数值积分和数值微分

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

由微积分学基本定理,当f(x)在[a,b]上连续时,存在原函数F(x) ,

由Newton-Leibnits公式 $I = \int_a^b f(x)dx = F(b) - F(a)$

有时,用上面的方法计算定积分有困难.

1.不易求f(x)的原函数F(x) e.g. $\frac{\sin x}{x}, \frac{1}{\ln x}, e^{-x^2}$

2.f(x)的原函数表达式很复杂(计算量大) e.g. $\int_a^b \frac{1}{1+x^4}dx$

3.f(x)用列表给出(观测所得数据表)

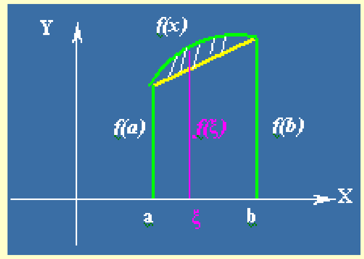
所以,讨论数值积分,即用数值方法计算定积分的近似值.

机械求积

对于 $I=\int_a^b f(x)dx$,若$f(x)>0$时,则I对应于曲边梯形的面积.

当$f(x)$在$[a,b]$上连续,由积分中值定理.

$\exists\ \xi\in[a,b]\quad \int_a^b f(x)dx=(b-a)f(\xi)$



I是以b-a为底,高为$f(\xi)$的矩形的面积.
$f(\xi)$称为$[a,b]$上的平均高度.

1. 梯形公式 取 $f(\xi)\approx\dfrac{f(a)+f(b)}{2}$

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

2. 中矩形公式

取 $f(\xi)\approx f(\dfrac{a+b}{2})\qquad \int_a^b f(x)dx\approx(b-a)f(\dfrac{a+b}{2})$

3. Simpson公式

$$\text{取}\quad f(\xi)\approx\frac{1}{6}[f(a)+4f(\frac{a+b}{2})+f(b)]$$

$$\int_a^b f(x)dx\approx\frac{(b-a)}{6}[f(a)+4f(\frac{a+b}{2})+f(b)]$$

$$\therefore\ \int_a^b f(x)dx\approx\frac{b-a}{6}f(a)+\frac{4(b-a)}{6}f(\frac{a+b}{2})+\frac{b-a}{6}f(b)$$

机械求积公式:

在[a,b]中有n+1个互异的节点$x_0, x_1, x_2, \ldots, x_n$.

$$\int_a^b f(x)dx \approx A_0 f(x_0) + A_1 f(x_1) + \ldots + A_n f(x_n) \qquad (3.1)$$

称上式为机械求积公式,其中x_0~ x_n为**求积节点**,
$A_i(i=0,1,\ldots,n)$为**求积系数**(权).

注:1. 求积系数A_k仅与节点x_i的选取有关,而不依赖于被积函数f(x)的具体形式.

2.通过机械求积,把求积分值转化为求函数值,避免了Newton-Leibnits求原函数的困难.

3. 机械求积是求定积分的近似方法.

$R_n(f) = I - \sum_{i=0}^{n} A_i f(x_i)$ 求积公式(3.1)的截断误差或余项.

代数精度

对于机械求积公式 $\int_a^b f(x)dx \approx \sum_{i=0}^{n} A_i f(x_i)$

$$R_n(f) = I - \sum_{i=0}^{n} A_i f(x_i)$$

定义 若上述公式对所有次数不超过m的多项式$P_m(x)$都精确成立,即$R_n(P_m)=0$,而对某一个m+1次多项式$P_{m+1}(x)$近似成立,即$R_n(P_{m+1})\neq 0$.则称机械求积公式具有**m次代数精度**.

梯形公式 $\int_a^b f(x)dx \approx \frac{(b-a)}{2}[f(a)+f(b)]$ 的代数精度为1.

判断代数精度的方法

当$f(x)=1,x,x^2,\dots,x^m$时,求积公式精确成立,

而$f(x)= x^{m+1}$时公式近似成立, ó 求积公式的代数精度为m次.

证明: 必要性显然.下证充分性

∵对任意次数低/等于m的多项式$P_m(x)=a_0+a_1x+ a_2x^2+\dots+ a_mx^m$,

由于求积公式 $\int_a^b f(x)dx \approx \sum_{i=0}^{n} A_i f(x_i)$ **对于** $f(x)=1,x,x^2,\dots,x^m$**时精确成立**

$$\therefore \int_a^b 1dx = \sum_{k\neq0}^{n} A_k, \quad \int_a^b xdx = \sum_{k\neq0}^{n} A_k x_k, \quad \dots, \int_a^b x^m dx = \sum_{k\neq0}^{n} A_k x_k^m$$

$$\int_a^b P_m(x)dx = a_0\int_a^b dx + a_1\int_a^b xdx + \dots + a_m\int_a^b x^m dx = a_0\sum_{k=0}^{n} A_k + a_1\sum_{k=0}^{n} A_k x_k + \dots + a_m\sum_{k=0}^{n} A_k x_k^m$$

$$= \sum_{k=0}^{n} A_k (a_0 + a_1 x_k + \dots + a_m x_k^m) = \sum_{k=0}^{n} A_k P_m(x_k)$$

∴求积公式对$P_m(x)$**精确成立. 但对m+1次多项式, 公式近似成立**

($R\neq0$), 由定义知该公式的代数精度是m次。

例 验证梯形公式的代数精度为1.

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

令$f(x)=1$ $左=\int_a^b 1dx = b-a, 右=\frac{b-a}{2}[1+1]=b-a, 左=右$

公式对 $f(x)=1$精确成立.

令$f(x)=x$ $\int_a^b xdx = \frac{b^2-a^2}{2}, 右=\frac{b-a}{2}[a+b]=\frac{b^2-a^2}{2}, 左=右$

公式对 $f(x)=x$精确成立

令$f(x)=x^2$ $左=\int_a^b x^2dx = \frac{b^3-a^3}{3}, 右=\frac{b-a}{2}[a^2+b^2]\neq 左$

公式对 $f(x)=x^2$不再精确成立

∴梯形公式代数精度为1.

例 Simpson公式的代数精度为3

$$\int_a^b f(x)dx \approx \frac{b-a}{6}[f(a)+4f(\frac{a+b}{2})+f(b)]$$

Review

机械求积公式 $\int_a^b f(x)dx \approx \sum_{i=0}^{n} A_i f(x_i)$

$$R_n(f) = I - \sum_{i=0}^{n} A_i f(x_i)$$

定义 **若上述公式对所有次数不超过m的多项式$P_m(x)$都精确成立,即$R_n(P_m)=0$,而对某一个$m+1$次多项式$P_{m+1}(x)$近似成立,即$R_n(P_{m+1})\neq 0$.则称机械求积公式具有m次代数精度.**

定理 **对上述机械求积公式，代数精度为m次的充分必要条件是：当$f(x)=1,x,x^2,\ldots,x^m$时,求积公式精确成立，而$f(x)= x^{m+1}$时公式近似成立.**

求积公式的构造方法一

例 **设有求积公式** $\int_{-1}^{1} f(x)dx \approx A_0 f(-1) + A_1 f(0) + A_2 f(1)$

试确定系数 A_0,A_1,A_2,**使这个公式具有最高的代数精度.**

分析: 要确定公式中3个待定常数A_0,A_1,A_2，可令公式对$1,x,x^2$都精确成立.

解: 令$f(x)= 1, x, x^2$公式都精确成立,则

$$\begin{cases} A_0 + A_1 + A_2 = 2 \\ -A_0 + A_2 = 0 \\ A_0 + A_2 = \frac{2}{3} \end{cases}$$

解得$A_0=1/3,A_1=4/3,A_2=1/3$

∴该求积公式为

$$\int_{-1}^{1} f(x)dx \approx \frac{1}{3}[f(-1) + 4f(0) + f(1)]$$

易验证:$f(x)= x^3$时, 求积公式也精确成立

而$f(x)= x^4$时 $\int_{-1}^{1} x^4 dx = \frac{2}{5} \neq \frac{2}{3} = \frac{1}{3}[(-1)^4 + 4\times 0 + 1^4]$

∴该求积公式具有3次代数精度,它是[-1,1]上的Simpson公式.

一般,对于n+1给节点上的机械求积公式 $\int_a^b f(x)dx \approx \sum_{k=0}^{n} A_k f(x_k)$

若使其代数精度至少为n,则可确定A_k**,构造出求积公式.**

只需令上式对$f(x)=1, x, x^2,\ldots, x^n$ **都精确成立,则**

$$\begin{cases} A_0 + A_1 + A_2 + \ldots + A_n = b - a \\ A_0x_0 + A_1x_1 + \ldots + A_nx_n = \dfrac{b^2 - a^2}{2} \\ \ldots \qquad \ldots \\ A_0x_0^n + A_1x_1^n + \ldots + A_nx_n^n = \dfrac{b^{n+1} - a^{n+1}}{n+1} \end{cases} \qquad (3.4)$$

上面是关于$A_0, A_1, A_2, \ldots, A_n$**的线性方程组,**
其系数行列式为范德蒙行列式, 其值非零,
可求得唯一解.

