{cases} a_0 + a_1 x_0 + a_2 x_0^2 + a_3 x_0^3 = y_0 \\ a_0 + a_1 x_1 + a_2 x_1^2 + a_3 x_1^3 = y_1 \\ a_1 + 2a_2 x_0 + 3a_3 x_0^2 = m_0 \\ a_1 + 2a_2 x_1 + 3a_3 x_1^2 = m_1 \end{cases}$$

因为方程组的系数行列式

$$\begin{vmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 1 & x_1 & x_1^2 & x_1^3 \\ 0 & 1 & 2x_0 & 3x_0^2 \\ 0 & 1 & 2x_1 & 3x_1^2 \end{vmatrix} = -(x_0 - x_1)^4 \neq 0$$

所以方程组有解, 可惟一解出  $a_0, a_1, a_2, a_3$ , 即关于节点  $x_0, x_1$  的埃尔米特插值多项式存在惟一。类似于拉格朗日插值多项式的构造手法, 也可通过插值基函数作出  $H_3(x)$ 。

$$\text{设 } H_3(x) = h_0(x)y_0 + h_1(x)y_1 + g_0(x)m_0 + g_1(x)m_1$$

$$\text{要 } H_3(x_0) = h_0(x_0)y_0 = y_0$$

$$\text{可设 } h_0(x_0) = 1, h_1(x_0) = 0, g_0(x_0) = 0, g_1(x_0) = 0$$

$$\text{同理要 } H_3(x_1) = h_1(x_1)y_1 = y_1$$

$$\text{有 } h_0(x_1) = 0, h_1(x_1) = 1, g_0(x_1) = 0, g_1(x_1) = 0$$

$$\text{要 } H'_3(x_0) = m_0$$

$$\text{有 } h'_0(x_0) = 0, h'_1(x_0) = 0, g'_0(x_0) = 1, g'_1(x_0) = 0$$

$$\text{要 } H'_3(x_1) = m_1$$

$$\text{有 } h'_0(x_1) = 0, h'_1(x_1) = 0, g'_0(x_1) = 0, g'_1(x_1) = 1$$

由上述要求, 对  $h_0(x)$  来说, 至多为三次多项式,  $x_1$  是它的二重根, 可设

$$h_0(x) = (a_0 + b_0 x) \left[ \frac{x - x_1}{x_0 - x_1} \right]^2 = (a_0 + b_0 x) l_0^2(x) \quad (1.20)$$

$$\text{利用 } h_0(x_0) = (a_0 + b_0 x_0) l_0^2(x_0) = (a_0 + b_0 x_0) = 1$$

$$h'_0(x_0) = b_0 l_0^2(x_0) + (a_0 + b_0 x_0) 2 l_0(x_0) l'_0(x_0) = 0$$

$$\text{解出 } h_0(x) = \left[ 1 + 2 \frac{x - x_0}{x_1 - x_0} \right] \left[ \frac{x - x_1}{x_0 - x_1} \right]^2$$

同理可得

$$h_1(x) = \left[ 1 + 2 \frac{x - x_1}{x_0 - x_1} \right] \left[ \frac{x - x_0}{x_1 - x_0} \right]^2$$

由  $g_0(x_0)=0, g(x_1)=g'(x_1)=0$ , 可设  $g_0(x)=a(x-x_0)l_0^2(x)$ 。

由  $g'_0(x_0)=1$ , 算出  $a=1$ 。所以

$$g_0(x) = (x - x_0)l_0^2(x) \quad (1.21)$$

同理

$$g_1(x) = (x - x_1)l_1^2(x)$$

综上, 得到以  $x_0, x_1$  为节点的埃尔米特插值为

$$\begin{aligned} H_3(x) &= h_0(x)f(x_0) + h_1(x)f(x_1) + g_0(x)f'(x_0) + g_1(x)f'(x_1) \\ H_3(x) &= \left[ 1 + 2 \frac{x - x_0}{x_1 - x_0} \right] \left[ \frac{x - x_1}{x_0 - x_1} \right]^2 f(x_0) + \left[ 1 + 2 \frac{x - x_1}{x_0 - x_1} \right] \left[ \frac{x - x_0}{x_1 - x_0} \right]^2 f(x_1) \\ &\quad + (x - x_0) \left[ \frac{x - x_1}{x_0 - x_1} \right]^2 f'(x_0) + (x - x_1) \left[ \frac{x - x_0}{x_1 - x_0} \right]^2 f'(x_1) \end{aligned} \quad (1.22)$$

容易证明, 当  $f \in C^4[a, b]$  时插值误差为

$$R(x) = f(x) - H_3(x) = \frac{f^{(4)}(\zeta)}{4!} (x - x_0)^2 (x - x_1)^2, \zeta \in [a, b] \quad (1.23)$$

如果要构造  $f(x)$  关于  $n+1$  个节点  $x_i, i=0, 1, \dots, n$  的  $2n+1$  次埃尔米特插值多项式, 手法与构造两个节点的方法类似。

$$H_{2n+1}(x) = \sum_{i=0}^n h_i(x)f(x_i) + \sum_{i=0}^n g_i(x)f'(x_i)$$

这里  $h_i(x), g_i(x), i=0, 1, \dots, n$  分别为不高于  $2n+1$  次插值多项式, 分别满足

$$\begin{cases} h_i(x_j) = \delta_{ij} \\ h'_i(x_j) = 0 \end{cases} \text{ 及 } \begin{cases} g_i(x_j) = 0 \\ g'_i(x_j) = \delta_{ij} \end{cases}$$

由此可得到

$$\begin{aligned} h_i(x) &= \left[ 1 - 2(x - x_i) \sum_{j \neq i} \frac{1}{x_i - x_j} \right] l_i^2(x) \\ g_i(x) &= (x - x_i) l_i^2(x) \end{aligned} \quad (1.24)$$

这里  $\{l_i(x)\}$  为关于节点  $x_i, i=0, 1, \dots, n$  的拉格朗日基函数。

容易证明, 当  $f \in C^{2n+2}[a, b]$  时, 误差为:

$$R(x) = f(x) - H_{2n+1}(x) = \frac{f^{(2n+2)}(\zeta)}{(2n+2)!} (x - x_0)^2 (x - x_1)^2 \cdots (x - x_n)^2, \zeta \in [a, b] \quad (1.25)$$

**例 1.8:** 给定  $f(-1)=0, f(1)=4, f'(-1)=2, f'(1)=0$ , 求埃尔米特插值多项式, 并计算  $f(0.5)$ 。

解:  $H_3(x) = h_0(x) \cdot 0 + h_1(x) \cdot 4 + g_0(x) \cdot 2 + g_1(x) \cdot 0$

显然本题不必计算  $h_0(x), g_1(x)$ 。

$$h_1(x) = \left[ 1 + 2 \frac{(x-1)}{-1-1} \right] \left[ \frac{x-(-1)}{1-(-1)} \right]^2 = (2-x)(x+1)^2/4$$

$$g_0(x) = (x - (-1)) \left[ \frac{x-1}{-1-1} \right]^2 = (x+1)(x-1)^2/4$$

$$H_3(x) = h_1(x) \cdot 4 + g_0(x) \cdot 2$$

$$H_3(x) = (2-x)(x+1)^2 + \frac{1}{2}(x+1)(x-1)^2$$

$$H_3(0.5) = 3.5625$$

利用构造基函数方法做插值多项式被广泛地应用在不同的插值条件中。

**例 1.9:** 给定  $f(x_0) = y_0, f'(x_0) = m_0, f(x_1) = y_1$ , 构造二次插值多项式函数。

**解:** 设  $P_2(x) = t_0(x)y_0 + t_1(x)y_1 + t_2(x)m_0$

这里  $t_0(x), t_1(x), t_2(x)$  均为不高于二次的多项式, 它们分别满足

$$\begin{cases} t_0(x_0) = 1 \\ t'_0(x_0) = 0 \\ t_0(x_1) = 0 \end{cases} \quad \begin{cases} t_1(x_0) = 0 \\ t'_1(x_0) = 0 \\ t_1(x_1) = 1 \end{cases} \quad \begin{cases} t_2(x_0) = 0 \\ t_2(x_1) = 0 \\ t'_2(x_0) = 1 \end{cases}$$

于是  $P_2(x)$  可表示为

$$P_2(x) = (a_0x + b_0) \frac{x-x_1}{x_0-x_1} y_0 + a_1 \left[ \frac{x-x_0}{x_1-x_0} \right]^2 y_1 + a_2(x-x_0)(x-x_1)m_0$$

由  $t_0(x_0) = 1$ , 得  $a_0x_0 + b_0 = 1$

由  $t'_0(x_0) = 0$ , 得  $a_0 \frac{x_0-x_1}{x_0-x_1} + (a_0x_0 + b_0) \frac{1}{x_0-x_1} = 0$

所以  $a_0 = -\frac{1}{x_0-x_1}, b_0 = \frac{2x_0-x_1}{x_0-x_1}$

所以  $t_0(x) = \frac{(2x_0-x_1-x)(x-x_1)}{(x_0-x_1)(x_0-x_1)}$

同理  $t_1(x) = \frac{(x-x_0)^2}{(x_1-x_0)^2}$

由  $t'_2(x_0) = 1, a_2(x_0-x_1) = 1$

所以  $a_2 = \frac{1}{x_0-x_1}$

所以  $t_2(x) = \frac{(x-x_0)(x-x_1)}{x_0-x_1}$

$$P_2(x) = \frac{2x_0-x_1-x}{x_0-x_1} \cdot \frac{x-x_1}{x_0-x_1} y_0 + \left[ \frac{x-x_0}{x_1-x_0} \right]^2 y_1 + \frac{(x-x_0)(x-x_1)}{x_0-x_1} m_0$$

用牛顿差商插值也能构造埃尔米特插值。对给定的插值点的函数值和一阶导数值  $(x_i, f(x_i), f'(x_i)), i=0, 1, \dots, n$ , 定义序列  $z_0 = x_0, z_1 = x_0, z_2 = x_1, z_3 = x_1, \dots$ , 即

$$z_{2i} = z_{2i+1} = x_i, i = 0, 1, \dots, n$$

计算一阶差商时:

$$f[z_{2i-1}, z_{2i}] = \frac{f(z_{2i}) - f(z_{2i-1})}{z_{2i} - z_{2i-1}}$$

由  $f[x_0, x_1] = f'(\xi)$  和  $\lim_{x_1 \rightarrow x_0} f[x_0, x_1] = f'(x_0)$ , 取

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

即构造差商表中用  $f'(x_0), f'(x_1), \dots, f'(x_n)$  代替

$$f[z_0, z_1], f[z_2, z_3], \dots, f[z_{2n}, z_{2n+1}]$$

其余差商计算公式不变,得到差商型埃尔米特插值公式:

$$H_{2n+1}(x) = f[z_0] + \sum_{k=1}^{2n+1} f[z_0, z_1, \dots, z_k](x - z_0) \cdots (x - z_{k-1}) \tag{1.26}$$

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

**例 1.10:** 用下列数据构造埃尔米特插值多项式,并计算  $f(1.36)$ 。

$x$	1.2	1.4	1.6
$f(x)$	0.6	0.9	1.1
$f'(x)$	0.5	0.7	0.6

**解:** 计算差商。

1.2	0.6							
1.2	0.6	0.5						
1.4	0.9	1.5	5					
1.4	0.9	0.7	-4	-45				
1.6	1.1	1.0	1.5	15	145			
1.6	1.1	0.6	2	-17.5	-76.25	-553.125		

$$\begin{aligned} H_5(x) &= 0.6 + 0.5(x - 1.2) + 5(x - 1.2)^2 - 45(x - 1.2)^2(x - 1.4) \\ &\quad + 145(x - 1.2)^2(x - 1.4)^2 - 553.125(x - 1.2)^2(x - 1.4)^2(x - 1.6) \\ H_5(1.36) &= 0.8655 \end{aligned}$$

## 1.5 分段插值

### 1.5.1 龙格(Runge)现象

在构造插值多项式时,根据误差表达式(1.9),你是否认为多取插值点总比少取插值点的效果好呢? 答案是不一定。如果被插函数是高次多项式,那么多取插值点比少取插值点效果好;但对有些函数来说,有时点取的越多,效果越不近人意。请看下面的例子。

给定函数  $f(x) = \frac{1}{1+25x^2}, x \in [-1, 1]$ 。构造 10 次插值多项式  $L_{10}(x)$ 。

对  $[-1, 1]$  作等距分割,取  $h = \frac{2}{10} = 0.2, x_i = -1 + 0.2i$ , 构造  $\left[ x_i, \frac{1}{1+25x_i^2} \right], i=0, 1, \dots, 10$ , 10 次插值多项式  $L_{10}(x)$  如图 1.3 所示。从图中可以看到,在零点附近,  $L_{10}(x)$  对  $f(x)$  的逼近效果较好,在  $x = -0.90, -0.70, 0.70, 0.90$  时误差较大。

下面列出  $L_{10}(x)$  和  $f(x)$  的几个插值点的数值:

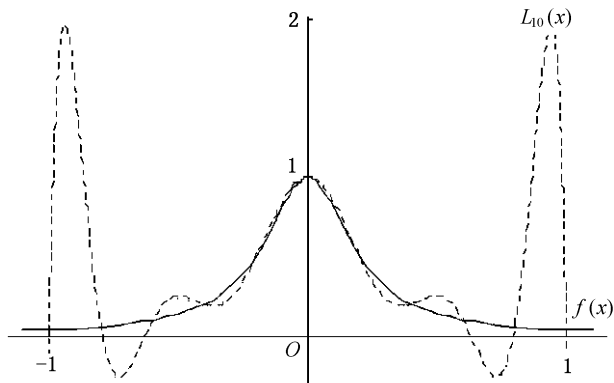


图 1.3  $L_{10}(x)$  和  $f(x)$

$x$	-0.90	-0.70	-0.50	-0.30
$f(x)$	1.57872	0.07547	0.13793	0.30769
$L_{10}(x)$	0.04706	-0.22620	0.25376	0.23535

这个例子是由龙格提出的,也称插值多项式在插值区间内发生剧烈振荡的现象为龙格现象。龙格现象揭示了插值多项式的缺陷。它说明高次多项式的插值效果并不一定优于低次多项式的插值效果。

在插值过程中,误差由截断误差和舍入误差组成。式(1.9)给出的是截断误差,它是插值函数  $\varphi(x)$  与原函数  $f(x)$  的误差。另外由节点  $\gamma_i$  和计算产生的舍入误差,在插值计算过程中可能被扩散或放大,这就是插值的稳定性问题。而高次多项式的稳定性一般比较差,这从另一角度说明了高次插值多项式的缺陷。

### 1.5.2 分段线性插值

既然增加插值点并不能提高插值函数的逼近效果,那么采用分段插值的效果又如何呢?

