Numerical Integration

by Loïc Quertenmont, PhD

LINFO1113 - 2019-2020

Programme

Cours 1	Librairies mathématiques, représentation des nombres en Python et erreurs liées
Cours 2,3	Résolution des systèmes linéaires
Cours 4,5	Interpolation et Régression Linéaires
Cours 6,7	Racines d'équations
Cours 8	Différentiation numérique
Cours 9	**Intégration numérique**
Cours 10	Problème a valeur initiale (ODE)
Cours 11	Introduction à l'optimisation
Cours 12,13	Rappel / Répétition

Outline

- **Introduction**
- **Newton-Cotes Formulas**
- **Romberg Integration**
- **Gaussian Integration**
- ~~**Multiple Integral**~~

Introduction

Introduction

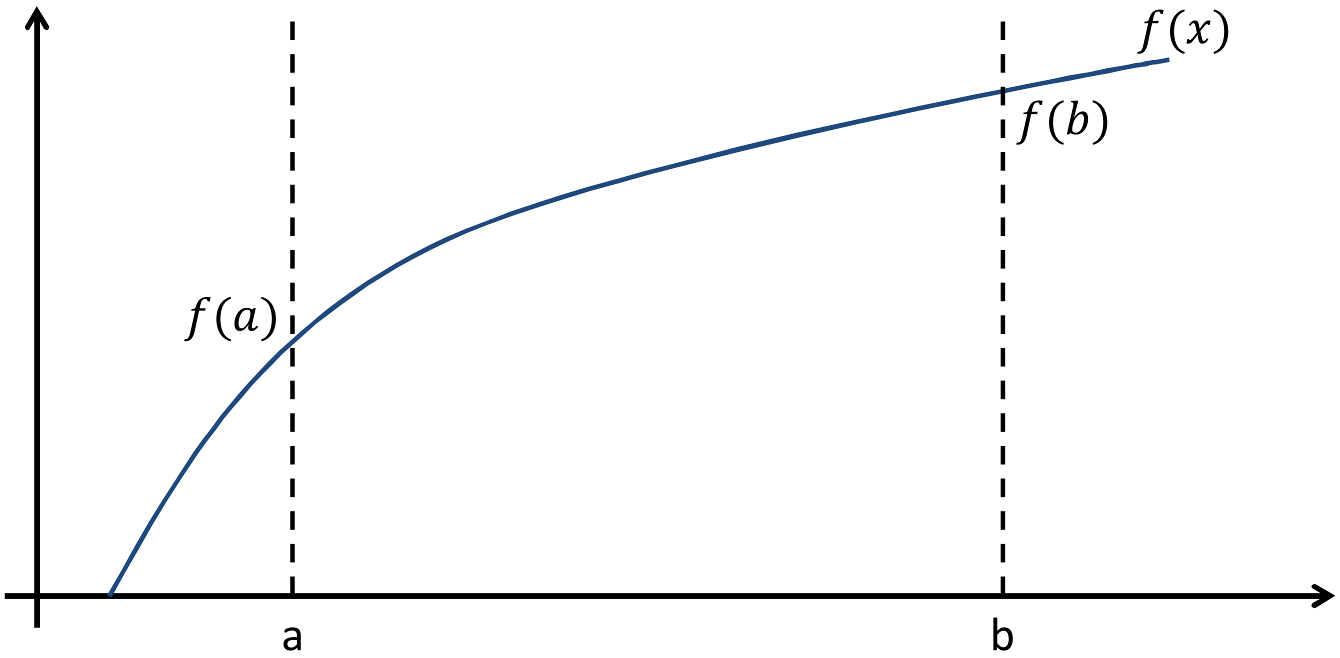
- Goal of numerical integration:
  - Also named "quadrature"
  - Given the function $f(x)$, compute :

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

- Find I such that

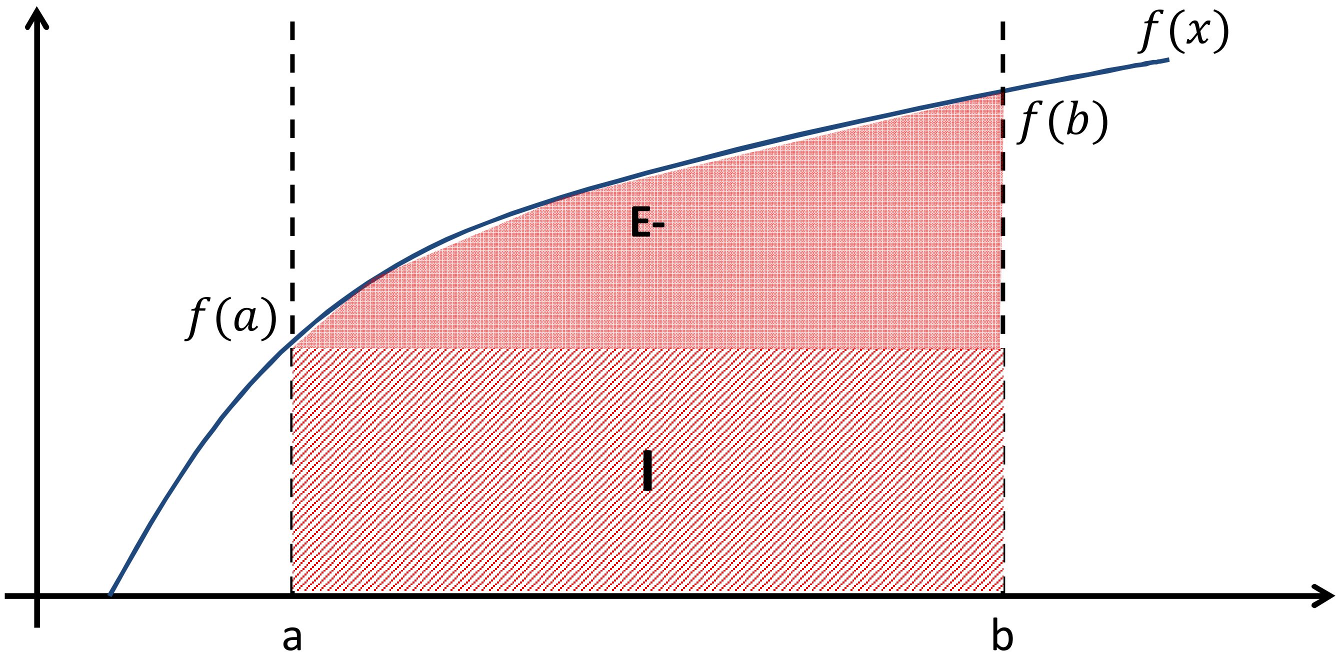
$$\int_a^b f(x)dx \approx I$$

Naïve approach



By what could you approximate the Integral of $f(x)$ over the domain $[a, b]$?
Let's start by the simplest things…

Naïve approach



By what could you approximate the Integral of $f(x)$ over the domain $[a, b]$?
Let's start by the simplest things…

Rectangle method

A Rectangle ?