Chapter 3
Numerical Integration
and Differentiation

求积公式的构造方法二——插值法

Problem 已知给定的一组节点$a\le x_0<x_1<x_2<\ldots<x_n\le b$
及函数值 $f(x_0),f(x_1),f(x_2),\ldots,f(x_n)$

构造:求积公式 $\int_a^b f(x)dx \approx \sum_{k=0}^{n} A_k f(x_k)$

思想: 构造$f(x)$**在n+1个插值节点上的Lagrange插值多项式**

$$P_n(x) = \sum_{k=0}^{n} y_k l_k(x) \text{ ,其中Lagrange插值基函数 } l_k(x) = \prod_{j=0,j\ne k}^{n} \frac{x - x_j}{x_k - x_j}$$

$$f(x) \approx P_n(x) \Rightarrow \int_a^b f(x)dx \approx \int_a^b P_n(x)dx$$

$$\mathbf{Q}\ \int_a^b P_n(x)dx = \int_a^b [\sum_{k=0}^{n} f(x_k)l_k(x)]dx = \sum_{k=0}^{n} f(x_k)\int_a^b l_k(x)dx$$

$$\therefore \int_a^b f(x)dx \approx \sum_{k=0}^{n} \left(\int_a^b l_k(x)dx\right) f(x_k) \qquad (*)$$

(*)式为所求的求积公式.(称为插值型求积公式)

求积系数 $A_k = \int_a^b l_k(x)dx$

考虑: 插值型求积公式(*)的代数精度是多少?

1. **∵任意次数≤n的多项式f(x),其n次Lagrange插值多项式**

$P_n(x)= f(x)$

$$\therefore \int_a^b f(x)dx = \int_a^b P_n(x)dx$$

∴插值型求积公式对f(x)精确成立，其至少具有n次代数精度.

2. **反之,假设** $\int_a^b f(x)dx \approx \sum_{k=0}^{n} A_k f(x_k)$ **至少具有n次代数精度.**

∴求积公式对任意次数≤n的多项式精确成立

又在$x_0, x_1, x_2,\dots, x_n$上的Lagrange插值基函数$l_k(x)$为n次多项式.

$$\therefore \int_a^b l_k(x)dx = \sum_{j=0}^{n} A_j l_k(x_j) \quad \text{而} \quad l_k(x_j) = \delta_{kj} \begin{cases} 1 & k=j \\ 0 & k \neq j \end{cases}$$

$$\therefore \int_a^b l_k(x)dx = A_k$$

∴该求积公式就是(*),为插值型的.

综合1,2 有:

Th3.2 **求积公式** $\int_a^b f(x)dx \approx \sum_{k=0}^{n} A_k f(x_k)$

至少具有n次代数精度的充要条件是：它是插值型的.

小结:已知f(x)的函数表

x_i互异,$x_i \in [a,b]$

x	x_0	x_1	...	x_n
y	$f(x_0)$	$f(x_1)$	...	$f(x_n)$

构造其求积公式，有两种方法:

1. **解线性方程组, 求A_k**
2. **利用插值型公式**

$$A_k = \int_a^b l_k(x)dx$$

下面介绍一种特殊的插值型求积公式:等距节点的求积公式.

对于[a,b]中的n+1个互异节点$x_0, x_1, x_2, \ldots, x_n$.

可构造插值型求积公式: $\int_a^b f(x)dx \approx \sum_{k=0}^{n} A_k f(x_k)$ n次代数精度.

$$A_k = \int_a^b l_k(x)dx = \int_a^b \frac{(x-x_0)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_1)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}dx$$

现在取$x_0, x_1, x_2, \ldots, x_n$ 为[a,b]的n等分点.

即 $x_k = a+kh \ (k=0,1,\ldots,n),\ h=\frac{b-a}{n}=x_k-x_{k-1}$,则

$$A_k = \int_a^b \prod_{\substack{j=0\\ j\neq k}}^{n} \frac{(x-x_j)}{(x_k-x_j)}dx \overset{x=a+th}{=\!=\!=\!=} \int_0^n \prod_{\substack{j=0\\ j\neq k}}^{n} \frac{(t-j)}{(k-j)}hdt$$

$$= \int_0^n \frac{t(t-1)\cdots(t-k+1)(t-k-1)\cdots(t-n)}{k(k-1)\cdots(k-k+1)(k-k-1)\cdots(k-n)}hdt$$

$$A_k = \frac{(-1)^{n-k}h}{k!(n-k)!}\int_0^n t(t-1)\cdots(t-k+1)(t-k-1)\cdots(t-n)dt$$

$$= (b-a)\frac{(-1)^{n-k}}{n\cdot k!(n-k)!}\int_0^n \prod_{\substack{j=0\\ j\neq k}}^{n}(t-j)dt \ @\ (b-a)C_k$$

其中 $C_k = \frac{(-1)^{n-k}}{n\cdot k!(n-k)!}\int_0^n \prod_{\substack{j=0\\ j\neq k}}^{n}(t-j)dt$ 称为Cotes系数 .

$$\int_a^b f(x)dx \approx \sum_{k=0}^{n}(b-a)C_k f(x_k) = (b-a)\sum_{k=0}^{n}C_k f(x_k)$$

称 $I_n = (b-a)\sum_{k=0}^{n} C_k f(x_k)$ 为n阶Newton–Cotes公式.

注: Newton–Cotes公式为等距节点、插值型求积公式。

Cotes系数

$$C_k = \frac{(-1)^{n-k}}{n \cdot k!(n-k)!}\int_0^n \prod_{\substack{j=0\\ j\neq k}}^{n}(t-j)dt$$

注: Cotes**系数不仅与函数**f(x)**无关，而且与积分区间[a,b]无关。**

例:n=1**时**，$C_0^{(1)} = \frac{(-1)^1}{0!\ 1!}\int_0^1 (t-1)dt = \frac{1}{2}$ $C_1^{(1)} = \frac{(-1)^0}{1!\ 0!}\int_0^1 (t-0)dt = \frac{1}{2}$

例:n=3**时**，

$$C_0^{(3)} = \frac{(-1)^3}{0!\ 3!\times 3}\int_0^3 (t-1)(t-2)(t-3)dt = \frac{1}{8} \qquad C_1^{(3)} = \frac{(-1)^3}{1!\ 2!\times 3}\int_0^3 (t-0)(t-2)(t-3)dt = \frac{3}{8}$$

$$C_2^{(3)} = \frac{(-1)^3}{2!\ 1!\times 3}\int_0^3 (t-0)(t-1)(t-3)dt = \frac{3}{8} \qquad C_3^{(3)} = \frac{(-1)^3}{3!\ 0!\times 3}\int_0^3 (t-0)(t-1)(t-2)dt = \frac{1}{8}$$

当n=0,1.2,...8**时,** Cotes**系数见书本上第**65**页**

Cotes系数

n	$C_0(n)$	$C_1(n)$	$C_2(n)$	$C_3(n)$	$C_4(n)$	$C_5(n)$	$C_6(n)$	$C_7(n)$	$C_8(n)$
1	1/2	1/2							
2	1/6	4/6	1/6						
3	1/8	3/8	3/8	1/8					
4	7/90	16/45	2/15	16/45	7/90				
5	19/288	25/96	25/144	25/144	25/96	19/288			
6	41/840	9/35	9/280	34/105	9/280	9/35	41/840		
7	751/17280	3577/17280	1323/17280	2989/17280	2989/17280	1323/17280	3577/17280	751/17280	
8	989/28350	5888/28350	−928/28350	10496/28350	−4540/28350	10496/28350	−928/28350	5888/28350	989/28350

Cotes系数的性质

性质1. Cotes系数的和等于1，即 $\sum_{k=0}^{n} C_k^{(n)} = 1$

证明: 设f(x)=1. 则使用n次多项式插值时：f(x)=P_n(x)=1.

$$\therefore \int_a^b p_n(x)dx = \int_a^b f(x)dx = \int_a^b 1dx = b-a$$

$$而\int_a^b p_n(x)dx = (b-a)\sum_{k=0}^{n} C_k^{(n)} f(x_k) = (b-a)\sum_{k=0}^{n} C_k^{(n)}$$

$$\therefore \sum_{k=0}^{n} C_k^{(n)} = 1$$

性质2. Cotes系数具有对称性，即$C_k(n)=C_{n-k}(n)$,k=0,1,...,n.

性质3. 对n≤7时, $C_k(n)$都是正数, n≥8时不成立.