若对给定区间  $[a, b]$  作分割  $a = x_0 < x_1 < \cdots < x_n = b$ , 在每个小区间  $[x_i, x_{i+1}]$  上作  $f(x)$  以  $x_i, x_{i+1}$  为节点的线性插值,记这个插值函数为  $p(x) = g_i(x)$ , 则

$$g_i(x) = \frac{x - x_{i+1}}{x_i - x_{i+1}} f(x_i) + \frac{x - x_i}{x_{i+1} - x_i} f(x_{i+1}), \quad x_i \leq x \leq x_{i+1} \quad (1.27)$$

把每个小区间的线性插值函数连接起来,就得到了  $f(x)$  以剖分(节点)  $a = x_0 < x_1 < \cdots < x_n = b$  的分段线性函数  $p(x)$ 。  $p(x)$  在  $[x_i, x_{i+1}]$  上为一个不高于一次的多项式,事实上  $p(x)$  是平面上以点  $(x_i, f(x_i))$  为折点的折线。由线性插值误差公式,当  $x \in [x_i, x_{i+1}]$  时,有

$$f(x) - p(x) = f(x) - g_i(x) = \frac{f^{(2)}(\zeta)}{2!} (x - x_i)(x - x_{i+1}) \quad (1.28)$$

因而



$$|f(x) - p(x)| \leq \frac{M_2}{2} |(x - x_i)(x - x_{i+1})|$$

$$\leq \frac{M_2}{2} \cdot \frac{1}{4} (x_{i+1} - x_i)^2 = \frac{M_2}{8} (x_{i+1} - x_i)^2$$

其中  $M_2 = \max |f''(x)|$ ,  $a \leq x \leq b$ 。

于是,当区间分割加密,  $\max_i (x_{i+1} - x_i) \rightarrow 0$  时,分段线性插值收敛于  $f(x)$ 。事实上,只要  $f(x)$  连续,分段线性插值序列就能收敛于  $f(x)$ 。

分段线性插值算法简单,只要区间充分小,就能保证它的误差要求。它的一个显著优点是它的局部性质,如果修改了某节点  $(x_i, f(x_i))$  的值,仅在相邻的两个区间  $[x_{i-1}, x_i]$ ,  $[x_i, x_{i+1}]$  受到影响。分段线性插值的缺点是在插值节点处不光滑。

图 1.4 给出分段线性插值  $p(x)$  (虚线表示) 和  $f(x)$  的图形,可以看到分段线性插值的效果明显好于整体的拉格朗日插值的效果。

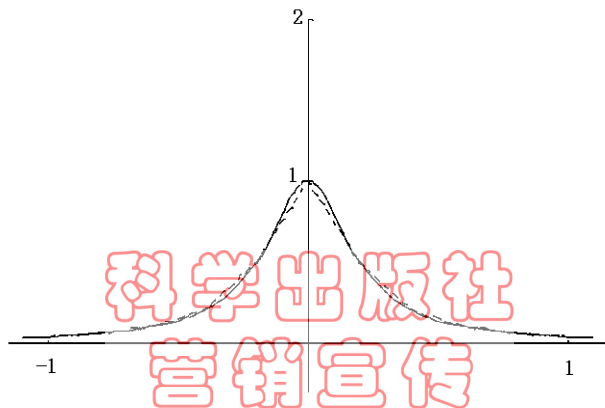


图 1.4 分段线性插值  $p(x)$  和  $f(x)$

**例 1.11:** 对下列数据作分段线性插值,并计算  $f(1.2)$ ,  $f(3.3)$ 。

$x_i$	-3	-1	2	3	9
$f(x_i)$	12	12	1	6	12

解:  $p(x) = g_i(x) = \frac{x - x_{i+1}}{x_i - x_{i+1}} f(x_i) + \frac{x - x_i}{x_{i+1} - x_i} f(x_{i+1}), x \in [x_i, x_{i+1}]$

$$\because 1.2 \in [-1, 2]$$

$$\therefore p(1.2) = g_1(x) = \frac{1.2 - 2}{-1 - 2} \times 5 + \frac{1.2 - (-1)}{2 - (-1)} \times 1 = 2.0667$$

$$\because 3.3 \in [3, 9]$$

$$\therefore p(3.3) = g_3(x) = \frac{3.3 - 9}{-6} \times 6 + \frac{3.3 - 3}{6} \times 12 = 6.3$$

## 1.6 三次样条函数

在制造船体和汽车外形等工艺中传统的设计方法是,首先由设计人员按外形要求,给出外形曲线的一组离散点值  $\{x_i, y_i\}, i = 0, 1, \dots, n$ , 施工人员准备好有弹性的样条(一般用竹条或有弹性的钢条)和压铁,将压铁放在点  $\{x_i, y_i\}$  的位置上,调整竹条的形状,使其自然光滑,这时竹条表示一条插值曲线,我们称为样条函数。从数学上看,这一条近似于分段的三次多项式,在节点处具有一阶和二阶连续微商。样条函数的主要优点是它的光滑程度较高,保证了插值函数二阶导数的连续性,对于三阶导数的间断,人类的眼睛已难以辨认了。样条函数是一种隐式格式,最后需要解一个方程组,它的工作量大于多项式拉格朗日型式或牛顿型式等显式插值方法。

**定义.** 给定区间  $[a, b]$  上  $n+1$  个节点  $a = x_0 < x_1 < \dots < x_n = b$  和这些点上的函数值  $f(x_i) = y_i, i = 0, 1, \dots, n$ 。若  $S(x)$  满足  $S(x_i) = y_i, i = 0, 1, \dots, n$ ;  $S(x)$  在每个小区间  $[x_i, x_{i+1}]$  上至多是一个三次多项式;  $S(x)$  在  $[a, b]$  上有连续的二阶导数,则称  $S(x)$  为  $f(x)$  关于剖分  $a = x_0 < x_1 < \dots < x_n = b$  的三次样条插值函数,称  $x_0, x_1, \dots, x_n$  为样条节点。

要在每个子区间  $[x_i, x_{i+1}]$  上构造三次多项式

$$S(x) = S_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i$$

$x \in [x_i, x_{i+1}], i = 0, 1, \dots, n-1$ , 共需要  $4n$  个条件,由插值条件  $S(x_i) = y_i, i = 0, 1, \dots, n$ , 提供了  $n+1$  个条件;用每个内点的关系建立条件

$$S(x_{i+0}) = S(x_{i-0}),$$

$$S'(x_{i+0}) = S'(x_{i-0}),$$

$$S''(x_{i+0}) = S''(x_{i-0}), i = 1, \dots, n-1$$

又得到  $3n-3$  个条件;再附加两个边界条件,即可惟一确定样条函数了。用待定系数法确定了构造样条函数的存在性和惟一性。在具体构造样条函数时一般都不使用计算量大的待定系数法。下面给出构造三次样条插值的  $M$  关系式和  $m$  关系式的方法。

### 1.6.1 三次样条插值的 $M$ 关系式

引入记号  $M_i = S''(x_i), m_i = S'(x_i), i = 0, 1, \dots, n$ 。用节点处二阶导数表示样条插值函数时称为大  $M$  关系式,用一阶导数表示样条插值函数时称为小  $m$  关系式。

**问题 1.8:** 给定插值点  $(x_i, y_i), i = 0, 1, \dots, n$ , 怎样构造用二阶导数表示的样条插值函数,即怎样构造  $M$  关系式?

假设  $S''(x_i) = M_i, i = 0, 1, \dots, n$ 。由于  $S''(x)$  在  $[x_i, x_{i+1}]$  上为线性函数,故在  $[x_i, x_{i+1}]$  上做  $M$  的分段线性插值函数:

$$S''(x) = \frac{x - x_{i+1}}{x_i - x_{i+1}} M_i + \frac{x - x_i}{x_{i+1} - x_i} M_{i+1}, i = 0, 1, \dots, n-1$$

令  $h_i = x_{i+1} - x_i$ , 得到

$$S''(x) = \frac{x_{i+1} - x}{h_i} M_i + \frac{x - x_i}{h_i} M_{i+1}, i = 0, 1, 2, \dots, n-1 \quad (1.29)$$

对  $S''(x)$  积分两次有

$$\begin{aligned} S(x) &= \frac{(x_{i+1} - x)^3}{6h_i} M_i + \frac{(x - x_i)^3}{6h_i} M_{i+1} + cx + d \\ &= \frac{(x_{i+1} - x)^3}{6h_i} M_i + \frac{(x - x_i)^3}{6h_i} M_{i+1} + C(x_{i+1} - x) + D(x - x_i) \quad (1.30) \end{aligned}$$

将  $S(x_i) = y_i, S(x_{i+1}) = y_{i+1}$  代入式(1.29)可解出

$$C = \frac{y_i}{h_i} - \frac{h_i M_i}{6}, D = \frac{y_{i+1}}{h_i} - \frac{h_i M_{i+1}}{6}$$

故在  $[x_i, x_{i+1}]$  上有

$$\begin{aligned} S(x) &= \frac{(x_{i+1} - x)^3 M_i + (x - x_i)^3 M_{i+1}}{6h_i} + \frac{(x_{i+1} - x)y_i + (x - x_i)y_{i+1}}{h_i} \\ &\quad - \frac{h_i}{6} [(x_{i+1} - x)M_i + (x - x_i)M_{i+1}] \quad x \in [x_i, x_{i+1}], i = 1, 2, \dots, n \quad (1.31) \end{aligned}$$

$S(x)$  在每个小区间上具有不同的表达式,但由于  $S(x)$  在整个区间  $[a, b]$  上是二阶光滑的,故有

$$S'(x_i + 0) = S'(x_i - 0), i = 1, 2, \dots, n-1$$

列出每一个关系式  $S'_i(x_i) = S'_{i-1}(x_i)$ , 再经计算得:

$$\mu_i M_{i-1} + 2M_i + \lambda_i M_{i+1} = d_i, i = 1, 2, \dots, n-1 \quad (1.32)$$

其中:

$$\begin{aligned} \lambda_i &= \frac{h_i}{h_i + h_{i-1}} \\ \mu_i &= 1 - \lambda_i \\ d_i &= \frac{6}{h_i + h_{i-1}} \left[ \frac{y_{i+1} - y_i}{h_i} - \frac{y_i - y_{i-1}}{h_{i-1}} \right] = 6y[x_{i-1}, x_i, x_{i+1}] \end{aligned}$$

由式(1.32)得到  $n+1$  个未知数的  $n-1$  个方程组。现补充两个边界条件,使方程组只有惟一解。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  $M_i, i=1, 2, \dots, n-1$ , 即

$$\begin{bmatrix} 2 & \lambda_1 & & & \\ \mu_2 & 2 & & & \\ & \ddots & \ddots & \ddots & \\ & & \mu_{n-2} & 2 & \lambda_{n-2} \\ & & & \mu_{n-1} & 2 \end{bmatrix} \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_{n-2} \\ M_{n-1} \end{bmatrix} = \begin{bmatrix} d_1 - \mu_1 M_0 \\ d_2 \\ \vdots \\ d_{n-2} \\ d_{n-1} - \lambda_{n-1} M_n \end{bmatrix} \quad (1.33)$$

