



---

# LINMA2171 Numerical Analysis

---

SIMON DESMIDT

Academic year 2024-2025 - Q1



UCLouvain

# Contents

<b>1</b>	<b>Polynomial interpolation</b>	<b>2</b>
1.1	Lagrange interpolation . . . . .	2
1.2	Hermite interpolation . . . . .	3
1.3	Neville's algorithm . . . . .	3
1.4	Newton's interpolation formula . . . . .	4
1.5	Linear algebra approach . . . . .	4
1.6	Barycentric interpolation formula . . . . .	4
1.7	Trigonometric interpolation . . . . .	5
1.8	Rational interpolation . . . . .	6
<b>2</b>	<b>Polynomial approximation</b>	<b>7</b>
2.1	Definitions . . . . .	7
2.2	Best approximation in the $\infty$ -norm . . . . .	7
2.3	Chebyshev polynomials . . . . .	8
2.4	Interpolation . . . . .	9
<b>3</b>	<b>Piecewise polynomial approximation</b>	<b>12</b>
3.1	Definition . . . . .	12
3.2	Linear interpolating splines . . . . .	12
3.3	B-splines . . . . .	13
3.4	Regression with splines . . . . .	13
3.5	Interpolation by natural cubic splines . . . . .	14
3.6	Hermite cubic splines . . . . .	15
3.7	Smoothing splines . . . . .	16
3.8	Error bounds of interpolation by natural splines . . . . .	16
3.9	Vector-valued splines . . . . .	17
3.10	Tensor product splines . . . . .	17
<b>4</b>	<b>Numerical integration</b>	<b>19</b>
4.1	Newton-Cotes formulae . . . . .	19
4.2	Composite formulae . . . . .	20
4.3	Gauss quadrature . . . . .	20

# Polynomial interpolation

$\mathcal{P}_n$  is the set of all real polynomials of degree at most  $n$ .

- The Runge phenomenon is the explosion of the polynomial near the boundary of the domain when the interpolation points are chosen to be equidistant. A solution to that is to put more points near the boundary and less in the middle of the domain, e.g. Chebyshev points.

## 1.1 Lagrange interpolation

Let  $x_0, \dots, x_n$  be distinct real numbers. The Lagrange polynomial  $L_k$  of degree  $n$  is such that it is equal to 0 for all  $x_i, i \neq k$  and 1 for  $x_k$ . This serves as a base for the next interpolations. The general formula for the Lagrange polynomial is

$$L_k(x) = \prod_{\substack{i=0 \\ i \neq k}}^n \frac{x - x_i}{x_k - x_i} \quad k = 0, 1, \dots, n \quad (1.1)$$

- Note: we usually denote  $L_k(x; x_0, \dots, x_n)$  or let  $\chi = (x_0, \dots, x_n)$  and  $L_k(x; \chi)$ .

**Theorem 1.1.** Assume that  $n \geq 0$ . Let  $x_i, i = 0, \dots, n$  be distinct real number and suppose that  $y_i, i = 0, \dots, n$  are real numbers. Then, there is a unique polynomial  $p_n \in \mathcal{P}_n$  such that  $p_n(x_i) = y_i$  for all  $i = 0, \dots, n$ .

**Definition 1.2.** The polynomial  $p_n$  defined by

$$p_n(x) = \sum_{k=0}^n y_k L_k(x) \quad (1.2)$$

with  $L_k$  the Lagrange polynomial ( $L_0(x) \equiv 1$ ), is the Lagrange interpolation polynomial of degree  $n$  for the points  $(x_i, y_i), i = 0, \dots, n$ .

**Theorem 1.3.** Let  $f : \mathbb{R} \mapsto \mathbb{R}, f \in \mathcal{C}^{n+1}[a, b]$ . Then, given that  $x \in [a, b]$ , there exists  $\xi = \xi(x)$  on  $(a, b)$  such that

$$f(x) - p_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \pi_{n+1}(x) \quad (1.3)$$

where  $\pi_{n+1} = \prod_{i=0}^n (x - x_i)$ . Moreover,

$$|f(x) - p_n(x)| \leq \frac{M_{n+1}}{(n+1)!} |\pi_{n+1}(x)| \quad M_{n+1} = \max_{\zeta \in [a, b]} |f^{(n+1)}(\zeta)| \quad (1.4)$$

## 1.2 Hermite interpolation

Let  $x_0, \dots, x_n$  be distinct real numbers. Then, given two sets of real numbers  $(y_0, \dots, y_n)$  and  $(z_0, \dots, z_n)$ , there is a unique polynomial  $p_{2n+1} \in \mathcal{P}_{2n+1}$  such that

$$p_{2n+1}(x_i) = y_i \quad p'_{2n+1}(x_i) = z_i \quad i = 0, \dots, n \quad (1.5)$$

The polynomial  $p_{2n+1}$  is called the Hermite interpolation polynomial of degree at most  $2n + 1$  for the data points  $(x_0, y_0, z_0), \dots, (x_n, y_n, z_n)$ . The expression is

$$p_{2n+1}(x) = \sum_{k=0}^n (H_k(x)y_k + K_k(x)z_k) \quad \begin{cases} H_k(x) = (L_k(x))^2(1 - 2L'_k(x_k)(x - x_k)) \\ K_k(x) = (L_k(x))^2(x - x_k) \end{cases} \quad (1.6)$$

where  $L_k(x)$  is the Lagrange polynomial.

- The  $H_k(x)$  are such that their derivative is zero for all  $x_i$ , and their value is zero for all  $x_i$  except  $x_k$ , where it is 1.

$$H_k(x_i) = \delta_{ik} \quad H'_k(x_i) = 0 \quad \forall i$$

- The  $K_k(x)$  are such that their derivative is zero for all  $x_i$  except  $x_k$  where it is one, and their value is zero for all  $x_i$ .

$$K_k(x_i) = 0 \quad K'_k(x_i) = \delta_{ik} \quad \forall i$$

→ Note: for  $n = 0$ , we define  $H_0(x) \equiv 1$  and  $K_0(x) \equiv x - x_0$ .

