Physics 77

Introduction to Computational Techniques in Physics

Spring 2019

Numerical Integration

Heather Gray, Amin Jazaeri, Yury Kolomensky

See also Python workbook Lecture08b.ipynb

What is numerical integration really calculating?

- Consider the definite integral

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

- The integral can be approximated by weighted sum

$$I \approx \sum_{i=1}^{n} \omega_i f(x_i)\, \Delta x_i$$

- The ω_ι are weights, and the x_i are abscissas
- Assuming that *f* is finite and continuous on the interval [*a*,*b*] numerical integration leads to a unique solution
- The goal of any numerical integration method is to choose abscissas and weights such that errors are minimized for the smallest n possible for a given function

Numerical Integration

- What are the choices of evaluation points, x_i's and weights w_i's such that:

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

- In general we have two sets of degree of freedom:
  - The spacing of evaluation points.
  - The weighted importance of each point.

Numerical Integration

- Numerical integration Methods:

1. Upper and Lower Sums
2. Newton-Cotes Methods:
   a) Trapezoid Rule
   b) Simpson Rules
3. Romberg Method
4. Gauss Quadrature

Upper and Lower Sums

The interval is divided into subintervals

$$Partition\ P = \{a = x_0 \le x_1 \le x_2 \le \ldots \le x_n = b\}$$

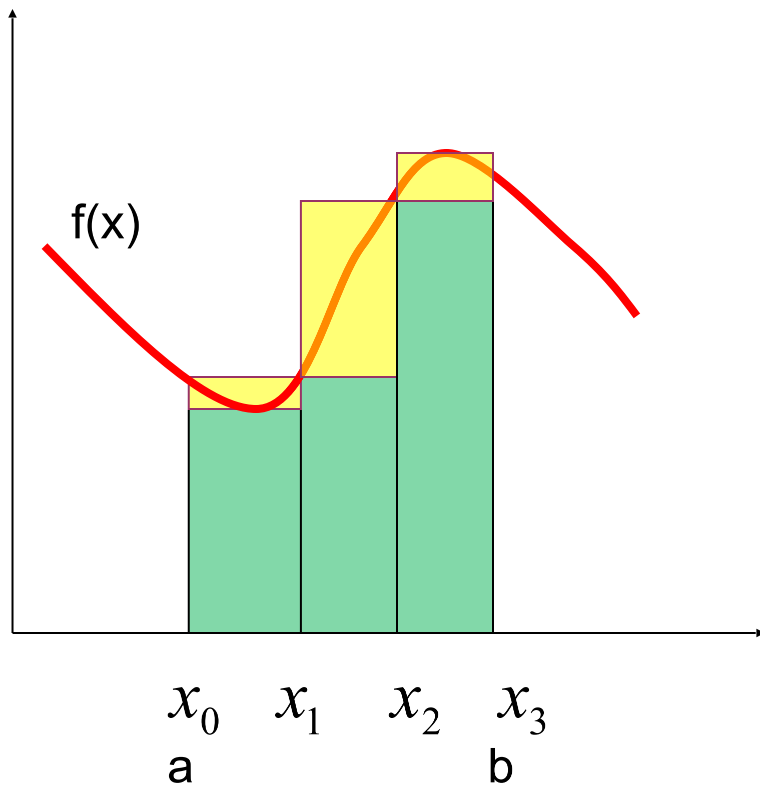
Define

$$m_i = \min\{f(x) : x_i \le x \le x_{i+1}\}$$

$$M_i = \max\{f(x) : x_i \le x \le x_{i+1}\}$$

Lower sum $\quad L(f,P) = \sum_{i=0}^{n-1} m_i (x_{i+1} - x_i)$

Upper sum $\quad U(f,P) = \sum_{i=0}^{n-1} M_i (x_{i+1} - x_i)$



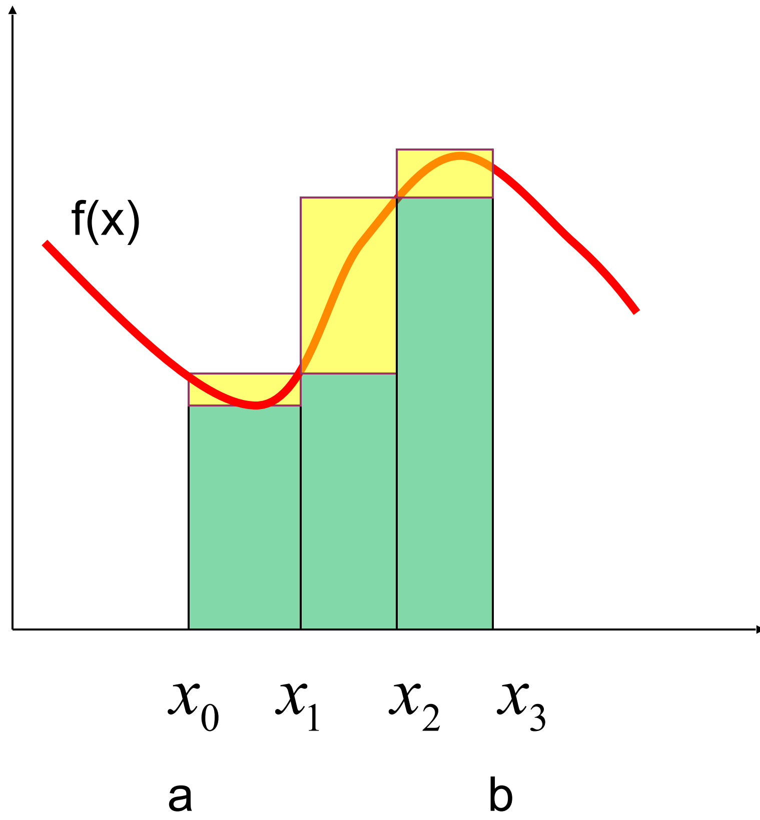
Upper and Lower Sums

Lower sum $\quad L(f,P) = \sum_{i=0}^{n-1} m_i \left(x_{i+1} - x_i\right)$

Upper sum $\quad U(f,P) = \sum_{i=0}^{n-1} M_i \left(x_{i+1} - x_i\right)$

$$\text{Estimate of the integral} = \frac{L+U}{2}$$

$$Error \leq \frac{U-L}{2}$$



Example

$\int_0^1 x^2 \, dx$ = 1/3 (true value)

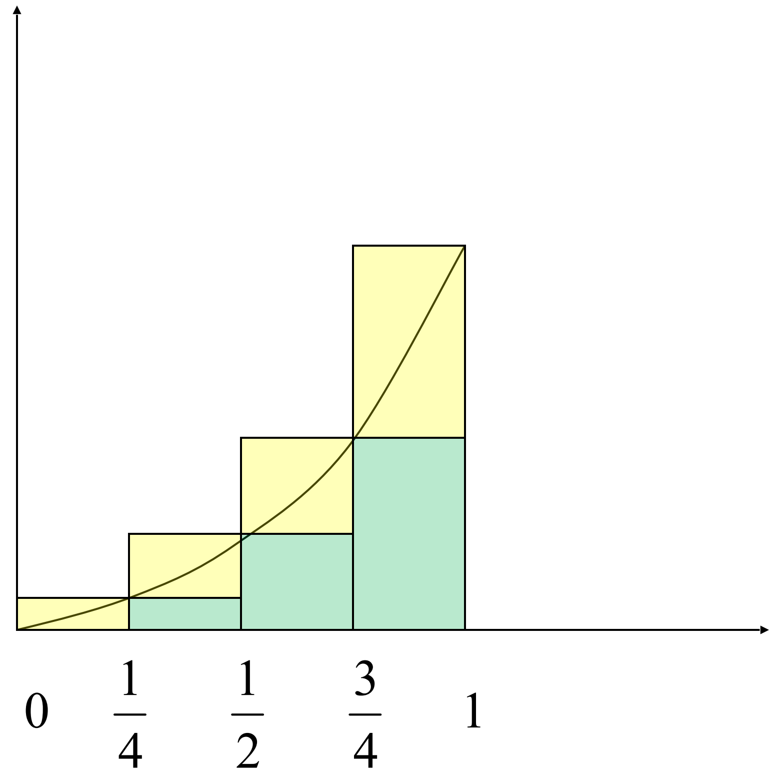
$$Partition\ P = \left\{0, \frac{1}{4}, \frac{2}{4}, \frac{3}{4}, 1\right\}$$

$n = 4$ (four equal intervals)

$$m_0 = 0, \quad m_1 = \frac{1}{16}, \quad m_2 = \frac{1}{4}, \quad m_3 = \frac{9}{16}$$

$$M_0 = \frac{1}{16}, \quad M_1 = \frac{1}{4}, \quad M_2 = \frac{9}{16}, \quad M_3 = 1$$

$$x_{i+1} - x_i = \frac{1}{4} \qquad for \quad i = 0, 1, 2, 3$$



Example

Lower sum $L(f,P)=\sum_{i=0}^{n-1} m_i\left(x_{i+1}-x_i\right)$

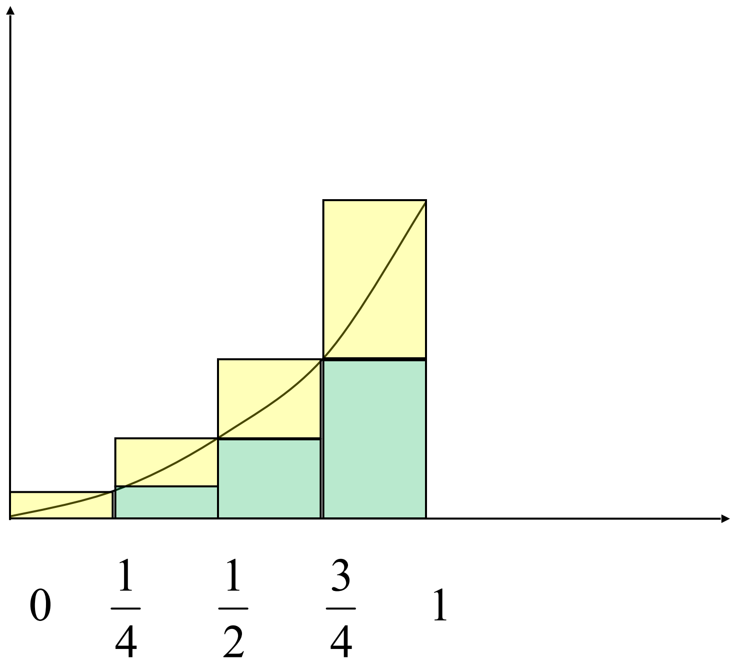
$$L(f,P)=\frac{1}{4}\left[0+\frac{1}{16}+\frac{1}{4}+\frac{9}{16}\right]=\frac{14}{64}$$