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

$$2M_0 + M_1 = \frac{6}{h_0} [y[x_0, x_1] - m_0] = d_0$$

$$M_{n-1} + 2M_n = \frac{6}{h_{n-1}} [m_n - y[x_{n-1}, x_n]] = d_n$$

于是需要解  $n+1$  阶的方程组:

$$\begin{bmatrix} 2 & 1 & & & \\ & u_1 & 2 & \lambda_1 & \\ & & u_2 & 2 & \lambda_2 \\ & & & \ddots & \ddots & \ddots \\ & & & & u_{n-2} & 2 & \lambda_{n-1} \\ & & & & & 1 & 2 \end{bmatrix} \begin{bmatrix} M_0 \\ M_1 \\ M_2 \\ \vdots \\ M_{n-1} \\ M_n \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \end{bmatrix} \tag{1.34}$$

(3) 被插函数以  $x_n - x_0$  为基本周期时,即  $y_0 = y_n$ ,即  $S(x_0) = S(x_n), S'(x_0) = S'(x_n), S''(x_0) = S''(x_n)$ ;即  $m_0 = m_n, M_0 = M_n$ 。此时化为  $n$  个变量、 $n$  个方程的方程组。

样条插值构造的  $M$  关系式是对角占优的三对角带状矩阵,可用第 5 章中的追赶法求解。

**例 1.12:** 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

$$\begin{aligned} h_0 &= 0.1, h_1 = 0.2, h_2 = 0.1 \\ \begin{cases} \lambda_1 = 0.6667, \\ \mu_1 = 0.3333, \end{cases} & \begin{cases} \lambda_2 = 0.3333 \\ \mu_2 = 0.6667 \end{cases} \\ d_1 &= 5, \quad d_2 = -55 \end{aligned}$$

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式为

$$S(x) = \begin{cases} 21.875x^3 - 72.1875x^2 + 83.1875x - 32.875 & x \in [1.1, 1.2] \\ -37.5x^3 + 141.5625x^2 - 173.3125x + 69.725 & x \in [1.2, 1.4] \\ 53.125x^3 - 239.0625x^2 + 359.5625x - 178.95 & x \in [1.4, 1.5] \end{cases}$$

特别地,  $f(1.25) \approx S(1.25) = 1.0336$ 。

详细的程序和算例请看 1.7 节。

### 1.6.2 三次样条插值的 $m$ 关系式

**问题 1.9:** 给定插值点  $(x_i, y_i), i = 0, 1, \cdots, n$ , 怎样构造用节点处一阶导数表示的样条插值函数, 即怎样构造  $m$  关系式?

对给定的插值点 $(x_i, y(x_i))$ ,  $i=0, \dots, n$ , 先假定已知  $S'(x_i) = m_i$ , 在每个小区间  $[x_i, x_{i+1}]$  上做埃尔米特插值, 那么在整个  $[x_0, x_n]$  上是分段的埃尔米特插值, 在  $[x_i, x_{i+1}]$  上  $S(x)$  的表达式为

$$S(x) = \left[ 1 + 2 \frac{x - x_i}{x_{i+1} - x_i} \right] \left[ \frac{x - x_{i+1}}{x_i - x_{i+1}} \right]^2 y_i + (x - x_i) \left[ \frac{x - x_{i+1}}{x_i - x_{i+1}} \right]^2 m_i \\ + \left[ 1 + 2 \frac{x - x_{i+1}}{x_i - x_{i+1}} \right] \left[ \frac{x - x_i}{x_{i+1} - x_i} \right]^2 y_{i+1} + (x - x_{i+1}) \left[ \frac{x - x_i}{x_{i+1} - x_i} \right]^2 m_{i+1}$$

通过  $S''(x_i+0) = S''(x_i-0)$  得到方程组

$$\lambda_i m_{i-1} + 2m_i + \mu_i m_{i+1} = c_i, \quad i = 1, 2, \dots, n-1$$

其中:

$$\lambda_i = \frac{h_i}{h_i + h_{i-1}}$$

$$\mu_i = 1 - \lambda_i$$

$$c_i = 3(\lambda_i y[x_{i-1}, x_i] + \mu_i y[x_i, x_{i+1}])$$

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  $M$  关系式中的讨论类似, 不再详述。

对  $f(x) = \frac{1}{1+25x^2}$  作样条插值, 插值效果见图 1.5。可以看到样条插值效果优于分段插值效果。

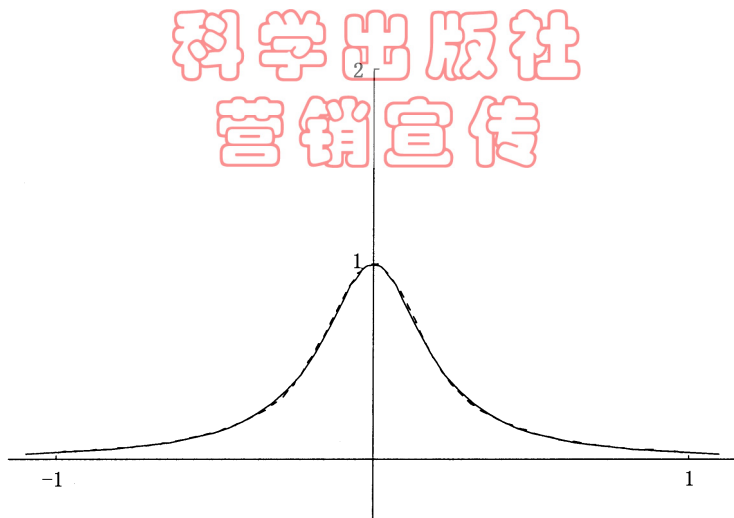


图 1.5 样条插值图示

## 1.7 程序示例

**程序 1** 给定  $(x_i, y_i)$ ,  $i=0, 1, \dots, n$ , 构造牛顿插值多项式  $N_n(x)$ ,  $x_i$  互不相同。

**算法描述**

输入  $n$  值, 及  $(x_i, y_i)$ ,  $i=0, 1, \dots, n$ ; 记  $f(x_i) = y_i$ ;

for  $i=0, 1, \dots, n$

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

其中  $f[x_i] = f(x_i)$

对给定的  $x$ , 由

$$N_n(x) = f(x_0) + (x - x_0)f[x_0, x_1] + (x - x_0)(x - x_1)f[x_0, x_1, x_2] + \dots + (x - x_0)(x - x_1) \dots (x - x_n)f[x_0, x_1, \dots, x_n]$$

计算  $N_n(x)$  的值;

输出  $N_n(x)$ 。

## 程序源码

```

////////////////////////////////////
//          Purpose: (x_i, y_i)的牛顿插值多项式          //
////////////////////////////////////

#include <stdio.h>
#define MAX_N 20          //定义(x_i, y_i)的最大维数
typedef struct tagPOINT   //点的结构
{
    double x;
    double y;
} POINT;
int main()
{
    int n;
    int i, j;
    POINT points[MAX_N+1]; double diff[MAX_N+1];
    double x, tmp, newton=0;
    printf("\nInput n value: ");          //输入被插值点的数目
    scanf("%d", &n);
    if (n>MAX_N)
    {
        printf("The input n is larger then MAX_N, please redefine the MAX_N. \n");
        return 1;
    }
    if (n<=0)
    {
        printf("Please input a number between 1 and %d. \n", MAX_N);
        return 1;
    }

    //输入被插值点(x_i, y_i)
    printf("Now input the (x_i, y_i), i=0,...,%d: \n", n);
    for (i=0; i<=n; i++)
        scanf("%lf%lf", &points[i].x, &points[i].y);
    printf("Now input the x value: ");          //输入计算牛顿插值多项式的 x 值
    scanf("%lf", &x);
    for (i=0; i<=n; i++) diff[i]=points[i].y;

```

科学出版社  
营销宣传

```

for (i=0; i<n; i++)
{
    for (j=n; j>i; j--)
    {
        diff[j]=(diff[j]-diff[j-1])/(points[j].x-points[j-1-i].x);
    }
    //计算 f(x_0,...,x_n)的差商
}

tmp=1; newton=diff[0];
for (i=0; i<n; i++)
{
    tmp=tmp*(x-points[i].x);
    newton=newton+tmp*diff[i+1];
}

printf("newton(%f)=%f\n", x, newton);    //输出
return 0;
}

```

## 计算实例

给定  $\sin 11^\circ = 0.190809$ ,  $\sin 12^\circ = 0.207912$ ,  $\sin 13^\circ = 0.224951$ , 构造牛顿插值函数并计算  $\sin 11^\circ 30'$ 。

## 程序输入输出

input n value: 2

Now input the (x\_i, y\_i), i=0,...,2:

11 0.190809 12 0.207912 13 0.224951

Now Input the x value: 11.5

newton(11.500000)=0.199369

**程序 2** 给定插值点  $(x_i, y_i)$ ,  $i=0, 1, \dots, n$  和二阶导数的端点值  $M_0, M_n$ , 用  $M$  关系式构造三次样条插值多项式  $S(x)$ , 求在给定点  $x$  处  $S(x)$  的值。

## 算法描述

1. 输入  $n$  值, 及  $(x_i, y_i)$ ,  $i=0, 1, \dots, n$ ,  $M_0, M_n$ , 要计算的函数点  $x$ ;
2. for  $i=1$  to  $n-1$

$$\text{计算 } \lambda_i = \frac{h_i}{h_i + h_{i-1}}, \mu_i = 1 - \lambda_i, d_i = \frac{6}{h_i + h_{i-1}} \left[ \frac{y_{i+1} - y_i}{h_i} - \frac{y_i - y_{i-1}}{h_{i-1}} \right],$$

其中  $h_i = x_{i+1} - x_i$ ;

3. 求解方程组

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

对给定的  $x$ , 由

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

4. 计算出  $S(x)$  的值;

5. 输出  $S(x)$ 。

## 程序源码

```

////////////////////////////////////
//      Purpose: 给定  $M_0, M_n$  值的三次样条插值多项式      //
////////////////////////////////////
#include <stdio.h>
#define MAX_N 20                //定义(x_i, y_i)的最大维数
typedef struct tagPOINT          //点的结构
{ double x; double y;
} POINT;

int main()
{ int n;
  int i, k;
  POINT points[MAX_N+1];
  double h[MAX_N+1], b[MAX_N+1], c[MAX_N+1], d[MAX_N+1], M[MAX_N+1];
  double u[MAX_N+1], v[MAX_N+1], y[MAX_N+1];
  double x, p, q, S;
  printf("\nInput n value: ");      //输入插值点的数目
  scanf("%d", &n);
  if (n>MAX_N)
  { printf("The input n is larger than MAX_N, please redefine the MAX_N. \n");
    return 1;
  }
  if (n<=0)
  { printf("Please input a number between 1 and %d. \n", MAX_N);
    return 1;
  }

  //输入插值点(x_i, y_i),  $M_0$  值和  $M_n$  值
  printf("Now input the (x_i, y_i), i=0,...,%d: \n", n);
  for (i=0; i<=n; i++)
    scanf("%lf%lf", &points[i].x, &points[i].y);
  printf("Now input the M[0] value: ");
  scanf("%lf", &M[0]);
  printf("Now input the M[n] value: ");
  scanf("%lf", &M[n]);
  printf("Now input the x value: ");      //输入计算三次样条插值函数的 x 值

```



```

scanf("%lf", &x);
if (x>points[n].x || x<points[0].x)
{ printf("Please input a number between %f and %f. \n", points[0].x, points[n].x);
  return 1;
}

//计算 M 关系式中各参数的值
h[0]=points[1].x-points[0].x;
for (i=1; i<n; i++)
{
    h[i]=points[i+1].x-points[i].x;
    b[i]=h[i]/(h[i]+h[i-1]);
    c[i]=1-b[i];
    d[i]=6*((points[i+1].y-points[i].y)/h[i]
            -(points[i].y-points[i-1].y)/h[i-1])/(h[i]+h[i-1]);
}

//用追赶法计算  $M_i, i=1, \dots, n-1$ 
d[1]=c[1]*M[0];
d[n-1]=b[n-1]*M[n];
b[n-1]=0; c[1]=0; v[0]=0;
for (i=1; i<n; i++)
{ u[i]=2-c[i]*v[i-1];
  v[i]=b[i]/u[i];
  y[i]=(d[i]-c[i]*y[i-1])/u[i];
}
for (i=1; i<n; i++)
{ M[n-i]=y[n-i]-v[n-i]*M[n-i+1];
}

//计算三次样条插值函数在 x 处的值
k=0;
while (x>=points[k].x) k++;
k=k-1;
p=points[k+1].x-x; q=x-points[k].x;
S=(p*p*p*M[k]+q*q*q*M[k+1])/(6*h[k])
  +(p*points[k].y+q*points[k+1].y)/h[k]-h[k]*(p*M[k]+q*M[k+1])/6;
printf("S(%f)=%f\n", x, S);

//输出
getchar();
return 0;
}

```

## 计算实例

给定离散点 $(1.1, 0.4), (1.2, 0.8), (1.4, 1.65), (1.5, 1.8)$ ,  $M_0 = M_n = 0$ , 用  $M$  关系式构造三次样条插值多项式  $S(x)$ , 计算  $S(1.25)$ 。

