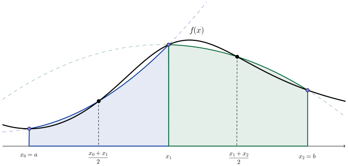
Algorithms and Data Structures

Quadrature
Newton-Côtes

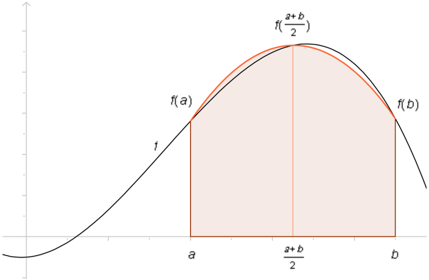


Learning goals

- Polynomial interpolation
- Newton-Côtes
- Composite rule

QUADRATURE WITH POLYNOMIALS

Idea: Approximate f in $[a, b]$ by polynomial interpolation of degree m



`https://de.wikipedia.org/wiki/Simpsonregel`

Approximation of the integral of f in $[a, b]$ using a polynomial of degree $m = 2$. Three grid points are needed.

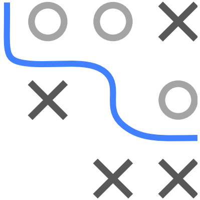
POLYNOMIAL INTERPOLATION

- **Find**: Polynomial interpolation $p_m(x) = \sum_{i=0}^{m} a_i x^i$ of degree m
- **Required**: Evaluation at $m + 1$ data points $(x_0 = a, x_1, ..., \ldots, x_m = b)$
- At these data points the function values of the polynomial and the function f must be identical, i.e. $p_m(x_k) = f(x_k)$ for $k = 0, 1, ..., m$ or equivalently using the **Vandermonde** matrix

$$\begin{pmatrix} 1 & x_0 & \ldots & x_0^m \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & \ldots & x_m^m \end{pmatrix} \begin{pmatrix} a_0 \\ \vdots \\ a_m \end{pmatrix} = \begin{pmatrix} f(x_0) \\ \vdots \\ f(x_m) \end{pmatrix}$$

- **Existence and uniqueness**: If the matrix is regular, the system of equations can be solved **uniquely**. The matrix is regular if the grid points $x_k, k = 1, ..., m$ are pairwise distinct.

POLYNOMIAL INTERPOLATION / 2



The polynomial interpolation can be determined by the solution of the equation system above. However, the effort is high (solution of the LES is $\mathcal{O}(n^3)$).

The polynomial interpolation can also be defined by using **Lagrange polynomials**:

$$L_{im}(x) = \prod_{j=0,\, j\neq i}^{m} \frac{x - x_j}{x_i - x_j}, \quad i = 0, 1, \ldots, m$$

The polynomial interpolation is:

$$p_m(x) = \sum_{i=0}^{m} L_{im}(x) f(x_i)$$

POLYNOMIAL INTERPOLATION / 3

With:
$$L_{im}(x_k) = \prod_{j=0,\, j\neq i}^{m} \frac{x_k - x_j}{x_i - x_j} = \begin{cases} 1 & \text{if } k = i \\ 0 & \text{if } k \neq i \end{cases}$$

Example: Let $m = 3$ and we calculate L_{i3} for $i = 2$

$$L_{23}(x_k) = \prod_{j=0,\, j\neq 2}^{3} \frac{x_k - x_j}{x_2 - x_j} = \frac{x_k - x_0}{x_2 - x_0} \cdot \frac{x_k - x_1}{x_2 - x_1} \cdot \frac{x_k - x_3}{x_2 - x_3}$$

For $k = i = 2$ all factors are 1

$$L_{23}(x_2) = \frac{x_2 - x_0}{x_2 - x_0} \cdot \frac{x_2 - x_1}{x_2 - x_1} \cdot \frac{x_2 - x_3}{x_2 - x_3} = 1 \cdot 1 \cdot 1 = 1$$

and for $k \neq 2$, e.g. $k = 1$, one factor is 0 and thus the whole product will be 0

$$L_{23}(x_1) = \frac{x_1 - x_0}{x_2 - x_0} \cdot \underbrace{\frac{x_1 - x_1}{x_2 - x_1}}_{=0} \cdot \frac{x_1 - x_3}{x_2 - x_3} = 0$$