Upper sum $U(f,P)=\sum_{i=0}^{n-1} M_i\left(x_{i+1}-x_i\right)$

$$U(f,P)=\frac{1}{4}\left[\frac{1}{16}+\frac{1}{4}+\frac{9}{16}+1\right]=\frac{30}{64}$$

$$\text{Estimate of the integral}=\frac{1}{2}\left(\frac{30}{64}+\frac{14}{64}\right)=\frac{11}{32}$$

$$Error<\frac{1}{2}\left(\frac{30}{64}-\frac{14}{64}\right)=\frac{1}{8}$$



Estimate - True_Value = 0.01 << Error_Estimate

Upper and Lower Sums

- Estimates based on Upper and Lower Sums are easy to obtain for **monotonic** functions (always increasing or always decreasing).
- For non-monotonic functions, finding maximum and minimum of the function can be difficult and other methods can be more attractive.

Newton-Cotes Methods

- In Newton-Cote Methods, the function is approximated by a polynomial of order *n*
- Computing the integral of a polynomial is easy.

$$\int_a^b f(x)dx \approx \int_a^b \left(a_0 + a_1 x + \ldots + a_n x^n\right)dx$$

$$\int_a^b f(x)dx \approx a_0(b-a) + a_1 \frac{(b^2 - a^2)}{2} + \ldots + a_n \frac{(b^{n+1} - a^{n+1})}{n+1}$$

Newton-Cotes Methods

- Trapezoid Method (First Order Polynomial are used)

$$\int_a^b f(x)dx \approx \int_a^b \left(a_0 + a_1 x\right)dx$$

- Simpson 1/3 Rule (Second Order Polynomial are used),

$$\int_a^b f(x)dx \approx \int_a^b \left(a_0 + a_1 x + a_2 x^2\right)dx$$

Trapezoid Method

Derivation-One interval

$$I = \int_a^b f(x)dx \approx \int_a^b \left(f(a) + \frac{f(b) - f(a)}{b - a}(x - a) \right) dx$$

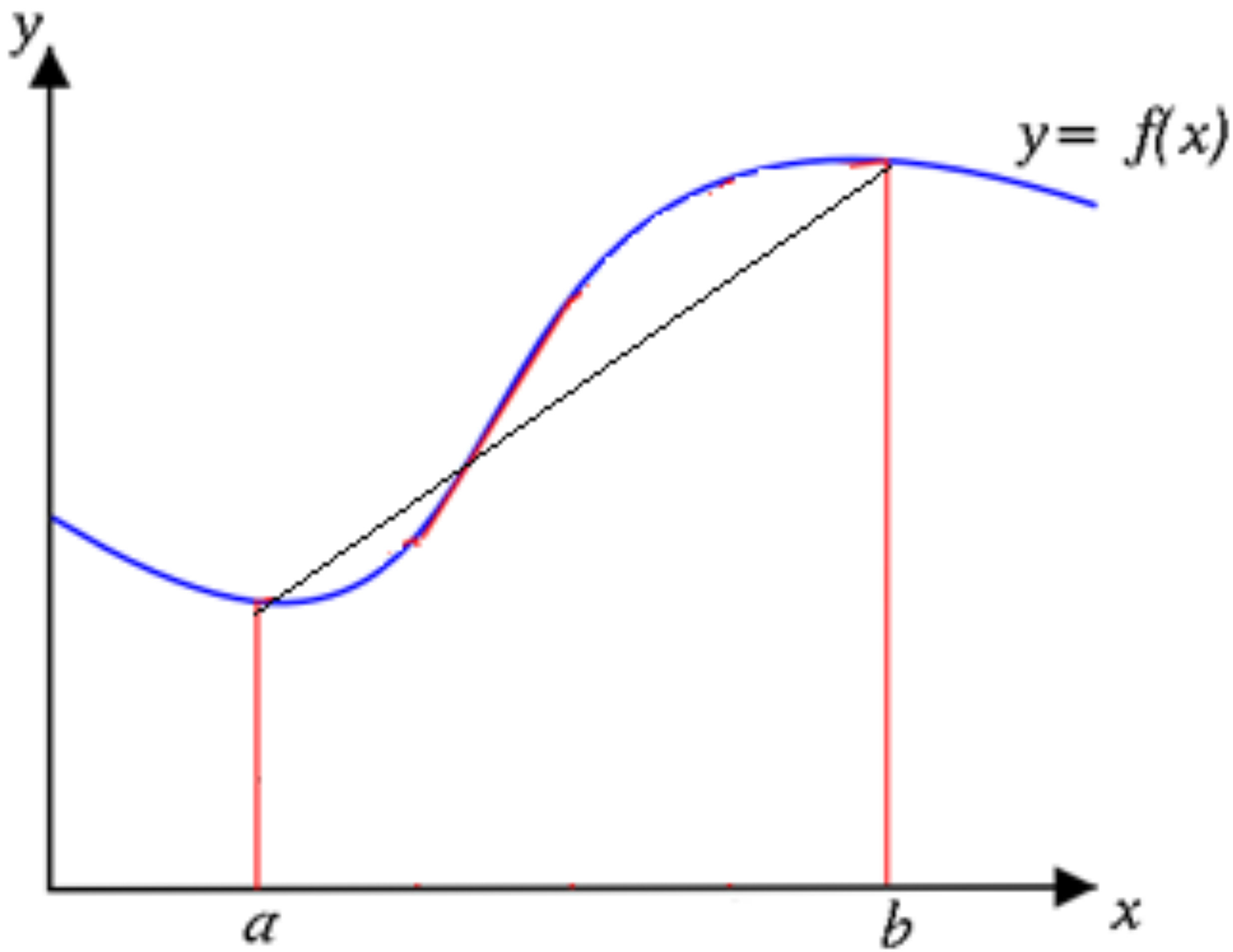
$$I \approx \int_a^b \left(f(a) - a\frac{f(b) - f(a)}{b - a} + \frac{f(b) - f(a)}{b - a} x \right) dx$$

$$= \left(f(a) - a\frac{f(b) - f(a)}{b - a} \right) x \Big|_a^b + \frac{f(b) - f(a)}{b - a} \frac{x^2}{2} \Big|_a^b$$

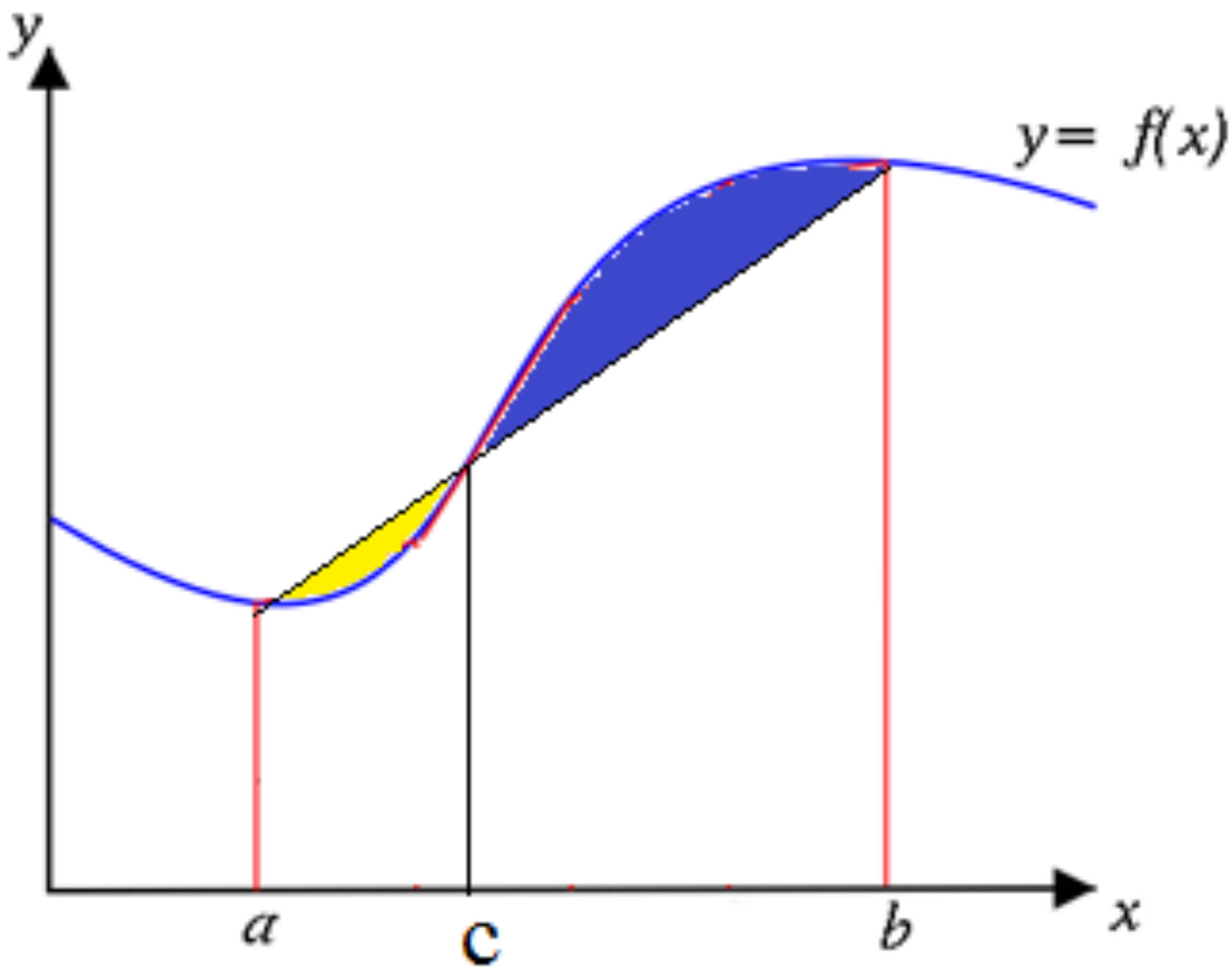
$$= \left(f(a) - a\frac{f(b) - f(a)}{b - a} \right) (b - a) + \frac{f(b) - f(a)}{2(b - a)} (b^2 - a^2)$$