低阶Newton–Cotes公式

n=1时, $I_1 = (b-a)[\frac{1}{2}f(a) + \frac{1}{2}f(b)] = \frac{1}{2}(b-a)[f(a)+f(b)]$

此即梯形公式, 即 $T = I_1 = \frac{1}{2}(b-a)[f(a)+f(b)]$

n=2时,

$$I_2 = (b-a)[\frac{1}{6}f(a) + \frac{2}{3}f(\frac{a+b}{2}) + \frac{1}{6}f(b)] = \frac{1}{6}(b-a)[f(a) + 4f(\frac{a+b}{2}) + f(b)]$$

此即Simpson公式 $S = I_2 = \frac{1}{6}(b-a)[f(a) + 4f(\frac{a+b}{2}) + f(b)]$

n=4时, 4阶Newton–Cotes公式称为Cotes公式.

$$C = I_4 = \frac{1}{90}(b-a)[7f(x_0) + 32f(x_1) + 12f(x_2) + 32f(x_3) + 7f(x_4)]$$

注: 梯形公式由线性插值推导而得.

Simpson公式由抛物插值推导而得.

Cotes公式由4次插值推导而得.

低阶Newton–Cotes公式的截断误差或余项

$$\because \int_a^b f(x)dx \approx \int_a^b P_n(x)dx$$

$\therefore R = I - I_n = \int_a^b [f(x) - P_n(x)]dx$ 为Newton-Cotes**公式的余项**

又$f(x) - P_n(x) = \dfrac{f^{(n+1)}(\xi_x)}{(n+1)!}(x-x_0)(x-x_1)\cdots(x-x_n) \quad \xi_x \in [a,b]$

$$\therefore R = I - I_n = \int_a^b \frac{f^{(n+1)}(\xi_x)}{(n+1)!}(x-x_0)(x-x_1)\cdots(x-x_n)dx$$

其中 $x_k = x_0 + kh \quad (k=0,1,\ldots,n) \quad x_0 = a$

对以上积分进行变量代换$x=x_0+th$，**并使用积分定理，有**

Chapter 3
Numerical Integration
and Differentiation

低阶Newton–Cotes公式的截断误差或余项

$$\therefore R = I - I_n = \int_a^b \frac{f^{(n+1)}(\xi_x)}{(n+1)!}(x-x_0)(x-x_1)\cdots(x-x_n)dx$$

其中 $x_k = x_0 + kh \quad (k=0,1,\ldots,n) \quad x_0 = a$

Th: 若$f(x)$**在**$[a,b]$**有连续的**$n+2$**阶导数,则**Newton-Cotes**公式余项为**

$$R = \begin{cases} \dfrac{h^{n+2} \cdot f^{(n+1)}(\xi)}{(n+1)!}\displaystyle\int_0^n t(t-1)\cdots(t-n)dt & n \text{ 是奇} \\ \dfrac{h^{n+3} \cdot f^{(n+2)}(\xi)}{(n+2)!}\displaystyle\int_0^n (t-\frac{n}{2})t(t-1)\cdots(t-n)dt & n \text{ 是偶} \end{cases} \quad \text{其中} \quad h = \frac{b-a}{n} \quad \xi \in [a,b]$$

由定理易知:

n**阶**Newton-Cotes**公式至少有**n**次代数精度(因该公式为插值型);**
而当n**为偶数时,它有**n+1**次代数精度.**

积分中值定理：若f(x)在[a，b]上连续，g(x)是在[a，b]上保号的可积函数，则存在ξ∈(a,b)使 $\int_a^b f(x)g(x)dx = f(x)\int_a^b g(x)dx$

梯形公式的余项(n=1) $R_T = I - T = \int_a^b \frac{f^{(2)}(\xi_x)}{2!}(x-a)(x-b)dx$

$$Q \begin{cases} 若 f^{(2)}(x) \in C[a,b] \\ (x-a)(x-b) \le 0 \quad x \in [a,b] \end{cases}$$

$$\exists \xi \in (a,b) \qquad R_T = \frac{f^{(2)}(\xi)}{2}\int_a^b (x-a)(x-b)dx$$

$$R_T = -\frac{f^{(2)}(\xi)}{12}(b-a)^3 = O(b-a)^3 \quad \xi \in (a,b)$$

注：此结论可由余项定理直接得到

Chapter 3
Numerical Integration
and Differentiation

Simpson公式的余项

直接由定理得Simpson公式(n=2)的余项

$$R_S = I - S = \frac{h^5 f^{(4)}(\xi)}{4!}\int_0^2 (t-1)t(t-1)(t-2)dt = -\frac{h^5}{90}f^{(4)}(\xi)$$

$$R_S = -\frac{(b-a)}{180}\left(\frac{b-a}{2}\right)^4 f^{(4)}(\xi) = O(b-a)^5$$

分析：Simpson公式是由a，b及其中点c进行抛物插值得到的，其代数精度是3，为证明以上余项公式，构造f(x)的3次插值多项式$H_3(x)$，即考虑如下插值问题：

已知f(x)的函数表 $c = \frac{1}{2}(a+b)$

x	a	c	b ...
y	f(a)	f(c)	f(b)...
f′(x)		f′(c)	

求 f(x)的Hermite插值多项式$H_3(x)$,使

$H_3(a)=f(a)$, $H_3(b)=f(b)$, $H_3(c)=f(c)$, $H'_3(c)=f'(c)$.

Simpson公式的余项的证明

证明: $f(x)-H_3(x)$有根 a、b、c（二重），易知其插值余项

$$f(x)-H_3(x)=\frac{f^{(4)}(\eta)}{4!}(x-a)(x-b)(x-c)^2$$ η依赖于x,且 $\eta\in[a,b]$

又Simpson公式代数精度为3.

$$\therefore\int_a^b H_3(x)dx=\frac{b-a}{6}[H_3(a)+4H_3(c)+H_3(b)]=\frac{b-a}{6}[f(a)+4f(c)+f(b)]=S$$

$$R_S=I-S=\int_a^b f(x)dx-\int_a^b H_3(x)dx=\int_a^b[f(x)-H_3(x)]dx$$

$$\therefore R_S=\int_a^b\frac{f^{(4)}(\eta)}{4!}(x-a)(x-b)(x-c)^2dx$$ 根据积分中值定理

$$R_S=\frac{f^{(4)}(\xi)}{4!}\int_a^b(x-a)(x-b)(x-c)^2dx=-\frac{b-a}{180}\left(\frac{b-a}{2}\right)^4 f^{(4)}(\xi),\quad \xi\in(a,b).$$

Cotes公式的余项(n=4，代数精度为5)

$$R_C=I-C=-\frac{8}{945}h^7 f^{(6)}(\xi)=-\frac{2(b-a)}{945}\left(\frac{b-a}{4}\right)^6 f^{(6)}(\xi)=O(b-a)^7,\quad \xi\in[a,b]$$

例. 分别用梯形公式,Simpson公式,Cotes公式和n=8的Newton-Cotes公式计算 $\int_{0.5}^{1}\sqrt{x}dx$ $(=0.430964406)$

解:(1)利用梯形公式 $\int_{0.5}^{1}\sqrt{x}dx\approx\frac{1-0.5}{2}(\sqrt{0.5}+\sqrt{1})\approx 0.4267767$

(2)利用Simpson公式 $\int_{0.5}^{1}\sqrt{x}dx\approx\frac{1-0.5}{6}(\sqrt{0.5}+4\sqrt{0.75}+\sqrt{1})\approx 0.4309403$

(3) 利用Cotes公式

$$\int_{0.5}^{1}\sqrt{x}dx\approx\frac{1-0.5}{90}\left(7\sqrt{0.5}+32\sqrt{\frac{5}{8}}+12\sqrt{\frac{6}{8}}+32\sqrt{\frac{7}{8}}+7\sqrt{1}\right)\approx 0.43096407$$

(4)利用n=8的Newton-Cotes公式计算

$$\int_{0.5}^{1}\sqrt{x}dx\approx\frac{1}{2\times 28350}\left[989(\sqrt{0.5}+\sqrt{1})+5888\left(\sqrt{\frac{9}{16}}+\sqrt{\frac{15}{16}}\right)\right.$$
$$\left.-928\left(\sqrt{\frac{10}{16}}+\sqrt{\frac{14}{16}}\right)+10496\left(\sqrt{\frac{11}{16}}+\sqrt{\frac{13}{16}}\right)-4540\sqrt{\frac{12}{16}}\right]$$
$$\approx 0.430964406$$

n较大时,结果较精确

Review

机械求积公式 $\int_a^b f(x)dx \approx \sum_{i=0}^{n} A_i f(x_i)$ $\quad R_n(f) = I - \sum_{i=0}^{n} A_i f(x_i)$

上述求积公式代数精度为m次 ó 当$f(x)=1,x,x^2,\ldots,x^m$**时,求积公式精确成立, 而**$f(x)= x^{m+1}$**时公式近似成立.**

插值型求积公式 $A_k = \int_a^b l_k(x)dx$

n阶Newton–Cotes**公式** $I_n = (b-a)\sum_{k=0}^{n} C_k f(x_k)$

$$R_T = -\frac{f^{(2)}(\xi)}{12}(b-a)^3 = O(b-a)^3 \quad \xi \in (a,b)$$

$$R_C = I - C = O(b-a)^7$$

$$R_S = -\frac{(b-a)}{180}\left(\frac{b-a}{2}\right)^4 f^{(4)}(\xi) = O(b-a)^5$$

Newton-Cotes公式的算法数值稳定性

数值稳定性 **指舍入误差在运算中的传播强度,**
即舍入误差对计算结果的影响程度.