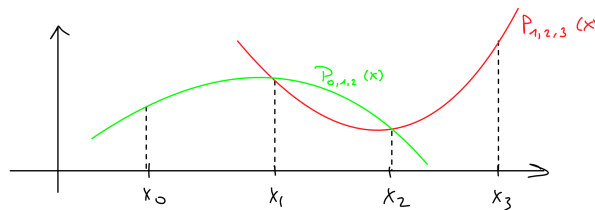
## 1.3 Neville's algorithm

Let us assume we are given a set of support points  $(x_i, y_i)$ ,  $i = 0, 1, \dots, n$ , and  $p_n$  is their Lagrange interpolation polynomial. Let us now define the notation  $P_{i_0 i_1 \dots i_k} \in \mathcal{P}_k$ , the polynomial for which  $P_{i_0 i_1 \dots i_k}(x_{i_j}) = y_{i_j}$  for all  $j = 0, 1, \dots, k$ . We work by recursion, with the following formula:

$$\begin{cases} P_i(x) = y_i \\ P_{i_0 i_1 \dots i_k} = \frac{(x - x_{i_0})P_{i_1 i_2 \dots i_k}(x) - (x - x_{i_k})P_{i_0 i_1 \dots i_{k-1}}(x)}{x_{i_k} - x_{i_0}} \end{cases} \quad (1.7)$$

Example:

Let us have four points  $(x_0, y_0), \dots, (x_3, y_3)$ . We want the polynomial interpolating all of them, using Neville's algorithm.



Here,

$$P_{0123}(x) = \frac{x - x_0}{x_3 - x_0} P_{123}(x) + \frac{x_3 - x}{x_3 - x_0} P_{012}(x) \quad (1.8)$$

## 1.4 Newton's interpolation formula

Newton's interpolation formula is used to evaluate polynomials with a computer, as it only needs to compute each operation  $(x - x_i)$  one time. We write it like:

$$p_n(x) = ((\dots (y_{0\dots n}(x - x_n) + y_{0\dots n-1})(x - x_{n-1}) + y_{0\dots n-2})(x - x_{n-2}) + \dots) + y_0 \quad (1.9)$$

with

$$y_{i_0 i_1 \dots i_k} = \frac{y_{i_1 \dots i_k} - y_{i_0 \dots i_{k-1}}}{x_{i_k} - x_{i_0}} \quad (1.10)$$

And the recursive formula is

$$P_{i_0 i_1 \dots i_k} = P_{i_0 i_1 \dots i_{k-1}}(x) + y_{i_0 i_1 \dots i_k}(x - x_{i_0})(x - x_{i_1}) \dots (x - x_{i_{k-1}}) \quad (1.11)$$

## 1.5 Linear algebra approach

Let  $(\phi_0, \dots, \phi_n)$  be a basis of  $\mathcal{P}_n$ , which is known to be an  $(n + 1)$ -dimensional linear space. The interpolation polynomial can thus be expressed in a unique way in the basis:

$$p_n(x) = \sum_{i=0}^n a_i \phi_i(x) \quad (1.12)$$

and the coefficient are obtained by solving the linear system

$$\begin{bmatrix} \phi_0(x_0) & \phi_1(x_0) & \dots & \phi_n(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \dots & \phi_n(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \dots & \phi_n(x_n) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix} \quad (1.13)$$

This is called a Vandermonde matrix, and its determinant is

$$\det(V) = \prod_{0 \leq i < j \leq n} (x_j - x_i) \quad (1.14)$$

which is always non zero, as the  $x_i$  are distinct, and the system has one unique solution.

→ Note: the condition number<sup>1</sup> of such a matrix grows exponentially with  $n$ .

## 1.6 Barycentric interpolation formula

This formula is interesting, because it is numerically stable, contrary to the linear algebra method described before. We use the following notation, called the nodal polynomial:

$$\pi_{n+1}(x) = \prod_{i=0}^n (x - x_i) \quad (1.15)$$

We now define

$$\lambda_j = \frac{1}{\prod_{k \neq j} (x_j - x_k)} \quad (1.16)$$

---

<sup>1</sup>It is a measure of the reaction of the system to a small perturbation

The modified Lagrange formula is then

$$p_n(x) = \pi_{n+1}(x) \sum_{j=0}^n \frac{\lambda_j}{x - x_j} y_j \quad (1.17)$$

For the polynomial  $p_n(x) = 1$ , we have the following expression:

$$1 = \pi_{n+1}(x) \sum_{j=0}^n \frac{\lambda_j}{x - x_j}$$

and thus we generally prefer to use the equivalent formula for equation (1.17):

$$p_n(x) = \sum_{j=0}^n \frac{\lambda_j y_j}{x - x_j} / \sum_{j=0}^n \frac{\lambda_j}{x - x_j} \quad (1.18)$$

for all  $x \notin \{x_0, \dots, x_n\}$ .

## 1.7 Trigonometric interpolation

Let us consider the evenly spaced points  $x_j = \frac{2\pi j}{N}$ ,  $j = 0, \dots, N$ , on the interval  $[0, 2\pi]$ , and the interpolation values  $f_0, \dots, f_N \in \mathbb{C}$ , with  $f_0 = f_N$ . The trigonometric interpolation problem consists of finding  $\beta_k$  such that

$$p(x) = \sum_{k=0}^{N-1} \beta_k e^{ikx} \text{ such that } p(x_j) = f_j \quad j = 0, \dots, N-1 \quad (1.19)$$

→ Note: the bound is  $N-1$  because the last condition  $p(x_N) = f_N$  is satisfied when the others are (periodicity).

This is equivalent to the generalization to  $\mathbb{C}$  of the polynomial interpolation problem: if we denote  $\omega := e^{ix}$ , the complex polynomial is

$$P(\omega) = \sum_{k=0}^{N-1} \beta_k \omega^k \quad (1.20)$$

The Vandermonde matrix in the complex case is defined as in the real case. We denote it  $W$ .

Theorem:  $W^*W = NI$  for a complex Vandermonde matrix in an interpolation problem.

From this, the solution to the interpolation problem is solved by multiplying both sides by  $W^*$ . We get

$$\beta = \frac{1}{N} W^* f \implies \beta_k = \frac{1}{N} \sum_{j=0}^{N-1} f_j e^{-i2\pi kj/N} \quad k = 0, \dots, N-1 \quad (1.21)$$

And that is the discrete Fourier transform (DFT)

## 1.8 Rational interpolation

Let the interpolation points be  $x_0 < x_1 < \dots < x_\sigma$ , with the values  $y_0, \dots, y_\sigma \in \mathbb{R}$ . We define the polynomial

$$\Phi(x) = \frac{p_\mu(x)}{q_\nu(x)} \quad p_\mu \in \mathcal{P}_\mu, q_\nu \in \mathcal{P}_\nu \quad (1.22)$$

$$\text{such that } \Phi(x_i) = y_i \quad i = 0, \dots, \sigma \quad (1.23)$$

The interpolation polynomial can be written

$$\Phi(x) = \frac{\sum_{k=0}^{\mu} a_k x^k}{\sum_{k=0}^{\nu} b_k x^k} = \frac{\lambda p_\mu(x)}{q_\nu(x)} \quad (1.24)$$

The number of constraints, i.e. points needed for the interpolation is then  $\sigma = \mu + \nu$ . This implies that if  $\Phi$  is a solution to the equation (1.22), then  $p_\mu, q_\nu$  are solutions of

$$p_\mu(x_i) - y_i q_\nu(x_i) = 0 \quad i = 0, \dots, \mu + \nu \quad (1.25)$$

$$\left( \sum_{k=0}^{\mu} a_k x_i^k \right) - y_i \left( \sum_{k=0}^{\nu} b_k x_i^k \right) = 0 \quad (1.26)$$

The theorem of existence states that the equation (1.25) always has a non trivial solution, i.e.  $(p_\mu, q_\nu) \neq (0, 0)$ .