$$= (b - a)\frac{f(b) + f(a)}{2}$$

Trapezoid Rule

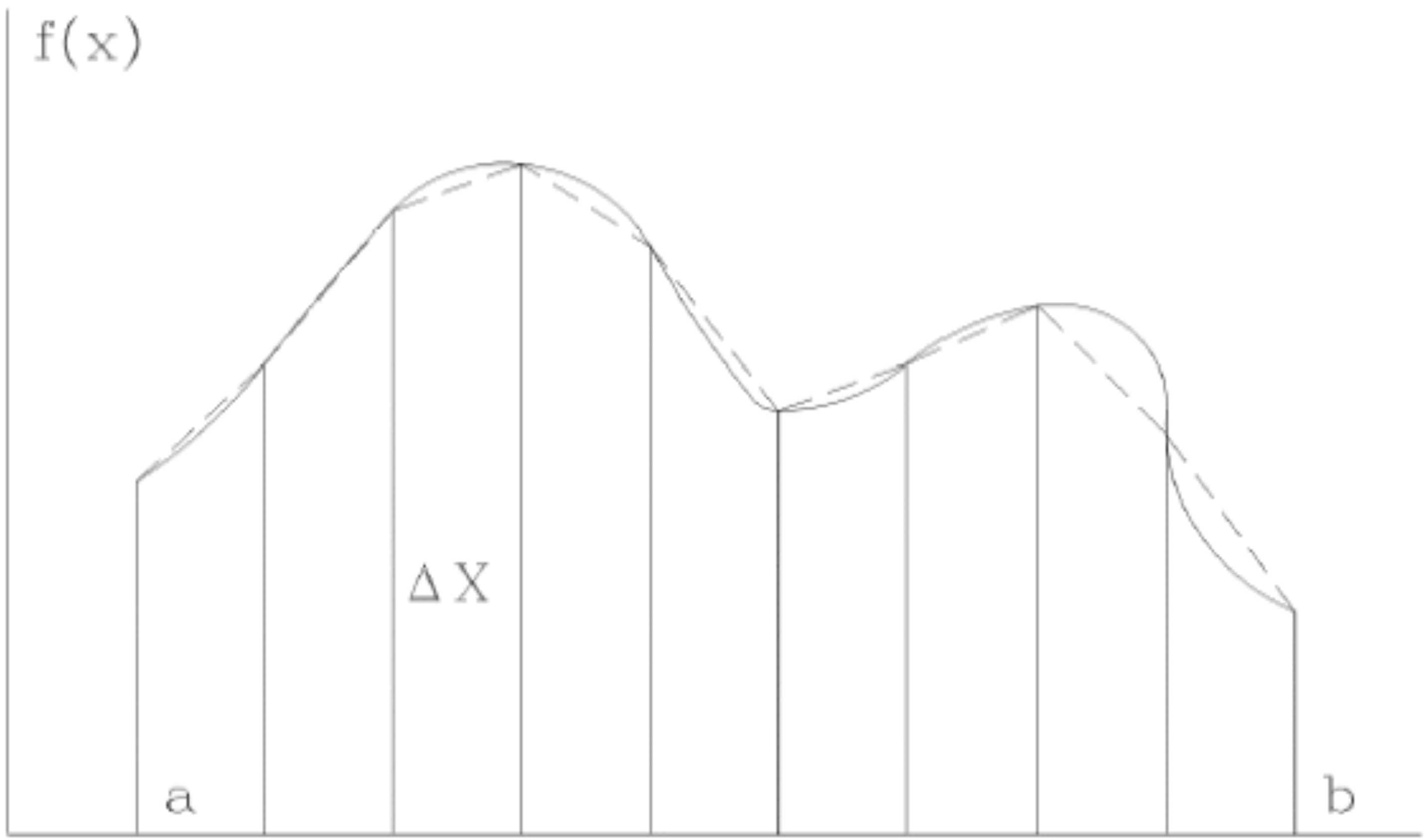


Trapezoid Rule

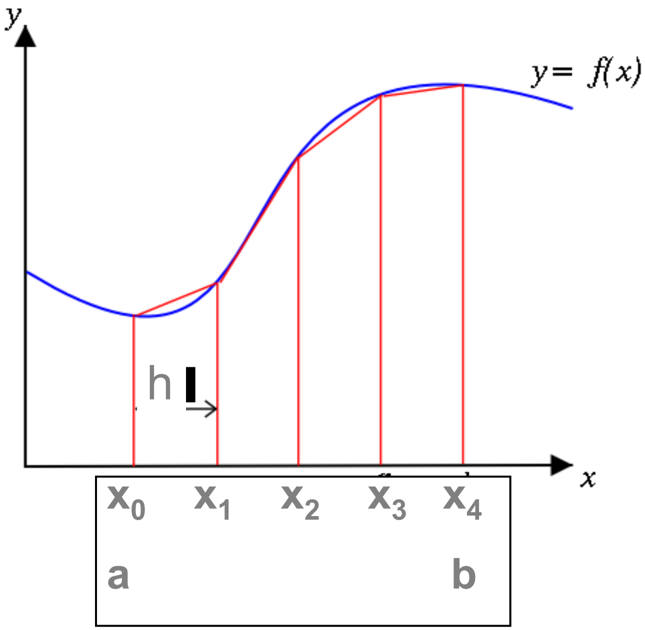


Numerical Integration

- Composite Trapezoid Rule



Trapezoid Rule



- Approximate integral using the areas of the trapezoids

$$\int_a^b f(x)dx \cong \sum \text{area of trapezoids}$$

$$= h\frac{f(x_0)+f(x_1)}{2} + h\frac{f(x_1)+f(x_2)}{2}$$

$$+ h\frac{f(x_2)+f(x_3)}{2} + h\frac{f(x_3)+f(x_4)}{2}$$

$$= \frac{h}{2}\left[f(x_0)+2f(x_1)+2f(x_2)+2f(x_3)+f(x_4)\right]$$

$$= \frac{h}{2}\left[f(x_0)+f(x_4)\right] + h\left[f(x_1)+f(x_2)+f(x_3)\right]$$

For n intervals this generalizes to

$$\int_a^b f(x)dx \cong \frac{h}{2}\left[f(a)+f(b)\right] + h\sum_{i=1}^{n-1} f(x_i) \qquad (1)$$

The *trapezoidal rule*.

Numerical Integration

- Trapezoid Rule

$$\int_a^b f(x)dx = \sum_{i=1}^{n-1} \frac{f(x_{i+1}) + f(x_i)}{2} \Delta x_i$$

$$\Delta x_i = (b-a)/(n-1)$$

- So the weights are:

$$\{w_i\}_{i=0}^{N} : w_i \begin{Bmatrix} 0.5 & i = 0, N \\ 1 & otherwise \end{Bmatrix}$$

Simpson's Rule

- Trapezoidal rule was based on approximating the integrand by a first order polynomial, and then integrating the line over the interval.
- Simpson's 1/3rd rule is an extension of Trapezoidal rule where the integrand is approximated by a second order polynomial.

$$I = \int_a^b f(x)dx \approx \int_a^b f_2(x)dx$$

Where $f_2(x)$ is a second order polynomial.

$$f_2(x) = a_0 + a_1 x + a_2 x^2$$

Simpson's Rule

- Simpson's rule:

$$\boxed{\int_a^b f(x)dx = \frac{\Delta x}{3}[f(a) + 4f(a+\Delta x) + f(b)]}$$

where $\Delta x = \dfrac{b-a}{2}$

- Simpson's rule can only be applied when there are an even number of subintervals:

$$\int_{x_1}^{x_n} f(x)dx \approx \sum_{i=1,3,5}^{n-2} \frac{x_{i+1}-x_i}{3}[f(x_i) + 4f(x_{i+1}) + f(x_{i+2})]$$

Composite Simpson's Rule

- The Simpson's rule for x0, x1, x2 is given as:

$$\int_{x_0}^{x_2} p_1(x)dx = \frac{h}{3}\left[f(x_0) + 4f(x_1) + f(x_2)\right]$$

- We can do the same on x_2,x_3,x_4 to get

$$\int_{x_2}^{x_4} p_1(x)dx = \frac{h}{3}\left[f(x_2) + 4f(x_3) + f(x_4)\right]$$