Def 3.2 **设给定的算法在执行某一步时产生误差**ε**,**
相继的n**步运算后结果的误差为**e_n **,且若其仅由**ε**引起,**

(1) **如**$|e_n| \approx Cn\varepsilon$**,其中**C**是与**n**无关的常数,**
则称误差的增长是线性级的.

(2) **如**$|e_n| \approx K^n\varepsilon$**, 其中**$K>1$**为常数,**
则称误差的增长是指数级的.

注: 误差线性级增长是可以控制的,这样的算法是数值稳定的, 其运算结果可靠.
误差指数级增长难于控制,这样的算法是数值不稳定的, 其运算结果不可靠.

Newton-Cotes公式的数值稳定性

$$I_n = (b-a)\sum_{k=0}^{n} C_k f(x_k)$$

若计算函数值$f(x_k)$有舍入误差$\varepsilon_k = \tilde{f}_k - f(x_k)$，$k=0,1,2,\ldots,n$.

设计算C_k没有误差,计算过程的舍入误差也不考虑,
则由ε_k引起的计算结果的误差为

$$e_n = (b-a)\sum_{k=0}^{n} C_k \tilde{f}_k - (b-a)\sum_{k=0}^{n} C_k f(x_k) = (b-a)\sum_{k=0}^{n} C_k \varepsilon_k$$

令 $\varepsilon = \max_{0\le k\le n} |\varepsilon_k|$ 应用 $\sum_{k=0}^{n} C_k = 1$,则

$$|e_n| \le (b-a)\sum_{k=0}^{n} |C_k|\cdot|\varepsilon_k| \le (b-a)\cdot\varepsilon\cdot\sum_{k=0}^{n}|C_k|$$

（1）当$n\le 7$时， $C_k > 0 \Rightarrow |e_n| \le (b-a)\varepsilon$

此时e_n有界,舍入误差的增长受到控制, 公式是数值稳定的.

（2）当$n \geqslant 8$时， C_k有正有负 ,$\sum_{k=0}^{n}|C_k| > 1$且随n增大而增大,

∴ 此时Newton-Cotes公式不能保证数值稳定性.

小结

$$I_n = (b-a)\sum_{k=0}^{n} C_k f(x_k)$$

$$R = \begin{cases} \dfrac{h^{n+2}\cdot f^{(n+1)}(\xi)}{(n+1)!}\displaystyle\int_0^n t(t-1)\cdots(t-n)dt & n\ \text{是奇} \\ \dfrac{h^{n+3}\cdot f^{(n+2)}(\xi)}{(n+2)!}\displaystyle\int_0^n (t-\frac{n}{2})t(t-1)\cdots(t-n)dt & n\ \text{是偶} \end{cases}$$

其中 $h = \dfrac{b-a}{n}$ $\xi \in [a,b]$

注:n越大,其Newton-Cotes公式I_n的截断误差越小,
那么是否n越大越好呢?

否! ① n大,计算量大,误差积累越严重.
② $n\ge 8$时,不能保证数值稳定性.

∴一般采用低阶的Newton-Cotes公式(T,S,C).

HW:
p.83 # 4(1) ,
6, 9(Simpson)

但使用T,S,C公式, 要控制其截断误差R. How?

Review

机械求积公式 $\int_a^b f(x)dx \approx \sum_{k=0}^{n} A_k f(x_k)$

构造其求积公式，有两种方法:

1. **解线性方程组, 求**A_k
2. **利用插值型公式** $A_k = \int_a^b l_k(x)dx$

机械求积公式至少具有n次代数精度的充要条件是：它是插值型的.

n**阶**Newton–Cotes**公式** $I_n = (b-a)\sum_{k=0}^{n} C_k f(x_k)$

$$R = \begin{cases} \dfrac{h^{n+2}\cdot f^{(n+1)}(\xi)}{(n+1)!}\displaystyle\int_0^n t(t-1)\cdots(t-n)dt & n \text{ 是奇} \\ \dfrac{h^{n+3}\cdot f^{(n+2)}(\xi)}{(n+2)!}\displaystyle\int_0^n (t-\frac{n}{2})t(t-1)\cdots(t-n)dt & n \text{ 是偶} \end{cases}$$

其中 $h = \dfrac{b-a}{n}$ $\xi \in [a,b]$

低阶Newton-Cotes**公式数值稳定性好.**

复化求积法

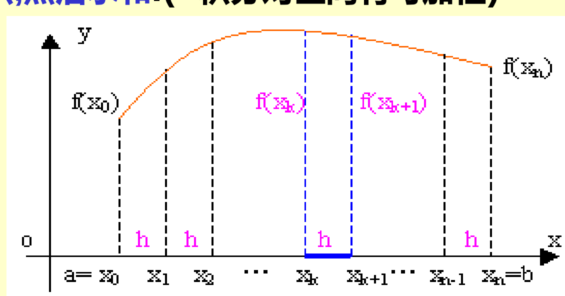
- **复化求积: 将积分区间划分成若干小区间,在每个小区间上构造相应的低阶求积公式,再把它们加起来作为整个区间的求积公式.----分段求积,然后求和.(∵积分对区间有可加性)**

复化梯形公式

把[a,b] n**等分,**

分点$x_k=a+kh$,

$k=0\sim n$,$h=\dfrac{b-a}{n}$



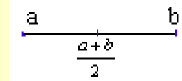
$$I=\int_a^b f(x)dx=\sum_{k=0}^{n-1}\int_{x_k}^{x_{k+1}} f(x)dx$$

$$\int_{x_k}^{x_{k+1}} f(x)dx \approx \frac{x_{k+1}-x_k}{2}[f(x_k)+f(x_{k+1})]$$

$$\therefore I \approx \sum_{k=0}^{n-1}\frac{h}{2}[f(x_k)+f(x_{k+1})]=\frac{h}{2}[f(a)+2\sum_{k=1}^{n-1} f(x_k)+f(b)]=T_n$$

复化Simpson公式

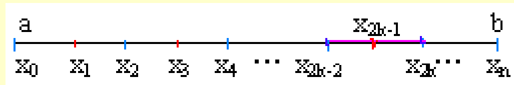
$$S = I_2 = \frac{1}{6}(b-a)[f(a) + 4f(\frac{a+b}{2}) + f(b)]$$



Ø构造复化Simpson公式时,应如何划分[a,b]?

必须将[a,b]偶数等分.

令n=2m ，m为正整数



$h = \frac{b-a}{n}$ 对每个区间$[x_{2k-2}, x_{2k}]$应用Simpson公式.(k=1,2,...m)

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

$$\int_{x_{2k-2}}^{x_{2k}} f(x)dx \approx \frac{1}{6}(x_{2k} - x_{2k-2})[f(x_{2k-2}) + 4f(x_{2k-1}) + f(x_{2k})]$$

$$\therefore\ I \approx \frac{h}{3}\sum_{k=1}^{m}[f(x_{2k-2}) + 4f(x_{2k-1}) + f(x_{2k})]$$

$$\therefore\ I \approx \frac{h}{3}[f(a) + 4\sum_{k=1}^{m} f(x_{2k-1}) + 2\sum_{k=1}^{m-1} f(x_{2k}) + f(b)] = S_n$$

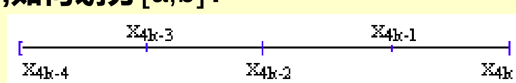
复化Cotes公式求积

Ø Cotes公式 (5点公式)

$$C = I_4 = \frac{b-a}{90}[7f(x_0) + 32f(x_1) + 12f(x_2) + 32f(x_3) + 7f(x_4)]$$

Ø构造复化Cotes公式时,如何划分[a,b]?

n=4m ，m为正整数



Ø复化Cotes公式如何推导?

$$C_n = \frac{4h}{90}[7f(a) + 32\sum_{k=1}^{m} f(x_{4k-3}) + 12\sum_{k=1}^{m} f(x_{4k-2})$$

$$+ 32\sum_{k=1}^{m} f(x_{4k-1}) + 14\sum_{k=1}^{m-1} f(x_{4k}) + 7f(b)]$$

小结:（1）对数值求积进行区间分段处理是一种有效的手段,
可以对许多公式进行复化处理.
（2）复化求积公式仍然是机械求积公式。

复化求积公式的截断误差

T_n的积分余项

在每个小区间$[x_k,x_{k+1}]$上,梯形公式的积分余项为

$$\int_{x_k}^{x_{k+1}} f(x)dx-\frac{h}{2}[f(x_k)+f(x_{k+1})]=-\frac{1}{12}f''(x_k)h^3 \qquad x_k\in(x_k,x_{k+1})$$

$$I-T_n=\sum_{k=0}^{n-1}\{\int_{x_k}^{x_{k+1}} f(x)dx-\tfrac{h}{2}[(f(x_k)+f(x_{k+1})]\}=\sum_{k=0}^{n-1}[-\frac{1}{12}h^3f''(x_k)]$$

