

Numerical Integration and Monte Carlo Integration

Elementary schemes for integration over one variable

Multi-dimensional integration

- dimension-by dimension

Problems with multi-dimensional numerical integrations

Monte Carlo sampling of high-dimensional integrals

- including some aspects of analysis of statistical data

Numerical integration in one dimension

Function of one variable x , assume no singularities

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

Discretize the x -axis

- $n+1$ equally spaced points including a, b :

$$[a, b] \rightarrow \{x_0, x_1, \dots, x_n\} \quad h \equiv x_i - x_{i-1}$$

Consider groups of $m+1$ points (m intervals of size h)

Construct the order- m polynomials fitting the $m+1$ points

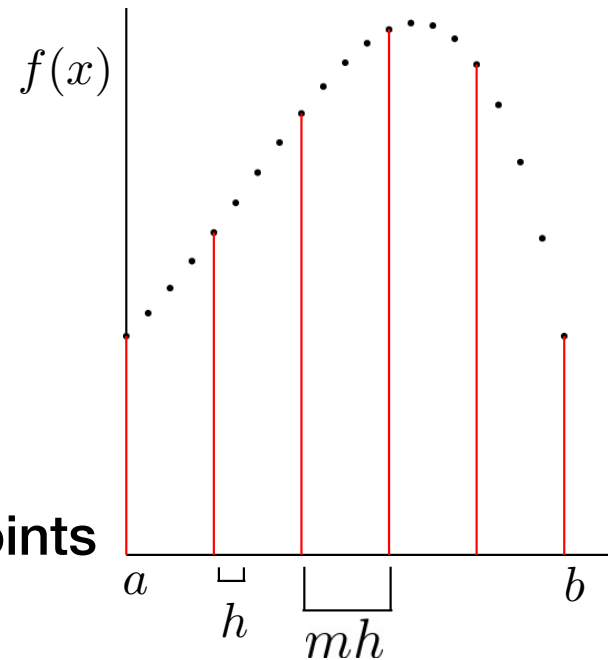
$$I = \sum_{i=1}^{n/m} I_i, \quad I_i = \int_{a+(i-1)mh}^{a+imh} P_i(x) dx$$

Simple formulas exist to construct the polynomials $P_i(x)$

Integrate the polynomials exactly and add up

Leads to simple integration formulas (sums) for small m

Error for one window typically of order $O(h^{m+1})$ or $O(h^{m+2})$



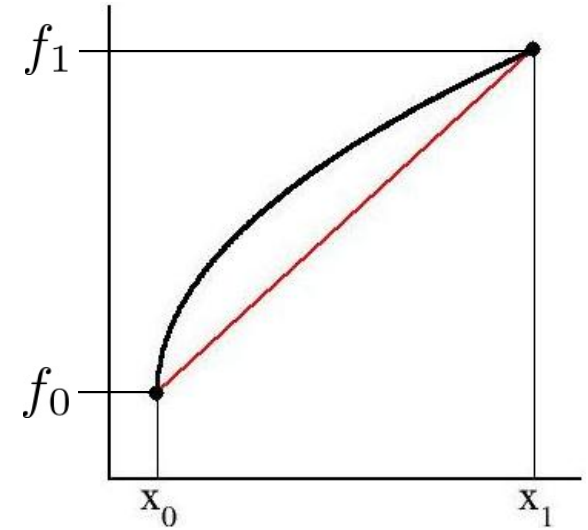
Simplest case; m=1 (trapezoidal rule)

$$f(x_0 + \delta) = a + b\delta, \quad 0 \leq \delta \leq h$$

$$f(x_0) = f_0, \quad f(x_0 + h) = f_1$$

$$a = f_0, \quad b = (f_1 - f_0)/h$$

$$\begin{aligned} I_1 &= \int_{x_0}^{x_1} P_1(x) dx = \int_0^h P_1(\delta) d\delta = [a\delta + b\delta^2/2]_0^h \\ &= h(a + bh/2) = \frac{h}{2}(f_0 + f_1) \end{aligned}$$



Here we also see that the error (“step error”) is $O(h^3)$

For the total error, we have to sum up step errors from (general m)

$$I = \sum_{i=1}^{n/m} I_i, \quad I_i = \int_{a+(i-1)mh}^{a+imh} P_i(x) dx \quad n/m = \frac{x_n - x_0}{mh}$$

Assuming no “lucky” error cancellations (from sign oscillations)

- error is $O(h^{m+2})O(h^{-1}) \sim O(h^{m+1})$

Second order; m=2 (Simpson's rule)

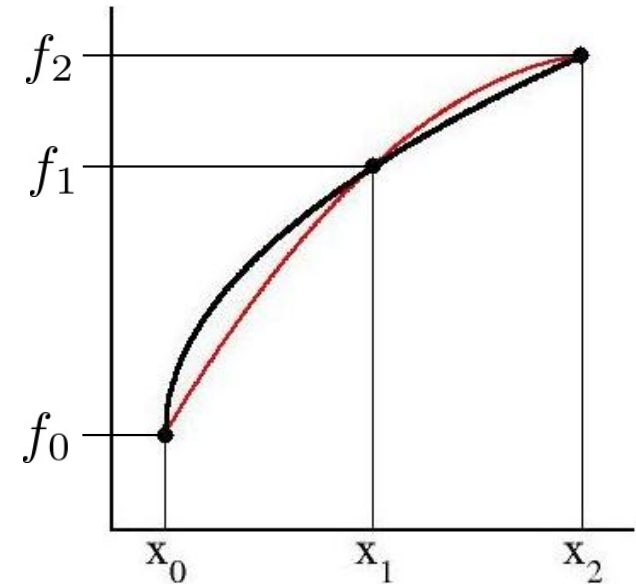
$$f(x_0 + \delta) = a + b\delta + c\delta^2, \quad 0 \leq \delta \leq 2h$$