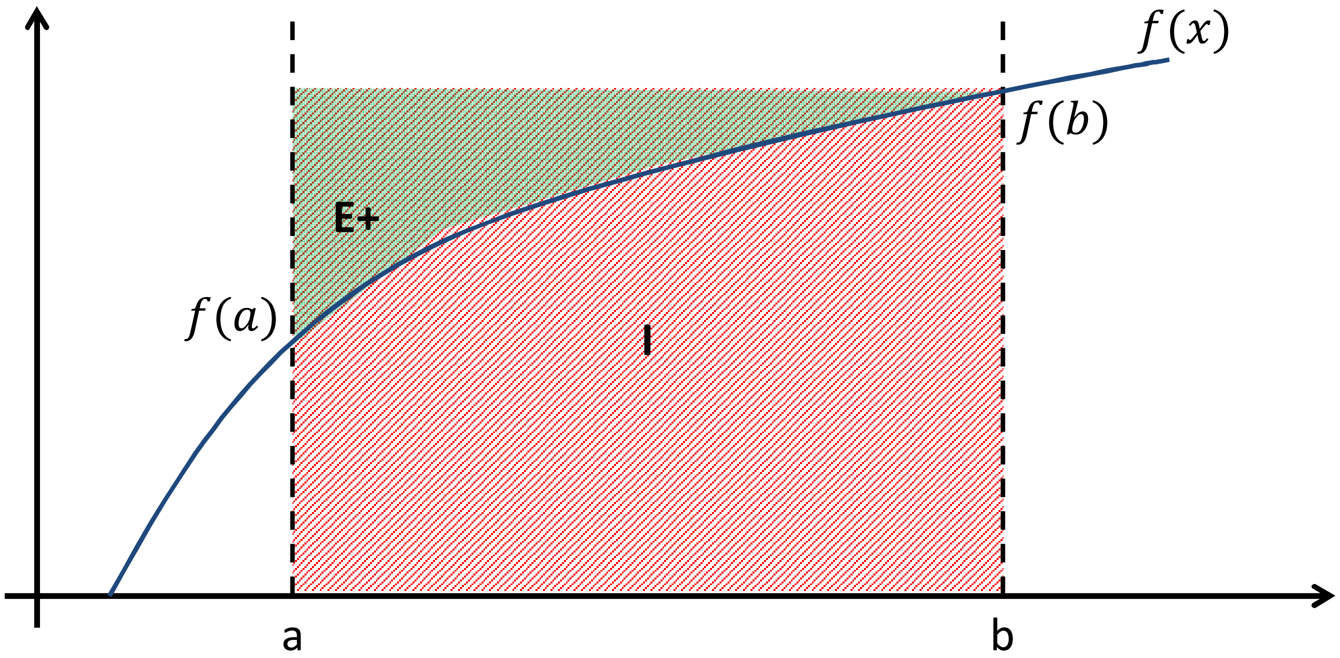
$$I = (b-a)f(a)$$

Method of order 0

$$E = E_{-} = \frac{(b-a)^2}{2} f'(\xi)$$

$E = 0$ if $f(x) = C$

Naïve approach



By what could you approximate the Integral of $f(x)$ over the domain $[a, b]$?
Let's start by the simplest things…

Rectangle method

A Rectangle ?

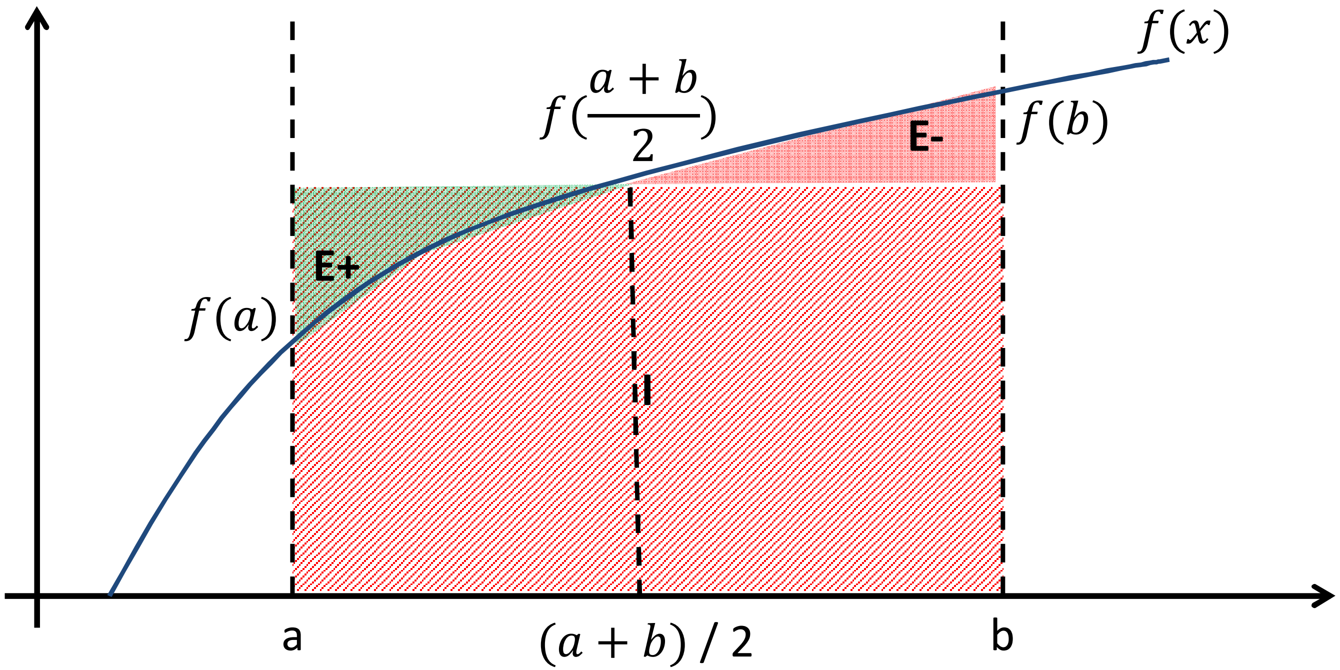
$$I = (b - a)f(b)$$

Method of order 0

$$E = E_+ = \frac{(b-a)^2}{2} f'(\xi)$$

$\boldsymbol{E = 0}$ **if** $\boldsymbol{f(x) = C}$

Naïve approach



By what could you approximate the Integral of $f(x)$ over the domain $[a, b]$?
Let's start by the simplest things…

Midpoint method

A Rectangle ?