而由定积分的定义和Newton-Leibnitz公式可得

$$\sum_{k=0}^{n-1}hf''(x_k)\approx\int_a^b f''(x)dx=f'(b)-f'(a)$$

$$\Rightarrow I-T_n\approx-\frac{h^2}{12}[f'(b)-f'(a)]=O(h^2)$$

类似可得

$$I-S_n\approx-\frac{1}{180}h^4[f'''(b)-f'''(a)]=O(h^4)$$

$$I-C_n\approx-\frac{2}{945}h^6[f^{(5)}(b)-f^{(5)}(a)]=O(h^6)$$

复化求积公式的截断误差(续)

$$I-T_n=\sum_{k=0}^{n-1}[\int_{x_k}^{x_{k+1}} f(x)dx-\tfrac{h}{2}(f(x_k)+f(x_{k+1}))]=\sum_{k=0}^{n-1}[-\frac{1}{12}h^3f''(x_k)]$$

$$\therefore\ I-T_n=-\frac{n}{12}h^3\sum_{k=0}^{n-1}\frac{f''(x_k)}{n}$$

如果$f''(x)\in C[a,b]$，**由连续函数的平均值定理**

$$I-T_n=-\frac{b-a}{12}h^2f''(x)=O(h^2),x\in(a,b)$$

类似可得

如果$f^{(4)}(x)\in C[a,b]$ $\qquad I-S_n=-\frac{b-a}{180}h^4f^{(4)}(x)$

如果$f^{(6)}(x)\in C[a,b]$ $\qquad I-C_n=-\frac{2}{945}(b-a)h^6f^{(6)}(x)$

小结

名称	阶	符号	代数精度	余项
梯形公式	1	T,I_1	1	$I-T=-\frac{(b-a)^3}{12}f''(x)=O(b-a)^3$
Simpson	2	S,I_2	3	$I-S=-\frac{b-a}{180}(\frac{b-a}{2})^4 f^{(4)}(x)=O(b-a)^5$
Cotes	4	C,I_4	5	$I-C=-\frac{2(b-a)}{945}(\frac{b-a}{12})^6 f^{(6)}(x)=O(b-a)^7$

代数精度+2

复化公式

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

$$I-T_n\approx O(h^2)$$
$$I-S_n\approx O(h^4)\qquad h=\frac{b-a}{n}$$
$$I-C_n\approx O(h^6)$$

代数精度+1

例题

例:分别用3种复化求积公式计算积分 $I=\int_0^1\frac{\sin x}{x}dx.$

要求误差不超过 $e=\frac{1}{2}\times 10^{-6}$

n=?

解: $\because f(x)=\frac{\sin x}{x}=\int_0^1\cos tx\,dt$

$$f^{(k)}(x)=\frac{d^k}{dx^k}(\frac{\sin x}{x})=\int_0^1\frac{d^k}{dx^k}(\cos tx)dt=\int_0^1 t^k\cos(tx+\frac{k\pi}{2})dt$$

$$\therefore \max_{0\le x\le 1}|f^{(k)}(x)|\le\int_0^1\max_{0\le x\le 1}|t^k\cos(tx+\tfrac{k\pi}{2})|dt$$

$$\le\int_0^1 t^k dt=\frac{1}{k+1}$$

$$\therefore \max_{0\le x\le 1}|f^{(2)}(x)|\le\frac{1}{3}.\quad \max_{0\le x\le 1}|f^{(4)}(x)|\le\frac{1}{5}.\quad \max_{0\le x\le 1}|f^{(6)}(x)|\le\frac{1}{7}.$$

1. 用复化梯形公式

$\because \ |I-T_n|=|-\frac{b-a}{12}h^2f''(x)|\le\frac{1}{12}h^2\cdot\max_{0\le x\le1}|f''(x)|\le\frac{h^2}{36} \quad (x\in[0,1])$

由 $e=\frac{1}{2}\times10^{-6}$ 得 $\frac{h^2}{36}<\frac{1}{2}\times10^{-6}$ ， $h=\frac{1}{n}$. $\therefore n>\frac{1}{\sqrt{18\times10^{-6}}}\approx 235.7$

$\therefore$取$n=236$. $h=\frac{1}{236}$.

$I\approx T_{236}=\frac{1}{2\times236}[1+2\sum_{k=1}^{235}\frac{\sin\frac{k}{236}}{\frac{k}{236}}+\sin 1]\approx 0.94608262.$

2. 用复化Simpson公式

$\because \ |I-S_n|=|-\frac{b-a}{180}h^4f^{(4)}(x)| \quad x\in[0,1] \qquad \therefore \ n>\frac{1}{\sqrt[4]{456\times10^{-6}}}\approx 6.8$

$\le\frac{1}{900}h^4<\frac{1}{2}\times10^{-6}$ ， $h=\frac{1}{n}$ 取 $n=8$ $(n\neq7?)$ $h=\frac{1}{8}$

$I\approx S_8=\frac{1}{24}[1+4\sum_{k=1}^{4}\frac{8}{2k-1}\sin\frac{2k-1}{8}+2\sum_{k=1}^{3}\frac{8}{2k}\sin\frac{2k}{8}+\sin 1]\approx 0.94608331$

3. 用复化Cotes公式

$\because |I-C_n|=|-\frac{2(b-a)}{945}h^6f^{(6)}(\xi)| \ ,x\in[0,1] \quad \therefore n>(\frac{4}{6615}\times10^6)^{\frac{1}{6}}\approx 2.9$

$\le\frac{2}{6615}h^6<\frac{1}{2}\times10^{-6}$ ， $h=\frac{1}{n}$ 取$n=4$, $h=0.25$

$I\approx C_4=\frac{1}{90}[7+7\sin1+32\times4\sin\frac{1}{4}+12\times2\sin\frac{1}{2}+\frac{32\times4}{3}\sin\frac{3}{4}]\approx 0.946083004$

事实上, I准确到小数点后7位的值是 $I=0.9460831$.

$\therefore |I-T_{236}|=0.48\times10^{-6}.\ |I-S_8|=0.21\times10^{-6}.\ |I-C_4|=0.096\times10^{-6}$

∴按同样精度要求,用复化Cotes公式优于其他两种算法,

其计算量最小,精度最高.

因此预先确定步长时,宜选用复化Cotes公式,其计算效率高.

例. 若用复化梯形公式,复化Simpson公式计算 $\int_{0.5}^{1}\sqrt{x}dx$，**要使精度达到** $\frac{1}{2}\times10^{-4}$ **,问n各取多少?**

解:

$$R_T=-\frac{b-a}{12}h^2f''(\eta)$$

$$R_s=-\frac{b-a}{180}h^4f^{(4)}(\eta)$$

$$\eta\in(a,b)$$

$$|f''(x)|=|-\tfrac{1}{4}x^{-\frac{3}{2}}|\le\tfrac{1}{4}(\tfrac{1}{2})^{-\frac{3}{2}}=\tfrac{\sqrt{2}}{2}<0.8$$

$$|f^{(4)}(x)|=|-\tfrac{15}{16}x^{-\frac{7}{2}}|\le\tfrac{15}{16}(\tfrac{1}{2})^{-\frac{7}{2}}=\tfrac{15\sqrt{2}}{2}<10.7$$

$$|R_T|\le\tfrac{1}{24}h^2\cdot0.8=\tfrac{1}{24}\times(\tfrac{0.5}{n})^2\cdot0.8\le\tfrac{1}{2}\times10^{-4}$$

$$n>12.91\qquad\therefore n=13$$

$$|Rs|<\frac{\frac{1}{2}}{180}(\tfrac{\frac{1}{2}}{n})^4\max|f^{(4)}(x)|<\tfrac{1}{2}\times10^{-4},$$

$$n^4>\frac{10.7\times10^4}{16\times180}\approx37.153$$

$$n>2.469\qquad\therefore n=4$$

Chapter 3
Numerical Integration
and Differentiation

用S_4计算,其分点为 $\frac{1}{2},\frac{5}{8},\frac{6}{8},\frac{7}{8},1,h=\frac{1}{8}$

$$S_4=\frac{h}{3}[f(\frac{1}{2})+4f(\frac{5}{8})+4f(\frac{7}{8})+2f(\frac{6}{8})+f(1)]$$

$$=\frac{1}{24}[\sqrt{\frac{1}{2}}+4\sqrt{\frac{5}{8}}+4\sqrt{\frac{7}{8}}+2\sqrt{\frac{6}{8}}+1]$$

$$=\frac{1}{24}[\sqrt{0.5}+\sqrt{10}+\sqrt{14}+\sqrt{3}+1]$$

$$=\frac{1}{24}[0.70711+3.16228+3.74166+1.73205+1]\approx0.4310$$

$$\int\sqrt{x}dx=\int\frac{\sqrt{x}}{\frac{1}{2}\cdot x^{-\frac{1}{2}}}d\sqrt{x}=\int2\left(\sqrt{x}\right)^2d\sqrt{x}\overset{y=\sqrt{x}}{=\!=\!=}\int2y^2dy=\tfrac{2}{3}y^3+C=\tfrac{2}{3}x^{\frac{3}{2}}+C$$

$$\therefore\int_{\frac{1}{2}}^{1}\sqrt{x}dx=\tfrac{2}{3}x^{\frac{3}{2}}\Big|_{\frac{1}{2}}^{1}=\tfrac{2}{3}[1-(\tfrac{1}{2})^{\frac{3}{2}}]=\tfrac{2}{3}-\tfrac{1}{3}\sqrt{\tfrac{1}{2}}\approx\tfrac{1}{3}(2-0.70711)\approx0.4310$$