- Hence on the entire region

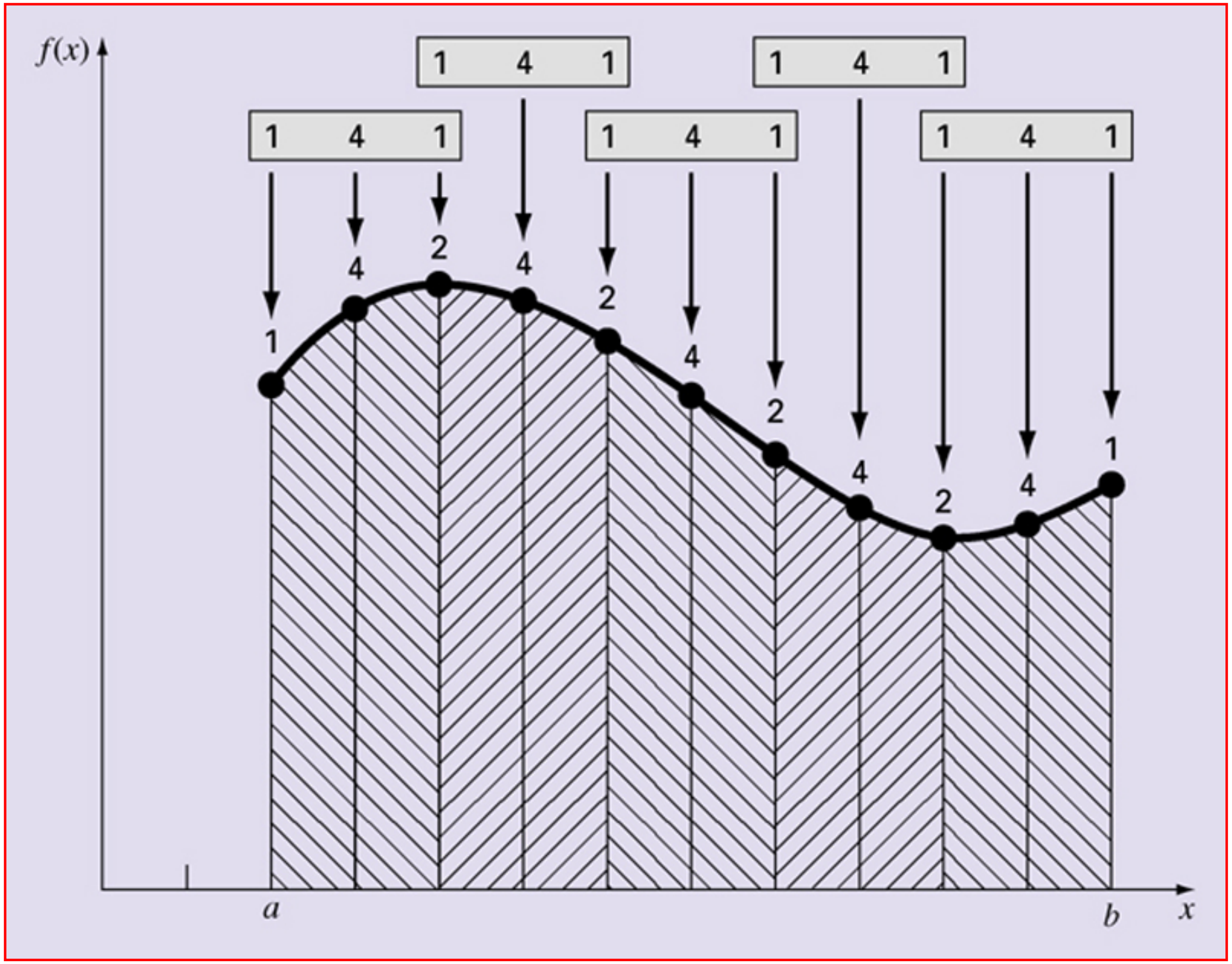
$$\int_{x_0}^{x_4} f(x)dx \cong \frac{h}{3}\left[f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + f(x_4)\right]$$

- In general for an even number of intervals n

$$\int_{a}^{b} f(x)dx \cong \frac{h}{3}\left[f(a) + f(b)\right] + \frac{4h}{3}\sum_{i=1}^{n/2} f(x_{2i-1}) + \frac{2h}{3}\sum_{i=1}^{n/2-1} f(x_{2i})$$

This is "Simpson's Composite Rule"

Composite Simpson's 1/3 Rule



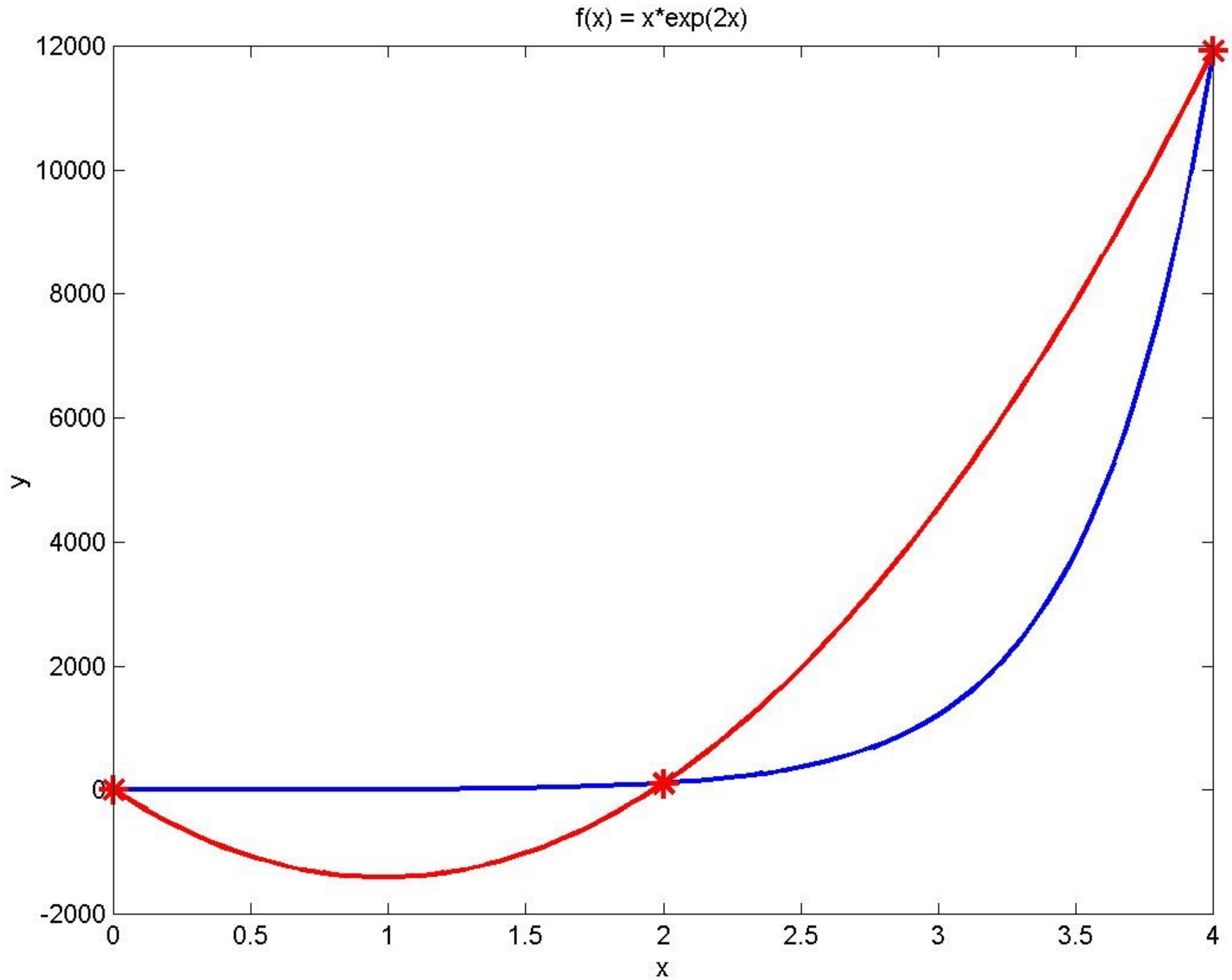
- Applicable only if the number of segments is even

