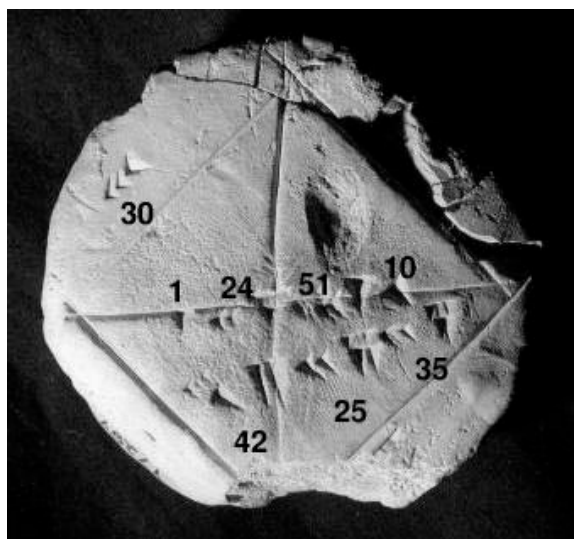


2017-2018年度第一学期 00151102 00151103

计算方法 (B)



童伟华 管理科研楼1205室

E-mail: tongwh@ustc.edu.cn

中国科学技术大学 数学科学学院

<http://math.ustc.edu.cn/>





第三章 数值微分和数值积分

- 函数 $f(x)$ 未知或非常复杂的情形下，如何求导数？
- 导数的逼近：差商

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

- 截断误差
- 步长的选取

■ 向前差商

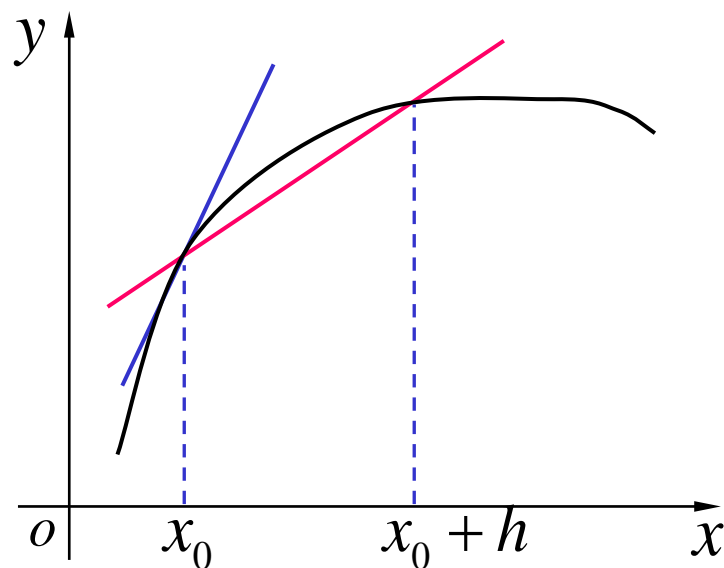
$$f'(x_0) \approx \frac{f(x_0 + h) - f(x_0)}{h}$$

■ 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

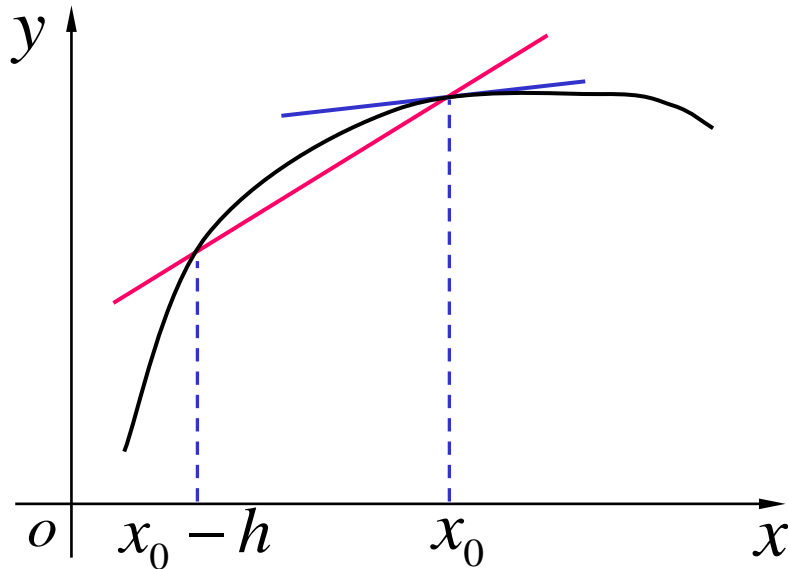
$$f'(x_0) \approx \frac{f(x_0) - f(x_0 - h)}{h}$$

