2 Spline Interpolation

Polynomial interpolation has a major setback: the difference between the values of the function f and the values of the interpolation polynomial outside the nodes' interval can be quite large. Choosing more nodes and finding a higher degree polynomial does not solve this problem, but increases the computational cost. So, even though polynomials are smooth and easy to work with functions, they are not always the best choice for approximating functions.

From these considerations came the idea of changing polynomials to *piecewise polynomials* that satisfy some continuity conditions (of the interpolation function and some of its derivatives). Such functions are called *splines*.

Historically, spline functions can be traced all the way back to ancient mathematics. The term "spline" was first used by I. J. Schoenberg in 1946, but a thorough spline function theory started developing in 1964, as their good approximating properties became more evident. They can be used in a large variety of ways in approximation theory, computer graphics, data fitting, numerical integration and differentiation, and the numerical solution of integral, differential, and partial differential equations.

Over time, there have been several world renowned research groups in spline theory, scattered all over the world. One such group, with remarkable contributions, was a Romanian research group (based especially in Cluj).

The basic idea of approximating a function on an interval $[a, b]$ with spline functions, is to use different polynomials (of lower degree) on different parts of the interval. The reason for this is the fact that on a sufficiently small interval, functions can be approximated arbitrarily well by polynomials of low degree, even degree 1, or 0.

Definition 2.1. *Let Δ be a grid of the interval $[a, b]$,*

$$\Delta \; : \; a = x_1 < x_2 < \cdots < x_{n-1} < x_n = b.$$

The set

$$\mathbb{S}_m^k(\Delta) \quad = \quad \{s \mid s \in C^k[a, b],\ s|_{[x_i, x_i+1]} \in \mathbb{P}_m,\ i = 1, 2, \ldots, n-1\} \tag{2.1}$$

*is called the **the space of polynomial spline functions** of degree m and class k on Δ.*

These are piecewise polynomial functions, of degree $\leq m$, continuous at $x_1, \ldots, x_{n-1}$, together with all their derivatives of order up to k. In general, we assume $0 \le k < m$. For $k = m$,

$$S_m^m \;=\; \mathbb{P}_m.$$

If $k = -1$, we allow discontinuities at the grid points.

2.1 Linear Splines

For $m = 1$ and $k = 0$, we have *linear spline functions*. We determine a function $s_1 \in \mathbb{S}_1^0(\Delta)$ such that

$$s_1(x_i) \;=\; f(x_i) \;=\; f_i,\; i = 1, 2, \ldots, n.$$

That means that on the interval $[x_i, x_{i+1}]$, the function s_1 is the interpolation polynomial of degree 1

$$s_1(f; x) \;=\; f_i + f[x_i, x_{i+1}](x - x_i) \;=\; f_i + \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}(x - x_i). \tag{2.2}$$

The graph of this function is shown in Figure 1.
The error is given by

$$|f(x) - s_1(f; x)| \;\le\; \frac{|(x - x_i)(x - x_{i+1})|}{2!} \max_{x \in [x_i, x_{i+1}]} |f''(x)| \le \frac{h_i^2}{8} \max_{x \in [x_i, x_{i+1}]} |f''(x)|, \tag{2.3}$$

where we denoted by $h_i = x_{i+1} - x_i$.
Hence, if $|\Delta|$ denotes

$$|\Delta| \;=\; \max_{i=\overline{1,n-1}} h_i,$$

we have

$$||f(\cdot) - s_1(f, \cdot)||_\infty \;\;\le\;\; \frac{|\Delta|^2}{8} ||f''||_\infty. \tag{2.4}$$

Obviously, $\mathbb{S}_1^0(\Delta)$ is a vector space. To find its dimension, we count the number of degrees of freedom and the number of constraints. There are $n - 1$ subintervals and 2 coefficients to be determined (i.e. 2 degrees of freedom) on each, for a total of $2(n-1)$. We have continuity conditions at each interior node, so $n - 2$ constraints. Thus, in the end we have