Weights in Simpson's Rule

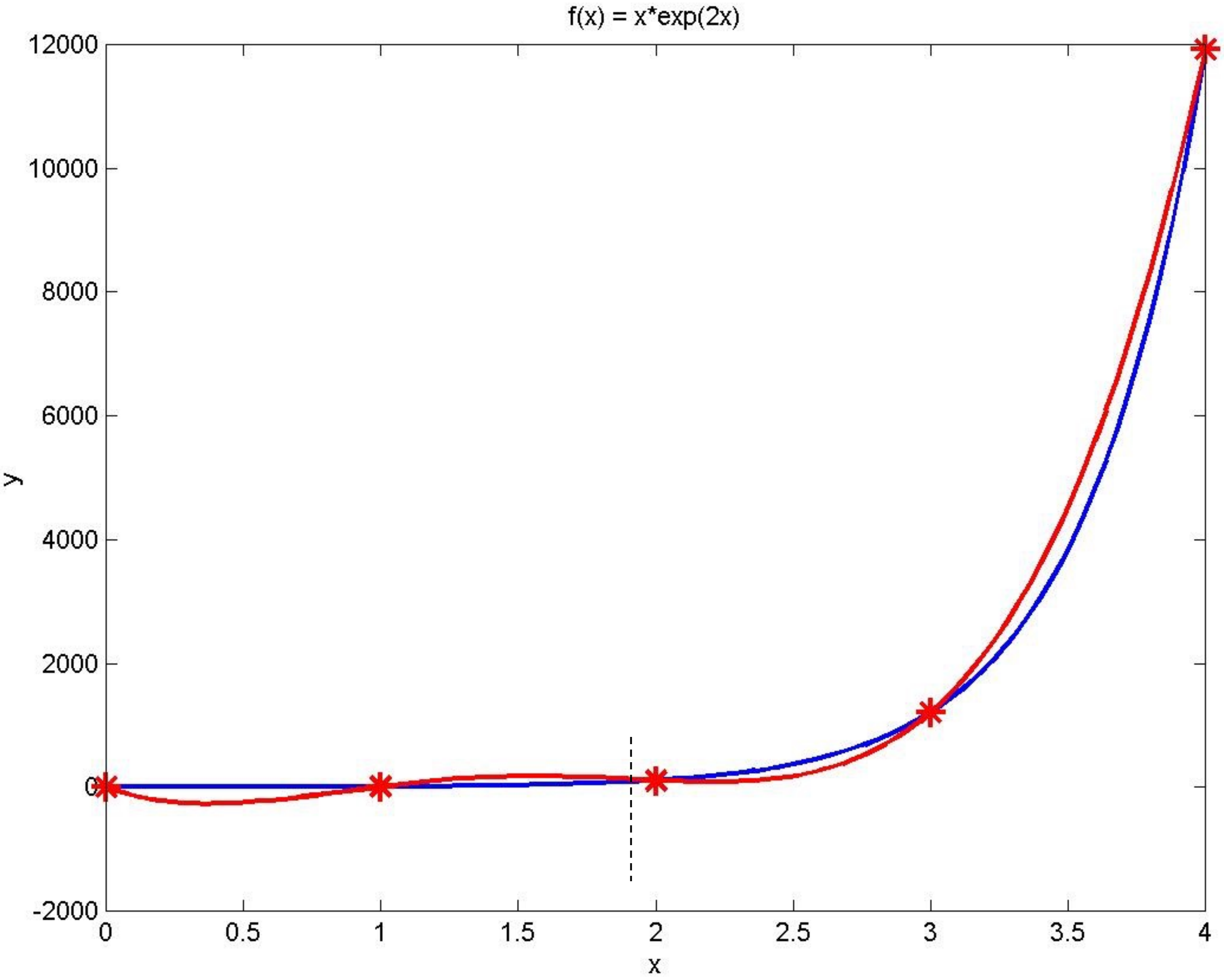
- For 1/3 Simpson' Rule the weights are:

$$\left\{w_i\right\}_{i=0}^{N} : w_i \left\{\begin{matrix} 1/3 & \text{end points} \\ 4/3 & \text{odd points} \\ 2/3 & \text{even points} \end{matrix}\right\}$$

Simpson’s 1/3 Rule



Composite Simpson’s 1/3 Rule



Higher order fits

- Can increase the order of the fit to cubic, quartic etc.
- For a cubic fit over x_0, x_1, x_2, x_3 we find

$$\int_{x_0}^{x_3} f(x)dx \cong \frac{3h}{8}\left[f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)\right]$$

Simpson's 3/8th Rule

- For a quartic fit over x_0, x_1, x_2, x_3, x_4

$$\int_{x_0}^{x_4} f(x)dx \cong \frac{2h}{45}\left[7f(x_0) + 32f(x_1) + 12f(x_2) + 32f(x_3) + 7f(x_4)\right]$$

Boole's Rule

- In practice these higher order formulas are not that useful, we can devise better methods if we first consider the errors involved

Error in the Trapezoid Rule

- Consider a Taylor expansions of f(x) about a

$$f(x) = f(a) + (x-a)f'(a) + \frac{(x-a)^2}{2!}f''(a) + \frac{(x-a)^3}{3!}f'''(a) + \ldots$$

- The integral of f(x) written in this form is then

$$\int_a^b f(x)dx = \left[xf(a) + \frac{(x-a)^2}{2!}f'(a) + \frac{(x-a)^3}{3!}f''(a) + \frac{(x-a)^4}{4!}f'''(a) \right.$$

$$\left. + \frac{(x-a)^5}{5!}f''''(a) + \ldots. \right]_a^b$$

$$= hf(a) + \frac{h^2}{2}f'(a) + \frac{h^3}{6}f''(a) + \frac{h^4}{24}f'''(a) + \ldots \qquad (1)$$ where h=b-a

Error in the Trapezoid Rule

- Perform the same expansion about b

$$\int_a^b f(x)dx = hf(b) - \frac{h^2}{2}f'(b) + \frac{h^3}{6}f''(b) - \frac{h^4}{24}f'''(b) + \ldots \qquad (2)$$

- If we take an average of (1) and (2) then

$$\int_a^b f(x)dx = \frac{h}{2}\left[f(a) + f(b)\right] + \frac{h^2}{4}\left[f'(a) - f'(b)\right] + \frac{h^3}{12}\left[f''(a) + f''(b)\right]$$
$$+ \frac{h^4}{48}\left[f'''(a) - f'''(b)\right] + \frac{h^5}{240}\left[f^{iv}(a) + f^{iv}(b)\right] + \ldots \quad (3)$$

- Notice that odd derivatives are differenced while even derivatives are added

Error in the Trapezoid Rule

- We also make Taylor expansions of f´ and f´´´ around both a & b, which allow us to substitute for terms in f´´ and f^{iv} and to derive

$$\int_a^b f(x)dx = \frac{h}{2}\left[f(a)+f(b)\right]+\frac{h^2}{12}\left[f'(a)-f'(b)\right]+\frac{h^4}{720}\left[f'''(a)-f'''(b)\right]+\ldots \quad (10)$$

- It takes *quite a bit of work* to get to this point, but the key issue is that we have now created correction terms which are all *differences*
- If we now use this formula in the composite trapezoid rule there will be a large number of cancellations

Error in the Composite Trapzoid

- We now sum over a series of trapezoids to get

$$\int_a^b f(x)dx = \frac{h}{2}\left[\left(f(a)+f(x_1)\right)+\left(f(x_1)+f(x_2)\right)+\ldots+\left(f(x_{n-2})+f(x_{n-1})\right)+\left(f(x_{n-1})+f(b)\right)\right]$$

$$+\frac{h^2}{12}\left[\left(f'(a)-f'(x_1)\right)+\left(f'(x_1)-f'(x_2)\right)+\ldots+\left(f'(x_{n-2})-f'(x_{n-1})\right)+\left(f'(x_{n-1})-f'(b)\right)\right]$$

$$+\frac{h^4}{720}\left[\left(f'''(a)-f'''(x_1)\right)+\left(f'''(x_1)-f'''(x_2)\right)+\ldots+\left(f'''(x_{n-2})-f'''(x_{n-1})\right)+\left(f'''(x_{n-1})-f'''(b)\right)\right]$$

$$+\ldots$$

$$=\frac{h}{2}\left[f(a)+f(b)\right]+h\sum_{i=1}^{n-1} f(a+ih)+\frac{h^2}{12}\left[f'(a)-f'(b)\right]+\frac{h^4}{720}\left[f'''(a)-f'''(b)\right]+\ldots \quad (11)$$

- Note now h=(b-a)/n
- The expansion is in powers of h^{2i}

Error in estimating the integral

Error in estimating the integral

Assumption : $f'(x)$ is continuous on *[a,b]*

Equal intervals (width $= h$)

Theorem : If Trapezoid Method is used to

approximate $\int_{\mathrm{a}}^{b} f(x)dx$ then

$$Error = \frac{b-a}{12}\ h^2\,(f'(b) - f'(a))$$

$$\left|Error\right| \le \frac{b-a}{12}\ h^2 \max_{x\in[a,b]} \left|f'(x)\right|$$

Estimating the Error

For Trapezoid method

- How many equally spaced intervals are needed to compute:

- To 5 decimal points accuracy?

Estimating the Error

For Trapezoid method

- How many equally spaced intervals are needed to compute:

$$\int_0^{\pi} \sin(x)dx$$

- To 5 decimal points accuracy?

Example

~750 equally-spaced points

Example

$$\boxed{\int_0^{\pi} \sin(x)dx, \quad \textit{find } h \textit{ so that } |\text{error}| \le \frac{1}{2}\times 10^{-5}}$$

$$|Error| \le \frac{b-a}{12} h^2 \max_{x\in[a,b]} |f'(x)|$$

$$b = \pi; \; a = 0; \quad f'(x) = \cos(x);$$

$$|f'(x)| \le 1 \quad \Rightarrow |Error| \le \frac{\pi}{12} h^2 \le \frac{1}{2}\times 10^{-5}$$

$$\Rightarrow \qquad h^2 \le \frac{6}{\pi}\times 10^{-5} = 1.91\times 10^{-5} \Rightarrow h \le 0.004$$

~750 equally-spaced points

Gaussian Quadrature

- Thus far we have considered regular spaced abscissas, although we have considered the possibility of adapting spacing
- We've also looked solely at closed interval formulas
- Gaussian quadrature achieves high accuracy and efficiency by optimally selecting the abscissas
- It is usual to apply a change of variables to make the integral map to [-1,1]
- There are also a number of different families of Gaussian quadrature, we'll look at Gauss-Legendre
- The following slides were adapted from:

http://numericalmethods.eng.usf.edu/topics/gauss_quadrature.html

See Numerical Recipes for a lengthy discussion

Theory of Gaussian Quadrature

Trapezoid Method

$$\int_a^b f(x)dx = h\left[\sum_{i=1}^{n-1} f(x_i) + \frac{1}{2}\big(f(x_0) + f(x_n)\big)\right]$$

It can be expressed as

$$\int_a^b f(x)dx = \sum_{i=0}^{n} c_i \; f(x_i)$$

$$where \quad c_i = \begin{cases} h & i = 1,2,\dots,n-1 \\ 0.5\,h & i = 0 \;\; and \;\; n \end{cases}$$

Basis of the Gaussian Quadrature Rule

Previously, the Trapezoidal Rule was developed by the method of undetermined coefficients. The result of that development is summarized below.

$$\int_a^b f(x)dx \approx c_1 f(a) + c_2 f(b)$$

$$= \frac{b-a}{2} f(a) + \frac{b-a}{2} f(b)$$

Basis of the Gaussian Quadrature Rule

The two-point Gauss Quadrature Rule is an extension of the Trapezoidal Rule approximation where the arguments of the function are not predetermined as a and b but as unknowns x_1 and x_2. In the two-point Gauss Quadrature Rule, the integral is approximated as

$$I = \int_a^b f(x)dx \approx c_1 f(x_1) + c_2 f(x_2)$$

Basis of the Gaussian Quadrature Rule

The four unknowns x_1, x_2, c_1 and c_2 are found by assuming that the formula gives exact results for integrating a general third order polynomial, $f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$.

