《数值分析》之

数值微分和数值积分

徐岩

中国科学技术大学数学系

yxu@ustc.edu.cn

http://staff.ustc.edu.cn/~yxu/

- 用$T_n(f)$表示在长度为$h=(b-a)/n$的n个子区间上积分$I=\int_a^b f(x)dx$的复化梯形法则. 对于区间$[a,b]=[0,1]$,

$$
\begin{aligned}
T_1(f) =& \frac{1}{2}f(0)+\frac{1}{2}f(1) \\
T_2(f) =& \frac{1}{4}f(0)+\frac{1}{2}f\left(\frac{1}{2}\right)+\frac{1}{4}f(1) \\
T_4(f) =& \frac{1}{8}f(0)+\frac{1}{4}\left[f\left(\frac{1}{4}\right)+f\left(\frac{1}{2}\right)+f\left(\frac{3}{4}\right)\right]+\frac{1}{8}f(1) \\
T_8(f) =& \frac{1}{16}f(0)+\frac{1}{8}\left[f\left(\frac{1}{8}\right)+f\left(\frac{1}{4}\right)+f\left(\frac{3}{8}\right)+f\left(\frac{1}{2}\right)+f\left(\frac{5}{8}\right)\right. \\
& \left.+f\left(\frac{3}{4}\right)+f\left(\frac{7}{8}\right)\right]+\frac{1}{16}f(1)
\end{aligned}
$$

- 为了计算$T(2n)$，可以利用$T(n)$计算中已有的结果，从而只需要计算那些出现在$T(2n)$，没有出现在$T(n)$中的项。

$$
\begin{aligned}
T_2(f) &= \frac{1}{2}T_1(f) + \frac{1}{2}f\left(\frac{1}{2}\right) \\
T_4(f) &= \frac{1}{2}T_2(f) + \frac{1}{4}\left[f\left(\frac{1}{4}\right) + f\left(\frac{3}{4}\right)\right] \\
T_8(f) &= \frac{1}{2}T_4(f) + \frac{1}{8}\left[f\left(\frac{1}{8}\right) + f\left(\frac{3}{8}\right) + f\left(\frac{5}{8}\right) + f\left(\frac{7}{8}\right)\right]
\end{aligned}
$$

- 令$h = \frac{b-a}{2n}$，则在一般区间$[a,b]$上的一般公式为

$$
\begin{aligned}
T_{2n}(f) &= \frac{1}{2}T_n(f) + h[f(a+h) + f(a+3h) + \cdots + f(a+(2n-1)h)] \\
&= \frac{1}{2}T_n(f) + h\sum_{i=1}^{n} f(a+(2i-1)h)
\end{aligned}
$$

中国科学技术大学

积分的自适应计算

- 函数变化有急有缓，为了照顾变化剧烈部分的误差，我们需要加密格点。
- 对于变化缓慢的部分，加密格点会造成计算的浪费。
- 以此我们介绍一种算法，可以自动在变化剧烈的地方加密格点计算，而变化缓慢的地方，则取稀疏的格点。

事后误差估计

- 复化梯形公式

$$I(f)-T_n(f)=-\frac{b-a}{12}h^2f''(\xi),\quad \text{n等分区间}$$

$$I(f)-T_{2n}(f)=-\frac{b-a}{12}(\frac{h}{2})^2f''(\eta),\quad \text{2n等分区间}$$

- 近似有：$f''(\eta)\approx f''(\xi)$
- 从而得到

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

- 误差可以用2组复化梯形公式的差来估计

中国科学技术大学

外推

由前面的事后误差估计式

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

则

$$I(f) \approx T_{2n}(f) + \frac{1}{3}(T_{2n}(f) - T_n(f)) = S_{2n}(f),$$

可以用低阶的公式组合后成为一个高阶的公式，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

记$I(h)$为以步长为h的数值积分公式，有

$$
\begin{aligned}
I(f) - I(h) &= ch^m + O(h^{m+1}), \\
I(f) - I(\frac{h}{2}) &= c(\frac{h}{2})^m + O((\frac{h}{2})^{m+1}), \\
I(f) - I(\frac{h}{2}) &\approx \frac{I(\frac{h}{2}) - I(h)}{2^m - 1}, \\
I(f) &\approx I(\frac{h}{2}) + \frac{I(\frac{h}{2}) - I(h)}{2^m - 1}.
\end{aligned}
$$

基于Euler-Maclaurin公式，可以建立起新的积分外推公式。

- Euler-Maclaurin公式：对于$f \in C^{2m}[0,1]$，

$$\int_0^1 f(t)dt = \frac{1}{2}[f(0)+f(1)] + \sum_{k=1}^{m-1} A_{2k}[f^{(2k-1)}(0) - f^{(2k-1)}(1)] - A_{2m}f^{(2m)}(\xi_0)$$