■ 截断误差

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

$$= \frac{h}{2!} f''(\xi)$$

$$= O(h)$$



■ 中心差商

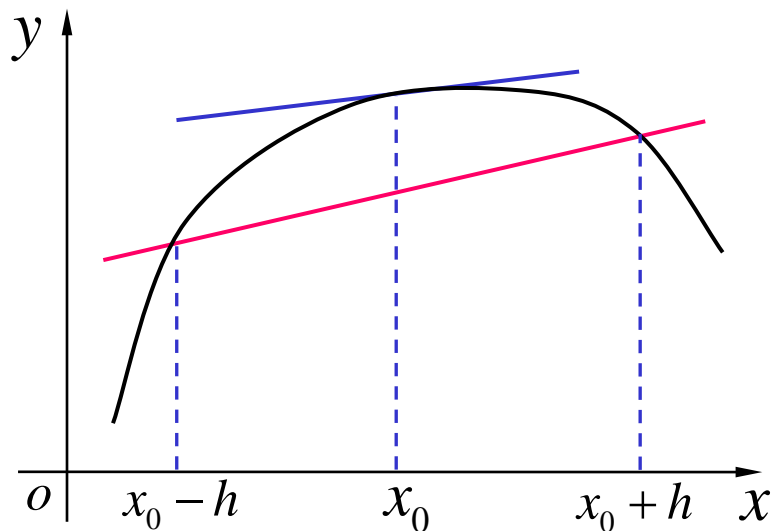
$$f'(x_0) \approx \frac{f(x_0 + h) - f(x_0 - h)}{2h}$$

■ 截断误差

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

$$= -\frac{h^2}{3!} f^{(3)}(\xi)$$

$$= O(h^2)$$



■ 如何设定步长?

■ 理论估计

- 截断误差: $h^2 \max |f^{(3)}(x)| / 6 = h^2 M_3 / 6$
 - 舍入误差: e / h
- $\left. \vphantom{\begin{matrix} \bullet \text{ 截断误差: } h^2 \max |f^{(3)}(x)| / 6 = h^2 M_3 / 6 \\ \bullet \text{ 舍入误差: } e / h \end{matrix}} \right\} \Rightarrow h = \sqrt[3]{\frac{3e}{M_3}}$

■ 事后估计

- 设 $D(h, x), D(h/2, x)$ 分别是取步长 $h, h/2$ 的差商计算公式 $f'(x)$, 对于给定的误差界 ε , 则当

$$|D(h, x) - D(h/2, x)| < \varepsilon$$

时, 步长 h 就是合适的步长 h

■ 插值型数值微分

- 对于给定 $f(x)$ 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

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

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

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

- 误差估计公式

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

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

■ 数值微分的三点公式

- 给定 $\{(x_i, f(x_i))\}_{i=0}^2$, 并有 $x_2 - x_1 = x_1 - x_0 = h$, 则

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

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

\Downarrow

$$f'(x_0) \approx L'_2(x_0) = \frac{1}{2h} (-3f(x_0) + 4f(x_1) - f(x_2)) + \frac{h^2}{3} f'''(\xi),$$

$$f'(x_1) \approx L'_2(x_1) = \frac{1}{2h} (-f(x_0) + f(x_2)) - \frac{h^2}{6} f'''(\xi),$$

$$f'(x_2) \approx L'_2(x_2) = \frac{1}{2h} (f(x_0) - 4f(x_1) + 3f(x_2)) + \frac{h^2}{3} f'''(\xi)$$

■ 类似可得数值微分的五点公式

■ 用其它插值函数方法类似, 如三次样条函数插值

数值积分



- Newton-Leibniz公式, Green公式, Gauss公式, Stokes公式
- 很多积分无解析解, 必须使用数值方式求解
- 从积分的定义出发

$$I(f) = \int_a^b f(x) dx \approx \lim_{\Delta x_i \rightarrow 0} \left(\sum_{i=0}^n f(x_i) \Delta x_i \right)$$

$$\approx \sum_{i=0}^n \alpha_i f(x_i) = I_n(f)$$

- 求积节点 $\{x_i\}$
- 求积系数 $\{\alpha_i\}$

■ 代数精度：衡量数值积分公式优劣的重要指标之一

- 设 $[a, b]$ 上以 $x_i, i = 0, 1, \dots, n$ 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

$$I_n(x^i) = I(x^i), i = 0, \dots, k;$$

$$I_n(x^{k+1}) \neq I(x^{k+1}),$$

则称 $I_n(f)$ 具有 k 阶代数精度

- ## ■ 性质：当 $I_n(f)$ 具有 k 阶代数精度时，对任意不高于 k 次的多项式 $p(x)$ ，有 $I_n(p) = I(p)$

■ 插值型数值积分

- 思想：用插值函数，譬如Lagrange插值函数，代替被积函数，进行积分

$$\begin{aligned} I(f) \approx I_n(f) &= \int_a^b L_n(x) dx = \int_a^b \sum_{i=0}^n l_i(x) f(x_i) dx = \sum_{i=0}^n \left(\int_a^b l_i(x) dx \right) f(x_i) \\ &= \sum_{i=0}^n \alpha_i f(x_i) \end{aligned}$$

$$\text{其中 } \alpha_i = \int_a^b l_i(x) dx$$

■ 误差估计公式

$$E_n(f) = I(f) - I_n(f) = \int_a^b R_n(x) dx = \int_a^b \frac{f^{(n+1)}(\xi_x)}{(n+1)!} \omega_n(x) dx$$

■ 代数精度

- n 次插值多项式形式的数值积分公式至少有 n 阶代数精度

数值积分



■ Newton-Cotes积分：取等距节点

■ 设节点步长 $h = \frac{b-a}{n}$, $x_i = a + ih, i = 0, \dots, n$, 有

$$\begin{aligned} a_i &= \int_a^b l_i(x) dx \stackrel{x=a+th}{=} \int_0^n \frac{t(t-1)\cdots(t-i+1)(t-i-1)\cdots(t-n)}{i!(n-i)!(-1)^{n-i}} h dt \\ &= nh \cdot \frac{(-1)^{n-i}}{i!(n-i)!n} \int_0^n t(t-1)\cdots(t-i+1)(t-i-1)\cdots(t-n) dt \\ &= (b-a) \cdot C_i^{(n)} \end{aligned}$$