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

Method of order 1

$$E = E_{-} + E_{+} = \frac{(b-a)^3}{24}\, f''(\xi)$$

$\boldsymbol{E = 0}$ **if** $\boldsymbol{f(x) = Ax + b}$

Introduction

- Integral is approximated by the sum

$$I = \sum_{i=0}^{n} A_i f(x_i)$$

  - Nodes / Abscissas (x_i)
  - weights (A_i)
  - Choice of x_i and A_i depends on the rules used

- All rules are derived from polynomial interpolations
  - The integration work better if $f(x)$ can be approximated by a polynomial

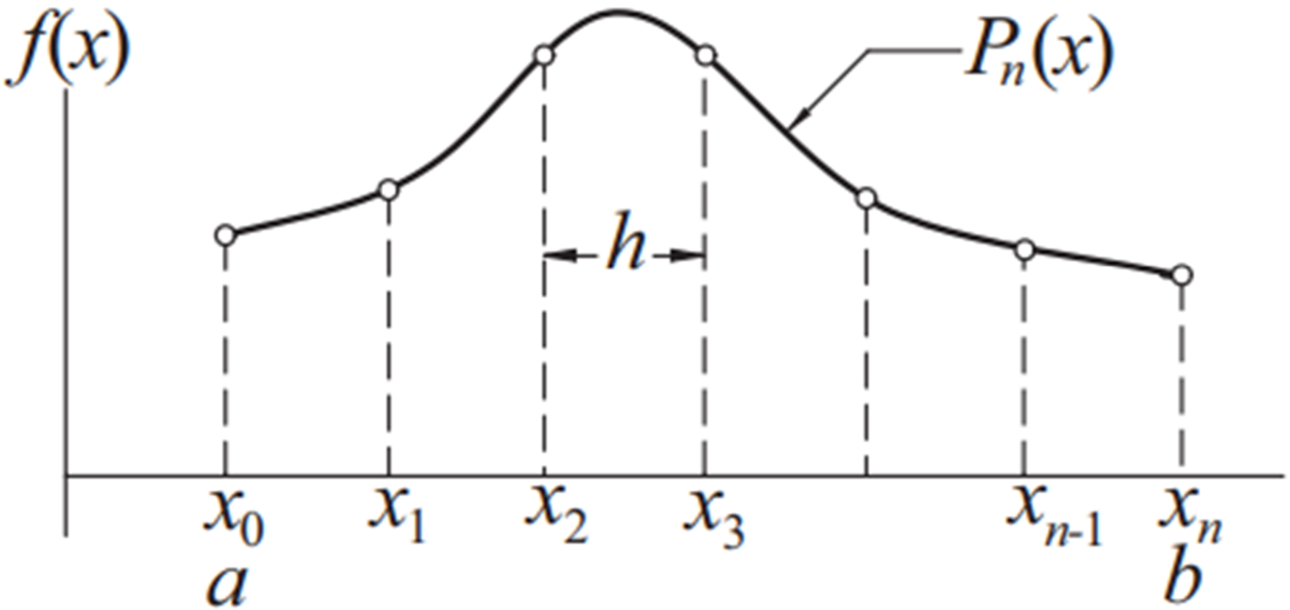
Introduction

- Two group of rules:
  - **Newton-Cotes**
    - Equally spaced nodes
    - Good if $f(x_i)$ can be evaluated at low cost
  - **Gaussian Quadrature**
    - Require less evaluations of $f(x_i)$
    - Can handle integrable singularities

Newton-Cotes Formulas

General Newton-Cotes Formula (1)

- Consider $\mathrm{I} = \int_a^b f(x)dx$
- We divide the range of integration in n intervals of equal length $h = (b-a)/n$
- Nodes $x_0, x_1, \ldots, x_n$ are given by: $x_i = a + ih$
- We approximate $f(x)$ by a polynomial of degree n : $\mathrm{P_n(x)}$



General Newton-Cotes Formula (2)

- Consider $\mathrm{I} = \int_a^b f(x)dx$
- We divide the range of integration in n intervals of equal length $h = (b-a)/n$
- Nodes $x_0, x_1, \ldots, x_n$ are given by: $x_i = a + ih$
- We approximate $f(x)$ by a polynomial of degree n : $\mathrm{P_n(x)}$
  - Lagrange form of polynomials

$$f(x) \approx \mathrm{P_n(x)} = \sum_{i=0}^{n} f(x_i) l_i(x) \qquad \text{with } l_i(x) = \prod_{j \neq i} \frac{(\mathrm{x} - \mathrm{x_j})}{(\mathrm{x_i} - \mathrm{x_j})}$$

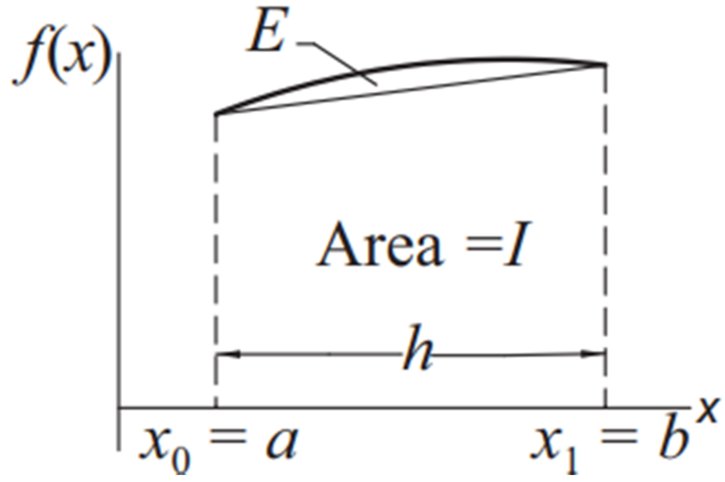
- The integral is therefore approximated by:

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

$$I \approx \sum_{i=0}^{n} A_i f(x_i) \qquad \text{with } A_i = \int_a^b l_i(x)dx$$

Trapezoidal rule (1)

- **Newton-Cotes Formula for n=1**
- Only one panel



- $l_0 = \frac{x-x_1}{x_0-x_1} = \frac{(x-b)}{-h}$
- $A_0 = \frac{1}{-h}\int_a^b (x-b)dx = \frac{1}{2h}(b-a)^2 = \frac{h}{2}$