$$\dim \mathbb{S}_1^0(\Delta) \;\;=\;\; 2(n-1) - (n-2) \;=\; n.$$

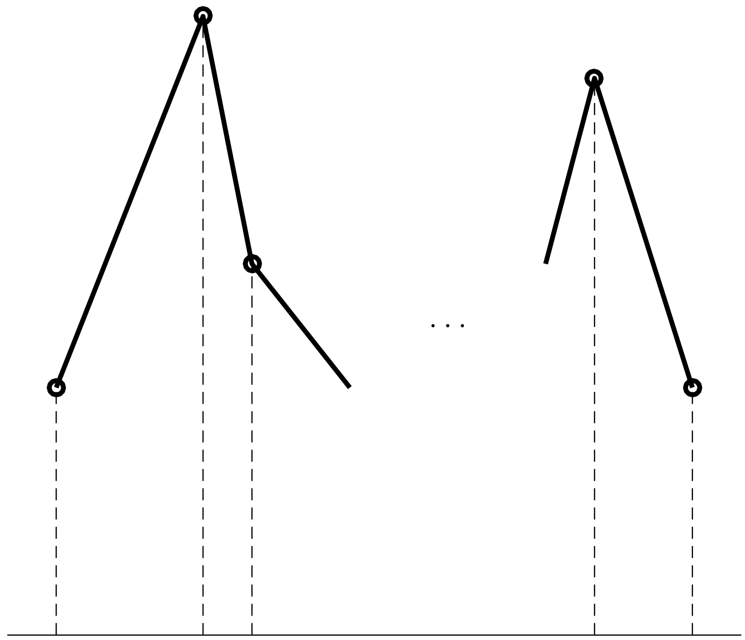

Fig. 1: Linear splines

A basis for this space is given by the so-called *B-spline functions*. Taking $x_0 = x_1 = a,\ x_{n+1} = x_n = b$, for $i = \overline{1, n}$, we define

$$B_i(x) = \begin{cases} \dfrac{x - x_{i-1}}{x_i - x_{i-1}}, & \text{If } x_{i-1} \le x \le x_i \\ \dfrac{x_{i+1} - x}{x_{i+1} - x_i}, & \text{If } x_i \le x \le x_{i+1} \\ 0, & \text{in rest.} \end{cases} \tag{2.5}$$

Note that the first equation for $i = 1$, and the second equation for $i = n$, are to be ignored. The functions B_i are sometimes referred to as "hat functions" (Chinese hats), but note that the first and the last hat are cut in half. Their graphs are depicted in Figure 2. They are linearly independent and have the property

$$B_i(x_j) = \delta_{ij}.$$

Any function $s \in \mathbb{S}_1^0(\Delta)$ can be written uniquely as

$$s(x) \;=\; \sum_{i=1}^{n} c_i B_i(x).$$

B-spline functions play the same role as fundamental Lagrange polynomials l_i.

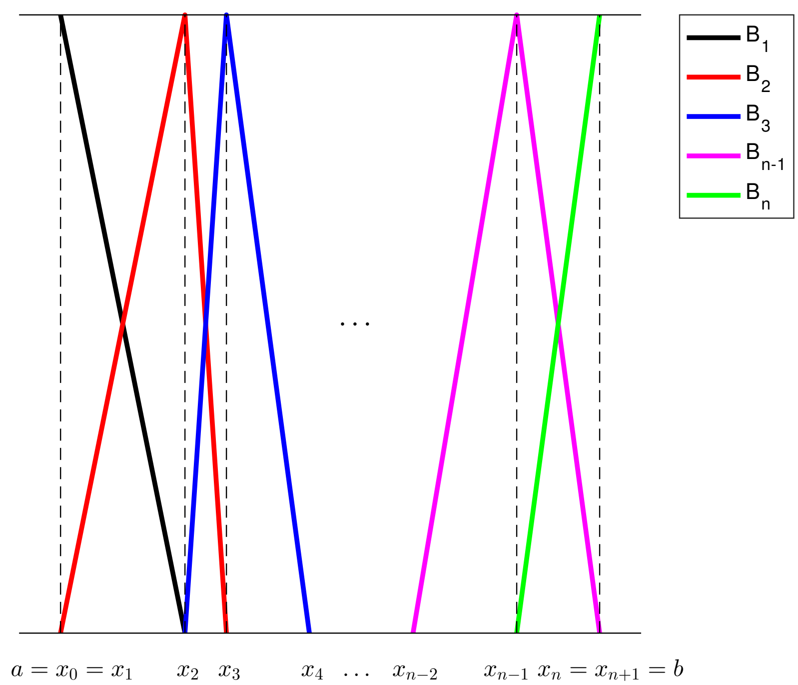

Fig. 2: Linear B-splines

A linear spline agrees with the data, but it has the disadvantage of not having a smooth graph. Most data will represent a smooth curved graph, one without the corners of a linear spline. Consequently, we usually want to construct a smooth curve that interpolates the given data points, but one that follows the shape of the linear spline.