其中 $C_i^{(n)}$ 为仅依赖于 n 的常量，可事先计算

■ 性质：

$$\sum_{i=0}^n C_i^{(n)} = 1$$

数值积分



■ 梯形积分：取 $n = 1$

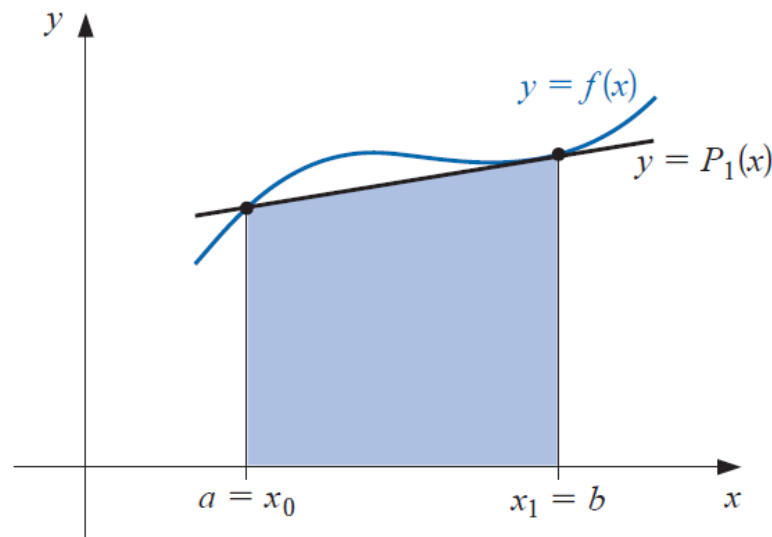
$$C_0^{(1)} = -\int_0^1 (t-1)dt = \frac{1}{2}, \quad C_1^{(1)} = \int_0^1 tdt = \frac{1}{2}$$

$$I_1(f) = (b-a) \left[\frac{1}{2} f(a) + \frac{1}{2} f(b) \right]$$

■ 性质：具有一阶代数精度

■ 误差估计公式

$$\begin{aligned} E_1(f) &= \int_a^b \frac{f''(\xi_x)}{2!} (x-a)(x-b)dx = \frac{f''(\eta)}{2!} \int_a^b (x-a)(x-b)dx \\ &= \frac{-(b-a)^3}{12} f''(\xi), \quad \xi \in [a, b] \end{aligned}$$



■ Simpson积分: 取 $n = 2$

$$C_0^{(2)} = \frac{1}{4} \int_0^2 (t-1)(t-2) dt = \frac{1}{6}, \quad C_1^{(2)} = -\frac{1}{2} \int_0^2 t(t-2) dt = \frac{4}{6}, \quad C_2^{(2)} = \frac{1}{4} \int_0^2 t(t-1) dt = \frac{1}{6}$$

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

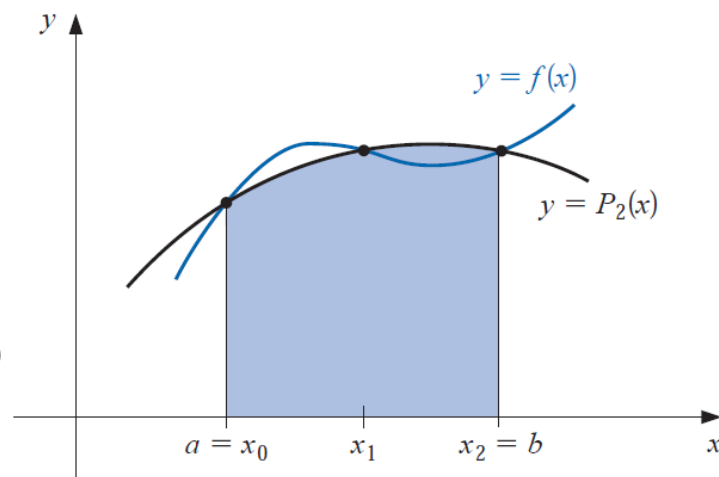
■ 性质: 具有三阶代数精度

■ 误差估计公式

$$E_2(f) = I(f) - I_2(f) = I(f) - I(P_3) + I_2(P_3) - I_2(f)$$

$$= \int_a^b \frac{f^{(4)}(\xi_x)}{4!} (x-a) \left(x - \frac{a+b}{2} \right)^2 (x-b) dx$$

$$= \frac{f^{(4)}(\xi)}{4!} \int_a^b (x-a) \left(x - \frac{a+b}{2} \right)^2 (x-b) dx = -\frac{(b-a)^5}{2880} f^{(4)}(\xi)$$



■ Newton-Cotes积分的误差估计公式

- 当 n 为奇数时, $f \in C^{n+1}[a, b]$

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

- 当 n 为偶数时, $f \in C^{n+2}[a, b]$

$$E_n(f) = I(f) - I_n(f) = \frac{f^{(n+2)}(\xi)}{(n+2)!} \int_a^b x(x - x_0)(x - x_1) \cdots (x - x_n) dx$$