The theorem of uniqueness states that if  $\Phi_1$  and  $\Phi_2$  are non trivial solutions of (1.25), then they are equivalent, i.e. they differ only by a common polynomial factor in the numerator and denominator.

- $p_\mu, q_\nu$  are relatively prime if they do not have zeros in common.

Given  $\Phi = \frac{p_\mu}{q_\nu}$ , let  $\tilde{\Phi} = \frac{\tilde{p}_\mu}{\tilde{q}_\nu}$  be the equivalent expression for which  $\tilde{p}_\mu$  and  $\tilde{q}_\nu$  are relatively prime.  $\Phi$  is the solution of (1.22)  $\iff \tilde{p}_\mu(x_i) - y_i \tilde{q}_\nu(x_i) = 0, i = 0, \dots, \mu + \nu$ .

# Polynomial approximation

## 2.1 Definitions

**Definition 2.1.** Suppose that  $w \in \mathcal{C}[a, b]$  is integrable on  $(a, b)$ . The set  $\mathcal{C}[a, b]$  is a normed linear space equipped with the norm

$$\|f\|_2^2 = \int_a^b w(x)|f(x)|^2 dx \quad (2.1)$$

**Lemma 2.2.** For a given interval  $[a, b]$ ,

$$\|f\|_2 \leq W\|f\|_\infty \quad W = \left[ \int_a^b w(x) dx \right]^{1/2} \quad (2.2)$$

→ Note: the 2- and  $\infty$ -norms are equivalent with vectors but not for functions. Hence they are not interchangeable in this course.

**Theorem 2.3.** Suppose  $f : \mathbb{R} \mapsto \mathbb{R}$ ,  $f \in \mathcal{C}[a, b]$ . Then, given any  $\epsilon > 0$ , there exists a polynomial  $p$  such that  $\|f - p\|_\infty < \epsilon$ . It also holds for the 2-norm if  $f$  is positive on  $[a, b]$ .

## 2.2 Best approximation in the $\infty$ -norm

**Definition 2.4.** A polynomial of best approximation of degree  $n$  to the function  $f$  in the  $\infty$ -norm is  $p_n \in \mathcal{P}_n$  such that

$$\|f - p_n\|_\infty = \inf_{q \in \mathcal{P}_n} \|f - q\|_\infty \quad (2.3)$$

**Theorem 2.5.** Given that  $f \in \mathcal{C}[a, b]$ , there exists a polynomial  $p_n \in \mathcal{P}_n$  such that  $\|f - p_n\|_\infty = \min_{q \in \mathcal{P}_n} \|f - q\|_\infty$ .

We call this polynomial the minimax polynomial.

**Theorem 2.6. De la Vallée Poussin's theorem** Let  $f \in \mathcal{C}[a, b]$  and  $r \in \mathcal{P}_n$ . Suppose that there exists  $n + 2$  points  $x_0 < \dots < x_{n+1}$  in the interval  $[a, b]$  such that  $f(x_i) - r(x_i)$  and  $f(x_{i+1}) - r(x_{i+1})$  have opposite signs. Then,

$$\min_{q \in \mathcal{P}_n} \|f - q\|_\infty \geq \min_{i=0, \dots, n+1} |f(x_i) - r(x_i)| \quad (2.4)$$



**Theorem 2.7.** Suppose that  $f \in \mathcal{C}[a, b]$ . A polynomial  $r \in \mathcal{P}_n$  is a minimax polynomial for  $f$  on  $[a, b]$  iff there exists a sequence of  $n + 2$  points  $x_i, i = 0, \dots, n + 1$  such that  $a \leq x_0 < \dots < x_{n+1} \leq b$ ,

$$\begin{aligned} |f(x_i) - r(x_i)| &\leq \|f - r\|_\infty & i = 0, \dots, n + 1 \\ f(x_i) - r(x_i) &= -[f(x_{i+1}) - r(x_{i+1})] & i = 0, \dots, n \end{aligned} \quad (2.5)$$

This means that  $f - r$  attains its maximum absolute value with alternating signs at the points  $x_i$ . These are called the critical points.

**Theorem 2.8.** Suppose that  $[a, b]$  is a bounded closed interval. Each  $f \in \mathcal{C}[a, b]$  has a unique minimax polynomial  $p_n \in \mathcal{P}_n$  on  $[a, b]$ .

## 2.3 Chebyshev polynomials

**Definition 2.9.** The Chebyshev polynomial  $T_n$  of degree  $n$  is defined, for  $x \in [-1, 1]$ , by

$$T_n(x) = \cos(n \arccos(x)) \quad n = 0, 1, \dots \quad (2.6)$$

Using some trigonometric formulae, there exists a recurrence relation for the Chebyshev polynomials:

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) \quad n = 1, \dots \quad x \in [-1, 1] \quad (2.7)$$

with  $T_0(x) \equiv 1$  and  $T_1(x) = x$ .

**Lemma 2.10.** The Chebyshev polynomials have the following properties:

- For  $n \geq 1$ ,  $T_n$  is a polynomial in  $x$  of degree  $n$  on the interval  $[-1, 1]$ , with leading coefficient  $2^{n-1}x^n$ ;
- $T_n$  is an even (resp. odd) function on  $[-1, 1]$  if  $n$  is even (resp. odd);
- For  $n \geq 1$ , the zeros of  $T_n$  are

$$x_j = \cos \frac{(2j-1)\pi}{2n} \quad j = 1, \dots, n \quad (2.8)$$

- $|T_n(x)| \leq 1$  for all  $x \in [-1, 1]$  and all  $n \geq 0$ ;
- For  $n \geq 1$ ,  $T_n(x) = \pm 1$ , alternately at the  $n + 1$  points  $x_k = \cos(\frac{k\pi}{n}), k = 0, \dots, n$ .

