Deriving Romberg Algorithm

composite trapezoid rule on 2^{n-1} subintervals

$$\int_a^b f(x)\,dx = R(n-1,0)+a_2h^2+a_4h^4+a_6h^6+\cdots$$

with $h = (b-a)/2^{n-1}$ and the coefficients a_i depend on f but not on h

after one refinement and replacing $n-1$ with n and h with $h/2$, we have

$$\int_a^b f(x)\,dx = R(n,0) + \frac{1}{4}a_2h^2 + \frac{1}{16}a_4h^4 + \frac{1}{64}a_6h^6 + \cdots$$

subtracting the 1st equation from 4 times the 2nd equation

$$\int_a^b f(x)\,dx = R(n,1) - \frac{1}{4}a_4h^4 - \frac{5}{15}a_6h^6 + \cdots$$

where for $n \geq 1$

$$R(n,1) = R(n,0) + \frac{1}{3}[R(n,0) - R(n-1,0)]$$

General Extrapolation

extrapolation processes can be applied in more general cases where the error term can be represented as

$$E = a\,h^{\alpha} + b\,h^{\beta} + c\,h^{\gamma} + \cdots$$

with $0 < \alpha < \beta < \gamma$, we show how the first term of the error expansion is annihilated. Let

$$L = \phi(h) + a\,h^{\alpha} + b\,h^{\beta} + c\,h^{\gamma} + \cdots \quad (1)$$

replacing h by $h/2$ yields

$$L = \phi\left(\frac{h}{2}\right) + a\left(\frac{h}{2}\right)^{\alpha} + b\left(\frac{h}{2}\right)^{\beta} + c\left(\frac{h}{2}\right)^{\gamma} + \cdots \quad (2)$$

multiplying (2) by 2^{α}

$$2^{\alpha}L = 2^{\alpha}\phi\left(\frac{h}{2}\right) + a\,h^{\alpha} + 2^{\alpha}b\left(\frac{h}{2}\right)^{\beta} + 2^{\alpha}c\left(\frac{h}{2}\right)^{\gamma} + \cdots \quad (3)$$

General Extrapolation - Cont.

subtracting the previous two equations, we can remove the h^α term

$$(2^\alpha - 1)L = 2^\alpha \phi\left(\frac{h}{2}\right) - \phi(h)$$

$$+\ (2^{\alpha-\beta} - 1)\, b\, h^\beta + (2^{\alpha-\gamma} - 1)\, c\, h^\gamma + \cdots$$

we can write the new approximation formula as

$$L = \frac{2^\alpha}{2^\alpha - 1}\,\phi\left(\frac{h}{2}\right) - \frac{1}{2^\alpha - 1}\,\phi(h) + \tilde{b}h^\beta + \tilde{c}h^\gamma + \cdots$$

this approximation formula raises the order of truncation error from $O(h^\alpha)$ to $O(h^\beta)$ with $\alpha < \beta$

please read the book on p. 217 for a concrete example to show how the approximation accuracy is improved using extrapolation

Basic Simpson's Rule

a three point numerical integration rule using the middle point of the interval is known as the Simpson's rule with different weights for each point

$$\int_a^{a+2h} f(x)\,dx \approx \frac{h}{3}[f(a)+4f(a+h)+f(a+2h)]$$

using Taylor's expansion, we can find the error term of this approximation is

$$-\frac{h^5}{90} f^{(4)}(\xi)$$

for some point ξ in $(a, a+2h)$. This should be compared to the error term of the simple trapezoid rule $O(h^3)$

it is desirable to subdivide the interval adaptively so that refinement is only placed at the area of large fluctuation of function value