其中$\xi_0 \in (0,1)$，$k!A_k$称为Bernoulli常数，由下式定义：

$$\frac{x}{e^x - 1} = \sum_{k=0}^{\infty} A_k x^k$$

- 进行变量代换，Euler-Maclaurin公式变为

$$\int_{x_i}^{x_{i+1}} f(x)dx = \frac{h}{2}[f(x_i) + f(x_{i+1})]$$
$$+ \sum_{k=1}^{m-1} A_{2k} h^{2k}[f^{(2k-1)}(x_i) - f^{(2k-1)}(x_{i+1})]$$
$$- A_{2m} h^{2m+1} f^{(2m)}(\xi_i)$$

- 令$x_i = a + ih$, $i = 0, 1, \ldots, 2^n$, $h = (b-a)/2^n$, 进行求和：

$$\int_a^b f(x)dx = \frac{h}{2} \sum_{i=0}^{2^n-1} [f(x_i) + f(x_{i+1})]$$
$$+ \sum_{k=1}^{m-1} A_{2k} h^{2k}[f^{(2k-1)}(a) - f^{(2k-1)}(b)]$$
$$- A_{2m}(b-a) h^{2m} f^{(2m)}(\xi)$$

- 从而有

$$I = T(2^n) + c_2 h^2 + c_4 h^4 + \cdots + c_{2m-2} h^{2m-2} + c_{2m} h^{2m} f^{(2m)}(\xi)$$

这样我们可以应用Richardson外推技术，得到公式：

Euler-Maclaurin定理

若$I(f) = I^{(m)}(h) + O(h^{2m})$为$2m$阶公式，则

$$I^{(m+1)}(\frac{h}{2}) = I^{(m)}(\frac{h}{2}) + \frac{I^{(m)}(\frac{h}{2}) - I^{(m)}(h)}{2^{2m} - 1} + O(h^{2m+2})$$

Romberg 积分就是不断地用如上定理组合低阶公式为高阶公式，进而计算积分

$$R(n,0) = T_{2^n}(f)$$

$$R(n,m) = R(n,m-1) + \frac{1}{4^m - 1}[R(n,m-1) - R(n-1,m-1)]$$

中国科学技术大学

- 对一个适当的M, 按如下公式计算$R(i,j)$, $i=0,1,\ldots,M$, $j=0,1,\ldots,M$:

$$R(0,0)=\frac{1}{2}(b-a)[f(a)+f(b)]$$

$$R(n,0)=\frac{1}{2}R(n-1,0)+h_n\sum_{i=1}^{2^{n-1}}f(a+(2i-1)h_n)$$

$$R(n,m)=R(n,m-1)+\frac{1}{4^m-1}[R(n,m-1)-R(n-1,m-1)]$$

其中$h_0=b-a$, $h_n=h_{n-1}/2$

- 在精致的算法中，应当选取适度的M, 并且加上一个自动终止程序，当达到指定的误差标准时停止计算

$$
\begin{array}{ccccccc}
R(0,0) & & & & & & \\
R(1,0) & R(1,1) & & & & & \\
R(2,0) & R(2,1) & R(2,2) & & & & \\
R(3,0) & R(3,1) & R(3,2) & R(3,3) & & & \\
R(4,0) & R(4,1) & R(4,2) & R(4,3) & R(4,4) & & \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \\
R(M,0) & R(M,1) & R(M,2) & R(M,3) & R(M,4) & \cdots & R(M,M)
\end{array}
$$

收敛性定理

- 在Romberg算法中，为了应用Euler-Maclaurin公式，我们需要$f \in C^{2m}[a,b]$, 从而$R(n,m)$收敛于f的积分，并且误差为$\mathcal{O}(h^{2m})$
- 如果f只是连续，我们有下面的定理

Theorem

若$f \in C[a,b]$, 则$Romberg$阵列中每一列都收敛于f的积分，即对每个m,

$$\lim_{n\to\infty} R(n,m) = \int_a^b f(x)dx := I$$

证明

采用归纳法。对于第一列，它是积分I的梯形估计。而具有k个子区间的梯形法则可以写成

$$\frac{1}{2}h\sum_{i=0}^{k-1} f(a+ih) + \frac{1}{2}h\sum_{i=1}^{k} f(a+ih)$$

这是两个Riemann和的平均。由于$h=(b-a)/k$, 所以当$k\to\infty$时子区间的长度趋向于零。从而根据Riemann积分理论，两个Riemann和都趋向于I, 从而它们的平均值也趋向于I. 这就证明了

$$\lim_{n\to\infty} R(n,0) = I$$

假设对于$m-1$结论成立，则由于

$$R(n,m) = R(n,m-1) + \frac{1}{4^m-1}[R(n,m-1) - R(n-1,m-1)]$$

所以

$$\lim_{n\to\infty} R(n,m) = \frac{4^m}{4^m-1}I - \frac{1}{4^m-1}I = I$$

中国科学技术大学