**Theorem 2.11.** Suppose that  $n \geq 0$ . The polynomial  $p_n \in \mathcal{P}_n$  defined by

$$p_n(x) = x^{n+1} - 2^{-n}T_{n+1}(x) \quad x \in [-1, 1] \quad (2.9)$$

is the minimax approximation of degree  $n$  to the function  $x \mapsto x^{n+1}$  on the interval  $[-1, 1]$ .

This implies that among all monic polynomials of degree  $n + 1$ , the polynomials  $2^{-n}T_{n+1}(x)$  and  $-2^{-n}T_{n+1}(x)$  have the smallest  $\infty$ -norm on the interval  $[-1, 1]$ .

---

**Algorithm 1** Exchange algorithm

---

- 1: Choose a reference  $\{\xi_i : i = 0, \dots, n+1\}$  that satisfies the condition  $a \leq \xi_0 < \xi_1 < \dots < \xi_{n+1} \leq b$ .
- 2: **while**  $\delta = |f(\eta) - p(\eta)| - |h|$  not small enough **do**
- 3:     Solve, for  $h \in \mathbb{R}$  and  $p \in \mathcal{P}_n$ ,

$$f(\xi_i) - p(\xi_i) = (-1)^i h \quad i = 0, \dots, n+1 \quad (2.10)$$

- 4:     Compute a point  $\eta \in [a, b]$  that realizes the maximum error, i.e.  $|f(\eta) - p(\eta)| = \|f - p\|_\infty$ .
- 5:     Replace a point of the reference by  $\nu$  in such a way that the new reference satisfies

$$\text{sign}(f(\xi_{i+1}) - p(\xi_{i+1})) = -\text{sign}(f(\xi_i) - p(\xi_i)) \quad i = 0, \dots, n \quad (2.11)$$

- 6: **end while**
- 

### 2.3.1 The exchange algorithm

Let  $f \in \mathcal{C}[a, b]$ . The exchange algorithm calculates the element  $p^* \in \mathcal{P}_n$  that minimizes the maximum error  $\|f - p\|_\infty$ .

## 2.4 Interpolation

**Theorem 2.12.** Suppose that  $f \in \mathcal{C}^{n+1}[a, b]$ . Let  $p_n \in \mathcal{P}_n$  denote the Lagrange interpolation polynomial of  $f$ , with interpolation points being the zeros of the Chebyshev polynomial:

$$\xi_j = \frac{1}{2}(b-a) \cos \frac{(j + \frac{1}{2})\pi}{n+1} + \frac{1}{2}(b+a) \quad j = 0, \dots, n \quad (2.12)$$

then,

$$\|f - p_n\|_\infty \leq \frac{(b-a)^{n+1}}{2^{2n+1}(n+1)!} M_{n+1} \quad (2.13)$$

### 2.4.1 Best approximation in the 2-norm

The polynomial of best approximation of degree  $n$  to the function  $f$  in the 2-norm on  $(a, b)$  is  $p_n \in \mathcal{P}_n$  such that

$$\|f - p_n\|_2 = \inf_{q \in \mathcal{P}_n} \|f - q\|_2 \quad (2.14)$$

#### Inner product

The set  $\mathcal{C}[a, b]$  is an inner product space with

$$\langle f, g \rangle = \int_a^b w(x) f(x) g(x) dx \quad (2.15)$$

where  $w(x) \in \mathcal{C}[a, b]$  is the weight function, positive and integrable on  $(a, b)$ .

**Definition 2.13.** A norm is strictly convex if

$$\forall p, \forall q \neq p, \quad \|p\| \leq r, \|q\| \leq r \implies \left\| \frac{p+q}{2} \right\| < r \quad (2.16)$$

The max-norm is not strictly convex.

The objective of this section is to minimise the error of the approximation in the 2-norm, defined as

$$E(c) = \|e_n\|_2^2 = \|f - p_n\|_2^2 = \int_0^1 |f(x) - p_n(x)|^2 dx \quad (2.17)$$

$c$  being the coefficient vector for the polynomial  $p_n$ .

To minimise this quantity, we solve

$$\sum_{k=0}^n M_{jk} c_k = b_j \quad j = 0, \dots, n \quad (2.18)$$

where  $M_{jk} = \int_0^1 x^{k+j} dx = \langle x^j, x^k \rangle = \frac{1}{k+j+1}$  and  $b_j = \int_0^1 f(x) x^j dx = \langle f, x^j \rangle$ .

### 2.4.2 Orthogonal polynomials

Suppose  $\varphi_j, j = 0, \dots, n$  form a basis for  $\mathcal{P}_n, n \geq 0$ . We want to express the polynomial  $p_n$  as

$$p_n(x) = \sum_{j=0}^n \gamma_j \varphi_j(x) \quad (2.19)$$

where  $\gamma_j$  are the coefficients. We get the system (as before):

$$\sum_{k=0}^n M_{jk} \gamma_k = b_j \quad j = 0, \dots, n \quad (2.20)$$

where  $M_{jk} = \langle \varphi_j, \varphi_k \rangle$  and  $b_j = \langle f, \varphi_j \rangle$ , the inner product being defined in (2.15). The matrix  $M$  will be diagonal if the basis is orthogonal.

**Definition 2.14.** Given a weight function  $w \in \mathcal{C}[a, b]$  positive and integrable on  $(a, b)$ , the sequence of polynomials  $\varphi_j, j = 0, 1, \dots$  is a system of orthogonal polynomials on the interval  $[a, b]$  with respect to  $w$  if each  $\varphi_j$  is exactly of degree  $j$  and if

$$\int_a^b w(x) \varphi_k(x) \varphi_j(x) dx \begin{cases} = 0 & k \neq j \\ \neq 0 & k = j \end{cases} \quad (2.21)$$

We can use the Gram-Schmidt procedure to create an orthogonal basis from a general set of polynomials.

**Theorem 2.15.** Given a function  $f$ , there exists a unique polynomial  $p_n \in \mathcal{P}_n$  such that

$$\|f - p_n\|_2 = \min_{q \in \mathcal{P}_n} \|f - q\|_2 \quad (2.22)$$

**Theorem 2.16.** A polynomial  $p_n \in \mathcal{P}_n$  is the polynomial of best approximation of degree  $n$  to a function  $f$  in the 2-norm iff the difference  $f - p_n$  is orthogonal to every element of  $\mathcal{P}_n$ , i.e.  $\langle f - p_n, q \rangle = 0 \forall q \in \mathcal{P}_n$ .