Adaptive Simpson's Algorithm

given an interval $[a, b]$, we can use the basic Simpson's rule to compute an approximation to the integral as

$$I \equiv \int_a^b f(x)\,dx = S(a,b) + E(a,b)$$

where the approximation part is

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

and the error term is

$$E(a,b) = -\frac{1}{90}\left(\frac{b-a}{2}\right)^5 f^{(4)}(\xi)$$

for simplicity, we assume $f^{(4)}(x)$ remains constant on (a,b). Let $h = b - a$, we have

$$I = S^{(1)} + E^{(1)}$$

for the first step approximation with

$$S^{(1)} = S(a,b)$$

Adaptive Simpson - (I)

and

$$E^{(1)} = -\frac{1}{90}\left(\frac{h}{2}\right)^5 f^{(4)}$$

we then subdivide the interval $[a, b]$ and apply the basic Simpson's rule on the subintervals $[a, c]$ and $[c, b]$ respectively. We have a new approximation on $[a, b]$ as the sum of two separate approximations

$$I = S^{(2)} + E^{(2)},$$

where $c = (a + b)/2$ with

$$S^{(2)} = S(a, c) + S(c, b)$$

and

$$E^{(2)} = -\frac{1}{90}\left(\frac{h/2}{2}\right)^5 f^{(4)} - \frac{1}{90}\left(\frac{h/2}{2}\right)^5 f^{(4)} = \frac{1}{16}E^{(1)}$$

this is certainly a better approximation since the subintervals are smaller than the original interval

Adaptive Simpson - (III)

subtracting the two approximations yields

$$S^{(2)} - S^{(1)} = E^{(1)} - E^{(2)} = 15E^{(2)}$$

hence the numerical integration can be

$$I = S^{(2)} + E^{(2)} = S^{(2)} + \frac{1}{15}(S^{(2)} - S^{(1)})$$

the error term is then computable and can be used for building the adaptive process

$$\frac{1}{15}|S^{(2)} - S^{(1)}| < \epsilon$$

if this test shows that the error is larger than ϵ, the interval $[a, b]$ is split into two subintervals $[a, c]$ and $[c, b]$ with $c = (a + b)/2$. The previous described procedure is repeated on the two subintervals with error tolerance is replaced by $\epsilon/2$ to make sure that the error sum is smaller than ϵ

Adaptive Simpson - (IV)

numerical integration on subintervals

$$I = \int_a^b f(x)\,dx = \int_a^c f(x)\,dx + \int_c^b f(x)\,dx = I_{\rm L} + I_{\rm R}$$

let S be the sum of $S_{\rm L}^{(2)}$ on $[a, c]$ and $S_{\rm R}^{(2)}$ on $[c, b]$, we have

$$\begin{aligned} |I - S| &= |I_{\rm L} + I_{\rm R} - S_{\rm L}^{(2)} - S_{\rm R}^{(2)}| \\ &\leq |I_{\rm L} - S_{\rm L}^{(2)}| + |I_{\rm R} - S_{\rm R}^{(2)}| \\ &= \frac{1}{15}|S_{\rm L}^{(2)} - S_{\rm L}^{(1)}| + \frac{1}{15}|S_{\rm R}^{(2)} - S_{\rm R}^{(1)}| \end{aligned}$$

if we want to have

$$|I - S| \leq \epsilon$$

it is more than enough to have

$$\frac{1}{15}|S_{\rm L}^{(2)} - S_{\rm L}^{(1)}| \leq \frac{\epsilon}{2}$$

and

$$\frac{1}{15}|S_{\rm R}^{(2)} - S_{\rm R}^{(1)}| \leq \frac{\epsilon}{2}$$

Computational Procedure

the interval $[a, b]$ is divided into four subintervals of equal length. Two Simpson approximations are computed using two double-width subintervals and four single-width subintervals

$$S_1 = \frac{h}{6}\left[f(a) + 4\,f\left(\frac{a+b}{2}\right) + f(b)\right]$$

$$S_2 = \frac{h}{12}\left[f(a) + 4\,f\left(\frac{a+c}{2}\right)\right.$$
$$\left. + 2\,f(c) + 4\,f\left(\frac{c+b}{2}\right) + f(b)\right]$$

if $|S_2 - S_1| \le \epsilon$, we have done, and set

$$S = \frac{1}{15}\,[16\,S_2 - S_1]$$

otherwise the interval $[a, b]$ is divided in half and the recursive procedure is applied on the two subintervals $[a, c]$ and $[c, b]$, until either the error tolerance is satisfied or the maximum number of subdivisions is reached