程序输入输出

Input n value: 3  
Now input the (x\_i, y\_i), i=0,...,3:  
1.1 0.4 1.2 0.8 1.4 1.65 1.5 1.8  
Now Input the M0 value: 0  
Now Input the Mn value: 0  
Now Input the x value: 1.25  
 $S(1.25000)=1.033594$

习题 1

- 1.1 作出插值点  $(-1.00, 3.00), (2.00, 5.00), (3.00, 7.00)$  的二次拉格朗日插值多项式  $L_2(x)$ , 并计算  $L_2(0)$ 。
- 1.2 作出插值点  $(-2.00, 0.00), (2.00, 3.00), (5.00, 6.00)$  的二次拉格朗日插值多项式  $L_2(x)$ , 并计算  $L_2(-1.2), L_2(1.2)$ 。
- 1.3 作出下列插值点的三次拉格朗日插值多项式。  
(1)  $(-1, 3), (0, -1/2), (1/2, 0), (1, 1)$   
(2)  $(-1, 2), (0, 0), (2, 1), (3, 3)$
- 1.4 设  $f \in C^2[a, b]$ , 且  $f(a) = f(b) = 0$ , 证明:  $\|f(x)\| \leq \frac{1}{8}(b-a)^2 M_2, a \leq x \leq b$ , 其中  $M_2 = \max_{a \leq x \leq b} |f''(x)|$ 。
- 1.5  $f(x) = \sqrt{x}$  在离散点有  $f(81)=9, f(100)=10, f(121)=11$ , 用插值方法计算  $x=105$  的近似值, 并由误差公式给出误差界, 同时与实际误差作比较。
- 1.6 给出函数表:

$x$	-1.00	2.00	3.00	4.00
$f(x)$	3.00	5.00	7.00	5.00

作出差商表, 写出牛顿插值公式, 用  $N_3(1.2)$  计算  $f(1.2)$  的近似值。

- 1.7 要作三角函数  $\log x$  的函数值表, 已知表值有五位小数的近似值, 要求用线性插值引起的截断误差不超过表值的舍入误差, 试决定其最大允许步长。
- 1.8  $f(x) = x^7 - 125x^5 + 237x^3 - 999$ , 计算差商  $f[2^0, 2^1], f[2^0, 2^1, \dots, 2^7]$  以及  $f[2^0, 2^1, \dots, 2^8]$ 。
- 1.9 给出函数表:

$x$	1.05	1.10	1.15	1.20
$f(x)$	2.12	2.20	2.17	2.32

构造分段线性函数,并计算  $f(1.075)$  和  $f(1.175)$  的近似值。

- 1.10 给定数据  $f(0), f(1), f'(1)$ , 作出二次插值多项式, 并写出插值余项。
- 1.11 给定数据  $f(3)=5.00, f(5)=15.00, f'(5)=7.00$ , 作出二次插值多项式, 写出插值余项, 并计算  $f(3.7)$  的近似值。
- 1.12 给定数据  $f(0), f(1), f(3), f'(3)$ , 作出三次插值多项式, 并写出插值余项。
- 1.13 给定数据  $f(0)=1.0, f(1)=0.75, f(3)=0.25, f'(3)=0.56$ , 作出三次插值多项式, 并写出插值余项。
- 1.14 给定数据  $f(0)=0, f(1)=1, f'(0)=0, f'(1)=1, f''(1)=0$ , 构造四次插值多项式, 并写出插值余项。
- 1.15 给出下列数据:

$x$	-2.00	-1.00	1.00	2.00
$f(x)$	-4.00	3.00	5.00	12.00

试求满足上列数据和  $S''(-2)=0, S''(2)=0$  的三次样条函数, 并计算  $S(0)$  的值。

- 1.16 给出下列数据:

$x$	-1.00	0.00	1.00	3.00
$f(x)$	2.00	3.00	4.00	29.00

试求满足上列数据和  $S'(-1)=5, S'(3)=29.00$  的三次样条函数, 并计算  $S(2)$  的值。

## 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

当函数  $f(x)$  是以离散点列给出时,当函数的表达式过于复杂时,常用数值微分近似计算  $f(x)$  的导数  $f'(x)$ 。在微积分中,导数表示函数在某点上的瞬时变化率,它是平均变化率的极限;在几何上可解释为曲线的斜率;在物理上可解释为物体变化的速率。

以下是导数  $f'(x)$  的三种定义形式:

$$\begin{aligned} f'(x) &= \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} \\ &= \lim_{h \rightarrow 0} \frac{f(x) - f(x-h)}{h} \\ &= \lim_{h \rightarrow 0} \frac{f(x+h) - f(x-h)}{2h} \end{aligned} \quad (2.1)$$

在微积分中,用差商的极限定义导数;在数值计算中返朴归真,导数取用差商(平均变化率)作为其近似值。

最简单的计算数值微分的方法是用函数的差商近似函数的导数,即取极限的近似值。下面是与式(2.1)相应的三种差商形式的数值微分公式以及相应的截断误差。

#### 向前差商

用向前差商(平均变化率)近似导数有:

$$f'(x_0) \approx \frac{f(x_0+h) - f(x_0)}{h} \quad (2.2)$$

其中  $x_0+h$  的位置在  $x_0$  的前面,因此称为向前差商。同理可得向后差商、中心差商的定义。

由泰勒展开

$$f(x_0+h) = f(x_0) + hf'(x_0) + \frac{h^2}{2!} \times f''(\xi), \quad x_0 \leq \xi \leq x_0+h$$

得向前差商的截断误差:

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

#### 向后差商

用向后差商近似导数有:

$$f'(x_0) \approx \frac{f(x_0) - f(x_0-h)}{h} \quad (2.3)$$

与计算向前差商的方法类似,由泰勒展开得向后差商的截断误差:

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

中心差商

用中心差商(平均变化率)近似导数有：

$$f'(x_0) \approx \frac{f(x_0 + h) - f(x_0 - h)}{2h} \tag{2.4}$$

由泰勒展开

$$\begin{aligned} f(x_0 + h) &= f(x_0) + hf'(x_0) + \frac{h^2}{2!}f''(x_0) + \frac{h^3}{3!}f'''(\xi_1) \\ f(x_0 - h) &= f(x_0) - hf'(x_0) + \frac{h^2}{2!}f''(x_0) - \frac{h^3}{3!}f'''(\xi_2) \end{aligned}$$

得中心差商的截断误差：

$$\begin{aligned} R(x) &= f'(x_0) - \frac{f(x_0 + h) - f(x_0 - h)}{2h} = \frac{h^2}{12}[f'''(\xi_1) + f'''(\xi_2)] \\ &= \frac{h^2}{6}f'''(\xi) = O(h^2) \qquad x_0 - h \leq \xi \leq x_0 + h \end{aligned}$$

差商的几何意义

微积分中的极限定义  $f'(x_0) = \lim_{h \rightarrow 0} \frac{f(x_0 + h) - f(x_0)}{h}$ ，表示  $f(x)$  在  $x = x_0$  处切线的斜率，即图 2.1 中直线  $P$  的斜率；差商  $\frac{f(x_0 + h) - f(x_0)}{h}$  表示过  $(x_0, f(x_0))$  和  $(x_0 + h, f(x_0 + h))$  两点直线  $Q$  的斜率，是一条过  $x_0$  的割线。可见数值微分是用近似值内接弦的斜率代替准确值切线的斜率。

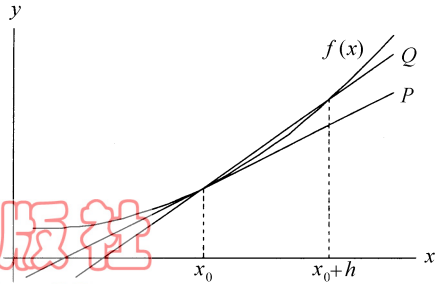


图 2.1 微商与差商

例 2.1: 给出下列数据, 计算  $f'(0.02), f'(0.06), f'(0.10), f''(0.08)$ 。

$x$	0.02	0.04	0.06	0.08	0.10
$f(x)$	5.06	5.07	5.065	5.05	5.055

$$\begin{aligned} \text{解: } f'(0.02) &\approx (5.07 - 5.06) / (0.04 - 0.02) = 0.5 \\ f'(0.06) &\approx (5.05 - 5.07) / (0.08 - 0.04) = -0.5 \\ f'(0.10) &\approx (5.05 - 5.055) / (0.08 - 0.10) = 0.25 \\ f''(0.08) &\approx (f'(0.10) - f'(0.06)) / (0.10 - 0.06) = 18.75 \end{aligned}$$

设定最佳步长

在计算数值导数时, 它的误差由截断误差和舍入误差两部分组成。用差商或插值公式近似导数产生截断误差, 由原始值  $y_i$  的数值近似产生舍入误差。在差商计算中, 从截断误差的逼近阶的角度看,  $|h|$  越小, 则误差也越小; 但是太小的  $|h|$  会带来较大的舍入误差。怎样选择最佳步长, 使截断误差与舍入误差之和最小呢?

一般对计算导数的近似公式进行分析可得到误差的表示式, 以中心差商为例, 截断误差不超过

h^2/6 M3 = h^2/6 max | f'''(x) |

而舍入误差可用量 e/h 估计(证明略),其中 e 是函数 yi 的原始值的绝对误差限,总误差为

h^2/6 M3 + e/h

当 [ h^2/6 M3 + e/h ]' = h/3 M3 - e/h^2 = 0 时,总误差达到最小值,即

h = cube root of (3e/M3) (\*)

可以看到用误差的表达式确定步长,难度较大,难以实际操作。

通常用事后估计方法选取步长 h,例如,记 D(h), D[h/2] 为步长等于 h, h/2 的差商计算公式,给定误差界 ε,当 | D(h) - D[h/2] | < ε 时, h/2 就是合适的步长。

例 2.2:对函数 y = e^x,取不同的步长计算 f'(1.15),观察误差变化规律,从而确定最佳步长。

解:

h	f'(1.15)	误差	h	f'(1.15)	误差
0.10	3.1630	-0.0048	0.05	3.1590	-0.0008
0.09	3.1622	-0.0040	0.04	3.1588	-0.0006
0.08	3.1613	-0.0031	0.03	3.1583	-0.0001
0.07	3.1607	-0.0025	0.02	3.1575	0.0007
0.06	3.1600	-0.0018	0.01	3.1550	0.0032

表中数据显示,当步长 h 从 0.10 减少到 0.03 时,数值微分误差的绝对值从 0.0048 减少到 0.0001,而随着 h 的进一步减少,误差的绝对值又有所反弹,表明当步长 h 小于 0.03 时,舍入误差起了主要作用。函数值取了四位数值,表明 e = 0.5 × 10^-4,由 (\*) 式可算出 h = 0.035。

在实际计算中是无法得到误差的准确数值的,这时以 | D(hi) - D(hj) | 最小为标准确定步长,本例中取 h = 0.04。

2.1.2 插值型数值微分

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。

设 xi, i = 0, 1, ..., n 为 [a, b] 上的节点,给定 (xi, f(xi)), i = 0, 1, ..., n, 以 (xi, f(xi)) 为插值点构造插值多项式 Ln(x), 以 Ln(x) 的各阶导数近似 f(x) 的相应阶的导数,即

f(x) = Ln(x) = sum from i=0 to n of li(x) f(xi)

$$f'(x) = L'_n(x) = \sum_{i=0}^n l'_i(x) f(x_i)$$

当  $x = x_j$  时,

$$f'(x_j) \approx \sum_{i=0}^n l'_i(x_j) f(x_i), j = 0, 1, \dots, n \quad (2.5)$$

误差项为:

$$R(x) = \frac{d}{dx} \left[ \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{i=0}^n (x - x_i) \right]$$

$$R(x_j) = \prod_{\substack{i=0 \\ i \neq j}}^n (x_j - x_i) \frac{f^{(n+1)}(\xi)}{(n+1)!}$$

**例 2.3:** 给定  $(x_i, f(x_i)), i=0, 1, 2$ , 并有  $x_2 - x_1 = x_1 - x_0 = h$ , 计算  $f'(x_0), f'(x_1), f'(x_2)$ 。

解: 作过  $(x_i, f(x_i)), i=0, 1, 2$  的插值多项式:

$$L_2(x) = \frac{(x - x_1)(x - x_2)}{2h^2} f(x_0) + \frac{(x - x_0)(x - x_2)}{-h^2} f(x_1) + \frac{(x - x_0)(x - x_1)}{2h^2} f(x_2)$$

$$f'(x) \approx L'_2(x) = \frac{f(x_0)}{2h^2} (x - x_1 + x - x_2) - \frac{f(x_1)}{h^2} (x - x_0 + x - x_2) + \frac{f(x_2)}{2h^2} (x - x_0 + x - x_1)$$