By the previous results, the polynomial  $p_n$  is of the form

$$p_n(x) = \sum_{j=0}^n \gamma_j \varphi_j(x) \quad \gamma_j = \frac{\langle f, \varphi_j \rangle}{\langle \varphi_j, \varphi_j \rangle} \quad j = 0, \dots, n \quad (2.23)$$

**Theorem 2.17.** Suppose that  $\varphi_j, j = 0, 1, \dots$ , is a system of orthogonal polynomials on the interval  $(a, b)$  with respect to  $w$ .  $\varphi_j$  are of exact degree  $j$ . Then, for  $j \geq 1$ , the zeros of the polynomial  $\varphi_j$  are real and distinct, and lie in the interval  $(a, b)$ .

# Piecewise polynomial approximation

## 3.1 Definition

Let  $\mathcal{S} = \mathcal{S}(k) = \mathcal{S}(k; x_0, \dots, x_m) = \{s \in \mathcal{C}^{k-1}[a, b] : s|_{[x_{i-1}, x_i]} \in \mathcal{P}_k, i = 1, \dots, m\}$  denote the linear space of splines of degree  $k \geq 1$ , with knots  $a = x_0 < x_1 < \dots < x_m = b$ . The conditions at the knots are the following:

$$s^{(j)}(x_i^-) = s^{(j)}(x_i^+) \quad j = 0, \dots, k-1 \quad (3.1)$$

$s^{(j)}$  denoting the  $j$ th derivative of the spline  $s$ . A basis of that set  $\mathcal{S}$  is

$$\{x^0, \dots, x^k, (x - x_1)_+^k, \dots, (x - x_{m-1})_+^k\} \quad (3.2)$$

where  $(x - x_j)_+^k = (\max\{0, x - x_j\})^k$ .

**Theorem 3.1.** The dimension of the linear space  $\mathcal{S}(k; x_0, \dots, x_m)$  is  $m + k$ .

## 3.2 Linear interpolating splines

Suppose  $f \in \mathcal{C}^0[a, b]$ . Let  $K = \{x_0, \dots, x_m\}$  such that  $a = x_0 < x_1 < \dots < x_m = b$ ,  $m \geq 2$ . The linear spline  $s_L$ , interpolating  $f$  at the points  $x_i$ , is defined by

$$s_L(x) = \frac{x_i - x}{x_i - x_{i-1}} f(x_{i-1}) + \frac{x - x_{i-1}}{x_i - x_{i-1}} f(x_i) \quad x \in [x_{i-1}, x_i], \quad i = 1, \dots, m \quad (3.3)$$

The points  $x_i$  are the knots.

**Theorem 3.2.** Suppose that  $f \in \mathcal{C}^2[a, b]$  and let  $s_L$  be the linear spline that interpolates  $f$  at the knots. Then, the error is bounded:

$$\|f - s_L\|_\infty \leq \frac{1}{8} h^2 \|f''\|_\infty \quad (3.4)$$

where  $h = \max_i (x_i - x_{i-1})$ .

### 3.3 B-splines

The basis defined above is not well suited for computation, we will instead use the B-splines. These are functions  $\phi$  such that

$$\phi(x) = 0 \quad \forall x \in [x_0, x_p] \cup [x_q, x_m] \quad (3.5)$$

with  $0 < p < q < m$  and  $q - p$  as small as possible. We will use  $\phi$  of the form

$$\phi(x) = \sum_{j=p}^q d_j (x - x_j)_+^k, \quad a \leq x \leq b \quad (3.6)$$

where the parameters  $d_j$  satisfy

$$r_k(x) := \sum_{j=p}^q d_j (x - x_j)^k = 0, \quad x_q \leq x \leq b \quad (3.7)$$

Playing with arithmetics and algebra, we finally get the general formula for a B-spline:

$$B_p(x) = \sum_{j=p}^{p+k+1} \left( \prod_{\substack{l=p \\ l \neq j}}^{p+k+1} \frac{1}{x_l - x_j} \right) (x - x_j)_+^k, \quad x \in \mathbb{R} \quad (3.8)$$

It belongs to  $\mathcal{S}$  and verifies the condition (3.5). It is well-defined for  $(p = 0, \dots, m - k - 1)$  and thus gives  $m - k$  B-splines. To define a basis of  $\mathcal{S}$ , we need  $2k$  more functions. We are going to add  $k$  knots on the left of  $x_0$  and on the right of  $x_m$ :

$$x_{-k} < x_{-k+1} < \dots, x_{-1} < x_0 = a < x_1 < \dots < x_m = b < x_{m+1} < \dots < x_{m+k} \quad (3.9)$$

and we will now define  $B_{-k}, \dots, B_{m-1}$  on these dots. We now have  $m + k$  linearly independent functions, and thus a basis of  $\mathcal{S}$ .

**Theorem 3.3.** Let  $x_{-k}, \dots, x_{m+k}$  satisfy (3.9). Then, the  $m + k$  functions  $B_p, p = -k, \dots, m - 1$ , given by (3.8) form a basis of the space  $\mathcal{S}(k; x_0, x_m)$ , with small support, meaning that  $B_p$  is null outside the interval  $(x_p, x_{p+k+1})$ .

The recurrence formula for B-splines is the following, for  $k > 1$ :

$$\begin{cases} B_p^k(x) = \frac{(x - x_p)B_p^{k-1}(x) + (x_{p+k+1} - x)B_{p+1}^{k-1}(x)}{x_{p+k+1} - x_p} \\ B_p^0(x) = 1_{[x_p, x_{p+1})} \end{cases} \quad (3.10)$$

### 3.4 Regression with splines

Let  $B_{-k}, \dots, B_{m-1}$  be a basis of the linear space of splines  $\mathcal{S}(k; u_0, \dots, u_m)$ . We have a function  $f \in \mathcal{C}[a, b]$  and sampling points  $w_0, \dots, w_q$ , assuming  $q + 1 \geq k + m$ . The goal of this section is to find a spline function  $s \in \mathcal{S}$  that is the closest to the data points  $(w_i, f(w_i)), i = 0, \dots, q$ , i.e.

$$\arg \min_{s \in \mathcal{S}} \sum_{i=0}^q |f(w_i) - s(w_i)|^2 \quad (3.11)$$

Using  $s = \sum_{j=-k}^{m-1} c_j B_j$ , we must solve the system

$$\underbrace{\begin{bmatrix} B_{-k}(w_0) & \dots & B_{m-1}(w_0) \\ \vdots & \ddots & \vdots \\ B_{-k}(w_q) & \dots & B_{m-1}(w_q) \end{bmatrix}}_{=:A} \underbrace{\begin{bmatrix} c_{-k} \\ \vdots \\ c_{m-1} \end{bmatrix}}_{=:c} = \underbrace{\begin{bmatrix} f(w_0) \\ \vdots \\ f(w_q) \end{bmatrix}}_{=:F} \quad (3.12)$$

This is solved using the normal equations:  $A^T A c = A^T F$ .