- 对于Newton-Cotes积分, 奇数有 n 阶代数精度, 偶数有 $n+1$ 阶精度

■ Newton-Cotes积分系数表

n	$C_i^{(n)}$	ns	$E_n(f)$	Name
1	1 1	2	$h^3 \frac{1}{12} f^{(2)}(\xi)$	Trapezoidal rule
2	1 4 1	6	$h^5 \frac{1}{90} f^{(4)}(\xi)$	Simpson's rule
3	1 3 3 1	8	$h^5 \frac{3}{80} f^{(4)}(\xi)$	3/8 rule
4	7 32 12 32 7	90	$h^7 \frac{8}{945} f^{(6)}(\xi)$	Milne's rule
5	19 75 50 50 75 19	288	$h^7 \frac{275}{12096} f^{(6)}(\xi)$	
6	41 216 27 272 27 216 41	840	$h^9 \frac{9}{1400} f^{(8)}(\xi)$	Weddle's rule

Table 2: Newton-Cotes formulas.

其中 $h = \frac{b-a}{n}$

复化数值积分



- 高阶Newton-Cotes积分：数值不稳定
- 解决方法：将积分区间分割为若干小区间，在每个小区间上进行数值积分（类似于分段插值的思想）
- 复化梯形积分：取等距节点 $h = \frac{b-a}{n}, x_i = a + ih, i = 0, \dots, n$

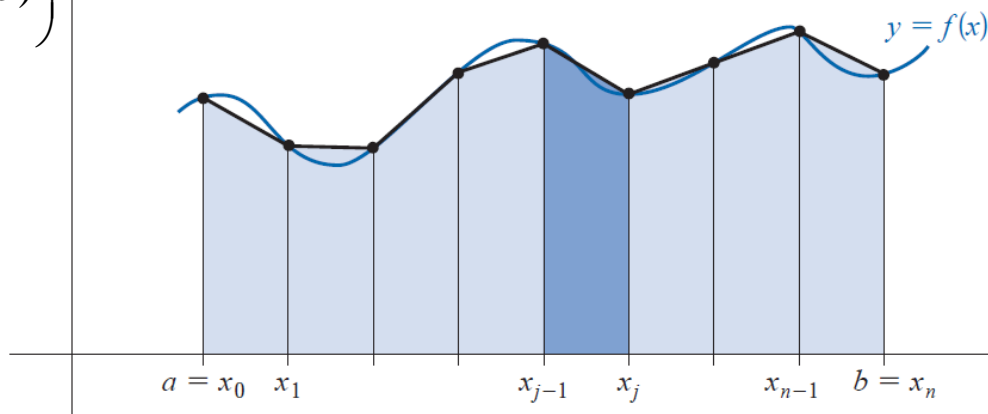
$$\int_{x_i}^{x_{i+1}} f(x) dx \approx \frac{h}{2}(f(x_i) + f(x_{i+1})) - \frac{h^3}{12} f''(\xi_i)$$

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

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

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

- 收敛性： $O(h^2)$ 或 $O(\frac{1}{n^2})$



复化数值积分



- 复化Simpson积分：把积分区间分成偶数 $n = 2m$ 等分，取等距节点 $h = \frac{b-a}{n}$, $x_i = a + ih$, $i = 0, \dots, n$

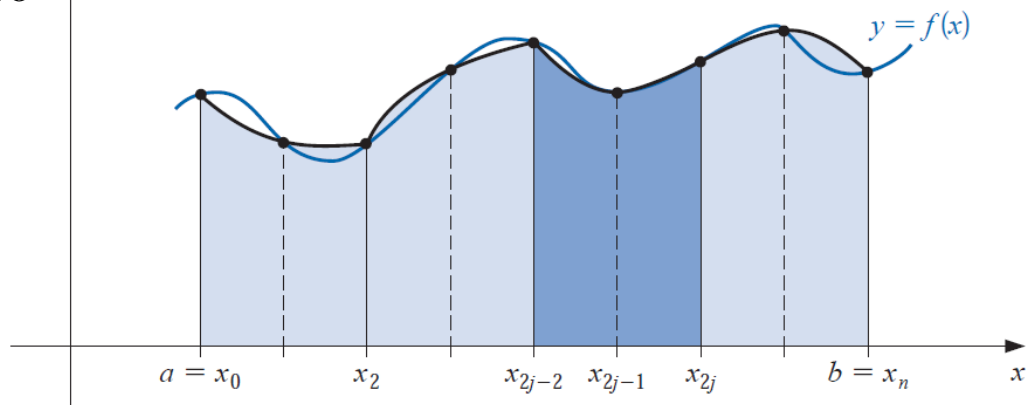
$$\int_{x_{2i}}^{x_{2i+2}} f(x) dx \approx \frac{2h}{6} (f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})) - \frac{(2h)^5}{2880} f^{(4)}(\xi_i)$$

$$I(f) = \sum_{i=0}^{n-1} \left\{ \frac{2h}{6} (f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})) - \frac{(2h)^5}{2880} f^{(4)}(\xi_i) \right\}$$
$$= \frac{h}{3} \left(f(a) + 4 \sum_{i=0}^{m-1} f(x_{2i+1}) + 2 \sum_{i=1}^{m-1} f(x_{2i}) + f(b) \right) - \sum_{i=0}^{m-1} \frac{(2h)^5}{2880} f^{(4)}(\xi_i)$$