将  $x = x_i$  代入  $f'(x)$  得三点端点公式和三点中点公式:

$$f'(x_0) \approx \frac{1}{2h} (-3f(x_0) + 4f(x_1) - f(x_2))$$

$$f'(x_1) \approx \frac{1}{2h} (-f(x_0) + f(x_2))$$

$$f'(x_2) \approx \frac{1}{2h} (f(x_0) - 4f(x_1) + 3f(x_2))$$

利用泰勒(Taylor)展开进行比较和分析, 可得三点公式的截断误差是  $O(h^2)$ 。

类似地, 可得到五点中点公式和五点端点公式:

$$f'(x_0) = \frac{1}{12h} [f(x_0 - 2h) - 8f(x_0 - h) + 8f(x_0 + h) - f(x_0 + 2h)] + \frac{h^4}{30} f^{(5)}(\xi), \quad \xi \in [x_0 - 2h, x_0 + 2h]$$

$$f'(x_0) = \frac{1}{12h} [f(x_0) + 48f(x_0 + h) - 36f(x_0 + 2h) + 16f(x_0 + 3h) - 3f(x_0 + 4h)] + \frac{h^4}{5} f^{(5)}(\xi), \quad \xi \in [x_0, x_0 + 4h]$$

### 2.1.3 样条插值数值微分

把离散点按大小排列成  $a = x_0 < x_1 < \dots < x_n = b$ , 用  $m$  关系式构造插值点  $(x_i, f(x_i)), i=0, 1, 2, \dots, n$  的样条函数  $S(x)$ :

$$S(x) = \left[ 1 + 2 \frac{x - x_i}{x_{i+1} - x_i} \right] \left[ \frac{x - x_{i+1}}{x_i - x_{i+1}} \right]^2 y_i + (x - x_i) \left[ \frac{x - x_{i+1}}{x_i - x_{i+1}} \right]^2 m_i \\ + \left[ 1 + 2 \frac{x - x_{i+1}}{x_i - x_{i+1}} \right] \left[ \frac{x - x_i}{x_{i+1} - x_i} \right]^2 y_{i+1} + (x - x_{i+1}) \left[ \frac{x - x_i}{x_{i+1} - x_i} \right]^2 m_{i+1}$$

若  $x = x_i$  则  $f'(x_i) \approx m_i$ ; 当  $x \in (x_i, x_{i+1})$  时, 可用  $S'(x) \approx f'(x)$  计算导数。

## 2.2 数值积分

在微积分中用牛顿-莱布尼兹(Newton-Leibniz)公式计算连续函数  $f(x)$  的定积分:

$$\int_a^b f(x) dx = F(b) - F(a)$$

但是, 当被积函数是以点列  $(x_i, f(x_i))$ ,  $i=0, 1, \dots, n$  的形式给出时, 当被积函数  $f(x)$  的原函数  $F(x)$  难以得到时, 例如  $\int_1^2 \sin x^2 dx$ , 则无法用牛顿-莱布尼兹积分公式计算。有时当被积函数的原函数过于复杂时, 也不宜套用积分公式计算积分, 而用数值积分公式计算定积分。

在微积分中, 定积分是黎曼(Rimann)和的极限, 它是分割小区间长度趋于零时的极限, 即

$$\int_a^b f(x) dx = \lim_{\lambda \rightarrow 0} \left[ \sum_{i=0}^{n-1} f(x_i) \Delta x_i \right]$$

在数值积分公式中, 只能用有限项的和近似上面的极限, 通常由函数在离散点函数值的线性组合形式给出。

记  $I(f) = \int_a^b f(x) dx$ ,  $I_n(f) = \sum_{i=0}^n \alpha_i f(x_i)$ , 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,  $\{x_i\}$  称为积分节点,  $\alpha_i$  称为积分系数。确定  $I_n(f)$  中积分系数  $\alpha_i$  的过程就是构造数值积分公式的过程。

怎样判断数值积分公式的效果? 代数精度是衡量数值积分公式优劣的重要标准之一。

### 代数精度

记  $[a, b]$  上以  $(x_i, f(x_i))$ ,  $i=0, 1, \dots, n$  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  $E_n(x^k) = I(x^k) - I_n(x^k) = 0$ ,  $k=0, 1, \dots, m$ , 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  $m$  阶多项式都有  $I(f) = I_n(f)$ 。

### 2.2.1 插值型数值积分

对给定的被积函数在  $[a, b]$  上的点列  $(x_i, f(x_i))$ ,  $i=0, 1, \dots, n$ , 作拉格朗日插值多项式  $L_n(x)$ , 以  $\int_a^b L_n(x) dx$  近似计算  $\int_a^b f(x) dx$ , 即



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

记  $\alpha_i = \int_a^b l_i(x) dx$ , 则有

$$I_n(f) = \int_a^b L_n(x) dx = \sum_{i=0}^n \alpha_i f(x_i)$$

数值积分误差,也就是对插值误差的积分值

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

或

$$E_n(f) = \int_a^b f[x_0, x_1, \dots, x_n, x] \prod_{i=0}^n (x - x_i) dx$$

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。

**例 2.4:** 建立  $[0, 2]$  上节点为  $x_0 = 0, x_1 = 0.5, x_2 = 2$  的数值积分公式。

解: 由  $\alpha_i = \int_a^b l_i(x) dx$  得

$$\alpha_0 = \int_0^2 l_0(x) dx = \int_0^2 \frac{(x-0.5)(x-2)}{(0-0.5)(0-2)} dx = -\frac{1}{3}$$

$$\alpha_1 = \int_0^2 l_1(x) dx = \int_0^2 \frac{(x-0)(x-2)}{(0.5-0)(0.5-2)} dx = \frac{16}{9}$$

$$\alpha_2 = \int_0^2 l_2(x) dx = \int_0^2 \frac{(x-0)(x-0.5)}{(2-0)(2-0.5)} dx = \frac{5}{9}$$

得到数值积分公式:

$$I_2(f) = \frac{1}{9} [-3f(0) + 16f(0.5) + 5f(2)]$$

## 2.2.2 牛顿-柯特斯(Newton-Cote's)积分

把积分区间  $[a, b]$  分成  $n$  等分, 记步长为  $h = \frac{b-a}{n}$ , 取等分点  $x_i = a + ih (i = 0, 1, \dots, n)$  作为数值积分节点, 构造拉格朗日插值多项式  $L_n(x)$ , 取

$$\int_a^b f(x) dx \approx \int_a^b L_n(x) dx$$

由此得到的数值积分称为牛顿-柯特斯积分。下面可以看到, 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

以  $(a, f(a))$  和  $(b, f(b))$  为插值节点构造线性函数  $L_1(x)$ , 有

$$\int_a^b f(x) dx \approx \int_a^b L_1(x) dx$$

那么,

$$\int_a^b L_1(x) dx = \int_a^b (l_0(x)f(x_0) + l_1(x)f(x_1)) dx$$

$$\alpha_0 = \int_a^b l_0(x) dx = \int_a^b \frac{x-b}{a-b} dx = \frac{1}{2}(b-a) = (b-a)c_0^{(1)}$$

$$\alpha_1 = \int_a^b l_1(x) dx = \int_a^b \frac{x-a}{b-a} dx = \frac{1}{2}(b-a) = (b-a)c_1^{(1)}$$

提取公因子  $(b-a)$  后, 得到牛顿-柯特斯积分的组合系数:  $c_0^{(1)} = \frac{1}{2}$ ,  $c_1^{(1)} = \frac{1}{2}$ , 它们已与积分区间没有任何关系了。

$$\int_a^b f(x) dx \approx \frac{b-a}{2} [f(a) + f(b)]$$

$$\text{记} \quad T(f) = \frac{b-a}{2} [f(a) + f(b)] \quad (2.6)$$

称  $T(f)$  为梯形积分公式。它的几何意义是用梯形面积近似代替积分值(图 2.2)。

怎样确定梯形积分公式的代数精度?

取  $f(x) = x$  时, 有

$$I(f) = \int_a^b x dx = \frac{b-a}{2} (f(a) + f(b)) = \frac{b^2 - a^2}{2} = T(f)$$

取  $f(x) = x^2$  时, 有

$$I(f) = \int_a^b x^2 dx \neq \frac{b-a}{2} (f(a) + f(b)) = T(f)$$

得梯形求积公式具有一阶代数精度。由

$$f(x) = L_1(x) + \frac{f''(\xi)}{2!} (x-a)(x-b), \quad a \leq \xi \leq b$$

$$E_1(x) = \int_a^b \frac{f''(\xi)}{2!} (x-a)(x-b) dx$$

因为  $(x-a)(x-b)$  在  $[a, b]$  上不变号, 由积分中值定理得到梯形求积公式的截断误差:

$$E_1(x) = \frac{f''(\eta)}{2!} \int_a^b (x-a)(x-b) dx = -\frac{f''(\eta)}{12} (b-a)^3, \quad a \leq \eta \leq b \quad (2.7)$$

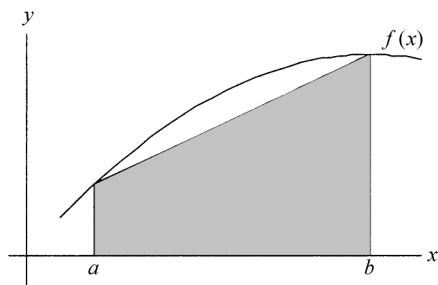


图 2.2 梯形面积

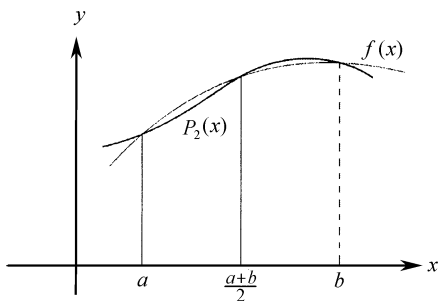


图 2.3 抛物线积分面积

### 辛普森(Simpson)积分

对区间  $[a, b]$  作二等分, 记  $x_0 = a$ ,  $x_1 = (a+b)/2$ ,  $x_2 = b$ 。以  $(a, f(a))$ ,  $((a+b)/2, f((a+b)/2))$  和  $(b, f(b))$  为插值节点构造二次插值函数  $L_2(x)$ , 那么, 有

$$\int_a^b L_2(x) dx = \int_a^b (l_0(x)f(x_0) + l_1(x)f(x_1) + l_2(x)f(x_2)) dx$$

$$\alpha_0 = \int_a^b l_0(x) dx = \int_a^b \frac{(x - (a+b)/2)(x-b)}{(a - (a+b)/2)(a-b)} dx = \frac{1}{6}(b-a) = (b-a)c_0^{(2)}$$

$$\alpha_1 = \int_a^b l_1(x) dx = \frac{4}{6}(b-a) = (b-a)c_1^{(2)}$$

$$\alpha_2 = \int_a^b l_2(x) dx = \frac{1}{6}(b-a) = (b-a)c_2^{(2)}$$

计算得到积分组合系数:  $c_0^{(2)} = \frac{1}{6}, c_1^{(2)} = \frac{4}{6}, c_2^{(2)} = \frac{1}{6}$ 。

$$\int_a^b f(x) dx \approx I_2(f) = S(f) = \frac{b-a}{6} \left[ f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right] \quad (2.8)$$

$S(f)$ 称为辛普森或抛物线积分公式。它的几何意义是用过三点的抛物线面积近似代替积分的曲边面积(图 2.3)。

分别将  $f(x)=1, x, x^2, x^3$  代入到  $I(f)$ 和  $S(f)$ 中,可以得到

$$S(f) = \frac{b-a}{6} \left[ f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right] = I(f)$$

表明辛普森公式对于次数不超过三次的多项式准确成立,  $S(f)$ 具有三阶代数精度。因此可设一个三次多项式满足条件:

$$P_3(a) = f(a), P_3(b) = f(b) \\ P_3\left[\frac{a+b}{2}\right] = f\left[\frac{a+b}{2}\right], P_3'\left[\frac{a+b}{2}\right] = f'\left[\frac{a+b}{2}\right]$$

计算得到误差为:

$$f(x) - P_3(x) = \frac{f^{(4)}(\xi)}{4!}(x-a)\left[x - \frac{a+b}{2}\right]^2(x-b), \quad a \leq \xi \leq b$$

于是有

$$E_2(f) = I(f) - S(f) = (I(f) - I(P_3)) + (I(P_3) - S(f))$$

但

$$I(P_3) = S(P_3) = \frac{b-a}{6} \left[ P_3(a) + 4P_3\left[\frac{a+b}{2}\right] + P_3(b) \right] = S(f)$$

故辛普森求积公式的截断误差:

$$E_2(f) = I(f) - I(P_3) = \int_a^b \frac{f^{(4)}(\eta)(x-a)\left[x - \frac{a+b}{2}\right]^2(x-b)}{4!} dx \\ = \frac{f^{(4)}(\eta)}{4!} \int_a^b (x-a)\left[x - \frac{a+b}{2}\right]^2(x-b) dx \\ = -\frac{(b-a)^5}{2880} f^{(4)}(\eta), \quad a \leq \eta \leq b \quad (2.9)$$

**牛顿-柯特斯积分系数**

$n$  等分区间  $[a, b]$ , 取等分点为积分节点,  $x_i = a + ih, i=0, 1, \dots, n$ , 其中  $h = \frac{b-a}{n}$ 。