Gaussian Quadrature Formulas

a general numerical integration formula is

$\int_a^b f(x)\,dx \approx A_0 f(x_0) + A_1 f(x_1) + \cdots + A_n f(x_n)$

it suffices to know the nodes $x_0, x_1, \ldots, x_n$ and the weights $A_0, A_1, \ldots, A_n$. For important special functions, they are listed in some reference books

suppose a set of nodes is given, how to find the weights. This can be done using Lagrange interpolation polynomial as

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

with

$$l_i(x) = \prod_{j=0, j\neq i}^{n} \left(\frac{x - x_j}{x_i - x_j} \right)$$

if p is a good approximate to f, we anticipate $\int_a^b p(x)dx$ is a good approximate to $\int_a^b f(x)dx$

Gaussian Quadrature

we integrate over $p(x)$ as

$$\int_a^b f(x)\,dx \approx \int_a^b p(x)\,dx$$

$$= \sum_{i=0}^{n} f(x_i) \int_a^b l_i(x)dx = \sum_{i=1}^{n} A_i f(x_i)$$

where we can compute

$$A_i = \int_a^b l_i(x)\,dx$$

note that the polynomial interpolation is exact for a polynomial of degree at most n. It follows that the integration will be exact for such polynomials

if the nodes can be chosen carefully, it is possible to increase the order of polynomial with the exact integration remarkably. This was discussed by Karl Gauss

Gaussian Quadrature Theorem

let q be a nontrivial polynomial of degree $n+1$ such taht

$$\int_a^b x^k q(x)\,dx = 0 \qquad (0 \leq k \leq n)$$

let $x_0, x_1, \ldots, x_n$ be the zeros of q. Then the formula

$$\int_a^b f(x)dx \approx \sum_{i=0}^{n} A_i f(x_i), \quad A_i = \int_a^b l_i(x)\,dx$$

with these x_i's as nodes will be exact for all polynomials of degree at most $2n+1$. All these nodes lie in the open interval (a, b)

we can first figure out the quadrature for

$$\int_{-1}^{1} f(t)\,dt \approx \sum_{i=0}^{n} A_i\, f(t_i)$$

then use transformation $t = [2x-(b-a)]/(b-a)$ for a Gaussian quadrature on the general interval $[a, b]$

Gaussian Theorem Proof

the transformed integral is

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

Proof of Gaussian Quadrature Theorem: let f be any polynomial of degree at most $(2n+1)$. Dividing f by q with a quotient p and a reminder r

$$f = p\,q + r$$

both q and r are of degree at most n

by hypothesis, we have

$$\int_a^b q(x)p(x)\,dx = 0$$

since x_i are roots of q, we have

$$f(x_i) = p(x_i)\,q(x_i) + r(x_i) = r(x_i)$$

Gaussian Theorem Proof (II)

since the degree of r is at most n, the integration $\int_a^b r(x)\,dx$ is exact

$$\int_a^b f(x)\,dx = \int_a^b p(x)q(x)\,dx + \int_a^b r(x)$$
$$= \int_a^b r(x)\,dx = \sum_{i=0}^{n} A_i r(x_i) = \sum_{i=0}^{n} A_i f(x_i)$$

Gaussian Quadrature Theorem gaurantees high accuracy numerical integration with few nodes. However, finding these nodes is not an easy task. The roots of Legendre polynomials are the nodes for Gaussian quadrature on the interval $[-1, 1]$. With $q_0(x) = 1$, $q_1(x) = x$, we have for $n \geq 2$

$$q_n(x) = \left(\frac{2n-1}{n}\right) x q_{n-1}(x) - \left(\frac{n-1}{n}\right) q_{n-2}(x)$$