So the polynomial is actually the polynomial interpolation through $(x_k, f(x_k))$

$$p_m(x_k) = \sum_{i=0}^{m} L_{im}(x_k) f(x_i) = f(x_k) \quad \text{for k = 0, 1, ..., m}$$

NEWTON-CÔTES

Instead of f we now integrate the polynomial p_m:

$$\begin{aligned} I(p_m) &= \int_a^b p_m(x)\, dx = \int_a^b \sum_{i=0}^{m} L_{im}(x) f(x_i)\, dx \\ &= \sum_{i=0}^{m} f(x_i) \underbrace{\int_a^b L_{im}(x)\, dx}_{:= w_{im}} \end{aligned}$$

So the integral of p_m on $[a, b]$ is defined by

$$I(p_m) = \sum_{i=0}^{m} w_{im} f(x_i)$$

with weights $w_{im} = \int_a^b L_{im}(x)\, dx$.

NEWTON-CÔTES / 2

Using equidistant grid points, i.e. $x_i = a + i \cdot h, i = 0, 1, ..., m$ with $h = \frac{b-a}{m}$, the formula can be further simplified.

When calculating the weights, we use integration by substitution $\int_{\varphi(0)}^{\varphi(m)} L_{im}(x)dx = \int_0^m L_{im}(\varphi(x)) \cdot \varphi'(x)\, dx$ with $\varphi(x) = x \cdot h + a$.
Since $\varphi(0) = a$ and $\varphi(m) = b$, the following holds

$$w_{im} \quad = \quad \int_{\varphi(0)}^{\varphi(m)} L_{im}(x)\, dx = \int_0^m L_{im}(x \cdot h + a) \cdot h\, dx = \int_0^m \prod_{j=0,\, j \neq i}^{m} \frac{x-j}{i-j} \cdot \frac{b-a}{m}\, dx$$

In the last step the fact that $x_i = i \cdot h + a$ was exploited

$$L_{im}(x \cdot h + a) \quad = \quad \prod_{j=0,\, j \neq i}^{m} \frac{x \cdot h + a - x_j}{x_i - x_j} = \prod_{j=0,\, j \neq i}^{m} \frac{x \cdot h + a - (j \cdot h + a)}{i \cdot h + a - (j \cdot h + a)} = \prod_{j=0,\, j \neq i}^{m} \frac{x-j}{i-j}$$

NEWTON-CÔTES / 3

The Newton-Côtes formula for equidistant grid points is given by

$$Q_m(f) = \int_a^b p_m(x)\,dx = (b-a)\sum_{i=0}^{m} w_{im} f(x_i),$$

with weights

$$w_{im} = \frac{1}{m}\int_0^m \prod_{j=0,\,j\neq i}^{m} \frac{x-j}{i-j}\,dx, \quad \text{for} \quad 0 \leq i \leq m.$$

For a given polynomial of degree m the weights have to be calculated (or looked up) only once, and the formula can be generalized to all possible intervals $[a, b]$.

NEWTON-CÔTES / 4

Example: For $m = 1$, two grid points are needed $\rightarrow x_0 = a$, $x_1 = b$.

Calculation of the weights for the integral $[0, 1]$:

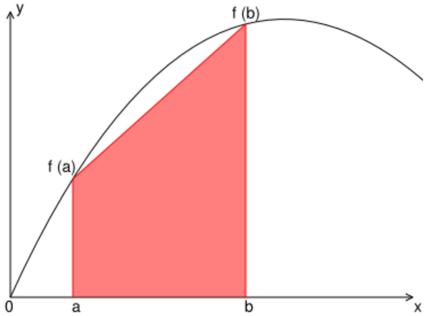
$$
\begin{aligned}
w_{01} &= \int_0^1 \left(\frac{x-1}{0-1} \right) dx = \int_0^1 (1-x)\, dx = \frac{1}{2} \\
w_{11} &= \int_0^1 \left(\frac{x-0}{1-0} \right) dx = \int_0^1 x\, dx = \frac{1}{2}
\end{aligned}
$$

So the formula is given by

$$
\begin{aligned}
I(f) \approx I(p_1) &= (b-a) \cdot (w_{01} \cdot f(x_0) + w_{11} \cdot f(x_1)) \\
&= (b-a) \cdot \frac{f(a) + f(b)}{2}
\end{aligned}
$$

NEWTON-CÔTES / 5

The approach is also called **trapezoidal rule**.