以  $(x_i, f(x_i)), i=1, 2, \dots, n$  为插值节点构造插值函数  $L_n(x)$ 。

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

其中

$$\alpha_i = \int_a^b l_i(x) dx = \int_a^b \frac{(x-x_0)(x-x_1)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)}{(x_i-x_0)(x_i-x_1)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)} dx$$

令  $x = a + th, x_i = a + ih$ , 代入上式得

$$\begin{aligned} \alpha_i &= \int_0^1 \frac{t(t-1)\cdots(t-i+1)(t-i-1)\cdots(t-n)}{i!(n-i)!(-1)^{n-i}} h dt \\ &= \frac{(b-a)}{n} \frac{(-1)^{n-i}}{i!(n-i)!} \int_0^n t(t-1)\cdots(t-i+1)(t-i-1)\cdots(t-n) dt \\ &= (b-a) c_i^{(n)} \end{aligned} \tag{2.10}$$

这里称  $c_i^{(n)} = \frac{(-1)^{n-i}}{i!(n-i)!} \int_0^n t(t-1)\cdots(t-i+1)(t-i-1)\cdots(t-n) dt$  为牛顿-柯特斯系数。

可见在取等距节点时, 积分系数  $c_i^{(n)}$  与积分节点和积分区间无直接关系, 只与插值的节点总数有关, 而在例 2.3 中的积分系数是待定系数, 这就简化了数值积分公式, 而不必对每一组插值节点  $x_i$  都要计算一组相应的积分系数  $\alpha_i$ 。在公式(2.10)中取  $n=1$ , 可算出梯形积分系数; 取  $n=2$ , 可算出辛普森积分系数。在表 2.1 中列出  $n$  从 1 到 6 的牛顿-柯特斯系数。

表 2.1

$n$	$c_0^{(n)}$	$c_1^{(n)}$	$c_2^{(n)}$	$c_3^{(n)}$	$c_4^{(n)}$	$c_5^{(n)}$	$c_6^{(n)}$
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	$\frac{41}{840}$	$\frac{9}{35}$	$\frac{9}{280}$	$\frac{34}{105}$	$\frac{9}{280}$	$\frac{9}{35}$	$\frac{41}{840}$

2.3 复化数值积分

由插值的龙格现象可知, 高阶牛顿-柯特斯积分不能保证等距数值积分系列的收敛性, 同时可证(略)高阶牛顿-柯特斯积分的计算是不稳定的。因此, 实际计算中常用低阶复化梯形等积分公式。

2.3.1 复化梯形积分

把积分区间分割成若干小区间, 在每个小区间  $[x_i, x_{i+1}]$  上用梯形积分公式, 再将这

些小区间上的数值积分累加起来,称为复化梯形公式。复化梯形公式用若干个小梯形面积逼近积分  $\int_a^b f(x) dx$  比用一个大梯形公式效果显然更好,如图 2.4 所示。这种作法使我们想起定积分定义,即它为被积函数无限分割的代数和。这也正是计算定积分最朴素的算法。

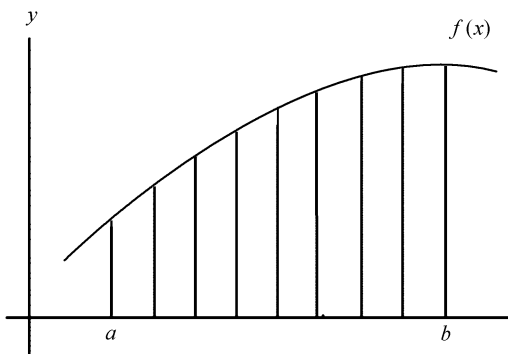


图 2.4 复化梯形公式积分视图

## 复化梯形积分计算公式

对  $[a, b]$  作等距分割, 有  $h = \frac{b-a}{n}, x_i = a + ih, i = 0, 1, \dots, n$ , 于是

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

在  $[x_i, x_{i+1}]$  上,  $\int_{x_i}^{x_{i+1}} f(x) dx = \frac{h}{2} (f(x_i) + f(x_{i+1})) - f''(\xi_i) \frac{h^3}{12}$ , 则有

$$\begin{aligned} I(f) &= \sum_{i=0}^{n-1} \left\{ \frac{h}{2} (f(x_i) + f(x_{i+1})) - f''(\xi_i) \frac{h^3}{12} \right\} \\ &= h \left[ \frac{1}{2} f(a) + \sum_{i=1}^{n-1} f(x_i) + \frac{1}{2} f(b) \right] - \sum_{i=0}^{n-1} f''(\xi_i) \frac{h^3}{12} \end{aligned}$$

记  $n$  等分的复化梯形公式为  $T_n(f)$  或  $T(h)$ , 有

$$T(h) = T_n(f) = h \left[ \frac{1}{2} f(a) + \sum_{i=1}^{n-1} f(a + ih) + \frac{1}{2} f(b) \right] \quad (2.11)$$

## 复化梯形公式截断误差

由  $E_n(f) = I(f) - T_n(f) = -\frac{h^3}{12} \sum_{i=0}^{n-1} f''(\xi_i)$ , 根据均值定理, 当  $f \in C^2[a, b]$  时, 存

在  $\xi \in [a, b]$ , 有  $\sum_{i=0}^{n-1} f''(\xi_i) = n f''(\xi)$ , 于是

$$E_n(f) = -\frac{nh^3}{12} f''(\xi) = -\frac{h^2}{12} (b-a) f''(\xi) = -\frac{(b-a)^3}{12n^2} f''(\xi), \quad a \leq \xi \leq b \quad (2.12)$$

由此看到复化梯形公式的截断误差按照  $h^2$  或者  $\frac{1}{n^2}$  的速度下降, 事实上, 可以证明,

只要  $f(x)$  在  $(a, b)$  上有界并黎曼可积, 当分点无限增多时, 复化梯形公式收敛到积分  $I(f) = \int_a^b f(x) dx$ 。

记  $M_2 = \max_{a \leq x \leq b} |f''(x)|$ , 则有

$$|E_n(f)| \leq \frac{(b-a)^3}{12n^2} M_2 = O\left(\frac{1}{n^2}\right)$$

对于任给的误差控制小量  $\epsilon > 0$ , 有

$$\frac{(b-a)^3}{12n^2} M_2 < \epsilon \quad \text{或} \quad n \geq \left\lceil \sqrt{\frac{(b-a)^3 M_2}{12\epsilon}} \right\rceil + 1$$

就有  $|E_n(f)| < \epsilon$ , 式中  $[\cdot]$  表示取其最大整数。

### 2.3.2 复化辛普森积分

把积分区间分成偶数等分  $2m$ , 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  $[x_{2i}, x_{2i+2}]$  上用辛普森数值积分公式计算, 则得到复化辛普森公式, 记为  $S_n(f)$ 。

#### 复化辛普森积分计算公式

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

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

$$S_n(f) = \sum_{i=0}^{m-1} \frac{2h}{6} (f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2}))$$

$$= \frac{h}{3} \left[ f(a) + 4 \sum_{i=1}^{m-1} f(x_{2i+1}) + 2 \sum_{i=1}^{m-1} f(x_{2i}) + f(b) \right] \tag{2.13}$$

为复化辛普森积分公式, 它是  $f(x)$  在  $[x_{2i}, x_{2i+2}]$  上采用辛普森积分公式叠加而得。下面用图 2.5 显示复化辛普森积分计算公式中节点与系数的关系, 取  $n=8$ , 在每个积分区间上提出因子  $\frac{2h}{6}$  后, 三个节点的系数分别是 1, 4, 1; 将 4 个积分区间的系数按节点的位置累加, 可以清楚地看到, 首尾节点的系数是 1, 奇数点的系数是 4, 偶数点的系数是 2。

$x_0$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$
1	4	1						
		1	4	1				
				1	4	1		
						1	4	1
1	4	2	4	2	4	2	4	1

图 2.5 复化辛普森积分系数

#### 复化辛普森公式的截断误差

设  $f \in C^4[a, b]$ , 在  $[x_{2i}, x_{2i+1}]$  上的误差为

$$-\frac{(2h)^5}{2880} f^{(4)}(\xi_i), \quad x_{2i} \leq \xi_i \leq x_{2i+2}$$

因此,

$$\begin{aligned} I(f) - S_n(f) &= -\frac{(2h)^5}{2880} \sum_{i=0}^{m-1} f^{(4)}(\zeta_i) = -\frac{(2h)^5}{2880} m f^{(4)}(\zeta) \\ &= -\frac{(b-a)^5}{2880 m^4} f^{(4)}(\zeta) = -\frac{(b-a)^5}{180 n^4} f^{(4)}(\zeta) \end{aligned}$$

$$\text{即} \quad E_n(f) = -\frac{(b-a)^5}{180 n^4} f^{(4)}(\zeta), \zeta \in [a, b] \quad (2.14)$$

与复化梯形公式类似,误差的截断误差按照  $h^4$  或者  $\frac{1}{n^4}$  的速度下降。可以证明,只要  $f(x)$  在  $(a, b)$  上有界并黎曼可积,当分点无限增多时,复化辛普森公式收敛到积分  $I(f) = \int_a^b f(x) dx$ 。

记  $M_4 = \max_{a \leq x \leq b} |f^{(4)}(x)|$ , 则有

$$|E_n(f)| \leq \frac{(b-a)^5}{2880 m^4} M_4 = O\left[\frac{1}{m^4}\right]$$

对任给的误差控制小量  $\epsilon > 0$ , 只要

$$\frac{(b-a)^5}{2880 m^4} M_4 < \epsilon \quad \text{或} \quad m \geq \left\lceil \sqrt[4]{\frac{(b-a)^5 M_4}{2880 \epsilon}} \right\rceil + 1$$

就有  $|E_n(f)| < \epsilon$ 。

**例 2.5:**  $I(f) = \int_0^1 e^x dx$ , 计算中要求有 5 位有效数字。用复化梯形和复化辛普森求积公式的分点应取多少?

解:

$$\begin{aligned} f(x) &= e^x, f''(x) = f^{(4)}(x) = e^x \\ |f''(x)| &= |f^{(4)}(x)| \leq e, \quad 0 \leq x \leq 1 \end{aligned}$$

由复化梯形误差公式得到:

$$|I(f) - T_n(f)| \leq \frac{(b-a)^3}{12 n^2} M_2 = \frac{1}{12 n^2} e \leq \frac{1}{2} 10^{-4}$$

计算出  $n = 67.3$ , 复化梯形公式至少要在  $[0, 1]$  等分  $n = 68$ 。

由复化辛普森误差公式, 有

$$|I(f) - S_n(f)| \leq \frac{1}{2880 m^4} e \leq \frac{1}{2} 10^{-4}$$

在复化辛普森公式中取  $m = \left\lceil \frac{3}{2} \right\rceil + 1 = 2$  或  $n = 4$ 。

### 2.3.3 复化积分的自动控制误差算法

复化积分的误差公式表明,截断误差随分点  $n$  的增大而减小,对于给定的误差量  $\epsilon$ , 用估计函数导数的界的方法可计算出  $n$ 。用误差公式计算满足精度的分点数,像是在做一道计算导数  $|f^{(n)}(\zeta)|$  上界的微积分习题(如例 2.5 所示)。但是在实际运算中,一般难以估计出函数的各阶导数界,也就无法确定分点数  $n$ 。在计算中常用误差的事后估计

方法,即用  $|T_{2n}(f) - T_n(f)|$  估计误差  $|I(f) - T_{2n}(f)|$ 。

### $T_{2n}(f)$ 的计算公式

对定积分  $\int_a^b f(x) dx$ , 取分点  $n=1$ , 计算得

$$T_1(f) = \frac{b-a}{2}(f(a) + f(b))$$

取分点  $n=2$ , 计算得

$$T_2(f) = \frac{b-a}{2} \left[ \frac{f(a)}{2} + \frac{f(b)}{2} + f(x_2) \right] = \frac{1}{2} T_1(f) + \frac{b-a}{2} f(x_2)$$

这里,  $x_2 = \frac{a+b}{2}$ 。可以看到,  $T_2(f)$  的值是  $T_1(f)$  与新增分点  $f(x_2)$  的组合。

取分点  $n=4$ , 计算得

$$\begin{aligned} T_4(f) &= \frac{b-a}{4} \left[ \frac{f(a)}{2} + \frac{f(b)}{2} + f(x_1) + f(x_2) + f(x_3) \right] \\ &= \frac{T_2}{2} + \frac{b-a}{4} (f(x_1) + f(x_3)) \end{aligned}$$

这里,  $x_1 = \frac{1}{2}(a + x_2)$ ,  $x_3 = \frac{1}{2}(x_2 + b)$ 。

同理, 计算  $T_4(f)$  时只要在  $T_2(f)$  的基础上计算新增分点  $f(x_1)$ ,  $f(x_3)$  的值再做组合, 如图 2.6 所示。