Romberg求积算法

Problem: 计算 $I=\int_a^b f(x)dx$ 使误差 $\varepsilon<10^{-8}$

如用复化公式求积分,则必须事先确定n=? (h=?).

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

（1）h大,不精确

$$I\text{-}S_n=O(h^4)\qquad I-C_n=O(h^6)$$

（2）h小,计算量大

而用误差公式确定h,有如下弊端:

（1）含f(x)高阶导数,估计 $|f^{(k)}(x)|$ 的最大值较困难.

（2）用此法估计的h很保守,偏小,增大了计算量.

实际上,可以让计算机自动选择数值积分的步长h.

即采用变步长求积公式.

变步长的梯形公式

变步长的思想:计算 $\int_a^b f(x)dx$ 的数值积分,先确定初始步长h,按某一复化公式求积,再将步长折半为h/2后,利用同一公式求积,反复进行,直到达到精度要求.

两个问题: （1）如何知道达到了精度要求 （ I未知, $I\text{-}T_n=?$）.

（2）步长折半前后的两次结果有何关系?

解1: 误差的事后估计法.

$$h=\frac{b-a}{n},\quad n\text{等分}[a,b],\quad I\text{-}T_n=O(h^2)$$

$$\frac{h}{2}=\frac{b-a}{2n},\quad 2n\text{等分}[a,b],\quad I\text{-}T_{2n}=O(\frac{h}{2})^2$$

$$\therefore\ \frac{I-T_{2n}}{I-T_n}=\frac{O(\frac{h}{2})^2}{O(h^2)}\approx\frac{1}{4}\qquad \therefore\ I-T_{2n}\approx\frac{1}{4}I-\frac{1}{4}T_n$$

$$\therefore\ \frac{3}{4}I-\frac{3}{4}T_{2n}\approx\frac{1}{4}T_{2n}-\frac{1}{4}T_n\ \Rightarrow\ I-T_{2n}\approx\frac{1}{3}(T_{2n}-T_n)$$

定理: 若 $|T_{2n}-T_n|<\varepsilon$,则 $|I-T_{2n}|<\varepsilon$

因此,对给定的误差限ε,计算机自动选择步长如下:

Algorithm: Step1 h=b-a; k=2; 算T_1; T_2;

Step2 while($|T_2-T_1|\geq\varepsilon$) {k=2k; 算T_k;

$T_1=T_2$; $T_2=T_k$}

Step3. $I\approx T_2$.

解2: 梯形公式的递推化.

(1) 先将[a,b]n等分 ,$h=\frac{b-a}{n}$,分点$x_k=a+kh$,k=0,1,...n.

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

(2) 将步长减半,将每个小区间$[x_k,x_{k+1}]$二等分,其中点为

$x_{k+\frac{1}{2}}=a+(k+\frac{1}{2})h$,k=0,1,…n-1.

Chapter 3
Numerical Integration
and Differentiation

这时[a,b]被分为2n个长度为 $\frac{h}{2}$ 的小区间.

$$T_{2n}=\frac{\frac{h}{2}}{2}[f(a)+2\sum_{k=1}^{n-1}f(x_k)+2\sum_{k=0}^{n-1}f(x_{k+\frac{1}{2}})+f(b)]$$

$$T_{2n}=\frac{1}{2}\times\frac{h}{2}[f(a)+2\sum_{k=1}^{n-1}f(x_k)+f(b)]+\frac{h}{2}\sum_{k=0}^{n-1}f(x_{k+\frac{1}{2}})$$

$$\therefore T_{2n}=\frac{1}{2}T_n+\frac{h}{2}\sum_{k=0}^{n-1}f(x_{k+\frac{1}{2}})=\frac{1}{2}[T_n+h\sum_{k=0}^{n-1}f(x_{k+\frac{1}{2}})]\quad\text{……递推公式}$$

注: (1) 计算T_{2n}时,只需在T_n的基础上,再计算n个点 $x_{k+0.5}$(k=0,1,...n—1)处f(x)的函数值.

(2) 将递推公式代入Algorithm,便可编制变步长梯形算法的程序.

HUST

例3.4 **用变步长算法计算** $I=\int_0^1 \frac{\sin x}{x}dx$ **,并要求误差** $\varepsilon<\frac{1}{2}\times 10^{-2}$.

解: **(1)取**h=1,n=1.

$$T_1=\frac{1}{2}[f(0)+f(1)]=\frac{1}{2}(1+0.8414710)=0.9207355$$

(2) 将步长折半为 $\frac{1}{2}$ **,分点为** $0,\frac{1}{2},1$ **.则**

$$T_2=\frac{1}{2}T_1+\frac{1}{2}f(\frac{1}{2})=\frac{1}{2}(0.9207355+0.958810)=0.9397933$$

而 $|T_2-T_1|=0.190578\times 10^{-1}>\frac{1}{2}\times 10^{-2}$

(3) 将步长折半为 $\frac{1}{4}$ **,分点为** $0,\frac{1}{4},\frac{1}{2},\frac{3}{4},1$

$$T_4=\frac{1}{2}T_2+\frac{\frac{1}{2}}{2}[f(\frac{1}{4})+f(\frac{3}{4})]=\frac{1}{2}\times 0.9397933+\frac{1}{4}(0.9896158+0.9088516)=0.9445135$$

其中 $|T_4-T_2|=0.0047203\times 10^{-2}<\frac{1}{2}\times 10^{-2}$ $\quad\therefore I\approx T_4=0.9445135$

Overview

名称	阶	符号	代数精度	余项
梯形公式	1	T, I_1	1	$I-T=-\frac{(b-a)^3}{12}f''(x)=O(b-a)^3$
Simpson	2	S, I_2	3	$I-S=-\frac{b-a}{180}(\frac{b-a}{2})^4 f^{(4)}(x)=O(b-a)^5$
Cotes	4	C, I_4	5	$I-C=-\frac{2(b-a)}{945}(\frac{b-a}{12})^6 f^{(6)}(x)=O(b-a)^7$

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

$$I-T_n\approx O(h^2)$$
$$I-S_n\approx O(h^4) \qquad h=\frac{b-a}{n}$$
$$I-C_n\approx O(h^6)$$

$$T_{2n}=\frac{1}{2}T_n+\frac{h}{2}\sum_{k=0}^{n-1}f(x_{k+\frac{1}{2}})=\frac{1}{2}[T_n+h\sum_{k=0}^{n-1}f(x_{k+\frac{1}{2}})] \qquad I-T_{2n}\approx\frac{1}{3}(T_{2n}-T_n)$$

Romberg公式

用误差的事后估计法得到复化梯形公式的误差:

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

$\therefore$ T_{2n} **的误差约为** $\frac{1}{3}(T_{2n} - T_n)$

若将此误差补偿给 T_{2n} **,可以得到更精确的结果.**

$$\bar{T} = T_{2n} + \frac{1}{3}(T_{2n} - T_n) \qquad \therefore I \approx \bar{T} = \frac{4}{3}T_{2n} - \frac{1}{3}T_n$$

这是积分I的一个更好的近似值, 称为外推公式。

通过外推公式可以加快收敛。可以验证: $S_{2n} = \frac{4}{3}T_{2n} - \frac{1}{3}T_n$

上式的意义: 复杂公式S_{2n}可以用T_n表示.

