

3.5 Adaptive quadrature

노트 제목

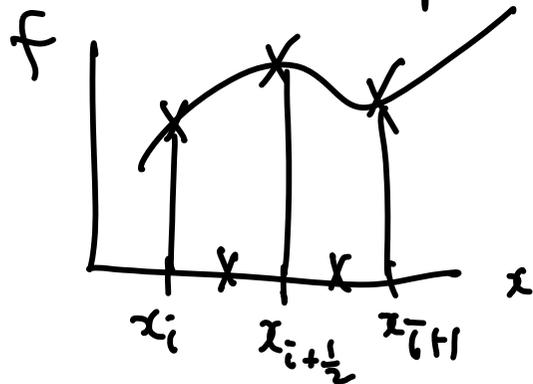
2019-10-02



→ user provides the error tolerance
then, the program automatically subdivides the interval
to achieve the prescribed accuracy.

$$\text{error } \epsilon \quad \left| I - \int_a^b f(x) dx \right| \leq \epsilon$$

Base: Simpson's rule



$$\text{For } (x_i, x_{i+1}), \quad S_i = \frac{h_i}{6} (f(x_i) + 4f(x_{i+1/2}) + f(x_{i+1}))$$

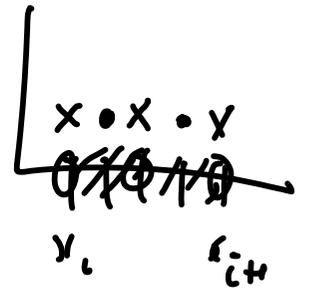
$$h_i = x_{i+1} - x_i$$

Subdivide into 4 panels.

$$S_i^{(2)} = \frac{h_i}{12} [f(x_i) + 4f(x_{i+\frac{1}{4}}) + 2f(x_{i+\frac{1}{2}}) + 4f(x_{i+\frac{3}{4}}) + f(x_{i+1})]$$

idea: Compare two estimates, S_i and $S_i^{(2)}$, and get an estimate for actual accuracy of $S_i^{(2)}$

Let I_i be the exact integral in (x_i, x_{i+1}) .



$$I_i - S_i = c h_i^5 f^{iv}(x_i + \frac{h_i}{2}) + \dots \quad (1)$$

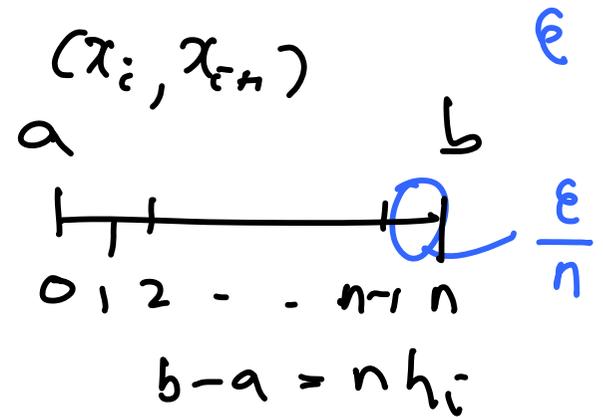
$$I_i - S_i^{(2)} = c \left(\frac{h_i}{2}\right)^5 \left[\underbrace{f^{iv}(x_i + \frac{h_i}{4})}_{=} + \underbrace{f^{iv}(x_i + \frac{3h_i}{4})}_{=} \right] + \dots$$

$$= c \left(\frac{h_i}{2}\right)^5 \cdot 2 f^{iv}(x_i + \frac{h_i}{2}) + \dots \quad (2)$$

$$\textcircled{1} - \textcircled{2} : S_i^{(2)} - S_i = \frac{15}{16} c h_i^5 f^{(4)}(x_i + \frac{1}{2}h_i) + \dots - \textcircled{3}$$