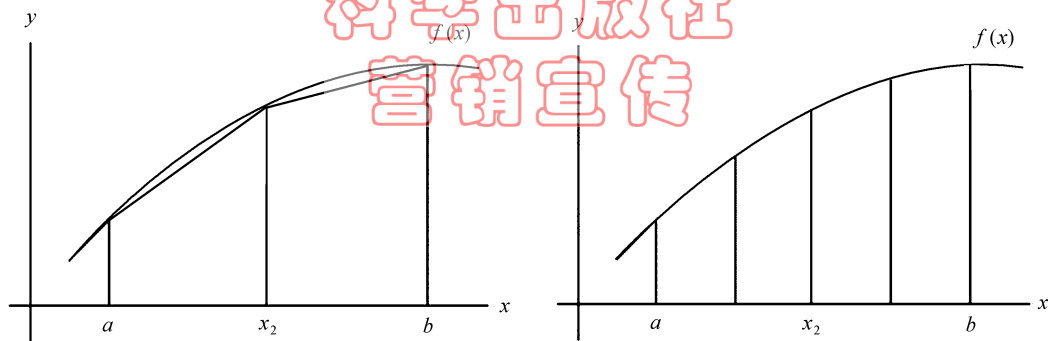


图 2.6  $T_2(f)$  与  $T_4(f)$

一般地, 每次总对前一次的小区间分半, 分点加密一倍, 并可充分利用老分点上的函数值, 每次只需计算新增分点的和。

对  $[a, b]$  上  $n$  等分,  $h_n = \frac{b-a}{n}$ , 则有

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

记  $[x_i, x_{i+1}]$  上的中点为  $x_{i+\frac{1}{2}}$ , 则

$$T_{2n}(f) = \frac{h_n}{2} \left[ \frac{1}{2} f(a) + \sum_{i=1}^{n-1} f(x_i) + \sum_{i=0}^{n-1} f(x_{i+1/2}) + \frac{1}{2} f(b) \right]$$



$$= \frac{h_n}{2} \left[ \frac{1}{2} f(a) + \sum_{i=1}^{n-1} f(x_i) + \frac{1}{2} f(b) \right] + \frac{h_n}{2} \sum_{i=0}^{n-1} f(x_{i+\frac{1}{2}}) \quad (2.15)$$

$$T_{2n}(f) = \frac{1}{2} (T_n(f) + H_n(f))$$

其中  $H_n(f) = h_n \sum_{i=0}^{n-1} f(x_{i+\frac{1}{2}})$ 。

或 
$$T_{2n} = \frac{T_n}{2} + h_{2n} \sum_{i=1}^n f(a + (2i-1)h_{2n})$$

其中  $h_{2n} = \frac{b-a}{2n}$ 。

类似地,可得积分节点为  $n, 2n$  的辛普森求积公式的关系式:

$$S_{2n}(f) = \frac{1}{2} S_n(f) + \frac{1}{6} (4H_{2n}(f) - H_n(f)) \quad (2.16)$$

$$|T_{2n}(f) - T_n(f)| \text{ 与 } |I(f) - T_{2n}(f)|$$

由误差公式:

$$I(f) - T_n(f) = -\frac{(b-a)}{12} h^2 f''(\xi)$$

$$I(f) - T_{2n}(f) = -\frac{(b-a)}{12} \left(\frac{h}{2}\right)^2 f''(\eta)$$

由于  $f''(\xi) = \frac{1}{n} \sum_{i=0}^{n-1} f''(\xi_i)$ ,  $f''(\eta) = \frac{1}{2n} \sum_{i=0}^{2n-1} f''(\eta_i)$ , 分别为  $n$  及  $2n$  个点上的均值, 可视  $f''(\xi) \approx f''(\eta)$ , 于是有

$$I(f) - T_n(f) \approx 4(I(f) - T_{2n}(f))$$

上式表明  $T_n(f)$  的误差大约是  $T_{2n}(f)$  误差的 4 倍。

或 
$$I(f) - T_{2n}(f) \approx \frac{1}{3} (T_{2n}(f) - T_n(f)) \quad (2.17)$$

由此得到启发,对任给的误差控制量  $\epsilon > 0$ , 要  $|I(f) - T_{2n}(f)| < \epsilon$ , 只需  $|T_{2n}(f) - T_n(f)| < 3\epsilon$  即可, 而用  $|T_{2n}(f) - T_n(f)|$  作为控制手段简单直接, 序列  $T_n(f)$ ,  $T_{2n}(f)$ , ... 在计算机上也不难实现。

### 复化积分的算法描述

从数值积分的误差公式可以看到,截断误差随分点  $n$  的增长而减少,控制计算的精度也就是确定分点数  $n$ 。在计算中不用数值积分的误差公式确定分点数  $n$  的理论模式, 而用  $|T_{2n}(f) - T_n(f)| < 3\epsilon$  作为控制,通过增加分点自动满足精度的方法称为数值积分公式的自动积分法。即在计算中构造序列  $T_n, T_{2n}, T_{4n}, \dots$ , 直到  $|T_{2m} - T_m| < \epsilon$  或  $\frac{|T_{2m} - T_m|}{|T_{2m}|} < \epsilon$  时停止计算,由分点数自动控制积分值的误差,并取  $I(f) \approx T_m(f)$ 。

下面描述复化数值积分公式的自动控制误差算法,详细程序和算例请看本章 2.6 节。

1. 输入:误差控制精度  $e = \text{eps}$ ; 初始分点值  $n = m$ 。

2. 计算  $n$  分点的复化梯形积分  $T_n, T2 = T_n$

$$T1 = T2 + 100$$

迭代计算中  $T1$  和  $T2$  分别表示  $T_n$  和  $T_{2n}$

3. while  $|T1 - T2| > e$

$$T1=T2$$

$$H=H_n \quad ! \text{ 计算新增节点的值 } H_n(f)=h_n \sum_{i=0}^{n-1} f(x_{i+\frac{1}{2}})$$

$$T2=(T1+H)/2$$

$$h=h/2, n=2n \quad ! \text{ 将区间一分为二}$$

end while

4. 输出积分值  $T2$ 。

在自动控制误差算法中初始分点值不宜过小,以防假收敛。

### 2.3.4 龙贝格(Romberg)积分

前面得到的关系式(2.17)将  $(T_n(f) - T_{2n}(f))$  作为  $T_{2n}(f)$  的修正值补充到  $I(f)$ , 即

$$I(f) \approx T_{2n}(f) + \frac{1}{3}(T_{2n}(f) - T_n(f)) = \frac{4}{3}T_{2n} - \frac{1}{3}T_n = S_n \quad (2.18)$$

其结果是将梯形求积公式组合成辛普森求积公式,截断误差由  $O(h^2)$  提高到  $O(h^4)$ 。这种手段称为外推算法。外推算法在不增加计算量的前提下提高了误差的精度,是计算方法中的一种常用手法。

不妨对  $S_{2n}(f), S_n(f)$  再做一次组合。由

$$I(f) - S_{2n}(f) = \frac{f^{(4)}(\eta)}{2880} \left(\frac{h}{2}\right)^4 \approx d \left[\frac{h}{2}\right]^4$$

$$I(f) - S_n(f) = \frac{f^{(4)}(\xi)}{2880} h^4 \approx dh^4$$

得到

$$I(f) - S_{2n}(f) \approx \frac{1}{15}(S_{2n}(f) - S_n(f))$$

$$I(f) \approx \frac{16}{15}S_{2n}(f) - \frac{1}{15}S_n(f) = C_n(f) \quad (2.19)$$

复化辛普森公式组成复化柯特斯公式,其截断误差是  $O(h^6)$ 。同理对柯特斯公式进行组合:

$$I(f) - C_n(f) = eh^6$$

$$I(f) - C_{2n}(f) = e\left[\frac{h}{2}\right]^6$$

得到具有 7 次代数精度和截断误差是  $O(h^8)$  的龙贝格公式:

$$R_n(f) = \frac{64}{63}C_{2n}(f) - \frac{1}{63}C_n(f)$$

还可以继续对  $R_n(f)$  做下去。

为了便于在计算机上实现龙贝格算法,将  $T_n, S_n, C_n, R_n, \dots$  统一用  $R_{k,j}$  表示,列标  $j=1, 2, 3, \dots$  分别表示梯形、辛普森、柯特斯积分,行标  $k$  表示分点数  $n \cdot 2^{k-j}$  或步长  $h_k = h/2^{k-j}$ 。

龙贝格计算公式:

$$R_{k,j} = R_{k,j-1} + \frac{R_{k,j-1} - R_{k-1,j-1}}{4^{j-1} - 1}, \quad k = 2, 3, \dots$$

对每一个  $k, j$  从 2 做到  $k$ , 一直做到  $|R_{k,k} - R_{k-1,k-1}|$  小于给定控制精度停止计算。  
**龙贝格算法**

1. 输入区间端点  $a, b$ , 精度控制值  $e$ , 循环次数  $M$ , 定义函数  $f(x)$ , 取  $n=1, h=b-a$ ;
2.  $R_{1,1}=(f(a)+f(b))/2$ ;
3. for  $k=2$  to  $M$ 

$$\{ R_{k,1}=(R_{k-1,1}+h_{k-1}\sum_{i=1}^{2^{k-2}} f(a+(2i-1)h_k))/2 \quad ! \quad h_k=h/2^{k-1}$$

for  $j=2$  to  $k$

$$\{ R_{k,j}=R_{k,j-1}+(R_{k,j-1}-R_{k-1,j-1})/(4^{j-1}-1)\}$$

if  $|R_{k,k}-R_{k-1,k-1}|<e$  退出循环
4. 输出  $R_{k,k}$ 。

表 2.2

$R_{1,1}$			
$R_{2,1}$	$R_{2,2}$		
$R_{3,1}$	$R_{3,2}$	$R_{3,3}$	
$\vdots$	$\vdots$	$\vdots$	$\ddots$
$R_{m,1}$	$R_{m,2}$	$R_{m,3}$	$R_{m,m}$

龙贝格算法按表 2.2 元素的行序进行运算,  $R_{1,1}, R_{2,1}, R_{2,2}, \dots$  在计算中每个元素只用到上一行和本行的元素。对上面的算法进一步优化, 对每  $k$  行可将计算定义在两行元素之间, 令  $R_{k,j}$  为  $R_{1,j}, R_{k-1,j}$  为  $R_{0,j}$ ; 在每计算一行元素后, 要将  $R_{1,j} \Rightarrow R_{0,j}, j=1, 2, \dots, k$ 。

## 2.4 重积分计算

在微积分中计算二重积分是用化为累次积分的方法进行的。计算二重数值积分也是计算累次数值积分的过程。为了简化问题, 我们仅讨论矩形域上的二重积分。有很多非矩形域上的二重积分可作变换将其转换到矩形域上。

$$\int_a^b \int_c^d f(x,y) dydx \tag{2.20}$$

其中:  $a, b, c, d$  是常数,  $f(x,y)$  在  $D$  上连续。像在微积分中一样, 将二重积分化为累次积分:

$$\int_a^b \int_c^d f(x, y) dy dx = \int_a^b \left[ \int_c^d f(x, y) dy \right] dx \quad (2.21)$$

或

$$\int_a^b \int_c^d f(x, y) dy dx = \int_c^d \left[ \int_a^b f(x, y) dx \right] dy$$

## 二重积分的复化梯形公式

对区间 $[a, b]$ 和 $[c, d]$ 分别选取正整数  $m$  和  $n$ , 在  $x$  轴和  $y$  轴上分别有步长

$$h = \frac{b-a}{m}, \quad k = \frac{d-c}{n}$$

用复化梯形公式计算  $\int_c^d f(x, y) dy$ , 计算中将  $x$  当作常数, 有

$$\int_c^d f(x, y) dy \approx k \left[ \frac{1}{2} f(x, y_0) + \frac{1}{2} f(x, y_n) + \sum_{j=1}^{n-1} f(x, y_j) \right] \quad (2.22)$$

再将  $y$  当作常数, 在  $x$  方向上计算式(2.23)中每一项的积分, 有

$$\frac{1}{2} \int_a^b f(x, y_0) dx \approx \frac{h}{2} \left[ \frac{1}{2} f(x_0, y_0) + \frac{1}{2} f(x_m, y_0) + \sum_{i=1}^{m-1} f(x_i, y_0) \right]$$

$$\frac{1}{2} \int_a^b f(x, y_n) dx \approx \frac{h}{2} \left[ \frac{1}{2} f(x_0, y_n) + \frac{1}{2} f(x_m, y_n) + \sum_{i=1}^{m-1} f(x_i, y_n) \right]$$

$$\begin{aligned} \int_a^b \sum_{j=1}^{n-1} f(x, y_j) dx &= \sum_{j=1}^{n-1} \int_a^b f(x, y_j) dx \\ &\approx h \sum_{j=1}^{n-1} \left[ \frac{1}{2} f(x_0, y_j) + \frac{1}{2} f(x_m, y_j) + \sum_{i=1}^{m-1} f(x_i, y_j) \right] \\ &= h \sum_{j=1}^{n-1} \left[ \frac{1}{2} f(x_0, y_j) + \frac{1}{2} f(x_m, y_j) \right] + h \sum_{j=1}^{n-1} \sum_{i=1}^{m-1} f(x_i, y_j) \end{aligned}$$