为便于后续描述，上式常表示为（等号左边的下标视为编号）：

$$S_n = \frac{4}{3}T_{2n} - \frac{1}{3}T_n \qquad (3.21)$$

Chapter 3
Numerical Integration
and Differentiation

类似 $\quad$ Q $I - S_n = O(h^4)$

$$\therefore \frac{I - S_{2n}}{I - S_n} = \frac{O(\frac{h}{2})^4}{O(h^4)} \approx \frac{1}{16} \qquad \therefore I - S_{2n} \approx \frac{1}{16}(I - S_n)$$

注：步长折半后,误差是原来的 $\frac{1}{16}$ ！可得误差事后估计式

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

从而,也有外推公式 $\quad I \approx \frac{16}{15}S_{2n} - \frac{1}{15}S_n$

易证 $\qquad C_{2n} = \frac{16}{15}S_{2n} - \frac{1}{15}S_n$

C_{2n}可由Simpson公式步长二分前后两值的线性组合表示，也记为

$$C_n = \frac{16}{15}S_{2n} - \frac{1}{15}S_n \qquad (3.22)$$

HUST

$$\because \quad I - C_n = O(h^6)$$

$$\therefore \frac{I - C_{2n}}{I - C_n} = \frac{O(\frac{h}{2})^6}{O(h^6)} \approx \frac{1}{64}$$

可导出如下加速收敛的外推公式：Romberg公式.

$$R_n = \frac{64}{63}C_{2n} - \frac{1}{63}C_n \qquad (3.23)$$

注:（1）R_n是一种变步长梯形公式的外推公式,其收敛速度快。

（2）如何用Romberg公式计算 $I = \int_a^b f(x)dx$,并要求误差 $\varepsilon < \frac{1}{2}\times 10^{-10}$ ？

Romberg算法:将T_n序列加工成Romberg序列R_n,从而加速收敛。

Chapter 3
Numerical Integration
and Differentiation

$$\begin{array}{cccc} T_1 & & & \\ T_2 & S_1 & & \\ T_4 & S_2 & C_1 & \\ T_8 & S_4 & C_2 & R_1 \\ T_{16} & S_8 & C_4 & R_2 \\ T_{32} & S_{16} & C_8 & R_4 \\ \cdots & \cdots & \cdots & \cdots \end{array}$$

if $|R_2-R_1|<\varepsilon$ ，则R_2即为所求

else 计算R_4, 判断$|R_4-R_2|<\varepsilon$? ......

Romberg算法　　注:可达到任意精度.

(1)置k=0；精度要求ε; h=b-a ; $T_1=\frac{h}{2}[f(a)+f(b)]$;

(2) $h\Leftarrow\frac{h}{2}$; $k=k+1$;

$T_2=\frac{T_1}{2}+hf(a+h)$; $S_1=\frac{4}{3}T_2-\frac{1}{3}T_1$;

(3) $h\Leftarrow\frac{h}{2}$; $k=k+1$; $T_4=\frac{T_2}{2}+h[f(a+h)+f(a+3h)]$;

$S_2=\frac{4}{3}T_4-\frac{1}{3}T_2$; $C_1=\frac{16}{15}S_2-\frac{1}{15}S_1$;

(4) $h\Leftarrow\frac{h}{2}$; $k=k+1$; $T_8=\frac{1}{2}T_4+h\sum_{i=1}^{4}f(a+(2i-1)h)$;

$S_4=\frac{4}{3}T_8-\frac{1}{3}T_4$; $C_2=\frac{16}{15}S_4-\frac{1}{15}S_1$; $R_1=\frac{64}{63}C_2-\frac{1}{63}C_1$;

(5) $h\Leftarrow\frac{h}{2}$; $k=k+1$; $T_{2^k}=\frac{1}{2}T_{2^{k-1}}+h\sum_{i=1}^{2^{k-1}}f[(a+(2i-1)h)]$;

$S_{2^{k-1}}=\frac{4}{3}T_{2^k}-\frac{1}{3}T_{2^{k-1}}$; $C_{2^{k-2}}=\frac{16}{15}S_{2^{k-1}}-\frac{1}{15}S_{2^{k-2}}$; $R_{2^{k-3}}=\frac{64}{63}C_{2^{k-2}}-\frac{1}{63}C_{2^{k-3}}$;

(6) if $|R_{2^{k-3}}-R_{2^{k-4}}|<\varepsilon,\Rightarrow I\approx R_{2^{k-3}}$, stop

else goto <5>

Chapter 3
Numerical Integration
and Differentiation

例 **用变步长计算** $I=\int_0^1\frac{4}{1+x^2}dx=\pi$,**并要求误差** $\varepsilon<\frac{1}{2}\times10^{-6}$.

解:

(1) $T_1=\frac{h}{2}[f(\frac{4}{1+0^2})+f(\frac{4}{1+1^2})]=3$

(2) $T_2=\frac{3}{2}+\frac{1}{2}\times\frac{4}{1+0.5^2}=3.1,\quad S_1=\frac{4}{3}\times3.1-\frac{1}{3}\times3=3.1333333$

(3) $T_4=\frac{3.1}{2}+\frac{1}{4}[\frac{4}{1+0.25^2}+\frac{4}{1+0.75^2}]=3.1311765$

$S_2=\frac{4}{3}T_4-\frac{1}{3}T_2=3.1415686,\quad C_1=\frac{16}{15}S_2-\frac{1}{15}S_1=3.1421177$

(4) $T_8=\frac{T_4}{2}+\frac{1}{8}[\frac{4}{1+(\frac{1}{8})^2}+\frac{4}{1+(\frac{3}{8})^2}+\frac{4}{1+(\frac{5}{8})^2}+\frac{4}{1+(\frac{7}{8})^2}]$

$=3.1389885$

$S_4=\frac{4}{3}T_8-\frac{1}{3}T_4=3.1415925,\ C_2=\frac{16}{15}S_4-\frac{1}{15}S_1=3.1415941,$

$R_1=\frac{64}{63}C_2-\frac{1}{63}C_1=3.1415858$

(5) $T_{16}=\frac{T_8}{2}+\frac{1}{16}\sum_{i=1}^{8}[\frac{4}{1+(\frac{2i-1}{16})^2}]=3.1409416$

$S_8=3.1415927, C_4=3.1415927,$

$R_2=3.141592639$

(6) $\because |R_2-R_1|=0.6839\times10^{-5}>\frac{1}{2}\times10^{-6}$ $\therefore$ go on (5)

$T_{32}=3.1414299, S_{16}=3.1415926, C_8=3.1415926,$

$R_4=3.141592644$

(7) $\because |R_4-R_2|=0.05\times10^{-7}<\frac{1}{2}\times10^{-6}$

$\therefore I\approx R_4=3.141592644$

HW:用至少三种方法求

$\int_0^{0.8}\sqrt{x^3}dx \;, e=0.5\times10^{-2}$

高斯型积分

牛顿—柯特斯型求积公式

(1)封闭型（区间$[a,b]$的两端点a,b均是求积节点）；

(2)求积节点等距。

受此限制，牛顿—柯特斯型求积公式的代数精度只能是n（n为奇数）或$n+1$（n为偶数）。

如果对求积节点也适当的选取,即在求积公式中不仅A_k而且x_k也加以选取, 这就可以增加自由度，从而可提高求积公式的代数精度。

高斯型积分

构造具有2n+1次代数精度的求积公式

$$\int_a^b f(x)dx \approx \sum_{k=0}^{n} A_k f(x_k)$$

将节点 $x_0 \ldots x_n$ 以及系数 $A_0 \ldots A_n$ 都作为待定系数。

令 $f(x) = 1, x, x^2, \ldots, x^{2n+1}$ 代入可求解，得到的公式具有 $2n+1$ 次代数精度。

这样的节点称为Gauss 点，公式称为Gauss 型求积公式。

Chapter 3
Numerical Integration
and Differentiation

例: $\int_{-1}^{1} f(x)dx \approx A_0 f(x_0) + A_1 f(x_1)$

其中，x_0, x_1 固定在-1，1，A_0, A_1可以适当选取, 只有两个自由度, 得到的是梯形公式, 其代数精确度只有1。

如对求积节点也适当选取, 则有四个自由度, 可得到如下公式:

$$\int_{-1}^{1} f(x)dx = f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)$$

这个积分公式的代数精确度为3, 是高斯型求积公式,

上面的求积节点 $\pm\frac{1}{\sqrt{3}}$ 称为高斯点。

定义 如果$n+1$个求积节点的求积公式的代数精度为$2n+1$, 则这$n+1$个求积节点称为高斯点。

HUST

数值微分

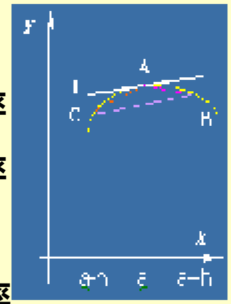
由微积分的知识 $f'(a)=\lim\limits_{h\to 0}\dfrac{f(a+h)-f(a)}{h}$ (*)

而实际中,通常f(x)会出现:

（1）f(x)由函数表给出；（2）f(x)非常复杂,不便求导

以上的f(x)难于用(*) 式求导, 通常用近似的方法. ---数值微分

一.差商法



向前差商 $f'(a)\approx\dfrac{f(a+h)-f(a)}{h}=f[a,a+h]$ AB的斜率

向后差商 $f'(a)\approx\dfrac{f(a)-f(a-h)}{h}=f[a-h,a]$ AC的斜率

将两式平均得:

中点法 $f'(a)\approx\dfrac{f(a+h)-f(a-h)}{2h}=G(h)$ BC的斜率

由泰勒公式,中点公式的截断误差为: $f'(a)-G(h)\approx O(h^2)$

Chapter 3
Numerical Integration
and Differentiation

中点公式分析

$f'(a)\approx G(h)=\dfrac{f(a+h)-f(a-h)}{2h}$ $f'(a)-G(h)\approx O(h^2)$

注：（1）由截断误差,步长h越小,精度越高.