**Theorem 3.4.** Under the above assumptions, the columns of  $A$  are linearly independent iff there exists a subset of  $m + k$  sampling times  $w_{i-k} < \dots < w_{i_{m-1}}$  such that

$$u_p < w_{i_p} < u_{p+k+1} \quad p = -k, \dots, m-1 \quad (3.13)$$

meaning that  $w_{i_p}$  must be in the support of  $B_p$ .

### 3.5 Interpolation by natural cubic splines

Let us define the set of cubic splines  $\mathcal{S}(k = 3; \xi_0, \dots, \xi_m)$ . The set of natural cubic splines with those knots is the set

$$\mathcal{S}_N(k = 3; \xi_0, \dots, \xi_m) = \{s \in \mathcal{C}^2[\xi_0, \xi_m] : s|_{[\xi_{i-1}, \xi_i]} \in \mathcal{P}_3, i = 1, \dots, m \text{ and } s''(\xi_0) = s''(\xi_m) = 0\} \quad (3.14)$$

For an arbitrary piece  $[\xi_{i-1}, \xi_i]$ , we have 4 conditions:

- $s|_{[\xi_{i-1}, \xi_i]}(\xi_{i-1}) = s_{i-1}$
- $s|_{[\xi_{i-1}, \xi_i]}(\xi_i) = s_i$
- $s''|_{[\xi_{i-1}, \xi_i]}(\xi_{i-1}) = \sigma_{i-1}$
- $s''|_{[\xi_i, \xi_{i+1}]}(\xi_i) = \sigma_i$

And we thus write

$$s|_{[\xi_{i-1}, \xi_i]} = s_{i-1}A(x) + s_iB(x) + \sigma_{i-1}C(x) + \sigma_iD(x) \quad (3.15)$$

where all functions are  $\in \mathcal{P}_3$  and satisfy 4 conditions themselves:

$A(\xi_{i-1}) = 1$	$B(\xi_{i-1}) = 0$	$C(\xi_{i-1}) = 0$	$D(\xi_{i-1}) = 0$
$A(\xi_i) = 0$	$B(\xi_i) = 1$	$C(\xi_i) = 0$	$D(\xi_i) = 0$
$A''(\xi_{i-1}) = 0$	$B''(\xi_{i-1}) = 0$	$C''(\xi_{i-1}) = 1$	$D''(\xi_{i-1}) = 0$
$A''(\xi_i) = 0$	$B''(\xi_i) = 0$	$C''(\xi_i) = 0$	$D''(\xi_i) = 1$

Defining  $h_i = \xi_i - \xi_{i-1}$ , the final formula is

$$s(x) = \frac{(x - \xi_{i-1})s_i + (\xi_i - x)s_{i-1}}{h_i} \quad (3.16)$$

$$- \frac{1}{6}(x - \xi_{i-1})(\xi_i - x) \left[ \left(1 + \frac{x - \xi_{i-1}}{h_i}\right) \sigma_i + \left(1 + \frac{\xi_i - x}{h_i}\right) \sigma_{i-1} \right] \quad x \in [\xi_{i-1}, \xi_i] \quad (3.17)$$

Now, we have the additional conditions that  $s'(\xi_j^-) = s'(\xi_j^+)$ ,  $j = 1, \dots, m-1$ , which we write in the following matrix form:

$$Q^T s = R\sigma \quad (3.18)$$

where the matrices are:

$$Q^T = \begin{pmatrix} h_1^{-1} & -h_1^{-1} - h_2^{-1} & h_2^{-1} & 0 & \dots & 0 \\ 0 & h_2^{-1} & -h_2^{-1} - h_3^{-1} & h_3^{-1} & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & h_{m-1}^{-1} & -h_{m-1}^{-1} - h_m^{-1} & h_m^{-1} \end{pmatrix} \quad (3.19)$$

$$R = \begin{pmatrix} \frac{1}{3}(h_1 + h_2) & \frac{h_2}{6} & 0 & \dots & 0 \\ \frac{h_2}{6} & \frac{1}{3}(h_2 + h_3) & \frac{h_3}{6} & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \frac{h_{m-1}}{6} & \frac{1}{3}(h_{m-1} + h_m) \end{pmatrix} \quad (3.20)$$

**Theorem 3.5.**  $s \in \mathcal{S}_N(k=3; \xi_0, \dots, \xi_m)$  iff  $Q^T s = R\sigma$ .

**Theorem 3.6.** Consider  $\xi_0, \dots, x_m$  distinct and  $y_0, \dots, y_m$ . The interpolation at the knots

$$s \in \mathcal{S}_N(k=3; \xi_0, \dots, \xi_m) \text{ such that } s(\xi_i) = y_i \quad i = 0, \dots, m \quad (3.21)$$

exists and is unique.

**Theorem 3.7.** Let  $s$  be a natural cubic spline. Then,

$$\int_{\xi_0}^{\xi_m} (s''(x))^2 dx = s^T K s \quad K = QR^{-1}Q^T \quad (3.22)$$

**Theorem 3.8.** Let  $s$  be the function in  $\mathcal{S}_N(k=3; \xi_0, \dots, \xi_m)$  such that  $s(\xi_i) = y_i$ ,  $i = 0, \dots, m$ . Let  $v$  be any function in  $H^2[a, b]$  that satisfies the same interpolation conditions. Then

$$\int_{\xi_0}^{\xi_m} (v''(x))^2 dx \geq \int_{\xi_0}^{\xi_m} (s''(x))^2 dx \quad (3.23)$$

with equality iff  $v = s$ .

### 3.6 Hermite cubic splines

We define the Hermite cubic spline as a function  $s \in \mathcal{C}^1[a, b]$  such that

- $s(x_i) = f(x_i)$ ,  $s'(x_i) = f'(x_i)$  for  $i = 0, \dots, m$ ;
- $s$  is a cubic polynomial on  $[x_{i-1}, x_i]$  for  $i = 1, \dots, m$ ;

It can be expressed as

$$s(x) = c_0 + c_1(x - x_{i-1}) + c_2(x - x_{i-1})^2 + c_3(x - x_{i-1})^3 \quad x \in [x_{i-1}, x_i] \quad (3.24)$$



with

$$\begin{aligned}
c_0 &= f(x_{i-1}) \\
c_1 &= f'(x_{i-1}) \\
c_2 &= 3 \frac{f(x_i) - f(x_{i-1})}{h_i^2} - \frac{f'(x_i) - 2f'(x_{i-1})}{h_i} \\
c_3 &= \frac{f'(x_i) - f'(x_{i-1})}{h_i^2} - 2 \frac{f(x_i) - f(x_{i-1})}{h_i^3}
\end{aligned} \tag{3.25}$$

**Theorem 3.9.** Let  $f \in C^4[a, b]$ , and let  $s$  be the Hermite cubic spline that interpolates  $f$  at the knots. Then, the error is bounded:

$$\|f - s\|_\infty \leq \frac{1}{384} h^4 \|f^{(4)}\|_\infty \tag{3.26}$$

### 3.7 Smoothing splines

The problem studied in this section is

$$\arg \min_{s \in H^2[a, b]} F_\lambda(s) := \sum_{i=0}^m (y_i - s(x_i))^2 + \lambda \int_a^b (s''(x))^2 dx \tag{3.27}$$

where  $a = x_0 < x_1 < \dots < x_m = b$ ,  $y_i \in \mathbb{R}$  are given and  $\lambda > 0$  is a parameter. The first term is the data-attachment and the second is the roughness penalty.