Hence

$$\int_a^b f(x)dx = \int_a^b \left(a_0 + a_1 x + a_2 x^2 + a_3 x^3\right)dx$$

$$= \left[a_0 x + a_1 \frac{x^2}{2} + a_2 \frac{x^3}{3} + a_3 \frac{x^4}{4}\right]_a^b$$

$$= a_0(b-a) + a_1\left(\frac{b^2 - a^2}{2}\right) + a_2\left(\frac{b^3 - a^3}{3}\right) + a_3\left(\frac{b^4 - a^4}{4}\right)$$

Basis of the Gaussian Quadrature Rule

It follows that

$$\int_a^b f(x)dx = c_1\left(a_0 + a_1x_1 + a_2x_1^{\;2} + a_3x_1^{\;3}\right) + c_2\left(a_0 + a_1x_2 + a_2x_2^{\;2} + a_3x_2^{\;3}\right)$$

Equating Equations the two previous two expressions yield

$$a_0(b-a) + a_1\left(\frac{b^2-a^2}{2}\right) + a_2\left(\frac{b^3-a^3}{3}\right) + a_3\left(\frac{b^4-a^4}{4}\right)$$

$$= c_1\left(a_0 + a_1x_1 + a_2x_1^{\;2} + a_3x_1^{\;3}\right) + c_2\left(a_0 + a_1x_2 + a_2x_2^{\;2} + a_3x_2^{\;3}\right)$$

$$= a_0(c_1 + c_2) + a_1(c_1x_1 + c_2x_2) + a_2\left(c_1x_1^{\;2} + c_2x_2^{\;2}\right) + a_3\left(c_1x_1^{\;3} + c_2x_2^{\;3}\right)$$

Basis of the Gaussian Quadrature Rule

Since the constants a_0, a_1, a_2, a_3 are arbitrary

$$b - a = c_1 + c_2$$

$$\frac{b^2 - a^2}{2} = c_1 x_1 + c_2 x_2$$

$$\frac{b^3 - a^3}{3} = c_1 x_1^{\ 2} + c_2 x_2^{\ 2}$$

$$\frac{b^4 - a^4}{4} = c_1 x_1^{\ 3} + c_2 x_2^{\ 3}$$

Basis of Gauss Quadrature

The previous four simultaneous nonlinear Equations have only one acceptable solution,

$$x_1 = \left(\frac{b-a}{2}\right)\left(-\frac{1}{\sqrt{3}}\right) + \frac{b+a}{2} \qquad x_2 = \left(\frac{b-a}{2}\right)\left(\frac{1}{\sqrt{3}}\right) + \frac{b+a}{2}$$

$$c_1 = \frac{b-a}{2} \qquad c_2 = \frac{b-a}{2}$$

Basis of Gauss Quadrature

Hence Two-Point Gaussian Quadrature Rule

$$\int_a^b f(x)dx \approx c_1 f(x_1) + c_2 f(x_2)$$

$$= \frac{b-a}{2} f\left(\frac{b-a}{2}\left(-\frac{1}{\sqrt{3}}\right) + \frac{b+a}{2}\right) + \frac{b-a}{2} f\left(\frac{b-a}{2}\left(\frac{1}{\sqrt{3}}\right) + \frac{b+a}{2}\right)$$

Higher Point Gaussian Quadrature Formulas

$$\int_a^b f(x)dx \approx c_1 f(x_1) + c_2 f(x_2) + c_3 f(x_3)$$

is called the three-point Gauss Quadrature Rule.

The coefficients c_1, c_2, and c_3, and the functional arguments x_1, x_2, and x_3 are calculated by assuming the formula gives exact expressions for integrating a fifth order polynomial

$$\int_a^b \left(a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + a_5 x^5\right)dx$$

General n-point rules would approximate the integral

$$\int_a^b f(x)dx \approx c_1 f(x_1) + c_2 f(x_2) + \ldots\ldots\ldots + c_n f(x_n)$$

Arguments and Weighing Factors for n-point Gauss Quadrature Formulas

In handbooks, coefficients and arguments given for n-point Gauss Quadrature Rule are given for integrals

$$\int_{-1}^{1} g(x)dx \cong \sum_{i=1}^{n} c_i g(x_i)$$

as shown in Table 1.

Table 1: Weighting factors c and function arguments x used in Gauss Quadrature Formulas.

Points	**Weighting Factors**	**Function Arguments**
2	c_1 = 1.000000000 c_2 = 1.000000000	x_1 = -0.577350269 x_2 = 0.577350269
3	c_1 = 0.555555556 c_2 = 0.888888889 c_3 = 0.555555556	x_1 = -0.774596669 x_2 = 0.000000000 x_3 = 0.774596669
4	c_1 = 0.347854845 c_2 = 0.652145155 c_3 = 0.652145155 c_4 = 0.347854845	x_1 = -0.861136312 x_2 = -0.339981044 x_3 = 0.339981044 x_4 = 0.861136312

Arguments and Weighing Factors for n-point Gauss Quadrature Formulas

So if the table is given for $\int_{-1}^{1} g(x)dx$ integrals, how does one solve $\int_{a}^{b} f(x)dx$? The answer lies in that any integral with limits of $\left[a, b\right]$ can be converted into an integral with limits $\left[-1, 1\right]$ Let

$$x = mt + c$$

If $x = a,$ then $t = -1$

If $x = b,$ then $t = 1$

Such that:

$$m = \frac{b-a}{2}$$

Arguments and Weighing Factors for n-point Gauss Quadrature Formulas

Then $$c = \frac{b+a}{2}$$ Hence

$$x = \frac{b-a}{2}t + \frac{b+a}{2} \qquad dx = \frac{b-a}{2}dt$$

Substituting our values of x, and dx into the integral gives us

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

See examples in Python workbook Lecture07b.ipynb

Basic Monte Carlo Algorithm

Basic Monte Carlo Algorithm

- Suppose we want to approximate

$$Z = \int_{\Omega} f\left(\mathbf{x}\right) d\mathbf{x}$$

Basic Monte Carlo Algorithm

- Suppose we want to approximate

$$Z = \int_{\Omega} f\left(\mathbf{x}\right) d\mathbf{x}$$

in a high-dimensional space

- For i = 1 to n
  - Pick a point x_i at random
  - Accept or reject the point based on criterion
  - If accepted, then add $f(x_i)$ to total sum
- Error estimates are “free” by calculating sums of squares

Basic Monte Carlo Algorithm

- Suppose we want to approximate

$$Z = \int_{\Omega} f\left(\mathbf{x}\right) d\mathbf{x}$$

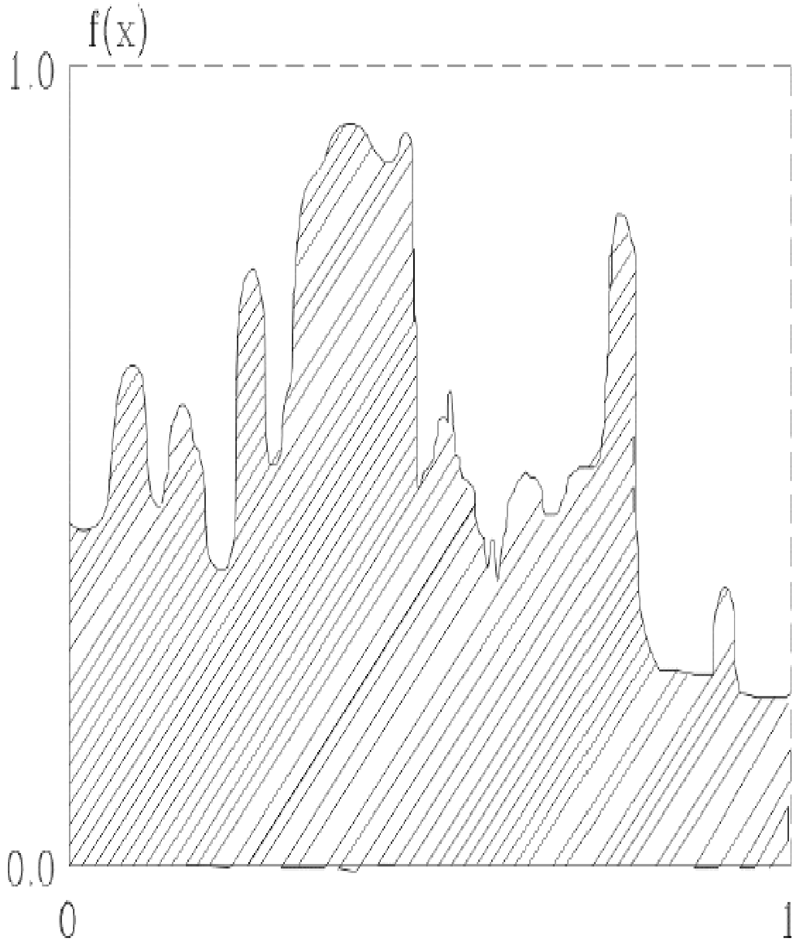
in a high-dimensional space

- For i = 1 to n
  - Pick a point x_i at random
  - Accept or reject the point based on criterion
  - If accepted, then add $f(x_i)$ to total sum
- Error estimates are “free” by calculating sums of squares
- Error typically decays as