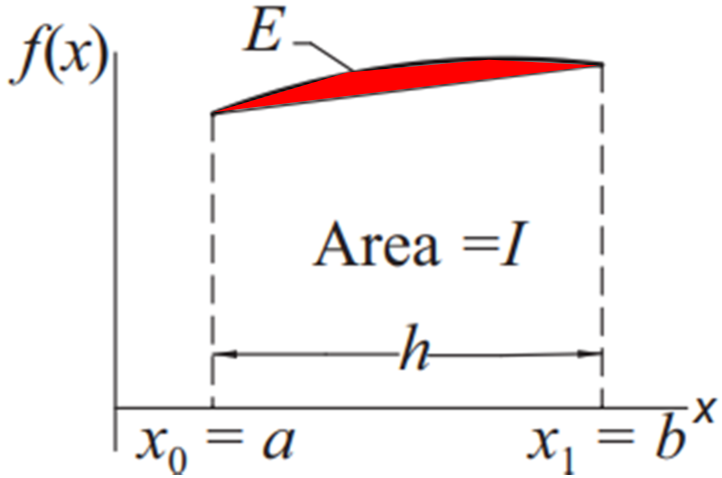
- $l_1 = \frac{x-x_0}{x_1-x_0} = \frac{(x-a)}{h}$
- $A_1 = \frac{1}{h}\int_a^b (x-a)dx = \frac{1}{2h}(b-a)^2 = \frac{h}{2}$
- $I = \sum_{i=0}^{n} A_i f(x_i) = \frac{h}{2}[f(a) + f(b)]$

This is the area of the trapezoid

Trapezoidal rule (2)

- The error in the trapezoidal rule:

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



- This is the area between $f(x)$ and the straight line interpolant

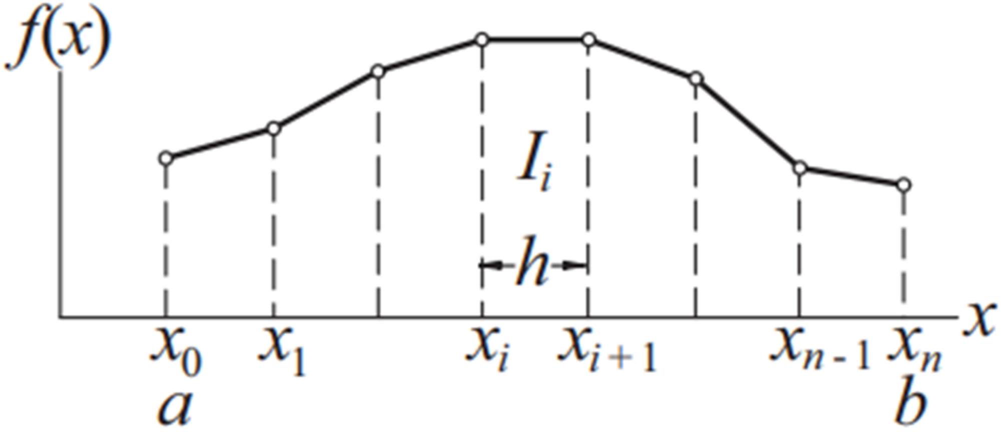
- We can compute the error by integrating the interpolation error

$$E = \frac{1}{2!}\int_a^b (x - x_0)(x - x_1) f''(\xi)dx = \frac{1}{2} f''(\xi) \int_a^b (x-a)(x-b)dx$$

$$= -\frac{1}{12}(b-a)^3 f''(\xi) = -\frac{h^3}{12} f''(\xi)$$

Composite Trapezoidal rule (1)

- In practice, the trapezoidal rule is applied piecewise



- The region (a,b) is divided in n panels each of width h
- The function $f(x)$ is approximated by a straight line in each panel
- The Integral (I_i) in each panel is given by: $I_i = \frac{h}{2}[f(x_i) + f(x_{i+1})]$

$$I = \sum_{i=0}^{n-1} I_i = \frac{h}{2}[f(x_0) + 2f(x_1) + \cdots + 2f(x_{n-1}) + f(x_n)]$$

Composite Trapezoidal rule (2)

- The truncation error on the integral of a panel is

$$E_i = \frac{-h^3}{12} f''(\xi_i)$$

- Thus, the total truncation error is

$$E = \sum_{i=0}^{n-1} E_i = \frac{-h^3}{12} \sum_{i=0}^{n-1} f''(\xi_i)$$

- But $\sum_{i=0}^{n-1} f''(\xi_i) = n\,\overline{f''}$ with $\overline{f''}$ the arithmetic mean of f''
- If f'' is continuous, the error can be rewritten:

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

  – Actually, the error is $E = c_1 h^2 + c_2 h^4 + c_3 h^6 + \ldots$
  because f'' has also some dependency on h
- Error is of the order of h^2 → so the more panels, the better

Recursive Trapezoidal rule (1)

- Let's define I_k the integral evaluated with the composite trapezoidal rule using 2^{k-1} panels
  - If k increase by one, the number of panel is doubled
  - Let's define $H = b - a$

- For $k = 1$ (one panel):
$$I_1 = [f(a) + f(b)]\frac{H}{2}$$

- For $k = 2$ (two panels):
$$I_2 = \left[f(a) + 2f\left(a + \frac{H}{2}\right) + f(b)\right]\frac{H}{4} = \frac{1}{2}I_1 + f\left(a + \frac{H}{2}\right)\frac{H}{2}$$

- For $k = 3$ (4 panels):
$$I_3 = \left[f(a) + 2f\left(a + \frac{H}{4}\right) + 2f\left(a + \frac{H}{2}\right) + 2f\left(a + \frac{3H}{4}\right) + f(b)\right]\frac{H}{8}$$
$$I_3 = \frac{1}{2}I_2 + \left[f\left(a + \frac{H}{4}\right) + f\left(a + \frac{3H}{4}\right)\right]\frac{H}{4}$$

Recursive Trapezoidal rule (2)

- For k (2^{k-1} panels):

$$I_k = \frac{1}{2} I_{k-1} + \frac{H}{2^{k-1}} \sum_{i=1}^{2^{k-2}} f\left(a + \frac{(2i-1)H}{2^{k-1}}\right)$$

- This is the recursive trapezoidal rule.
  - I_k is computed from I_{k-1} and new points only
  - There is no additional cost (CPU) w.r.t composite rule, BUT…
  - It allows to monitor convergence of the integral by comparing I_k and I_{k-1} and stop earlier.