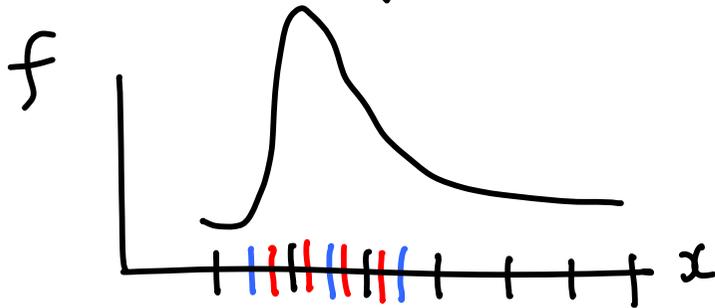
$$\textcircled{3} \rightarrow \textcircled{2} : I_i - S_i^{(2)} = \frac{1}{15} [S_i^{(2)} - S_i] + \dots$$

error in $S_i^{(2)}$ is $\frac{1}{15}$ of $S_i^{(2)} - S_i$



$$\frac{1}{15} |S_i^{(2)} - S_i| \leq \frac{\epsilon}{n} \rightarrow \text{ok}$$

" $> \frac{\epsilon}{n}$ \rightarrow subdivide further



'adaptive' quadrature

3.6 Gauss quadrature

Method that is optimum in the sense of maximum accuracy for a given number of function evaluations.

$$\int_a^b f(x) dx = \sum_{i=0}^n f_i w_i$$

$n+1$ grid pts $\left\{ \begin{array}{l} 2n+2 \text{ adjustable} \\ n+1 \text{ weights} \end{array} \right. \rightarrow$ parameters \rightarrow Construct a polynomial of degree $2n+1$.

Choose x_i and w_i for highest accuracy.

Let f be a polynomial of degree $2n+1$

Select grid points x_0, x_1, \dots, x_n (so far, we don't know where x_i 's are)
 $f(x_0), f(x_1), \dots, f(x_n)$

Interpolate w/ Lagrange polynomial of degree n

$$P(x) = \sum_{j=0}^n f_j L_j(x)$$

$f-p$: polynomial of degree $2n+1$

& $f-p = 0$ @ $x = x_0, x_1, \dots, x_n$

$\Rightarrow F(x) = (x-x_0)(x-x_1) \dots (x-x_n)$: polynomial of degree $n+1$.

$$\Rightarrow f-p = F(x) \sum_{l=0}^n g_l x^l$$

$$\int f dx - \int p dx = \int F \sum_{l=0}^n g_l x^l dx = \sum_{l=0}^n g_l \underbrace{\int F(x) x^l dx}_0$$

Demand $\int F(x) x^l dx = 0$ for $l=0, 1, 2, \dots, n$

$\rightarrow F(x)$ is orthogonal to all polynomials of degree n or less.

$$\text{Then } \int f(x) dx = \int p(x) dx = \int \sum_{j=0}^n f_j L_j dx = \sum_{j=0}^n f_j \underbrace{\int L_j(x) dx}_{w_j}$$

Here, F belongs to class of Legendre polynomials.

x_j are the zeros of " " " " .

$$\int_{-1}^1 F_n F_m dx = \delta_{nm} = \begin{cases} 0 & m \neq n \\ 1 & m = n \end{cases}$$

$$F_0(x) = 1, F_1(x) = x, F_2(x) = \frac{1}{2}(3x^2 - 1), F_3(x) = \frac{1}{2}(5x^3 - 3x),$$

$$F_4(x) = \frac{1}{8}(35x^4 - 30x^2 + 3), F_5(x) = \frac{1}{8}(63x^5 - 70x^3 + 15x), \dots$$

$$F_5 : x_0 = -0.9061, x_1 = -0.5384, x_2 = 0, x_3 = 0.5384, x_4 = 0.9061.$$

One can transform $a \leq x \leq b$ to $-1 \leq \xi \leq 1$

using $x = \frac{1}{2}(a+b) + \frac{1}{2}(b-a)\xi$.