2.2 Cubic Splines

Cubic splines are the most widely used. In general, cubic splines are fairly smooth functions that are convenient to work with, and they have come to be widely used in the past several decades in computer graphics and in many areas of Applied Mathematics and Statistics. There are several types of cubic spline functions, depending on the smoothness conditions they satisfy.

Interpolation with cubic splines $s \in \mathbb{S}_3^1(\Delta)$

We impose the continuity of the first order derivative of $s_3(f;\cdot)$ by prescribing the values of the first derivative at each node $x_i, i = 1, 2, \ldots, n$. Given n arbitrary numbers $m_1, m_2, \ldots, m_n$, we seek a function $s_3(f;\cdot)$ that satisfies the conditions

$$\begin{aligned} s_3|_{[x_i, x_i+1]} &= p_i(x) \in \mathbb{P}_3,\ i = 1, 2, \ldots, n-1, \\ s_3(f; x_i) &= f_i,\ i = 1, 2, \ldots, n, \\ s_3'(f; x_i) &= m_i,\ i = 1, 2, \ldots, n. \end{aligned} \tag{2.6}$$

This means that on each subinterval $[x_i, x_{i+1}]$, $s_3(f;\cdot)$ is the unique solution of the Hermite interpolation problem

$$\begin{aligned} p_i(x_i) &= f_i, \quad p_i(x_{i+1}) = f_{i+1}, \\ p_i'(x_i) &= m_i, \quad p_i'(x_{i+1}) = m_{i+1},\ i = \overline{1, n-1}. \end{aligned} \tag{2.7}$$

The divided differences are computed from the table

$$\begin{array}{llllllll}
x_i & f_i & \longrightarrow & m_i & \longrightarrow & \dfrac{f[x_i, x_{i+1}] - m_i}{h_i} & \longrightarrow & \dfrac{m_{i+1} - 2f[x_i, x_{i+1}] + m_i}{h_i^2} \\
 & & \nearrow & & \nearrow & & \nearrow & \\
x_i & f_i & \longrightarrow & f[x_i, x_{i+1}] & \longrightarrow & \dfrac{m_{i+1} - f[x_i, x_{i+1}]}{h_i} & & \\
 & & \nearrow & & \nearrow & & & \\
x_{i+1} & f_{i+1} & \longrightarrow & m_{i+1} & & & & \\
 & & \nearrow & & & & & \\
x_{i+1} & f_{i+1}, & & & & & &
\end{array}$$

Using the Newton form of the Hermite polynomial, we have

$$\begin{aligned} p_i(x) &= f_i + m_i(x - x_i) + \frac{f[x_i, x_{i+1}] - m_i}{h_i}(x - x_i)^2 \\ &+ \frac{m_{i+1} - 2f[x_i, x_{i+1}] + m_i}{h_i^2}(x - x_i)^2(x - x_{i+1}). \end{aligned}$$

Alternatively, we can write it in Taylor's form around x_i. Considering that $x - x_{i+1} = x - x_i - h_i$, for $x \in [x_i, x_{i+1}]$, we get

$$p_i(x) = c_{i,0} + c_{i,1}(x - x_i) + c_{i,2}(x - x_i)^2 + c_{i,3}(x - x_i)^3, \tag{2.8}$$

with

$$
\begin{aligned}
c_{i,0} &= f_i, \\
c_{i,1} &= m_i, \\
c_{i,2} &= \frac{f[x_i, x_{i+1}] - m_i}{h_i} - c_{i,3} h_i \;=\; \frac{3f[x_i, x_{i+1}] - 2m_i - m_{i+1}}{h_i}, \\
c_{i,3} &= \frac{m_{i+1} - 2f[x_i, x_{i+1}] + m_i}{h_i^2}.
\end{aligned}
\tag{2.9}
$$

Hence, to compute $s_3(f;x)$ at a point $x \in [a,b]$ that is not a node, we first identify the interval $[x_i, x_{i+1}]$ that contains x, then compute the coefficients in (2.9) and evaluate the spline using (2.8).

Next, we discuss some possible choices for the parameters $m_1, m_2, \ldots, m_n$.