- A form that is easier to remember is:

$$I(h) = \frac{1}{2} I(2h) + h \sum f(x_{new})$$

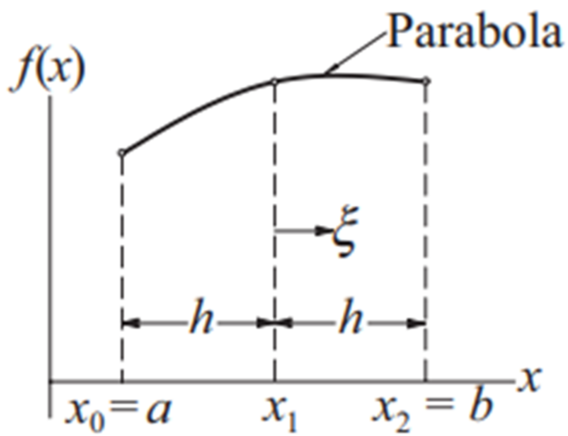
  - with $h = H/n$ the width of the panel

Algorithm

```
## module trapezoid
''' Inew = trapezoid(f,a,b,Iold,k).
   Recursive trapezoidal rule:
   old = Integral of f(x) from x = a to b computed by
   trapezoidal rule with 2^(k-1) panels.
   Inew = Same integral computed with 2^k panels.
'''
def trapezoid(f,a,b,Iold,k):
    if k == 1:Inew = (f(a) + f(b))*(b - a)/2.0
    else:
        n = 2**(k -2 )      # Number of new points
        h = (b - a)/n       # Spacing of new points
        x = a + h/2.0
        sum = 0.0
        for i in range(n):
            sum = sum + f(x)
            x = x + h
        Inew = (Iold + h*sum)/2.0
    return Inew
```

Simpson's rules

- **Simpson 1/3 rule**
- **Newton-Cotes Formula for n=2**
- Two panels
- Fit, using a parabola

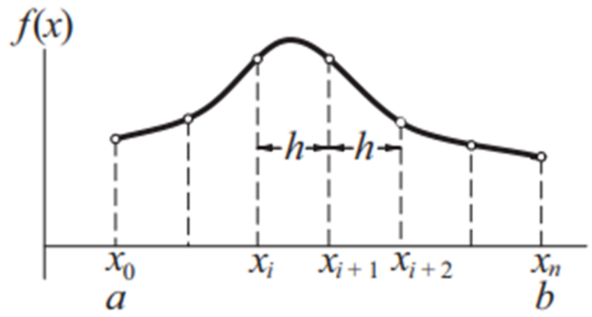


- $l_0 = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} = \frac{(x-\frac{a+b}{2})(x-b)}{2h^2}$ → $A_0 = \int_a^b l_0(x)dx = \frac{h}{3}$
- $l_1 = \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} = \frac{(x-a)(x-b)}{-h^2}$ → $A_1 = \int_a^b l_1(x)dx = \frac{4h}{3}$
- $l_2 = \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} = \frac{(x-a)(x-\frac{a+b}{2})}{2h^2}$ → $A_2 = \int_a^b l_2(x)dx = \frac{h}{3}$

This is the Simpson 1/3 rule

$$I = \sum_{i=0}^{n} A_i f(x_i) = \frac{h}{3}\left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right]$$

Composite Simpson's rules (1)



- **Composite rule**
- We divide into n panels (with n even) of width h = (b − a)/n each

- We then apply Simpson rule to two adjacent panels:

$$\int_{x_i}^{x_{i+2}} f(x)dx \approx \frac{h}{3}[f(x_i) + 4f(x_{i+1}) + f(x_{i+2})]$$

- Therefore:

$$\int_a^b f(x)dx = \int_{x_0}^{x_m} f(x)dx = \sum_{i=0,2,..}^{n} \left(\int_{x_i}^{x_{i+2}} f(x)dx\right)$$

$$I \approx \frac{h}{3}[f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + \cdots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)]$$

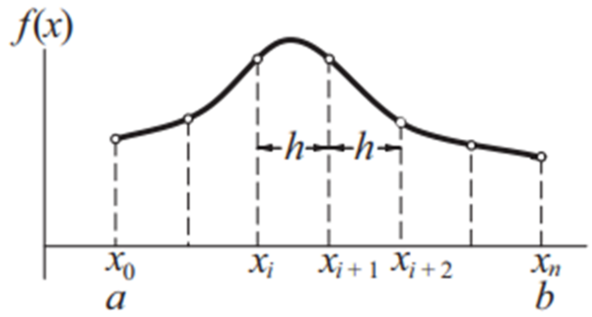
- The error in the composite Simpson Rule is:

$$E = \frac{(b-a)h^4}{180} f^{(4)}(\xi)$$

- The error is proportional to h^4, that's much better than trapezoidal rule

Composite Simpson's rules (2)

- **Simpson 1/3 rule requires the number of panel (n) to be even**



- **If this is not the case:**

- We can integrate over the first (or last) three panels using Simpson's 3/8 rules (derived from Newton-Cotes with m=3)

$$I = \sum_{i=0}^{n} A_i f(x_i) = \frac{3h}{8}[f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)]$$

- And then, integrate the remaining panels using the composite Simpson's 1/3 rule

Example

- **Estimate $\int_{0}^{2.5} f(x)dx$ from the following data:**

x	0	0.5	1.0	1.5	2.0	2.5
$f(x)$	1.5000	2.0000	2.0000	1.6364	1.2500	0.9565

- **Solution:**
  - We use Simpson Rule because it is more accurate
  - h = 0.5
  - They are 6 points ➔ 5 panels (odd)

  - We use Simpson's 3/8 on the first 3 and Simpson's 1/3 on the last 2

  - **Simpson's 3/8:**
  - $I_a = \frac{3(0.5)}{8}[f(0) + 3f(0.5) + 3f(1.0) + f(1.5)] = \frac{3(0.5)}{8}[1.5 + 3 \times 2 + 3 \times 2 + 1.6364] = 2.8381$

  - **Simpson's 1/8:**
  - $I_b = \frac{1(0.5)}{8}[f(1.5) + 4f(0.2) + f(2.5)] = \frac{1(0.5)}{8}[1.6364 + 4 \times 1.25 + 0.9565] = 1.2655$