**Theorem 3.10.** If  $\hat{s}$  is a solution of (3.27), then  $\hat{s} \in \mathcal{S}_N(k = 3; x_0, \dots, x_m)$ .

To find the solution of (3.27), we can rewrite the function to minimize:

$$F_\lambda(s) = (y - s)^T (y - s) + \lambda s^T K s \tag{3.28}$$

This function is strictly convex and quadratic and thus  $s$  is the solution of the linear system

$$(I + \lambda K)s = y \tag{3.29}$$

Meaning that (3.27) has one and only one solution. The easiest way to compute  $s$  is

$$s = y - \lambda Q \sigma \tag{3.30}$$

- When  $\lambda \rightarrow 0$ , we get a simple interpolation problem and there is an infinity of solutions.
- When  $\lambda \rightarrow \infty$ , we find the linear regression solution.

### 3.8 Error bounds of interpolation by natural splines

**Theorem 3.11.** Let  $s$  be the natural cubic spline interpolant of  $f \in C^4[a, b]$ , where the interpolation is at equally spaced knots. Then,

$$\|(f - s)^{(r)}\|_\infty \leq C_r \|f^{(4)}\|_\infty h^{4-r} \quad r = 0, 1, 2, 3 \tag{3.31}$$

with  $C_0 = 5/384, C_1 = 1/24, C_2 = 3/8, C_3 = 1$  and  $h$  the space between two knots. This means that the error between the interpolation and the function tends to 0 as the number of interpolation points goes to infinity.

### 3.9 Vector-valued splines

Vector-valued splines just work component-wise:

$$\mathbf{s}(x) = \sum_{j=-k}^{m-1} \mathbf{c}_j B_j(x) \quad (3.32)$$

where  $x$  and  $B_j(x)$  are real and scalar, but  $\mathbf{s}$  and  $\mathbf{c}_j$  are in  $\mathbb{R}^n$  for all  $j$ . We define the Bernstein polynomials of degree  $n$  as

$$b_i^n(x) = \binom{n}{i} x^i (1-x)^{n-i} \quad i = 0, \dots, n \quad (3.33)$$

They form a basis of  $\mathcal{P}_n$  and a partition of unity, i.e.  $\sum_{i=0}^n b_i^n(x) = 1$ . We can thus write any polynomial piece  $\mathbf{s}|_{[x_j, x_{j+1}]} \in \mathcal{P}_k$  in the Bernstein form:

$$\mathbf{s}|_{[x_j, x_{j+1}]} = \sum_{i=0}^k \mathbf{c}_i^j b_i^k \frac{x - x_j}{x_{j+1} - x_j} \quad (3.34)$$

aka the Bézier curve on  $[x_j, x_{j+1}]$  with control points  $\mathbf{c}_0^j, \dots, \mathbf{c}_k^j$ .

### 3.10 Tensor product splines

In the last section, thin plate splines are not piecewise polynomials. If we want the admissible functions restrict to piecewise polynomials, we need other approaches, as the tensor product splines.

**Definition 3.12. Tensor product** Let  $\mathcal{R}$  and  $\mathcal{S}$  be two linear spaces of real-valued univariate functions. The tensor product of two functions  $r \in \mathcal{R}$  and  $s \in \mathcal{S}$  is  $r \otimes s : \mathbb{R}^2 \rightarrow \mathbb{R} : (t_1, t_2) \rightarrow r(t_1)s(t_2)$ . The tensor product of the two spaces is the linear space

$$\mathcal{R} \otimes \mathcal{S} = \left\{ \sum_{i,j} c_{ij} r_i \otimes s_j \mid c_{ij} \in \mathbb{R} \right\} \quad (3.35)$$

where  $(r_1, \dots, r_{N_1})$  is a basis of  $\mathcal{R}$  and  $(s_1, \dots, s_{N_2})$  is a basis of  $\mathcal{S}$ .

Let us define the sets

$$\Delta = \{a = x_0 < x_1 < \dots < x_m = b\} \quad (3.36)$$

$$\bar{\Delta} = \{\bar{a} = y_0 < y_1 < \dots < y_{\bar{m}} = \bar{b}\} \quad (3.37)$$

and let  $\mathcal{S}_k(\Delta) = \mathcal{S}(k; x_0, \dots, x_m)$ , same for  $\bar{\Delta}$ . In this section, the set of admissible functions is

$$\mathcal{S}_k(\Delta) \otimes \mathcal{S}_{\bar{k}}(\bar{\Delta}) \quad (3.38)$$

The functions of that set are called tensor product splines of degree  $(k, \bar{k})$  and the points in  $\Delta, \bar{\Delta}$  are the knots. The dimension of the space is  $(m+k)(\bar{m}+\bar{k})$ , and a basis is given by

$$B_{ij}^{k,\bar{k}}(x, y) := B_i^k(x) B_j^{\bar{k}}(y) \quad i \in \{-k, \dots, m-1\}, j \in \{-\bar{k}, \dots, \bar{m}-1\} \quad (3.39)$$

where  $B$  functions are B-splines creating a basis of their respective  $\mathcal{S}_k(\Delta), \mathcal{S}_{\bar{k}}(\bar{\Delta})$ . Any admissible function can thus be uniquely written as a linear combination of those.

