Numerical Integration and Monte Carlo Integration

Elementary schemes for integration over one variable

Multi-dimensional integration
- dimension-by dimension

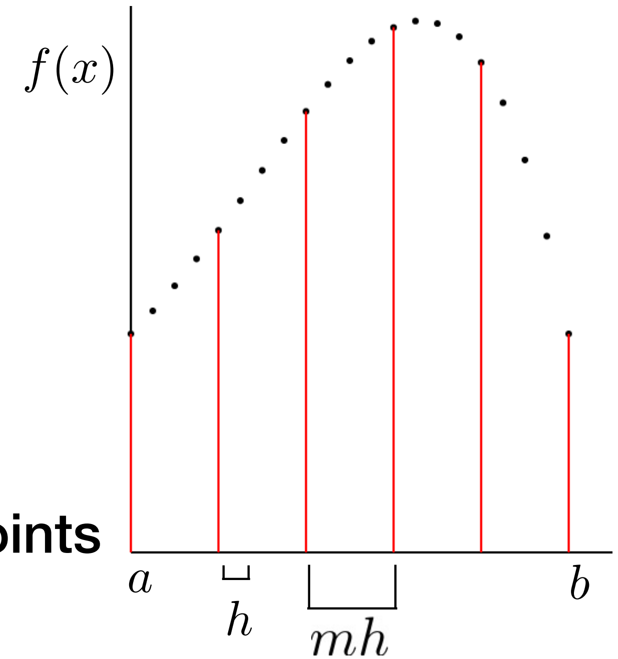
Problems with multi-dimensional numerical integrations

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:

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

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)

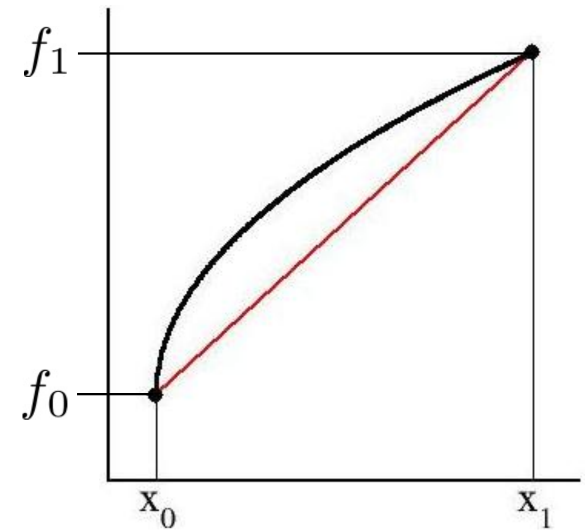
$$f(x_0+\delta) = a + b\delta, \quad 0 \le \delta \le h$$

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

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

$$I_1 = \int_{x_0}^{x_1} P_1(x)dx = \int_0^h P_1(\delta)d\delta = \left[a\delta + b\delta^2/2\right]_0^h$$

$$= h(a + bh/2) = \frac{h}{2}(f_0 + f_1)$$



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)

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

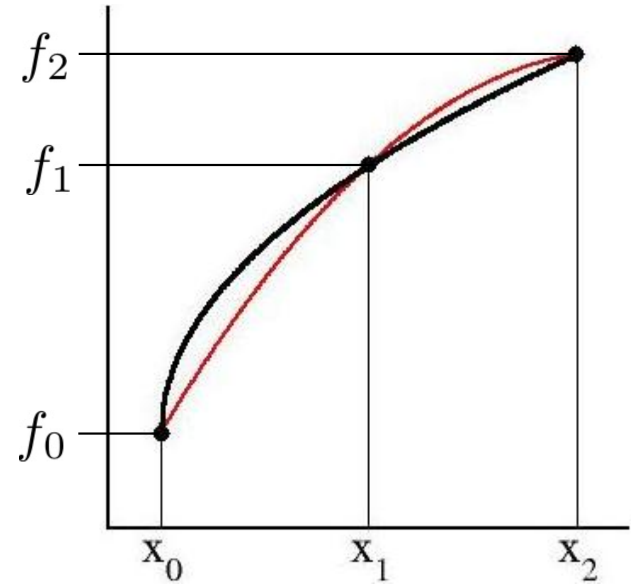
Assuming no "lucky" error cancellations (from sign oscillations)
- error is O(h^{m+2})O(h^{-1}) ~ O(h^{m+1})