$$S_n(f) = \frac{h}{3} \left(f(a) + 4 \sum_{i=0}^{m-1} f(x_{2i+1}) + 2 \sum_{i=1}^{m-1} f(x_{2i}) + f(b) \right)$$

$$E_n(f) = -\frac{(b-a)^5}{180n^4} f^{(4)}(\xi), \quad a \leq \xi \leq b$$

- 收敛性： $O(h^4)$ 或 $O(\frac{1}{n^4})$



复化数值积分



- 定义：若一个数值积分公式满足

$$\lim_{h \rightarrow 0} \frac{R[f]}{h^p} = C < \infty, C \neq 0,$$

则称该公式是 p 阶收敛的

- 性质：

$$T_n \sim O(h^2), S_n \sim O(h^4), C_n \sim O(h^6)$$

- 例：计算 $\pi = \int_0^1 \frac{4}{1+x^2} dx$

$$T_8 = \frac{1}{16} \left[f(0) + 2 \sum_{k=1}^7 f(x_k) + f(1) \right] = 3.138988494$$

$$S_4 = \frac{1}{24} \left[f(0) + 4 \sum_{\text{odd}} f(x_k) + 2 \sum_{\text{even}} f(x_k) + f(1) \right] = 3.141592502$$

运算量基本相同

■ 误差估计和控制

- 先验估计：截断误差公式
- 事后估计：递推关系

- ## ■ 自动误差控制：依据事后误差估计，采取自适应的方式加密积分区间，使得在变化剧烈的地方取稠密的格点，而变化缓慢的地方取稀疏的格点

复化数值积分



■ 递推关系：（复化梯形公式）

■ 对区间 $[a, b]$ 进行 n 等分, $h_n = \frac{b-a}{n}$

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

■ 对每个区间 $[x_i, x_{i+1}]$ 进行细分, 中点为 $x_{i+1/2}$, 则有

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

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

■ 递推关系：（复化Simpson公式）

$$S_{2n}(f) = \frac{1}{6} [3S_n(f) + 4H_{2n}(f) - H_n(f)]$$

复化数值积分



■ 后验误差估计：（复化梯形公式）

$$\left. \begin{aligned} I(f) - T_n(f) &= -\frac{(b-a)}{12} h^2 f''(\xi) \\ I(f) - T_{2n}(f) &= -\frac{(b-a)}{12} \left(\frac{h}{2}\right)^2 f''(\eta) \end{aligned} \right\} \xrightarrow{f''(\eta) \approx f''(\xi)} I(f) - T_{2n}(f) \approx \frac{1}{3} (T_{2n}(f) - T_n(f))$$

■ 因此任给 $\varepsilon > 0$ ，要使得 $|I(f) - T_{2n}(f)| < \varepsilon$ ，只需要

$$|T_{2n}(f) - T_n(f)| < 3\varepsilon$$

■ 后验误差估计：（复化Simpson公式）

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

复化数值积分



■ 自适应复化Simpson积分算法

Algorithm 4 Simpson's Adaptive Quadrature Algorithm

Input:

```
 $f(x), a, b, \varepsilon, N$ 
1:  $\Delta \leftarrow b - a; \Sigma \leftarrow 0; h \leftarrow \Delta/2; c \leftarrow (a + b)/2; k \leftarrow 1;$ 
2:  $\bar{a} \leftarrow f(a); \bar{b} \leftarrow f(b); \bar{c} \leftarrow f(c);$ 
3:  $S \leftarrow (\bar{a} + 4\bar{c} + \bar{b})h/3;$ 
4:  $v^{(1)} \leftarrow [a, h, \bar{a}, \bar{c}, \bar{b}, S];$ 
5: while  $1 \leq k \leq N$  do
6:    $h \leftarrow v_2^{(k)}/2;$ 
7:    $\bar{y} \leftarrow f(v_1^{(k)} + h);$ 
8:    $S^* \leftarrow (v_3^{(k)} + 4\bar{y} + v_4^{(k)})h/3;$ 
9:    $\bar{z} \leftarrow f(v_1^{(k)} + 3h);$ 
10:   $S^{**} \leftarrow (v_4^{(k)} + 4\bar{z} + v_5^{(k)})h/3;$ 
11:  if  $|S^* + S^{**} - v_6^{(k)}| < 30\varepsilon h/\Delta$  then
12:     $\Sigma \leftarrow \Sigma + S^* + S^{**} + [S^* + S^{**} - v_6^{(k)}]/15;$ 
13:     $k \leftarrow k - 1;$ 
14:    if  $k \leq 0$  then
15:      output  $\Sigma;$ 
16:      return true
17:    end if
18:  else
19:    if  $k \geq N$  then
20:      output failure;
21:      return false
22:    else
23:       $\bar{v} \leftarrow v_5^{(k)};$ 
24:       $v^k \leftarrow [v_1^{(k)}, h, v_3^{(k)}, \bar{y}, v_4^{(k)}, S^*];$ 
25:       $k \leftarrow k + 1;$ 
26:       $v^k \leftarrow [v_1^{(k-1)} + 2h, h, v_5^{(k-1)}, \bar{z}, \bar{v}, S^{**}];$ 
27:    end if
28:  end if
29: end while
```

Output:

Σ

复化数值积分



- 外推算法：在不显著增加计算量的前提下提高数值计算结果精度的技巧，是数值计算方法典型的技巧

- Romberg积分公式

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

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

- 效果：利用外推技巧，将梯形积分求积公式组合成Simpson求积公式，截断误差由 $O(h^2)$ 降低到 $O(h^4)$

- 类似地有

$$I(f) \approx S_{2n}(f) + \frac{1}{15}(S_{2n}(f) - S_n(f)) = \frac{16}{15}S_{2n}(f) - \frac{1}{15}S_n(f) = C_{2n}(f)$$

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

复化数值积分



- (Richardson Extrapolation) 若有数值逼近格式具有形式 $I(f) = I^{(m)}(h) + C_m h^{2m} + C_{m+1} h^{2m+2} + \dots = I^{(m)}(h) + O(h^{2m})$, 即为 $2m$ 阶公式, 则

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

- Romberg积分计算公式

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

- (收敛性) 若 $f(x) \in C[a, b]$, 则对任意固定的 m

$$\lim_{n \rightarrow \infty} R(n, m) = \int_a^b f(x) dx \quad \text{及} \quad \lim_{n \rightarrow \infty} R(n, n) = \int_a^b f(x) dx$$

■ Romberg积分算法

Algorithm 5 Romberg Algorithm

Input:

$f(x), a, b, M$

1: $h \leftarrow b - a;$

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

3: **for** $n = 1$ to M **do**

4: $h \leftarrow h/2;$

5: $R(n, 0) \leftarrow \frac{1}{2}R(n - 1, 0) + h \sum_{i=1}^{2^{n-1}} f(a + (2i - 1)h);$

6: **for** $m = 1$ to n **do**

7: $R(n, m) \leftarrow R(n, m - 1) + [R(n, m - 1) - R(n - 1, m - 1)]/(4^m - 1);$

8: **end for**

9: **end for**

Output:

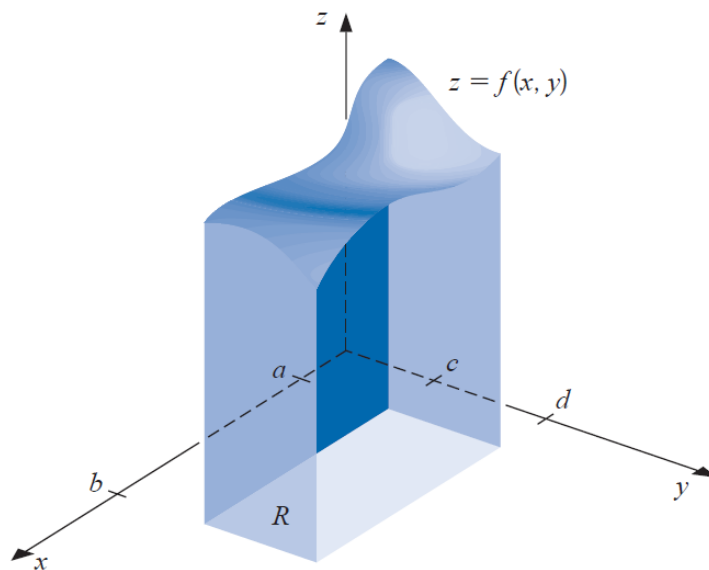
$R(n, m) \ (0 \leq n \leq M, 0 \leq m \leq n)$

重积分的计算



- 重积分：化为累次积分进行计算
- 积分区域逼近：非矩形区域可用一系列矩形区域进行逼近
- 矩形区域上的二重积分：若 $f(x) \in C([a, b] \times [c, d])$ ，则

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



重积分的计算



■ 二重积分的复化梯形公式

■ 将区间 $[a, b]$ 和 $[c, d]$ 分别进行 m 和 n 等分, 记 $h = \frac{b-a}{m}$, $k = \frac{d-c}{n}$

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

$$\begin{aligned} \Rightarrow \int_a^b \int_c^d f(x, y) dy dx &\approx h \left\{ \frac{1}{4} (f(x_0, y_0) + f(x_0, y_n) + f(x_m, y_0) + f(x_m, y_n)) \right. \\ &\quad \left. + \frac{1}{2} \left(\sum_{i=1}^{m-1} f(x_i, y_0) + \sum_{i=1}^{m-1} f(x_i, y_n) + \sum_{j=1}^{n-1} f(x_0, y_j) + \sum_{j=1}^{n-1} f(x_m, y_j) \right) + \sum_{j=1}^{n-1} \sum_{i=1}^{m-1} f(x_i, y_j) \right\} \\ &= hk \sum_{j=0}^n \sum_{i=0}^m c_{i,j} f(x_i, y_j) \quad \text{角点: } 1/4, \text{ 内部边界点 } 1/2, \text{ 内部点: } 1 \end{aligned}$$

■ 误差估计公式:

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

重积分的计算



■ 二重积分的复化Simpson公式

$$\int_a^b \int_c^d f(x, y) dy dx = hk \sum_{i=0}^m \sum_{j=0}^n \omega_{i,j} f(x_i, y_j)$$