（2）但步长h越小，f(a+h)与f(a-h)越接近.

（3）由舍入误差分析,应避免相近的数相减，h不宜太小.

用二分步长及误差的事后估计法自动选择步长——变步长算法

HUST

中点公式误差事后估计及外推

$$f'(a) \approx G(h) = \frac{f(a+h)-f(a-h)}{2h} \qquad f'(a) - G(h) \approx O(h^2)$$

记 $D_1 = G(h)$ ，$D_2 = G(\frac{h}{2})$

$$f'(a) - D_1 \approx O(h^2),\ f'(a) - D_2 \approx O(\tfrac{h}{2})^2 \Rightarrow \frac{f'(a)-D_2}{f'(a)-D_1} \approx \frac{1}{4}$$

$\Rightarrow f'(a) - D_2 \approx \frac{1}{3}(D_2 - D_1)$ ——事后误差估计法（$|D_2 - D_1| \le \varepsilon$）

$$\therefore f'(a) \approx \tfrac{4}{3}D_2 - \tfrac{1}{3}D_1 = G_1$$

$G_1(h) = \frac{4}{3}G(\frac{h}{2}) - \frac{1}{3}G(h)$ 且 $f'(a) - G_1(h) \approx O(h^4)$

根据Richardson外推法还可进一步外推

$G_2(h) = \frac{16}{15}G_1(\frac{h}{2}) - \frac{1}{15}G_1(h) \qquad G_3(h) = \frac{64}{63}G_1(\frac{h}{2}) - \frac{1}{63}G_1(h)$

例1.用变步长中点法求e^x在x=1处的导数值。初始h=0.8,精度要求 ε =0.5× 10^{-4}.

分析: $f(x) = e^x$ ，则$f'(x)= e^x$ ， $f'(1)= e$

解：由中点公式

$$f'(a) \approx G(h) = \frac{f(a+h)-f(a-h)}{2h} \qquad \therefore f'(1) = e \approx \frac{1}{2h}(e^{1+h} - e^{1-h})$$

h	G(h)	|G(0.5h)-G(h)|
0.8	3.01765	0.2263
0.4	2.79135	0.05491
0.2	2.73644	0.01363
0.1	2.72281	
⋮	⋮	⋮

HUST

二. 插值求导

已知,f(x)函数表:

x	x_0	x_1	x_2	…	x_n
f(x)	$f(x_0)$	$f(x_1)$	$f(x_2)$	…	$f(x_n)$

构造f(x)的 Lagrange插值公式$P_n(x)$.

$$f(x)\approx P_n(x) \quad 且 \quad f(x)-P_n(x)=\frac{f^{(n+1)}(x)}{(n+1)!}v(x)$$

于是,可构造如下插值型求导公式

$f^{(k)}(a) \approx P_n^{(k)}(a)$； 当 k=1, $f'(a) \approx P_n'(a)$,

注：即使f(x)与 $P_n(x)$相差不大,但可能它们的导数相差很大！

Chapter 3
Numerical Integration
and Differentiation

二. 插值求导余项

$$f(x)-P_n(x)=\frac{f^{(n+1)}(x)}{(n+1)!}v(x)$$

$$f'(a)-P_n'(a)=[f(x)-P_n(x)]'_{x=a}=[\frac{f^{(n+1)}(\xi)}{(n+1)!}w(x)]'_{x=a}$$

$$f'(a)-P_n'(a)=\{[\frac{f^{(n+1)}(\xi)}{(n+1)!}]'w(x)+\frac{f^{(n+1)}(\xi)}{(n+1)!}w'(x)\}_{x=a}$$

由于ξ是x的未知函数,上式无法估计.

若a为插值节点时,w(a)=0. $f'(a)-P_n'(a)=\frac{f^{(n+1)}(\xi)}{(n+1)!}w'(a)$

$f'(a) \approx P_n'(a)$，使:让a为插值节点；且用等距节点插值公式.

HUST

例. 三点公式 $n=2$,在$x_0,x_1=x_0+h,\ x_2=x_0+2h$**进行二次插值,**

$$P_2(x)=\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}f(x_0)+\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}f(x_1)+\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}f(x_2)$$

令$x=x_0+th$ **则**

$$P_2(x)=P_2(x_0+th)=\tfrac{1}{2}(t-1)(t-2)f(x_0)-t(t-2)f(x_1)+\tfrac{1}{2}(t-1)(t-2)f(x_2)$$

对t求导 $P_2'(x_0+th)\cdot h=\tfrac{1}{2}(2t-3)f(x_0)-(2t-2)f(x_1)+\tfrac{1}{2}(2t-1)f(x_2)$

$$P_2'(x_0+th)=\tfrac{1}{2h}[(2t-3)f(x_0)-4(t-1)f(x_1)+(2t-1)f(x_2)]$$

Chapter 3
Numerical Integration
and Differentiation

$$P_2'(x_0+th)=\tfrac{1}{2h}[(2t-3)f(x_0)-4(t-1)f(x_1)+(2t-1)f(x_2)]$$

将$t=0,1,2$**代入,得**

$$f'(x_0)\approx P_2'(x_0)=\frac{1}{2h}[-3f(x_0)+4f(x_1)-f(x_2)]$$

$$f'(x_1)\approx P_2'(x_1)=\frac{1}{2h}[-f(x_0)+f(x_2)]$$

$$f'(x_2)\approx P_2'(x_2)=\frac{1}{2h}[f(x_0)-4f(x_1)+3f(x_2)]$$

截断误差分别为:

$$f'(x_0)-P_2'(x_0)=\frac{h^2}{3}f'''(\xi)$$

$$f'(x_1)-P_2'(x_1)=-\frac{h^2}{6}f'''(\xi)$$

$$f'(x_2)-P_2'(x_2)=\frac{h^2}{3}f'''(\xi)$$

其中$\xi\in(x_0,x_2)$

HUST

例.已知$y=e^x$的函数表.

x	2.5	2.6	2.7	2.8	2.9
y	12.1825	13.4637	14.8797	16.446	18.1741

试用三点数值微分公式计算2.7处的导数值.

分析:用中点公式 $f'(x_1) \approx P_2'(x_1) = \frac{1}{2h}[-f(x_0)+f(x_2)],\ x_1 = 2.7$

解: $h=0.2$**时** $f'(2.7) \approx \frac{1}{2\times 0.2}(18.1741-12.1825) = 14.979$

$h=0.1$**时** $f'(2.7) \approx \frac{1}{2\times 0.1}(16.4446-13.4637) = 14.9045$

$\because f'(x_1)-P_2'(x_1) = -\frac{h^2}{6}f'''(\xi)$ 其中$\xi \in (x_0, x_2)$

$\max|f'''(x)| = e^{2.9} < 27$ 注: $f'(2.7) = 14.87973\ldots$

$\Rightarrow |f'(2.7)-p_2'(2.7)| < 0.045 \quad (h=0.1)$

小结

1.机械求积公式 $\int_a^b f(x)dx \approx A_0 f(x_0) + A_1 f(x_1) + \ldots + A_n f(x_n)$

2.求积公式的代数精度

3.插值型求积公式： $A_k = \int_a^b l_k(x)dx$ **iff 至少具有n次代数精度**

4.Newton-Cotes**求积公式：等距节点的插值型求积公式**

$$I_n = (b-a)\sum_{k=0}^{n} C_k f(x_k)$$

$T = I_1 = \frac{1}{2}(b-a)[f(a)+f(b)]$ $R_T = -\frac{f^{(2)}(\xi)}{12}(b-a)^3 = O(b-a)^3 \quad \xi \in (a,b)$

$S = I_2 = \frac{1}{6}(b-a)[f(a)+4f(\frac{a+b}{2})+f(b)]$ $R_S = O(b-a)^5$

5.复化求积公式 $T_n = \frac{h}{2}[f(a)+2\sum_{k=1}^{n-1} f(x_k)+f(b)]$

$I\text{-}T_n \approx -\frac{h^2}{12}[f'(b)-f'(a)] = O(h^2)$ $I\text{-}T_n = -\frac{b-a}{12}h^2 f''(x),\ x \in (a,b)$

6.变步长的梯形公式与 Romberg**算法**

7.插值型求导公式：中点公式