  - **Total Integral** → $I = I_a + I_b = 2.8381 + 1.2655 = 4.1036$

Romberg Integration

Principle

- **Romberg Integration**

- Combine the **trapezoidal Rule** with **Richardson Extrapolation**

- **Trapezoidal rule reminder:**
  - 1 panels: $I_1 = [f(a) + f(b)]\frac{H}{2}$
  - 2 panels: $I_2 = \frac{1}{2}I_1 + f\left(a + \frac{H}{2}\right)\frac{H}{2}$
  - 2^{k-1} panels: $I_k = \frac{1}{2}I_{k-1} + \frac{H}{2^{k-1}}\sum_{i=1}^{2^{k-2}} f\left(a + \frac{(2i-1)H}{2^{k-1}}\right)$
  - Error = $E = \frac{-(b-a)h^2}{12}f''(\xi) = c_1h^2 + c_2h^4 + c_3h^6 + \ldots$ with $h = \frac{(b-a)}{2^{k-1}}$

- **Richardson extrapolation reminder:**
  - $G = g(h) + E(h)$ with $E(h) = c{h_1}^p$
  - We can get a more accurate estimate of G (reduce the error), from two approximated measurements with h_1 and $h_1/2$
  - $G = \frac{2^p g(h_1/2) - g(h_1)}{2^p - 1}$

Romberg (1)

- **Let's define the notation:** $R_{i,1} = I_i$
  - where I_i is the approximate integral from the recursive trapezoidal rule computed with 2^{i-1} panels

- Romberg Integration starts by computing
  - $R_{1,1} = I_1$ (one panel)
  - $R_{2,1} = I_2$ (two panels)
- The leading error term ($c_1 h^2$) is then removed with the Richardson extrapolation with p=2. The result is called $R_{2,2}$
  - $R_{2,2} = \frac{2^2 R_{2,1} - R_{1,1}}{2^2 - 1} = \frac{4}{3} R_{2,1} - \frac{1}{3} R_{1,1}$
- The leading error on $R_{2,2}$ is $c_2 h^4$

- It is convenient to store the results in an array $\begin{bmatrix} R_{1,1} & \\ R_{2,1} & R_{2,2} \end{bmatrix}$

Romberg (2)

- Then, we can compute
  - $R_{3,1} = I_3$ (4 panels)
- We can repeat Richardson extrapolation with p=2 on $R_{2,1}$ and $R_{3,1}$ to compute $R_{3,2}$.

$$\begin{bmatrix} R_{1,1} & \\ R_{2,1} & R_{2,2} \\ R_{3,1} & R_{3,2} \end{bmatrix}$$

  - $R_{3,2} = \frac{2^2 R_{3,1} - R_{2,1}}{2^2 - 1} = \frac{4}{3} R_{3,1} - \frac{1}{3} R_{2,1}$
- Element of the second column have a leading error of the form $c_2 h^4$ We can use Richardson extrapolation with p=4 to eliminate it
  - $R_{3,3} = \frac{2^4 R_{3,2} - R_{2,2}}{2^4 - 1} = \frac{16}{15} R_{3,2} - \frac{1}{15} R_{2,2}$
  - The leading error on $R_{3,3}$ is of the form $c_3 h^6$
  - What we have computed so far is:

$$\begin{bmatrix} R_{1,1} & & \\ R_{2,1} & R_{2,2} & \\ R_{3,1} & R_{3,2} & R_{3,3} \end{bmatrix}$$

Romberg (3)

- One more round iteration would lead to:

$$\begin{bmatrix} R_{1,1} & & & \\ R_{2,1} & R_{2,2} & & \\ R_{3,1} & R_{3,2} & R_{3,3} & \\ R_{4,1} & R_{4,2} & R_{4,3} & R_{4,4} \end{bmatrix}$$

  - The leading error on $R_{4,4}$ is of the form $c_4 h^8$
  - Note the most precise estimate of the integral is always the last element of the diagonal.
  - The process is continued until the difference between two successive diagonal terms become sufficiently small.

Romberg (4)

- The general form is:

$$R_{i,j} = \frac{4^{j-1}R_{i,j-1} - R_{i-1,j-1}}{4^{j-1} - 1} \qquad \text{with } i > 1 \text{ and } j = 2, 3, \dots, i$$

- An illustration of the process is:

$$\begin{tikzcd}
\boxed{R_{i-1,j-1}} \arrow[dr] & & \\
& \alpha \arrow[dr] & \\
\boxed{R_{i,j-1}} \arrow[r] & \beta \arrow[r] & \boxed{R_{i,j}}
\end{tikzcd}$$

- With the coefficients α and β which depends on j

j	2	3	4	5	6
α	$-1/3$	$-1/15$	$-1/63$	$-1/255$	$-1/1023$
β	$4/3$	$16/15$	$64/63$	$256/255$	$1024/1023$

Romberg Vs Simpson

- Romberg $R_{k,2}$ is equivalent to Simpson with 2^{k-1} panels
- **Simpson:**

$$I = \frac{h}{3}\left[f(x_0) + 4\sum_{i=1,3,\ldots}^{n-1} f(x_i) + 2\sum_{i=2,4,\ldots}^{n-2} f(x_i) + f(x_n)\right]$$

- **Composite Trapezoidal Rule:**

$$R_{k,1} = I_k = \frac{h}{2}\left[f(x_0) + 2\sum_{i=1}^{n-1} f(x_i) + f(x_n)\right]$$

2x less panels
h→2h

$$R_{k-1,1} = I_{k-1} = h\left[f(x_0) + 2\sum_{i=2,4,\ldots}^{n-2} f(x_i) + f(x_n)\right]$$

- **Romberg $R_{k,2}$**

$$R_{k,2} = \frac{4}{3}R_{k,1} - \frac{1}{3}R_{k-1,1} = \frac{h}{3}\left[f(x_0) + 4\sum_{i=1,3,\ldots}^{n-1} f(x_i) + 2\sum_{i=2,4,\ldots}^{n-2} f(x_i) + f(x_n)\right]$$