其中 $\omega_{i,j} = u_i \cdot v_j$, $U = \{u_0, u_1, \dots, u_m\} = \left\{ \frac{1}{3}, \frac{4}{3}, \frac{2}{3}, \frac{4}{3}, \dots, \frac{2}{3}, \frac{4}{3}, \frac{1}{3} \right\}^T$,

$$V = \{v_0, v_1, \dots, v_n\} = \left\{ \frac{1}{3}, \frac{4}{3}, \frac{2}{3}, \frac{4}{3}, \dots, \frac{2}{3}, \frac{4}{3}, \frac{1}{3} \right\}^T$$

■ 误差估计公式

$$E(f) = -\frac{(d-c)(b-a)}{180} \left(h^4 \frac{\partial^4}{\partial x^4} f(\eta, \mu) + k^4 \frac{\partial^4}{\partial y^4} f(\bar{\eta}, \bar{\mu}) \right)$$

Gauss型积分



- 对于Newton-Cotes积分, 奇数有 n 阶代数精度, 偶数有 $n+1$ 阶精度
- 如果放弃等距节点的要求, 是否能建立更有高精度的公式?
- 例: 取区间 $[-1, 1]$, 考虑两点数值积分格式, 将积分点视为未知量, 则有

$$\left\{ \begin{array}{l} a_0 \cdot 1 + a_1 \cdot 1 = \int_{-1}^1 1 dx = 2 \\ a_0 \cdot x_0 + a_1 \cdot x_1 = \int_{-1}^1 x dx = 0 \\ a_0 \cdot x_0^2 + a_1 \cdot x_1^2 = \int_{-1}^1 x^2 dx = \frac{2}{3} \\ a_0 \cdot x_0^3 + a_1 \cdot x_1^3 = \int_{-1}^1 x^3 dx = 0 \end{array} \right. \quad \Rightarrow \quad \left\{ \begin{array}{l} a_0 = 1 \\ a_1 = 1 \\ x_0 = -\frac{1}{\sqrt{3}} \\ x_1 = \frac{1}{\sqrt{3}} \end{array} \right.$$

不难验证数值积分公式 $\int_{-1}^1 f(x) dx = f(-\frac{1}{\sqrt{3}}) + f(\frac{1}{\sqrt{3}})$ 具有三阶代数精度

Gauss型积分



■ **定理：** 设 $I(f) = \int_a^b f(x)dx$ 关于积分节点 x_1, x_2, \dots, x_n 的数值积分公式为 $I_n(f) = \sum_{i=1}^n a_i f(x_i)$ ，则 $I_n(f)$ 的代数精度不超过 $2n-1$ 阶

证明： 取多项式 $p(x) = (x-x_1)^2(x-x_2)^2 \cdots (x-x_n)^2 = \omega_n^2(x)$

易知 $I(p(x)) > 0$ 及 $I_n(p(x)) = 0$

故数值积分公式 $I_n(f)$ 的代数精度不可能达到 $2n$ 阶

■ **如何构造最高阶 $(2n-1)$ 精度的公式？**

■ **Gauss型积分：** 取一组特殊的积分节点（正交函数的零点）

Gauss型积分



- 定义：给定一般形式的积分和内积

$$I(f) = \int_a^b W(x) f(x) dx, \quad W(x) \geq 0,$$

$$\langle f, g \rangle = \int_a^b W(x) f(x) g(x) dx$$

其中 $W(x)$ 为权函数. 若 $\langle f, g \rangle = 0$, 则称函数 f 与 g 正交

- 如何将线性函数空间的一组基变为一组正交基?
- 利用Schmidt正交化过程:

$$\begin{cases} g_1(x) = f_1(x) \\ \vdots \\ g_n(x) = f_n(x) - \sum_{i=1}^{n-1} \frac{(f_n(x), g_i(x))}{(g_i(x), g_i(x))} g_i(x) \end{cases}$$

Gauss型积分



■ **定理：**以 n 次正交多项式的 n 个零点为积分的数值积分公式有 $2n-1$ 阶代数精度

■ **证明：**设 n 次正交多项式 $p_n(x)$ 的 n 个零点为 x_1, x_2, \dots, x_n ，记 $G_n(f) = \int_a^b L_n(x)W(x)dx = \sum_{i=1}^n \left(\int_a^b l_i(x)W(x)dx \right) f(x_i) = \sum_{i=1}^n \alpha_i f(x_i)$ ，则有

$$E(f) = I(f) - G_n(f) = \int_a^b f[x_1, x_2, \dots, x_n, x] \omega_n(x) W(x) dx$$

对于任意的 $p(x) \in P_{2n-1}(x)$ ，利用多项式带余除法得

$$p(x) = s(x)p_n(x) + r(x), \quad s(x), r(x) \in P_{n-1}(x),$$

$$\text{故 } \int_a^b p(x)W(x)dx = \int_a^b [s(x)p_n(x) + r(x)]W(x)dx = \int_a^b r(x)W(x)dx.$$

$$\text{因此 } E(p) = I(p) - G_n(p) = I(r) - G_n(r) = E(r) = 0$$

■ **收敛性：**若 $f(x) \in C[a, b]$ ，则 $\lim_{n \rightarrow \infty} G_n(f) = I(f)$

仅理论上可行，因为求 n 次正交多项式的所有零点是困难的！

■ Gauss型求积公式的构造方法

- 求出区间 $[a, b]$ 上权函数为 $W(x)$ 的正交多项式 $p_n(x)$;
- 求出 $p_n(x)$ 的 n 个零点 x_1, x_2, \dots, x_n 即为 Gauss 积分点;
- 计算积分系数 α_i ;