$$\text{ex) } I = \int_0^k e^x dx = 53.59815003$$

$$5 \text{ pts. } \quad x = \frac{1}{2}(a+b) + \frac{b-a}{2}\xi \rightarrow dx = \frac{b-a}{2} d\xi = 2 d\xi$$

$$\downarrow$$

$$5 \text{ zeros} \quad I = \int_0^k e^x dx = 2 \int_{-1}^1 f(\xi) d\xi.$$

ξ_j	w_j
-0.9061	0.2369
-0.5384	0.4786
0	0.5688
0.5384	0.4786
0.9061	0.2369

$$I = 2 \sum_{j=0}^4 f_j w_j = 53.59813663$$

$$\rightarrow \mathcal{E} = 0.0000134$$

cf $\mathcal{E} = 0.018$ w/ Simpson's rule
w/ 9 pts.

Disadvantage: lack of a simple method for systematic error reduction.

• $\int_0^{\infty} e^{-x} f(x) dx = \sum_j f_j w_j$ Gauss-Laguerre formula

orthogonality $\int_0^{\infty} L_n L_m e^{-x} dx = \delta_{nm}$

$$\cdot \int_{-\infty}^{\infty} e^{-x^2} f(x) dx = \sum_j f_j w_j \quad \text{Gauss-Hermite quadrature}$$

$$\text{orthogonality} \quad \int_{-\infty}^{\infty} H_n H_m e^{-x^2} dx = \delta_{nm}$$

$$\cdot \int_{-1}^1 \frac{1}{\sqrt{1-x^2}} f(x) dx \quad \text{Chebyshev-Gauss quadrature}$$

$$\text{orthogonality} \quad \int_{-1}^1 T_n T_m \frac{1}{\sqrt{1-x^2}} dx = \delta_{nm}$$

$$* \text{Singularity} \quad I = \int_0^1 \frac{e^x}{\sqrt{x}} dx$$

$$\textcircled{1} \text{ substitution: let } x = t^2 \rightarrow dx = 2t dt \rightarrow I = 2 \int_0^1 \sqrt{x} e^x dx$$

$$\textcircled{2} \text{ Integration by parts: } \begin{matrix} v = 2\sqrt{x} \\ u = e^x \end{matrix} \rightarrow I = 2\sqrt{x} e^x \Big|_0^1 - 2 \int_0^1 \sqrt{x} e^x dx$$

* Integrals w/ ∞ limits

$$I = \int_0^{\infty} e^{-x^2} dx \rightarrow \textcircled{1} \text{ Gauss quadrature}$$

$\textcircled{2}$ change the indep. variable

$$t = \frac{1}{1+s} \text{ maps } [0, \infty) \text{ to } [0, 1].$$

Ch. 4 Numerical solution of ordinary diff'l eqs (ODEs)

$$y'' + \omega^2 y = f(x)$$

$$\begin{cases} y(0) = y_0 \\ \left. \frac{dy}{dx} \right|_0 = v \end{cases}$$

or $\begin{cases} y(0) = y_0 \\ y(L) = y_L \end{cases}$

↑
initial value problems : all the conditions are given at one point.

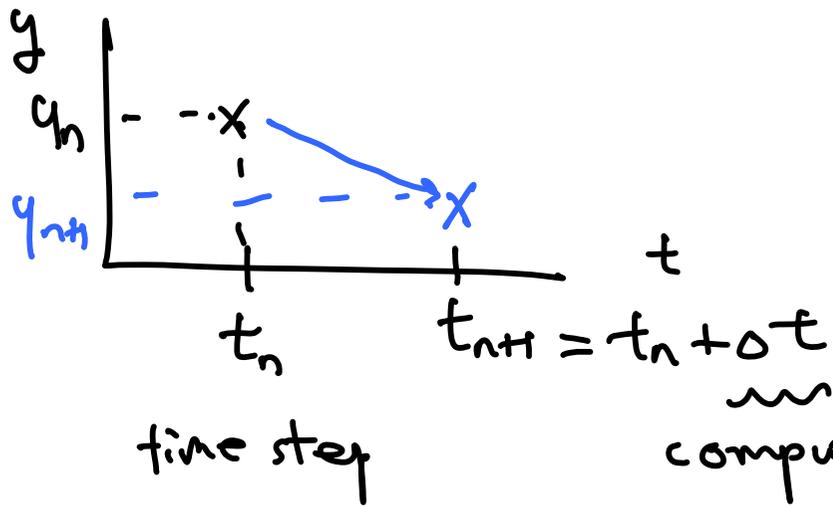
boundary value problems : conditions are given at more than one pt.