Example

- **Estimate $\int_0^{\pi} si^{n}(x)dx$ up to 4 decimals using Romberg Integration:**
- **Solution:**

$$R_{1,1} = I(\pi) = \frac{\pi}{2}\left[f(0) + f(\pi)\right] = 0$$

$$R_{2,1} = I(\pi/2) = \frac{1}{2}I(\pi) + \frac{\pi}{2}f(\pi/2) = 1.5708$$

$$R_{3,1} = I(\pi/4) = \frac{1}{2}I(\pi/2) + \frac{\pi}{4}\left[f(\pi/4) + f(3\pi/4)\right] = 1.8961$$

$$R_{4,1} = I(\pi/8) = \frac{1}{2}I(\pi/4) + \frac{\pi}{8}\left[f(\pi/8) + f(3\pi/8) + f(5\pi/8) + f(7\pi/8)\right]$$
$$= 1.9742$$

$$\begin{bmatrix} R_{1,1} & & & \\ R_{2,1} & R_{2,2} & & \\ R_{3,1} & R_{3,2} & R_{3,3} & \\ R_{4,1} & R_{4,2} & R_{4,3} & R_{4,4} \end{bmatrix} = \begin{bmatrix} 0 & & & \\ 1.5708 & 2.0944 & & \\ 1.8961 & 2.0046 & 1.9986 & \\ 1.9742 & 2.0003 & 2.0000 & 2.0000 \end{bmatrix}$$

j	2	3	4
α	$-1/3$	$-1/15$	$-1/63$
β	$4/3$	$16/15$	$64/63$

- **Convergence on:**

$$\int_0^{\pi} \sin x\, dx = R_{4,4} = 2.0000,$$

Gaussian Integration

Introduction

- **Newton-Cotes**
  - Used to integrate smooth functions
  - $I = \int_a^b f(x)dx = \sum_{i=0}^{n} A_i f(x_i)$ with $x_i = a + ih$ (nodes are equidistant)

- **Gaussian Quadrature**
  - Can also integrate functions with singularities
  - Integral of the form:

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

Where $w(x)$ is the weighting function can be $1, e^{-x}, e^{-x^2}, \frac{1}{\sqrt{1-x^2}}$

  - Gaussian quadrature can also accommodate casaes with a$=-\infty$ or b$=\infty$
  - $I = \sum_{i=0}^{n} A_i f(x_i)$ approximation has the same form as in Newton-Cotes
  - but x_i are at specific positions

Gaussian Quadrature

- Weights A_i and nodes x_i are such that $\sum_{i=0}^{n} A_i f(x_i)$ is the **exact integral if $\boldsymbol{f(x)}$ is a polynomial of degree 2n + 1** or less

$$I = \int_a^b w(x) P_n(x) dx = \sum_{i=0}^{n} A_i P_m(x_i), \qquad m < 2n + 1$$

- One way of determining the weights and abscissas is to substitute $P_0(x) = 1, P_1(x) = x, \dots, P_{2n+1}(x) = x^{2n+1}$ in the above equation and solve the resulting 2n + 2 equations

$$\int_a^b w(x) x^j dx = \sum_{i=0}^{n} A_i {x_i}^j, \qquad j = 0,1, \dots, 2n + 1$$

Example (1)

- Let's do the exercise for $\boldsymbol{w(x) = e^{-x}}$, $a = 0,\ b = \infty$ and n=1

$$\int_a^b w(x)x^j dx = \sum_{i=0}^{n} A_i x_i{}^j, \qquad j = 0,1,\dots,2n+1$$

- Lead to the 4 equations

$$\int_0^\infty e^{-x} dx = A_0 + A_1$$
$$\int_0^\infty e^{-x} x\, dx = A_0 x_0 + A_1 x_1$$
$$\int_0^\infty e^{-x} x^2 dx = A_0 x_0^2 + A_1 x_1^2$$
$$\int_0^\infty e^{-x} x^3 dx = A_0 x_0^3 + A_1 x_1^3$$

$\Rightarrow$

$$A_0 + A_1 = 1$$
$$A_0 x_0 + A_1 x_1 = 1$$
$$A_0 x_0^2 + A_1 x_1^2 = 2$$
$$A_0 x_0^3 + A_1 x_1^3 = 6$$

Example (2)

- Solving the system:

$$\begin{aligned} A_0 + A_1 &= 1 \\ A_0 x_0 + A_1 x_1 &= 1 \\ A_0 x_0^2 + A_1 x_1^2 &= 2 \\ A_0 x_0^3 + A_1 x_1^3 &= 6 \end{aligned} \quad \Rightarrow \quad \begin{aligned} x_0 &= 2 - \sqrt{2} \qquad A_0 = \frac{\sqrt{2}+1}{2\sqrt{2}} \\ x_1 &= 2 + \sqrt{2} \qquad A_1 = \frac{\sqrt{2}-1}{2\sqrt{2}} \end{aligned}$$

- Integral can be approximated by:

$$\int_a^b e^{-x} f(x)dx = \frac{1}{2\sqrt{2}}\left[\left(\sqrt{2}+1\right)f\left(2-\sqrt{2}\right) + \left(\sqrt{2}-1\right)f(2+\sqrt{2})\right]$$

Gaussian Quadrature

- Solving the system of equation is in general not easy
  - It is not linear
  - Difficult for large n

- Practical methods of finding x_i and A_i require some knowledge of orthogonal polynomials.
  - This will not be discussed in this course

- There are, however, several "classical" Gaussian integration formulas for which the abscissas and weights have been computed with great precision and then tabulated.

- These formulas can be used without knowing the theory behind them, because all one needs for Gaussian integration are the values of x_i and A_i.

Classical Gaussian Quadrature (1)

- **Gauss-Legendre:**
- Weighting function is 1 →

$$\int_{-1}^{1} f(\xi)d\xi \approx \sum_{i=0}^{n} A_i f(\xi_i)$$

- Nodes and weights are symmetric around 0
- Truncation error:

$$E = \frac{2^{2n+3}\left[(n+1)!\right]^4}{(2n+3)\left[(2n+2)!\right]^3} f^{(2n+2)}(c), \quad -1 < c < 1$$

- Tabulated nodes and weights:

$\pm\xi_i$	A_i	$\pm\xi_i$	A_i
$n = 1$		$n = 4$	
0.577 350	1.000 000	0.000 000	0.568 889
$n = 2$		0.538 469	0.478 629
0.000 000	0.888 889	0.906 180	0.236 927
0.774 597	0.555 556	$n = 5$	
$n = 3$		0.238 619	0.467 914
0.339 981	0.652 145	0.661 209	0.360 762
0.861 136	0.347 855	0.932 470	0.171 324

Classical Gaussian Quadrature (2)

- **Gauss-Legendre:**
- Usage on an integration domain [a,b] instead of [-1,1] can be achieved after a variable transformation

$$\int_a^b f(x)dx \qquad \longleftrightarrow \qquad \int_{-1}^{1} f(\xi)d\xi$$

- Transformation form [a,b] to [-1,1]:

$$x = \frac{b+a}{2} + \frac{b-a}{2}\xi \qquad\qquad dx = \frac{(b-a)}{2}d\xi$$

- The quadrature becomes:

$$\int_a^b f(x)dx \approx \frac{(b-a)}{2}\sum_{i=1}^{n} A_i f(x_i)$$

Classical Gaussian Quadrature (3)