■ 例：求积分 $\int_{-1}^1 x^2 f(x) dx$ 的两点 Gauss 积分公式

$$p_0(x) = 1$$

$$p_1(x) = x - \frac{(x, p_0(x))}{(p_0(x), p_0(x))} p_0(x)$$

$$p_2(x) = x^2 - \frac{(x^2, p_0(x))}{(p_0(x), p_0(x))} p_0(x) - \frac{(x^2, p_1(x))}{(p_1(x), p_1(x))} p_1(x)$$

$$= x^2 - \frac{\int_{-1}^1 x^4 dx}{\int_{-1}^1 x^2 dx} - \frac{\int_{-1}^1 x^5 dx}{\int_{-1}^1 x^4 dx} x = x^2 - \frac{3}{5}$$

Gauss型积分

$p_2(x)$ 的两个零点为 $x_1 = -\sqrt{\frac{3}{5}}$, $x_2 = \sqrt{\frac{3}{5}}$,

积分系数为

$$A_1 = \int_{-1}^1 x^2 l_1(x) dx = \int_{-1}^1 x^2 \frac{x - x_2}{x_1 - x_2} dx = \frac{1}{3}$$

$$A_2 = \int_{-1}^1 x^2 l_2(x) dx = \int_{-1}^1 x^2 \frac{x - x_1}{x_2 - x_1} dx = \frac{1}{3}$$

故两点Gauss积分公式为

$$\int_{-1}^1 x^2 f(x) dx \approx \frac{1}{3} [f(-\sqrt{\frac{3}{5}}) + f(\sqrt{\frac{3}{5}})]$$

Gauss型积分



■ 几种Gauss型求积公式

- Gauss-Legendre求积公式: 区间 $[a, b] = [-1, 1]$, $W(x) = 1$
- Gauss-Laguerre求积公式: 区间 $[a, b] = [0, +\infty)$, $W(x) = e^{-x}$
- Gauss-Hermite求积公式: 区间 $[a, b] = (-\infty, +\infty)$, $W(x) = e^{-x^2}$

■ 一般区间上的积分

$$\int_a^b f(x)dx = \frac{b-a}{2} \int_{-1}^1 f\left(\frac{a+b}{2} + \frac{b-a}{2}t\right)dt \quad \left(x = \frac{(a+b) + (b-a)t}{2}\right)$$

$$\int_a^b f(x)dx \approx \frac{b-a}{2} \sum_{i=1}^n A_i f\left(\frac{a+b}{2} + \frac{b-a}{2}x_i\right)$$

■ 误差估计公式: 若 $f(x) \in C^{2n}[a, b]$, 则

$$E_n(f) = I(f) - G_n(f) = \frac{f^{(2n)}(\xi)}{(2n)!} \int_a^b \omega^2(x)W(x)dx, \quad a < \xi < b$$

Gauss型积分



■ Gauss-Legendre积分节点与系数表

n	x_k	A_k	n	x_k	A_k
1	0	2	6	± 0.9324695142	0.1713244924
2	± 0.5773502692	1		± 0.6612093865	0.3607615730
3	± 0.7745966692	0.5555555556		± 0.2386191861	0.4679139346
	0	0.8888888889	7	± 0.9491079123	0.1294849662
4	± 0.8611363116	0.3478548451		± 0.7415311856	0.2797053915
	± 0.3399810436	0.6521451549		± 0.4058451514	0.3818300505
5	± 0.9061798459	0.2369268851	8	0	0.4179591837
	± 0.5384693101	0.4786286705		± 0.9602898565	0.1012285363
	0	0.5688888889		± 0.7966664774	0.2223810345
				± 0.5255324099	0.3137066459
				± 0.1834346425	0.3626837834

Gauss型积分



■ Gauss-Laguerre积分节点与系数表

n	x_k	A_k	n	x_k	A_k
2	0.5858864376	0.8535533905	5	0.2635603197	0.5217556105
	3.4142135623	0.1464466094		1.4134030591	0.3986668110
3	0.4157745567	0.7110930099		3.5964257710	0.0759424497
	2.2942803602	0.2785177335		7.0858100058	0.0036117587
	6.02899450829	0.0103892565		12.6408008442	0.0000233700
4	0.3225476896	0.6031541043	6	0.2228466041	0.4589646793
	1.7457611011	0.3574186924		1.1889321016	0.4170008307
	4.5366202969	0.0388879085		2.9927363260	0.1133733820
	9.3950709123	0.0005392947		5.7751435691	0.0103991975
				9.8374674183	0.0002610172
				15.9828739806	0.0000008985

Gauss型积分



■ Gauss-Hermite积分节点与系数表

n	x_k	A_k	n	x_k	A_k
2	± 0.7071067811	0.8862269254	6	± 0.4360774119	0.7246295952
3	± 1.2247448713 0	0.2954089751 1.8163590006		± 1.3358490704	0.1570673203
4	± 0.5246476232 ± 1.6506801238	0.8049140900 0.0813128354		± 2.3506049736	0.0045300099
5	± 0.9585724646 ± 2.0201828704 0	0.3936193231 0.0199532421 0.9453087204	7	± 0.8162878828	0.4256072526
				± 1.6735516287	0.0545155828
				± 2.6519613563 0	0.0009717812 0.8102646175