$$\frac{1}{\sqrt{N}}$$

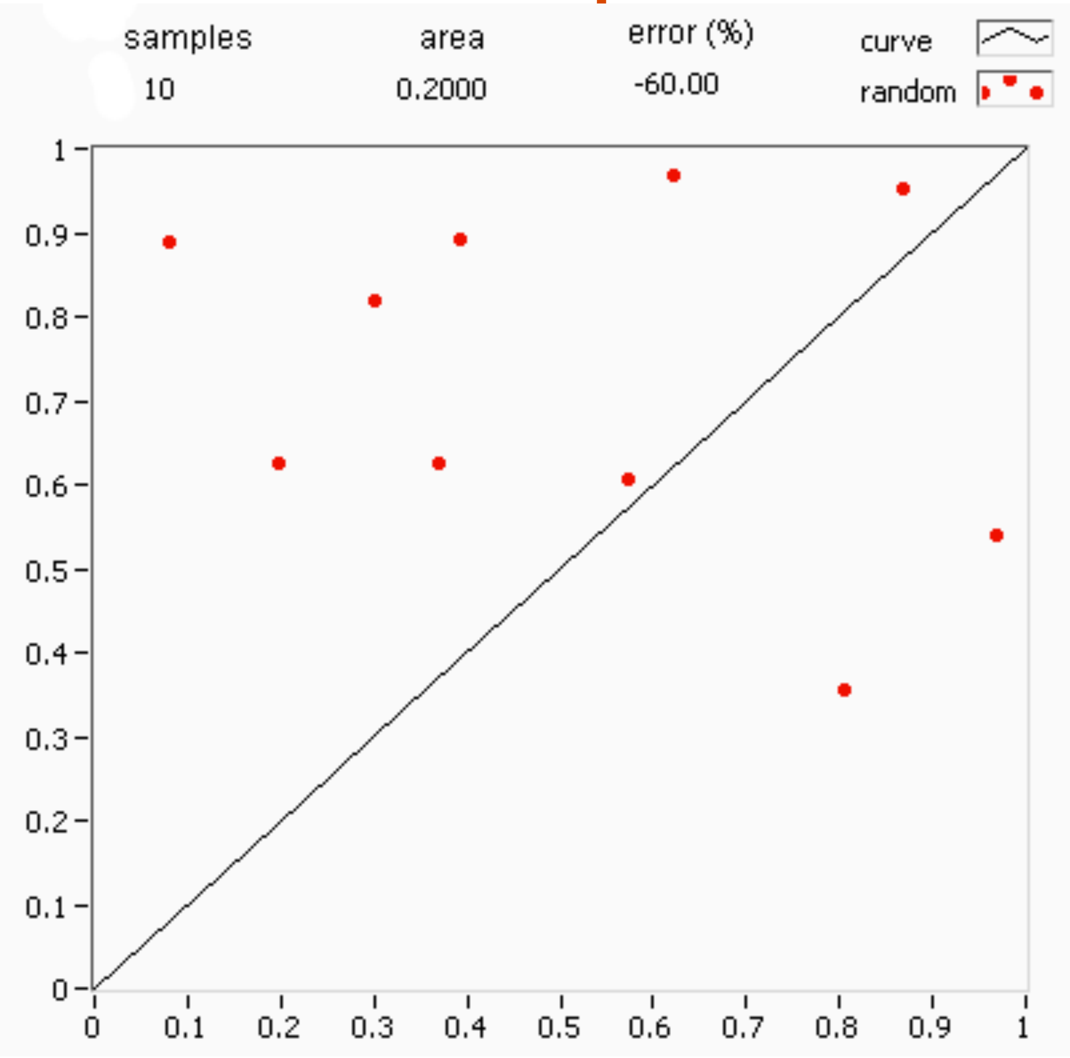
1-dimensional MC Integration

- Suppose we want to find the area under the curve for Y=f(X) for a given interval.
- Now let's assume we are throwing darts at the picture of f(x). Each dart will land on a point (x_i,y_i), where $0<x_i<1$ and $0<y_i<1$.
- If the value of $y_i<f(x_i)$, then we count it as a hit. Repeat the process for a large number of i.
- The area under the curve=number of hits/ total number of throws.



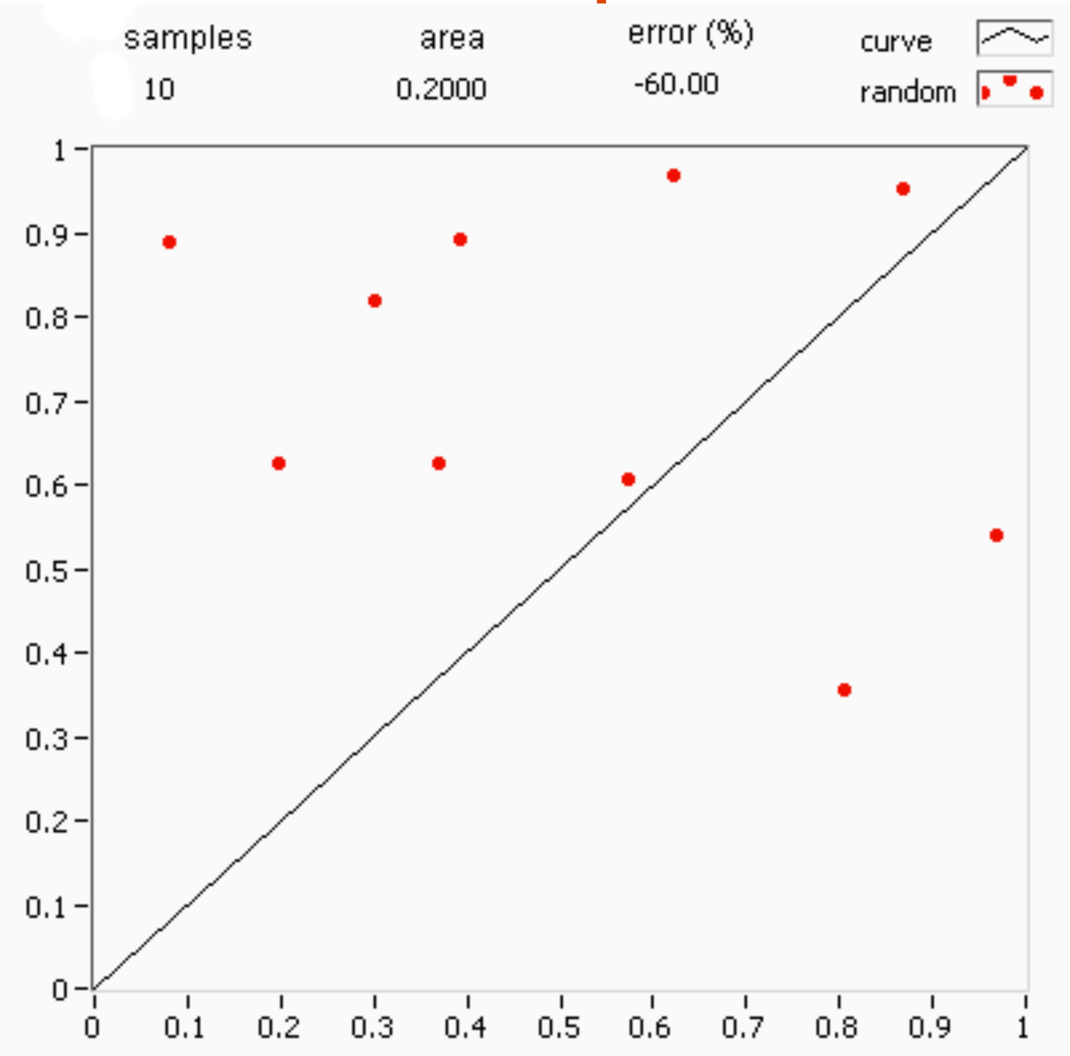
Amin Jazaeri

Example



https://en.wikipedia.org/wiki/Monte_Carlo_method

Example



https://en.wikipedia.org/wiki/Monte_Carlo_method

Error Estimate for MC Integrals

- Uncertainties for MC integrals follow the *binomial* distribution:
  - Let the area under the curve be S, and the area populated by random points ("darts") is S_0. Let N_0 be the number of points thrown, and N the number of points accepted. Then the estimator of S is:

$$\hat{S} = S_0 \frac{N}{N_0}$$

  - Define the "efficiency" of accepting the events as

$$\epsilon \equiv \frac{S}{S_0} \approx \hat{\epsilon} = \frac{N}{N_0}$$

Error Estimate for MC Integrals

- *N* is a random variable, which follows a binomial distribution ("choose *N* out of N_0"). Estimator of $\varepsilon = N/N_0$ is therefore also a random variable. Its uncertainty is given by