则

$$\begin{aligned} \int_a^b \int_c^d f(x, y) dy dx &\approx hk \left\{ \frac{1}{4} (f(x_0, y_0) + f(x_m, y_0) + f(x_0, y_n) + f(x_m, y_n)) \right. \\ &\quad + \frac{1}{2} \left[ \sum_{i=1}^{m-1} f(x_i, y_0) + \sum_{i=1}^{m-1} f(x_i, y_n) + \sum_{j=1}^{n-1} f(x_0, y_j) + \sum_{j=1}^{n-1} f(x_m, y_j) \right] \\ &\quad \left. + \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} f(x_i, y_j) \right\} = hk \sum_{i=0}^m \sum_{j=0}^n c_{i,j} f(x_i, y_j) \end{aligned}$$

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。

误差:

$$E(f) = -\frac{(d-c)(b-a)}{12} \left[ h^2 \frac{\partial^2 f}{\partial x^2}(\eta, u) + k^2 \frac{\partial^2 f}{\partial y^2}(\eta, u) \right]$$

$(\eta, u)$  和  $(\eta, u)$  在积分区间内。

**例 2.6:** 用复化梯形公式计算二重积分  $\int_0^1 \int_1^2 \sin(x^2 + y) dy dx$ , 取  $h = k = 0.25$ 。

解:  $f(x, y)$  如下表所示:

$\begin{matrix} y \\ x \end{matrix}$	1.00	1.25	1.50	1.75	2.00
0.00	0.841471	0.948985	0.997495	0.983986	0.909297
0.25	0.873575	0.966827	0.999966	0.970932	0.88153
0.50	0.948985	0.997495	0.983986	0.909297	0.778073
0.75	0.999966	0.970932	0.88153	0.737319	0.547265
1.00	0.909297	0.778073	0.598472	0.381661	0.14112

$c_{i,j}$ 的数值列表如下：

$\begin{matrix} j \\ i \end{matrix}$	0	1	2	3	4
0	1/4	1/2	1/2	1/2	1/4
1	1/2	1	1	1	1/2
2	1/2	1	1	1	1/2
3	1/2	1	1	1	1/2
4	1/4	1/2	1/2	1/2	1/4

$$\begin{aligned} \therefore \int_0^1 \int_1^2 \sin(x^2+y) dydx &= 0.25 \times 0.25 \bigg[ \frac{1}{4}(0.841471+0.909297+0.909297+0.14112) \\ &\quad + \frac{1}{2}(0.948985+0.997495+0.983986) \\ &\quad + \frac{1}{2}(0.778073+0.598472+0.381661) + \frac{1}{2}(0.873575+0.948985+0.999966) \\ &\quad + \frac{1}{2}(0.88153+0.778073+0.547265) + (0.966827+0.999966+0.970932 \\ &\quad + 0.997495+0.983986+0.909297+0.970932+0.88153+0.737319) \bigg] \\ &= 0.873601 \\ \int_0^1 \int_1^2 \sin(x^2+y) dydx \text{ 的准确值是 } &0.886176。 \end{aligned}$$

### 二重复化辛普森求积公式

对区间 $[a,b]$ 和 $[c,d]$ 分别  $m$  等分和  $n$  等分,在  $x$  轴和  $y$  轴上分别有步长

$$h=\frac{b-a}{m}, \quad k=\frac{d-c}{n}, \quad m,n \text{ 均为偶数}$$

类似于二重复化梯形公式推导,得

$$\int_a^b \int_c^d f(x,y) dydx = \frac{hk}{9} \sum_{i=0}^m \sum_{j=0}^n w_{i,j} f(x_i,y_j)$$

$$\text{记 } U=\{u_0,u_1,\cdots,u_m\}^T=\{1,4,2,4,\cdots,2,4,1\}^T$$

$$V=\{v_0,v_1,\cdots,v_n\}^T=\{1,4,2,4,\cdots,2,4,1\}^T$$

$$w_{i,j}=u_i \cdot v_j$$

误差：

$$E(f)=\frac{(d-c)(b-a)}{180}\bigg[h^4\frac{\partial^4 f}{\partial x^4}(\eta,u)+k^4\frac{\partial^4 f}{\partial y^4}(\eta,w)\bigg]$$

$(\eta, u)$ 和 $(\eta, \pi)$ 在积分区间内。

按例 2.6 的化分区间,  $w_{i,j}$ 的值列表如下:

$i \backslash j$	0	1	2	3	4
0	1	4	2	4	1
1	4	16	8	16	4
2	2	8	4	8	2
3	4	16	8	16	4
4	1	4	2	4	1

2.5\* 高斯(Gauss)型积分公式介绍

在牛顿-柯特斯积分公式中要求节点是等距的,其优点是计算积分系数的公式规则相同,缺点是制约了求积公式的代数精度。可以证明:当节点个数  $n$  为偶数时,求积公式具有  $n-1$  阶的代数精度;当节点个数  $n$  为奇数时,求积公式具有  $n$  阶的代数精度。

**例 2.7:**计算求积系数  $c_1, c_2$  和求积节点  $x_0, x_1$ ,使得  $\int_{-1}^1 f(x) dx \approx c_0 f(x_0) + c_1 f(x_1)$  至少具有 3 阶代数精度。

**解:**按照求积公式的代数精度定义,分别令  $f(x)=1, x, x^2, x^3$ ,得方程组:

$$\left\{ \begin{array}{l} c_1 \cdot 1 + c_2 \cdot 1 = \int_{-1}^1 1 dx = 2 \\ c_1 \cdot x_1 + c_2 \cdot x_2 = \int_{-1}^1 x dx = 0 \\ c_1 x_1^2 + c_2 x_2^2 = \int_{-1}^1 x^2 dx = \frac{2}{3} \\ c_1 x_1^3 + c_2 x_2^3 = \int_{-1}^1 x^3 dx = 0 \end{array} \right.$$

解方程得:

$$c_1 = 1, c_2 = 1, x_1 = -\frac{1}{\sqrt{3}}, x_2 = \frac{1}{\sqrt{3}} = 0.5773503$$

求积公式:

$$\int_{-1}^1 f(x) dx = f(-0.57735) + f(0.57735)$$

按例 2.7 的方式,构造更高阶的代数精度的求积公式,生成求积系数和求积节点的方程组并无困难,而求解该方程组则无一定的章法可循。一般地,通过计算正交多项式的零点作为求积节点。

当取积分节点为正交多项式的零点时,则节点个数是  $n$  的求积公式具有  $2n-1$  阶的代数精度。并称积分节点为正交多项式的零点的数值积分公式为高斯型积分公式。

为了一般性,考虑积分

$$I = \int_a^b W(x) f(x) dx$$

其中  $W(x)$  ( $W(x) \geq 0$ ) 称为权函数。当  $W(x) = 1$  时,即是普通的积分。

对于不同的权函数  $W(x)$  选定的节点也不相同。

对给定的  $[a, b]$  及权函数  $W(x)$ ,由施密特(Schmidt)正交化过程作出正交多项式  $P_0(x), P_1(x), \dots, P_n(x)$ ;解出正交多项式  $P_n(x)$  的  $n$  个零点  $x_1, x_2, \dots, x_n$ ,这  $n$  个零点就是积分节点;以这些节点构造插值多项式,计算积分系数

$$\alpha_i = \int_a^b l_i(x) W(x) dx, \quad i = 1, 2, \dots, n$$

其中  $l_i(x)$  是插值基函数。

高斯型求积公式为

$$G_n(f) = \sum_{i=1}^n \alpha_i f(x_i)$$

高斯型积分公式的优点是它的代数精度高,特别是对无穷区间或瑕积分更有效,但计算正交多项式的零点即积分节点有一定的工作量,好在数值学家们已算出一些特定的函数的积分节点和积分系数,在计算中我们可以查表直接得到这些数值。

本章并不构造各种高斯型积分公式,有关的详细内容请参考有关的教材。

下面给出  $[a, b] = [-1, 1]$  上,取权函数  $W(x) = 1$  的高斯型积分。

$[-1, 1]$  上权函数  $W(x) = 1$  的正交多项式为勒让德(Legendre)多项式:

$$L_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} [(x^2 - 1)^n], \quad x \in [-1, 1]$$

高斯-勒让德  $n=2, 4, 5$  相应的积分节点和积分系数表如下:

$n$	$x_k$	$\alpha_k$
2	$x_1 = -0.5773503, x_2 = 0.5773503$	1.0000000, 1.0000000
4	$x_1 = -0.8611363, x_2 = -0.3399810$ $x_3 = 0.3399810, x_4 = 0.8611363$	0.3478548, 0.6521452 0.6521452, 0.3478548
5	$-x_1 = x_5 = 0.9061798$ $-x_2 = x_4 = 0.538469$ $x_3 = 0.0$	0.2369269, 0.2369269 0.4786287, 0.4786287 0.5688889

要计算一般区间  $[a, b]$  上的积分  $\int_a^b f(x) dx$ , 只需作变量代换  $x = \frac{a+b}{2} + \frac{b-a}{2} t$  则有  $\int_a^b f(x) dx = \frac{b-a}{2} \int_{-1}^1 g(t) dt$ , 其中  $g(t) = f\left(\frac{a+b}{2} + \frac{b-a}{2} t\right)$ , 这样,  $I(f) = \int_a^b f(x) dx$  仍可用高斯积分求积,即

$$G_n(f) = \frac{b-a}{2} G_n(g) = \frac{b-a}{2} \sum_{i=1}^n \alpha_i g(x_i)$$

**例 2.8:**应用两点高斯-勒让德积分公式计算  $I = \int_{-1}^1 u^2 \cos u du$ 。

解:  $I = (-0.5773503)^2 \cos(-0.5773503) + (0.5773503)^2 \cos(0.5773503)$

$$=0.558608$$

**例 2.9:**应用两点高斯-勒让德积分公式计算  $I = \int_0^1 \frac{dx}{1+x^2}$ 。

解:令  $x = \frac{1}{2} + \frac{1}{2}t$ , 得到积分

$$I = 2 \int_{-1}^1 \frac{dt}{4 + (t+1)^2} = 2(f(x_1) + f(x_2)) = 0.786885$$

## 2.6 程序示例

**程序 3** 复化梯形公式  $T_n = h \left[ \frac{1}{2} f(a) + \sum_{i=1}^{n-1} f(a+ih) + \frac{1}{2} f(b) \right]$  的自动控制误差算法。

### 算法描述

输入  $m, b, a$  的值, 定义  $f(x)$ ;

$n = m, h = (b - a) / n$ ;

$T2 = T_n = h \left[ \frac{1}{2} f(a) + \sum_{i=1}^{n-1} f(a+ih) + \frac{1}{2} f(b) \right]$ ;

$T1 = T2 + 100$

while  $|T1 - T2| > \epsilon$

$T1 = T2$ ;

$H = h \sum_{i=0}^{n-1} f(x_{i+1/2})$  ! 或用复化梯形公式计算  $T2$

$T2 = (T1 + H) / 2$ ;  $h = h/2$ ;  $n = 2n$ ;

end while

输出  $T2$ 。

### 程序源码

```

////////////////////////////////////
//          Purpose: 梯形公式的自动控制误差算法          //
////////////////////////////////////
#include <stdio.h>
#include <math.h>
#define f(x) (sin(x))          //f(x)由 define 定义
#define m 100
#define a 1.0
#define b 2.0
#define epsilon 0.00001

```

void main()

```

{
    int n=m;
    int i;
    double T_n, H_n, T1, T2;
    double h=(b-a)/n;
    T_n=(f(a)+f(b))/2;

```



```

for (i=1; i<n; i++)
    T_n+=f(a+h*i);
T_n*=h;
T2=T_n;
T1=T2+100;
do
{
    T1=T2;
    for (i=0, H_n=0; i<n; i++)
        H_n+=f(a+h*i+h/2);
    H_n*=h;
    T2=(T1+H_n)/2;
    h=h/2; n=n*2;
} while (fabs(T1-T2) > epsilon);
printf("T=%lf\n", T2);
}
// ----- End of File -----

```

## 计算实例

对于  $f(x) = \sin(x)$ , 在区间  $[a, b]$  上验证梯形公式的自动控制误差公式。

## 程序输入输出

(对于不同  $f(x)$ , 修改程序 #define f(x) 项, 本例  $f(x) = \sin(x)$ , 初始  $m=100$ , 区间  $[a, b]=[1, 2]$ 。)

$T=0.956447$

## 程序 4 龙贝格积分算法

计算公式和算法描述如第 2.3.4 节所述。

## 程序源码

```

////////////////////////////////////
//                               Purpose: 龙贝格算法                               //
////////////////////////////////////
#include <stdio.h>
#include <math.h>
#define f(x)                (sin(x))
#define N_H                20
#define MAXREPT            10
#define a                  1.0
#define b                  2.0
#define epsilon            0.00001

double computeT(double aa, double bb, long int n)    //复化梯形公式
{
    int i; double sum, h=(bb-aa)/n; sum=0;
    for (i=1; i<n; i++)

```