$$f(x_0) = f_0, \quad f(x_0 + h) = f_1, \quad f(x_0 + 2h) = f_2$$

$$f_0 = a, \quad f_1 = a + bh + ch^2, \quad f_2 = a + 2bh + 4ch^2$$

Solve for a,b,c, integrate polynomial →

$$I_1 = \int_{x_0}^{x_2} P_1(x)dx = \frac{h}{3}(f_0 + 4f_1 + f_2)$$



What is the order of the error?

- from the polynomial it may seem $O(h^4)$ (from missing integrated δ^3 term)

Write expansion around x_1 instead:

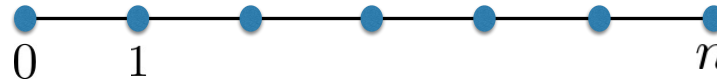
$$f(x_1 + \delta) = a + b\delta + c\delta^2 + d\delta^3 + e\delta^4, \quad -h \leq \delta \leq h$$

When integrated in the symmetric window, all odd powers give 0

Same formula for I_1 as above when a,b,c terms included, d term gives 0

Error is $O(h^5)$, becomes $O(h^4)$ for range $[x_0, x_n]$

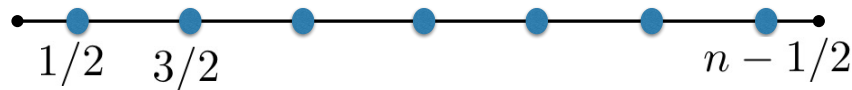
Extended formulas



$$\int_{x_0}^{x_n} f(x) dx = h \left(\frac{1}{2} f_0 + f_1 + f_2 + f_3 + \dots + f_{n-1} + \frac{1}{2} f_n \right) + O(h^2) \quad \text{trapezoid}$$

$$\int_{x_0}^{x_n} f(x) dx = \frac{h}{3} (f_0 + 4f_1 + 2f_2 + 4f_3 + \dots + 4f_{n-1} + f_n) + O(h^4) \quad \text{Simpson}$$

For integrands with singularities at the end point(s); open interval formulas

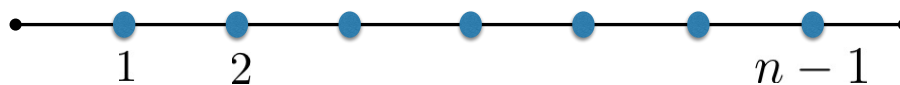


midpoint rule

$$\int_{-h/2}^{h/2} f(x_{i+1/2} + \delta) d\delta = \int_{-h/2}^{h/2} (f_{i+1/2} + b\delta + c\delta^2) d\delta = hf_{i+1/2} + O(h^3)$$

$$\int_{x_0}^{x_n} f(x) dx = h(f_{1/2} + f_{3/2} + \dots + f_{n-3/2} + f_{n-1/2}) + O(h^2)$$

Alternative:



$$\int_{x_0}^{x_n} f(x) dx = h \left(\frac{3}{2} f_1 + f_2 + f_3 + \dots + f_{n-2} + \frac{3}{2} f_{n-1} \right) + O(h^2)$$

- interior points have $O(h^3)$ errors, sum to $O(h^2)$, end points contribute $O(h^2)$ errors

Comments on singularities

Open-interval formulas can be used

- singular point(s) should be at end(s); divide up interval in parts if needed
- but convergence with number of points n may be very slow

Divergent part can some times be subtracted and solved analytically

More sophisticated methods exist for difficult cases

Other methods

Gaussian quadrature:

- non-uniform grid points; $n+1$ points $\rightarrow$ exact result for polynomial of order n
- **several Julia packages, e.g., [FastGaussQuadrature.jl](#)**

Gauss-Kronrod quadrature:

- uses two Gaussian quad. evaluations for different n
- **package [QuadGK.jl](#) uses a version of this method**

Idea similar to
“Romberg method”
(example tomorrow)

Adaptive grid (adaptive mesh):

- dynamically adapted to be more dense where most needed

Infinite integration range

Change variables to make range finite

Multi-Dimensional integration

$$I = \int_{a_n}^{b_n} dx_n \cdots \int_{a_2}^{b_2} dx_2 \int_{a_1}^{b_1} dx_1 f(x_1, x_2, \dots, x_n),$$

Can be carried out numerically dimension-by-dimension

Example, function of two variables

$$I = \int_{a_y}^{b_y} dy \int_{a_x(y)}^{b_x(y)} dx f(x, y)$$

Integrating numerically over x first, gives a function of y:

$$F(y) = \int_{a_x(y)}^{b_x(y)} dx f(x, y)$$

This has to be done for values of y on a grid, to be used in

$$I = \int_{a_y}^{b_y} dy F(y)$$

Very time consuming for large dimensionality D; scaling M^D of effort
- M represents ~mean number of grid points for 1D integrals