$$\sigma(\hat{\epsilon}) = \sqrt{\frac{\epsilon(1-\epsilon)}{N_0}}$$

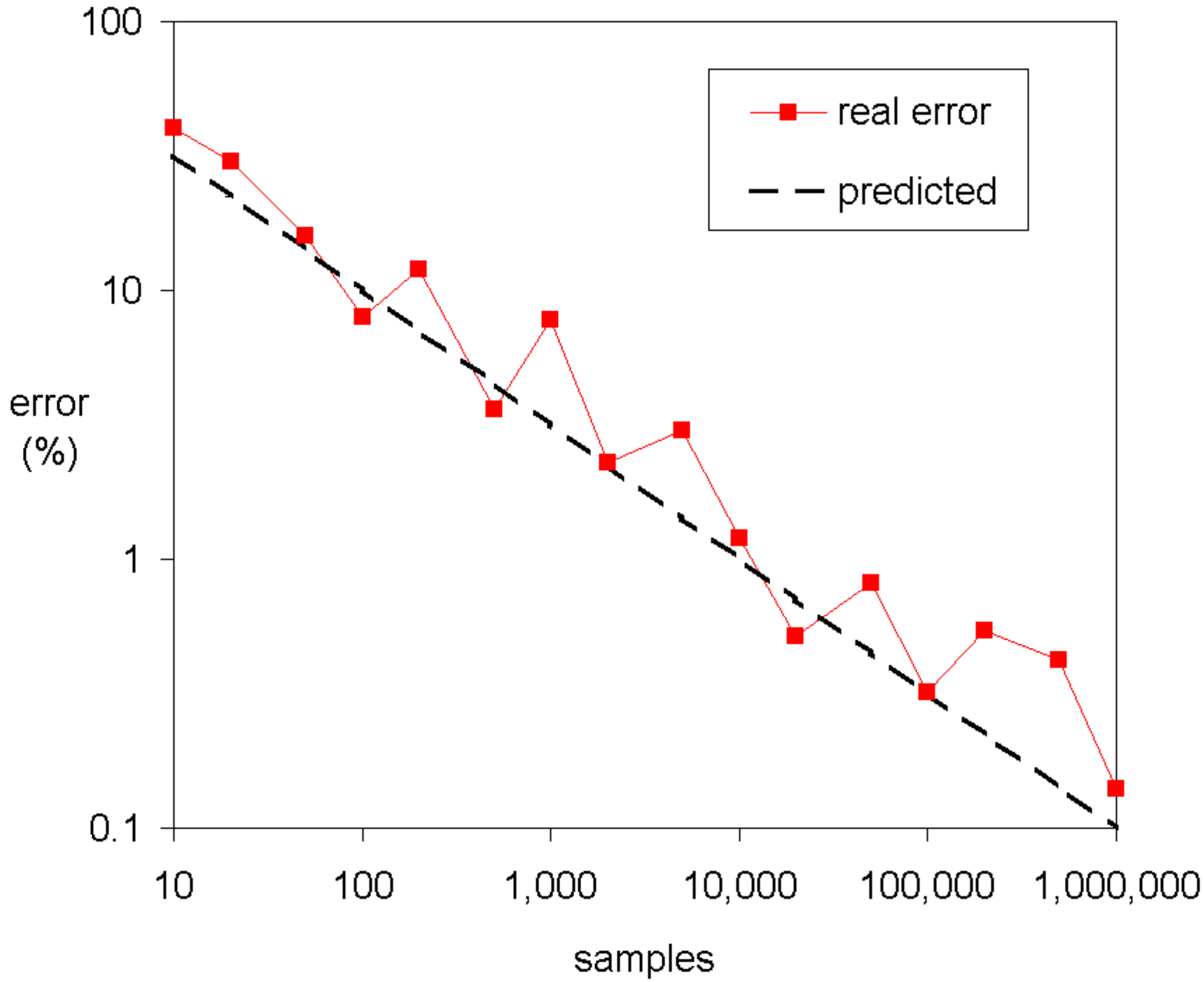
- You can show that in the limit $N \ll N_0$ ($\varepsilon \to 0$),

$$\sigma(\hat{\epsilon}) \approx \sqrt{\frac{\epsilon}{N_0}} \approx \frac{\epsilon}{\sqrt{N}}$$

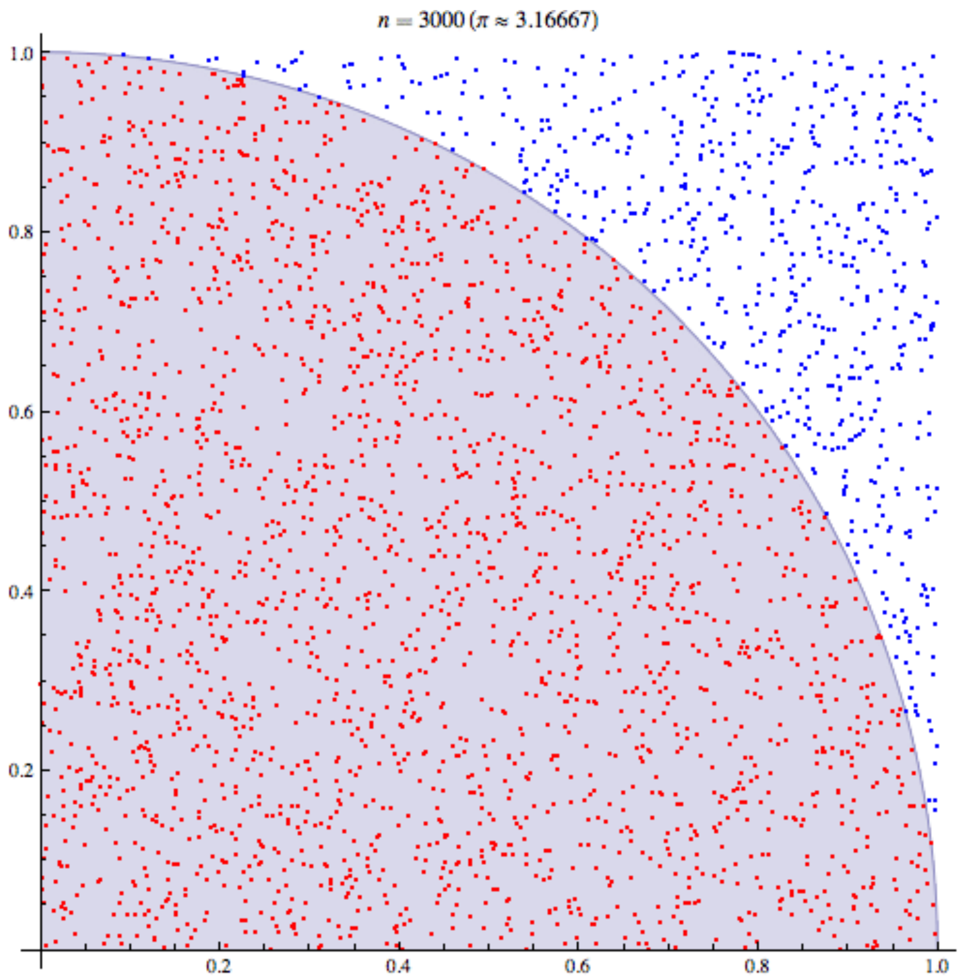
- and

$$\sigma(\hat{S}) = S_0 \sigma(\hat{\epsilon}) = \frac{S}{\sqrt{N}}$$

Error Estimate



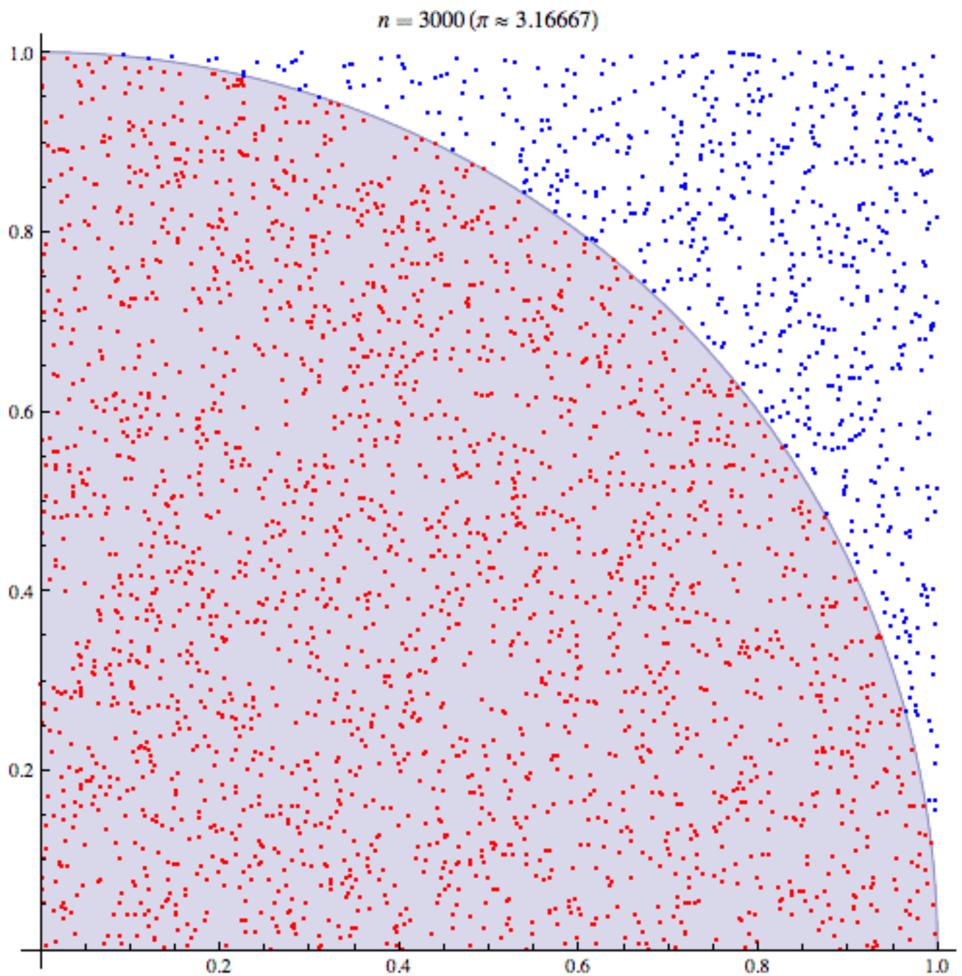
Example: Compute π by MC



Workshop 6 !

https://en.wikipedia.org/wiki/Monte_Carlo_method

Example: Compute π by MC



Workshop 6 !

https://en.wikipedia.org/wiki/Monte_Carlo_method