`https://de.wikipedia.org/wiki/Trapezregel`

Approximation of the integral of f on $[a, b]$ using polynomials of degree $m = 1$ (trapezoidal rule).

OPEN VS. CLOSED NEWTON-CÔTES

We distinguish between:

- **Closed** Newton-Côtes formulas: interval margins a and b are used as grid points for the polynomial interpolation. Usually equidistant nodes are used as grid points, $x_i = a + i \cdot h$, $i = 0, \ldots, m$ with $h = \frac{b-a}{m}$
- **Open** Newton-Côtes formulas: interval margins a and b are **not** used as grid points for the polynomial interpolation. Usually equidistant nodes $x_i = a + i \cdot h$, $i = 1, \ldots, m+1$, $h = \frac{b-a}{m+2}$ are used.

WEIGHTS OF THE NEWTON CÔTES

m	type	sampling points [(*)]	ω_{im}	
0	open	$\frac{1}{2}$	1	Riemann sum
1	closed	0, 1	$\frac{1}{2}, \frac{1}{2}$	trapezoidal rule
2	closed	$0, \frac{1}{2}, 1$	$\frac{1}{6}, \frac{4}{6}, \frac{1}{6}$	Simpson's rule
3	closed	$0, \frac{1}{3}, \frac{2}{3}, 1$	$\frac{1}{8}, \frac{3}{8}, \frac{3}{8}, \frac{1}{8}$	3/8-rule
4	closed	$0, \frac{1}{4}, \frac{2}{4}, \frac{3}{4}, 1$	$\frac{7}{90}, \frac{32}{90}, \frac{12}{90}, \frac{32}{90}, \frac{7}{90}$	Milne rule
⋮	⋮	⋮	⋮	

[(*)] The grid points are only valid for integration on $[0, 1]$. For general integration limits the grid points are $a + x_i \cdot (b - a)$.

NEWTON-CÔTES: QUADRATURE ERROR

The interpolation error can generally be represented by

$$f(x) - p_m(x) = \frac{1}{(m+1)!} f^{(m+1)}(\mathbf{x}^{(i)}) \cdot \prod_{i=0}^{m} (x - x_i),$$

for an intermediate point $\mathbf{x}^{(i)} \in [a, b]$. With this, the quadrature error can be generally derived by

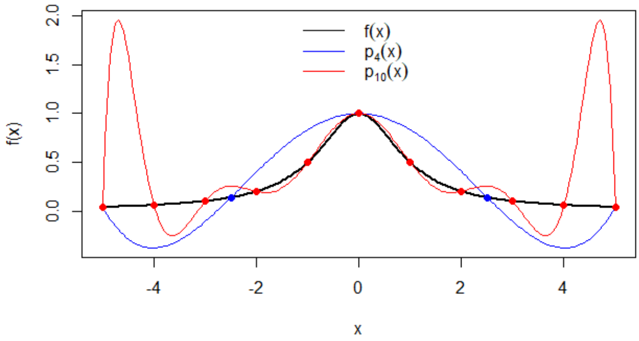
$$E(f) = \int_a^b p_m(x) - f(x)\, dx = -\frac{1}{(m+1)!} f^{(m+1)}(\mathbf{x}^{(i)}) \int_a^b \prod_{i=0}^{m} (x - x_i)\, dx$$

Example: Trapezoidal rule

$$\begin{aligned} E(f) &= \int_a^b p_m(x) - f(x)\, dx = -\int_a^b \frac{1}{2!} f^{(2)}(\mathbf{x}^{(i)})(x-a)(x-b)\, dx \\ &= -\frac{f^{(2)}(\mathbf{x}^{(i)})}{2} \int_a^b (x-a)(x-b)\, dx = \frac{1}{12}(b-a)^3 \cdot f^{(2)}(\mathbf{x}^{(i)}) \end{aligned}$$

COMPOSITE RULE

Interpolation with a polynomial of higher degree allows for more flexibility. However, the polynomial function **oscillates** stronger near the interval boundaries (Runge's phenomenon).