- **Gauss-Chebyshev:**
- Weighting function is $\frac{1}{\sqrt{1-x^2}}$ → $\int_{-1}^{1} (1 - x^2)^{-1/2} f(x) dx$
- Nodes are symmetric around 0:

$$x_i = \cos \frac{(2i+1)\pi}{2n+2}$$

- Weights are all equals:

$$A_i = \pi / (n+1).$$

- Truncation error:

$$E = \frac{2\pi}{2^{2n+2}(2n+2)!} f^{(2n+2)}(c), \quad -1 < c < 1$$

Classical Gaussian Quadrature (4)

- **Gauss-Laguerre:**
- Weighting function is e^{-x} →

$$\int_0^\infty e^{-x} f(x)\,dx \approx \sum_{i=0}^{n} A_i f(x_i)$$

- Tabulated nodes and weights:

x_i	A_i	x_i	A_i
$n=1$		$n=4$	
0.585 786	0.853 554	0.263 560	0.521 756
3.414 214	0.146 447	1.413 403	0.398 667
$n=2$		3.596 426	(−1)0.759 424
0.415 775	0.711 093	7.085 810	(−2)0.361 175
2.294 280	0.278 517	12.640 801	(−4)0.233 670
6.289 945	(−1)0.103 892	$n=5$	
$n=3$		0.222 847	0.458 964
0.322 548	0.603 154	1.188 932	0.417 000
1.745 761	0.357 418	2.992 736	0.113 373
4.536 620	(−1)0.388 791	5.775 144	(−1)0.103 992
9.395 071	(−3)0.539 295	9.837 467	(−3)0.261 017
		15.982 874	(−6)0.898 548

Table 6.4. Nodes and weights for Gauss–Laguerre quadrature (Multiply numbers by 10^k, where k is given in parentheses.)

Classical Gaussian Quadrature (5)

- **Gauss-Hermite:**
- Weighting function is e^{-x^2} → $$\int_{-\infty}^{\infty} e^{-x^2} f(x)dx \approx \sum_{i=0}^{n} A_i f(x_i)$$
- Tabulated nodes and weights:

$\pm x_i$	A_i	$\pm x_i$	A_i
$n=1$		$n=4$	
0.707 107	0.886 227	0.000 000	0.945 308
$n=2$		0.958 572	0.393 619
0.000 000	1.181 636	2.020 183	(−1) 0.199 532
1.224745	0.295 409	$n=5$	
$n=3$		0.436 077	0.724 629
0.524 648	0.804 914	1.335 849	0.157 067
1.650 680	(−1)0.813 128	2.350 605	(−2)0.453 001

Table 6.5. Nodes and weights for Gauss–Hermite quadrature (Multiply numbers by 10^k, where k is given in parentheses.)

Example (1)

- **Estimate $\int_{-1}^{1}(1-x^2)^{3/2}\,dx$ as accurately as possible**

- **Solution:**
  - $(1-x^2)^{3/2}$ is smooth and free of singularities
  - We can use Gauss-Legendre,
  - but we could get the exact integral with Gauss-Chebyshev:

$$\int_{-1}^{1}(1-x^2)^{3/2}\,dx = \int_{-1}^{1}\frac{1}{\sqrt{(1-x^2)}}(1-x^2)^2\,dx$$

  - $(1-x^2)^2$ is a polynomial of degree 4
  - Integral is exact if the polynomial degree is < 2n+1 → n=2

$$x_i = \cos\frac{(2i+1)\pi}{2n+2} \qquad \Rightarrow \qquad x_i = \cos\frac{(2i+1)\pi}{6}, \quad i = 0, 1, 2$$

$$A_i = \pi/(n+1). \qquad \Rightarrow \qquad A_i = \pi/3$$

Example (2)

- **Estimate $\int_{-1}^{1}(1-x^2)^{3/2}\,dx$ as accurately as possible**
- **Solution:**

$$x_0 = \cos\frac{\pi}{6} = \frac{\sqrt{3}}{2}$$

$$x_1 = \cos\frac{\pi}{2} = 0$$

$$A_i = \pi/3$$

$$x_2 = \cos\frac{5\pi}{6} = \frac{\sqrt{3}}{2}$$

$$\int_{-1}^{1}(1-x^2)^{3/2}\,dx \approx \frac{\pi}{3}\sum_{i=0}^{2}(1-x_i^2)^2$$

$$= \frac{\pi}{3}\left[\left(1-\frac{3}{4}\right)^2 + (1-0)^2 + \left(1-\frac{3}{4}\right)^2\right] = \frac{3\pi}{8}$$

Example 2 (1)

- **Estimate $\int_{1.5}^{3} f(x)\,dx$ where $f(x)$ is represented by**

x	1.2	1.7	2.0	2.4	2.9	3.3
$f(x)$	−0.362 36	0.128 84	0.416 15	0.737 39	0.970 96	0.987 48

- **Solution:**
  - Data are unevenly distributed !
  - We approximate $f(x)$ by a polynomial $P_5(x)$ because we have 6 points
  - Then we can evaluate $\int_{1.5}^{3} f(x)\,dx = \int_{1.5}^{3} P_5(x)dx$ with Gauss-Legendre
  - Polynomial Degree is 5, so we only need 3 nodes (n=2)
  - Gauss-Legendre (with domain transformation):

$\pm\xi_i$		A_i
	$n=1$	
0.577 350		1.000 000
	$n=2$	
0.000 000		0.888 889
0.774 597		0.555 556
	$n=3$	
0.339 981		0.652 145
0.861 136		0.347 855

$$x_0 = \frac{3+1.5}{2} + \frac{3-1.5}{2}(-0.774597) = 1.6691$$

$$x_1 = \frac{3+1.5}{2} = 2.25$$

$$x_2 = \frac{3+1.5}{2} + \frac{3-1.5}{2}(0.774597) = 2.8309$$

Example 2 (2)

- **Solution:**
  - From Newton or Neville interpolation algorithm, we can find an interpolant
  - And use it to evaluate the function at x_0, x_1 and x_2
  - We find:

$$P_5(x_0) = 0.098\,08 \qquad P_5(x_1) = 0.628\,16 \qquad P_5(x_2) = 0.952\,16$$

  - From Gauss-Legendre,

$$I = \int_{1.5}^{3} P_5(x)dx = \frac{3-1.5}{2}\sum_{i=0}^{2} A_i P_5(x_i)$$

  - We find:

$$I = 0.75\,[0.555\,556(0.098\,08) + 0.888\,889(0.628\,16) + 0.555\,556(0.952\,16)]$$

$$= 0.856\,37$$

  - The points were distributed on $f(x) = -\cos(x)$, so we can compare with the true value of the integral: 0. 85638. The difference is within roundoff error.