### 3.10.1 Matrix form

Any tensor-product spline  $s$  can be written in matrix form:

$$s(x, y) = [B_{-k}^k(x) \quad \dots \quad B_{m-1}^k(x)] C \begin{bmatrix} B_{-k}^{\bar{k}}(y) \\ \vdots \\ B_{\bar{m}-1}^{\bar{k}}(y) \end{bmatrix} \quad C = \begin{bmatrix} c_{-k, -\bar{k}} & \dots & c_{-k, \bar{m}-1} \\ \vdots & \ddots & \vdots \\ c_{m-1, -\bar{k}} & \dots & c_{m-1, \bar{m}-1} \end{bmatrix} \quad (3.40)$$

### 3.10.2 Interpolation and regression

Consider  $t_1 < \dots < t_n$  in  $[a, b]$  and  $\bar{t}_1 < \dots < \bar{t}_{\bar{n}}$  in  $[\bar{a}, \bar{b}]$ , and suppose that we are given real number  $\{z_{ij}\}_{i=1, j=1}^{n, \bar{n}}$ . The interpolation problem is to find a tensor product spline  $s$  such that

$$s(t_i, \bar{t}_j) = z_{ij} \quad i = 1, \dots, n, \quad j = 1, \dots, \bar{n} \quad (3.41)$$

In matrix form, this system of equations is

$$M C \bar{M}^T = Z = \begin{bmatrix} z_{1,1} & \dots & z_{1,\bar{n}} \\ \vdots & \ddots & \vdots \\ z_{n,1} & \dots & z_{n,\bar{n}} \end{bmatrix} \quad (3.42)$$

$$M = \begin{bmatrix} B_{-k}^k(t_1) & \dots & B_{m-1}^k(t_1) \\ \vdots & \ddots & \vdots \\ B_{-k}^k(t_n) & \dots & B_{m-1}^k(t_n) \end{bmatrix} \quad \bar{M} = \begin{bmatrix} B_{-k}^{\bar{k}}(\bar{t}_1) & \dots & B_{\bar{m}-1}^{\bar{k}}(\bar{t}_1) \\ \vdots & \ddots & \vdots \\ B_{-k}^{\bar{k}}(\bar{t}_{\bar{n}}) & \dots & B_{\bar{m}-1}^{\bar{k}}(\bar{t}_{\bar{n}}) \end{bmatrix} \quad (3.43)$$

Solving this in the least square sense, the normal equations are

$$M^T M C \bar{M}^T \bar{M} = M^T Z \bar{M} \quad (3.44)$$

# Numerical integration

## 4.1 Newton-Cotes formulae

Functions can be difficult to integrate. This is why we use polynomial approximations to compute it. One way is to use the interpolations seen before:

$$\int_a^b f(x)dx \approx \int_a^b p_n(x)dx \quad (4.1)$$

where  $p_n(x)$  is the Lagrange interpolation polynomial of  $f$  on  $[a, b]$ . Assuming that the interpolation points are equally spaced, using equation (1.2),

$$\int_a^b f(x)dx \approx \sum_{k=0}^n w_k f(x_k) \quad w_k = \int_a^b L_k(x)dx \quad k = 0, \dots, n \quad (4.2)$$

and the  $w_k$  are called the quadrature weights.

We call the Newton-Cotes formula with  $n = 1$  the trapezium rule:  $\int_a^b f(x)dx \approx \frac{b-a}{2}[f(a) + f(b)]$ . For  $n = 2$ , it is called Simpson's rule.

In general, we can make the change of variable  $x(t) = \frac{b-a}{2}t + \frac{b+a}{2}$  to reduce the integration interval to  $[-1, 1]$ .

→ Note: the weight values do not depend on the function to integrate.

### 4.1.1 Error estimation

**Theorem 4.1.** Let  $n \geq 1$ . Suppose that  $f : \mathbb{R} \mapsto \mathbb{R}$ ,  $f \in \mathcal{C}^{n+1}(a, b)$ . Then, the error  $E_n(f) := \int_a^b f(x)dx - \sum_{k=0}^n w_k f(x_k)$  is bounded:

$$|E_n(f)| \leq \frac{M_{n+1}}{(n+1)!} \int_a^b |\pi_{n+1}(x)|dx \quad (4.3)$$

where  $M_{n+1} = \max_{\zeta \in [a, b]} |f^{(n+1)}(\zeta)|$ .

→ Note: some Runge phenomenon can appear like in the case of interpolation, depending on the integration points  $x_i$ .

## 4.2 Composite formulae

To avoid Runge phenomenon, we usually use Newton-Cotes with very low degree ( $n = 1$  or  $2$ ). We rather divide the interval into  $m$  equal subintervals of width  $h = (b - a)/m$ :

$$\int_a^b f(x)dx = \sum_{i=1}^m \int_{x_{i-1}}^{x_i} f(x)dx \quad x_i = a + ih \quad i = 0, \dots, m \quad (4.4)$$

The integral over  $[x_i, x_{i+1}]$  is approximated by the trapezium rule or Simpson's rule. Using the usual formula for the bound on the error, we have for the trapezium rule

$$|E_1(f)| \leq \frac{1}{12}h^3 \sum_{i=1}^m \left( \max_{\xi \in [x_{i-1}, x_i]} |f''(\xi)| \right) \leq \frac{(b-a)^3}{12m^2} M_2 \quad (4.5)$$

The same reasoning can be done on the Simpson rule, and we find the bound

$$|E_2(f)| \leq \frac{(b-a)^5}{2880m^4} M_4 \quad (4.6)$$

→ Note: the error of the composite trapezium rule is in  $\mathcal{O}(1/m^2)$ , but can be improved by extrapolation methods.

## 4.3 Gauss quadrature

Gauss quadrature consists in approximating the integral of  $f$  over the interval  $[a, b]$  by the integral of its Hermite polynomial on the same interval.

$$\int_a^b w(x)f(x)dx \approx \int_a^b w(x)p_{2n+1}(x)dx = \sum_{k=0}^n W_k f(x_k) + \sum_{k=0}^n V_k f'(x_k) \quad (4.7)$$

where

$$W_k = \int_a^b w(x)H_k(x)dx \quad V_k = \int_a^b w(x)K_k(x)dx \quad (4.8)$$

We want to choose the points  $x_k$  such that all  $V_k$  are zero and the function  $f'$  does not appear.

$$\int_a^b w(x)f(x)dx \approx \mathcal{G}_n(f) = \sum_{k=0}^n W_k f(x_k) \quad (4.9)$$

The quadrature points are chosen as the zeros of the polynomial of degree  $n + 1$  from a system of orthogonal polynomials over the interval  $(a, b)$  with respect to the weight function  $w$ , i.e. they are the zeros of  $\phi_{n+1}(x)$ . This gives the formula

$$W_k = \int_a^b w(x)[L_k(x)]^2 dx \quad (4.10)$$

**Theorem 4.2.** Suppose that the weight function  $w \in \mathcal{C}[a, b]$  is integrable and positive on  $[a, b]$ , and that  $f \in \mathcal{C}^0[a, b]$ . Then,

$$\lim_{n \rightarrow \infty} \mathcal{G}_n(f) = \int_a^b w(x)f(x)dx \quad (4.11)$$