COMPOSITE RULE / 2

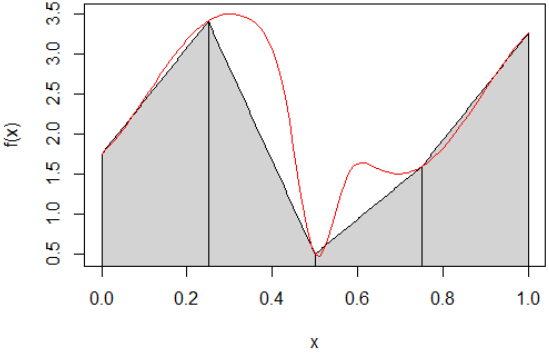
In addition, many Newton-Côtes formulas have negative weights for degree ≥ 8, which entails the risk of cancellation.

Therefore, it is common to divide larger integration intervals $[a, b]$ into n sub-intervals and apply the Newton-Côtes formula with a lower polynomial degree on each of these sub-intervals. Then, the individual results for the sub-intervals are added up.

In numerical integration this is known as the **composite rule**.

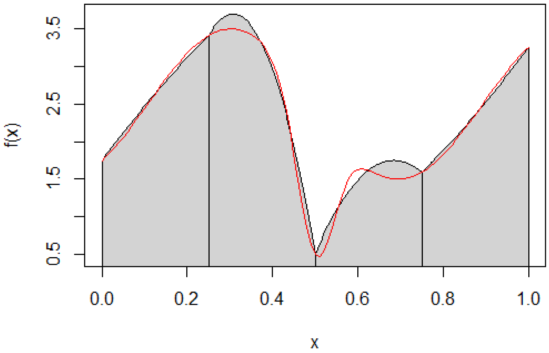
COMPOSITE RULE / 3

Degree $m = 1$**:** f is approximated in the intervals $[x_i, x_{i+1}]$ by linear functions (trapezoidal rule)



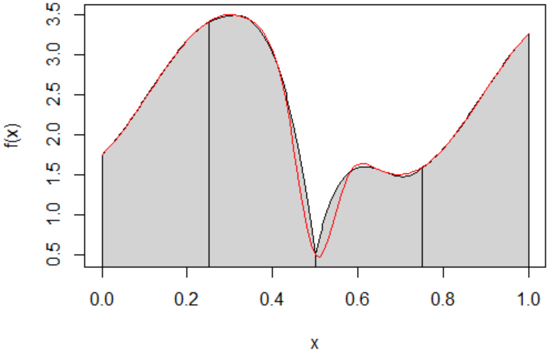
COMPOSITE RULE / 4

Degree $m = 2$**:** f is approximated in the intervals $[x_i, x_{i+1}]$ by quadratic functions (Simpson's rule)



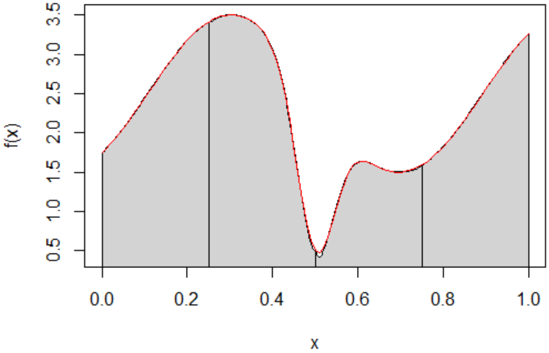
COMPOSITE RULE / 5

Degree $m = 3$: f is approximated in the intervals $[x_i, x_{i+1}]$ by polynomials of degree 3



COMPOSITE RULE / 6

Degree $m = 4$**:** f is approximated in the intervals $[x_i, x_{i+1}]$ by polynomials of degree 4



CONCLUSION: NUMERICAL INTEGRATION

- In practice, adaptive procedures are often used: the number of sub-intervals to which the Newton-Côtes formulas are applied is adaptively fine-tuned.
- The composite Simpson's rule no longer really corresponds to the state-of-the-art, but is certainly performant.
- There are better methods such as Gaussian quadrature or Gauss–Kronrod quadrature formula (not further discussed here).
- Impressive convergence rates of some procedures (in 1D), if f is sufficiently smooth, otherwise possibly problematic.
- In principle, the procedures discussed so far can also be generalized to **higher dimensions**.
- **But:** Computing effort increases exponentially with dimension d (*curse of dimensionality*).