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

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

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

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



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:

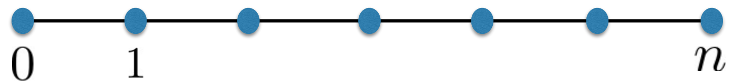
$$f(x_1+\delta) = a + b\delta + c\delta^2 + d\delta^3 + e\delta^4, \quad -h \le \delta \le h$$

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 $[x_0,x_n]$

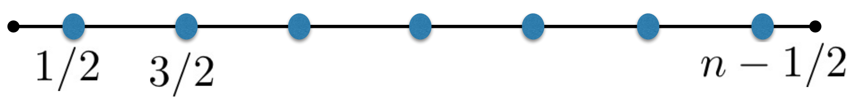
Extended formulas



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

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

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

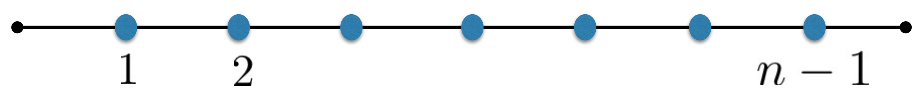


midpoint rule

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

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

Alternative:



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

- 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 → 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, similarly to Romberg
- **package QuadGK.jl uses a version of this method**

Adaptive grid (adaptive mesh):

- dynamically adapted to be more dense where most needed

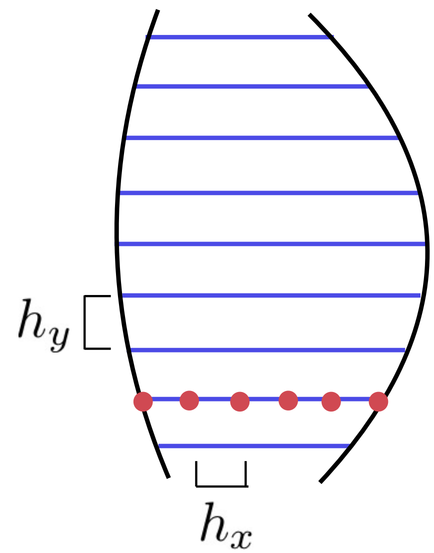
Infinite integration range

Change variables to make range finite

Multi-Dimensional integration

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

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

Monte Carlo Integration

An integral over a finite volume V:

- is (by definition) the mean value of the function times the volume

$$I = \int_a^b f(x)dx = (b-a)\langle f \rangle$$

The mean value <f> can be estimated by sampling

- generate N random (uniformly distributed) x values x_i in the range, then

$$\bar{f} = \frac{1}{N}\sum_{i=1}^{N} f(x_i) \to \langle f \rangle, \quad \text{when } N \to \infty$$

For finite N, there is a statistical error:

$$\langle \bar{f} - \langle f \rangle \rangle \propto \frac{1}{\sqrt{N}}$$

interepretation of the mean error:
If the "simulation" is repeated many times, the averaged squared error (variance) tends to a value ***a/N***, for with ***a*** some constant

The statistical result for the integral should be expressed as

$$I = \bar{I} \pm \sigma = V(\bar{f} + \sigma/V) \qquad \sigma \propto N^{-1/2}$$

Computing the "error bar" σ is an important aspect of the sampling method

Standard ilustration of MC integration; estimate of π

Consider a circle of radius 1, centered at (x,y)=0. Define a function:

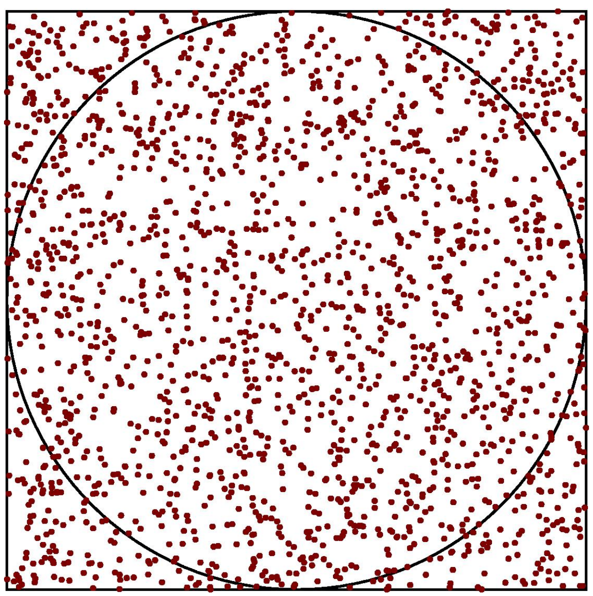
$$f(x,y) = \begin{cases} 1, & \text{if } x^2+y^2 \leq 1 \\ 0, & \text{if } x^2+y^2 > 1 \end{cases}$$

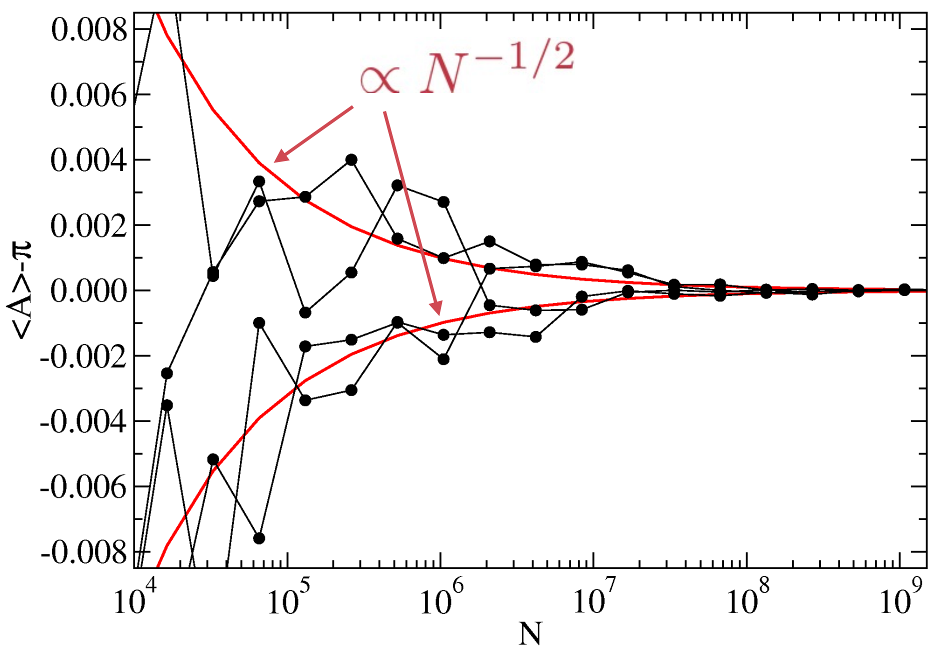
mean value inside the surrounding box

Use MC sampling to compute:

$$A = \int_{-1}^{1} dy \int_{-1}^{1} dx f(x,y) = \pi \quad = 4\langle f \rangle_{\square}$$

Expected fraction of "hits" inside circle = $\pi/4$





The error after N steps

Four repetitions of a simulation, dots showing partial results as the mean value evolves

We should compute the statistical error properly