4.1 Initial value problems

$$\begin{cases} y' = \frac{dy}{dt} = f(y, t) \\ y(0) = y_0 \end{cases}$$

Higher-order ODEs can be converted to a system of 1st-order ODEs

$$\begin{array}{lcl} y'' + \omega^2 y = f & & y_1 = y \\ \downarrow & \searrow & y_2 = y_1' = y' \\ y'' = y_2' & & y_2' + \omega^2 y_1 = f \end{array}$$



$$\Rightarrow \begin{cases} y_1' = y_1 \\ y_2' = -\omega^2 y_1 + f \end{cases}$$

All methods assume that solution is known at $a \leq t \leq t_n$,
and use it to get the Sol at $t = t_{n+1}$.

$$y' = f(y, t)$$

Taylor series expansion $\parallel h$

$$y_{n+1} = y(t_{n+1}) = y(t_n + \Delta t) = y_n + h \underline{y_n'} + \frac{h^2}{2} \underline{y_n''} + \frac{h^3}{6} \underline{y_n'''} + \dots$$

$$y'_n = f(y_n, t_n)$$

$$y''_n = \frac{d}{dt} y'_n = \frac{d}{dt} f(y, t) \Big|_n = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y} \frac{dy}{dt} \Big|_n = f_{t_n} + f_{y_n} f_n$$

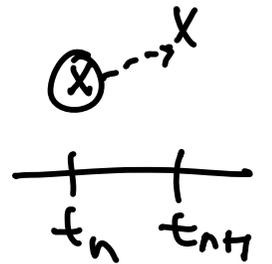
$$y'''_n = \frac{d}{dt} (f_t + f_y f) = \frac{\partial}{\partial t} (f_t + f_y f) + \frac{\partial}{\partial y} (f_t + f_y f) \frac{dy}{dt} \stackrel{f}{=} \\ = f_{tt_n} + f_{ty_n} f_n + f_{yt_n} f_t + f_{ty_n} f_{t_n} + f_{yy_n} f_n^2 + f_{tn} f_{y_n}^2$$

of terms increases very rapidly.

Hence, it is not very practical to include higher order terms than third order.

① Euler method (Explicit Euler or Forward Euler)

$$y' = f(y, t) \rightarrow y_{n+1} = y_n + h f(y_n, t_n) + \mathcal{O}(h^2)$$



$$y' = \lim_{h \rightarrow 0} \frac{y_{n+1} - y_n}{h}$$

or
$$\frac{y_{n+1} - y_n}{h} = f(y_n, t_n)$$

2nd-order accurate for one time step
globally (integration from t_0 to t_f),

1st-order accurate

$$T = nh \quad \text{error} \sim \sum h^2 y'' \rightarrow \frac{T}{h} h^2 y'' \rightarrow \mathcal{O}(h)$$

ex) $\frac{dy}{dt} = y' = -y^2$ $y(0) = 1$ exact sol. $y = \frac{1}{1+t}$

$h = 0.1$

EE : $y_{n+1} = y_n + hf(y_n, t_n) = y_n - hy_n^2$

$y_0 = 1$

$y_1 = y_0 - hy_0^2 = 0.9$

$y_2 = y_1 - hy_1^2 = \dots$

\vdots

$y_{10} = y_9 - hy_9^2 = 0.4629$

Exact

0.90909

\vdots

\vdots

0.5