Piecewise cubic Hermite interpolation

Assuming that the derivatives $f'(x_i),\ i = 1, \ldots, n$, are known, we choose $m_i = f'(x_i)$. This way, we obtain a strictly local scheme, where the polynomial on each subinterval $[x_i, x_{i+1}]$ is completely determined by the interpolation data at node points inside, independently of the other pieces. The error in this case (see Example 1.4, in Lecture 4) is

$$
||f(\cdot) - s_3(f, \cdot)||_\infty \;\leq\; \frac{1}{384}\, |\Delta|^4\, ||f^{(4)}||_\infty. \tag{2.10}
$$

For equally spaced nodes, we have

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

and, therefore,

$$
||f(\cdot) - s_3(f, \cdot)||_\infty \;=\; O(n^{-4}),\ n \to \infty. \tag{2.11}
$$

Interpolation with cubic splines $s \in \mathbb{S}_3^2(\Delta)$

To have $s_3(f;\cdot) \in S_3^2(\Delta)$, we require continuity of the second derivatives at the nodes, i.e.

$$
p''_{i-1}(x_i) \;=\; p''_i(x_i),\ i = 2, \ldots, n-1,
$$

which, for the Taylor coefficients in (2.8), means

$$
2c_{i-1,2} \;+\; 6c_{i-1,3} h_{i-1} \;=\; 2c_{i,2},\ i = 2, \ldots, n-1. \tag{2.12}
$$

Substituting in (2.9), we obtain the linear system

$$h_i m_{i-1} \ + \ 2(h_{i-1}+h_i)m_i \ + \ h_{i-1}m_{i+1} \quad = \quad b_i, \ i = 2, \ldots, n-1, \tag{2.13}$$

where

$$b_i \quad = \quad 3\Big(h_i f[x_{i-1}, x_i] + h_{i-1} f[x_i, x_{i+1}]\Big). \tag{2.14}$$

Thus, we have a system of $n-2$ linear equations with n unknowns, $m_1, m_2, \ldots, m_n$. Once m_1 and m_n are chosen, the system is tridiagonal and can be solved efficiently by several methods.

Next, we discuss possible choices for m_1 and m_n.

1. **Complete (clamped) splines.** We take

   $$m_1 = f'(a), \ m_n = f'(b).$$

   For this type of spline, it can be shown that, if $f \in C^4[a,b]$, then

   $$||f^{(r)}(\cdot) - s_3^{(r)}(f, \cdot)||_\infty \quad \leq \quad C_r |\Delta|^{4-r} \ ||f^{(4)}||_\infty, \ r = 0, 1, 2, 3, \tag{2.15}$$

   where
   $$C_0 \ = \frac{5}{384}, \ C_1 \ = \ \frac{1}{24}, \ C_2 \ = \ \frac{3}{8},$$
   and C_3 depends on the ratio $|\Delta| / \min\limits_i h_i$.

2. **Endpoint second derivative splines.** We require

   $$s_3''(f, a) \ = \ f''(a), \ s_3''(f, b) \ = \ f''(b).$$

   These lead to two more equations,

   $$\begin{aligned} 2m_1 \quad + \quad m_2 \qquad\qquad\qquad &= \ 3f[x_1, x_2] - \frac{1}{2} f''(a) h_1, \\ m_{n-1} \quad + \quad 2m_n \ &= \ 3f[x_{n-1}, x_n] - \frac{1}{2} f''(b) h_{n-1}. \end{aligned} \tag{2.16}$$

   We place the first equation at the beginning of the system (2.13) and the second at the end of it, thus preserving the tridiagonal structure of the system.

3. **Natural cubic splines.** Imposing

$$s_3''(f;a) \;=\; s_3''(f;b) \;=\; 0,$$

we get the same two equations as above, with $f''(a) = f''(b) = 0$:

$$\begin{array}{ccccccc} 2m_1 & + & m_2 & & & = & 3f[x_1, x_2], \\ & & & m_{n-1} & + \;\; 2m_n & = & 3f[x_{n-1}, x_n]. \end{array} \tag{2.17}$$

The advantage of this type of spline is that it requires only the function values of f – no derivatives – but the price paid is a decrease in the accuracy to $O(|\Delta|^2)$ near the endpoints (unless indeed $f''(a) = f''(b) = 0$).

4. **"Not-a-knot" (deBoor) splines.** Here we impose the conditions that the first two pieces and the last two, coincide, i.e.

$$p_1(x) \;\equiv\; p_2(x),\; p_{n-2}(x) \;\equiv\; p_{n-1}(x).$$

This means that the first interior node, x_2, and the last one, x_{n-1}, are both inactive (hence, the name). We get two more equations expressing the continuity of $s_3'''(f;x)$ at $x = x_2$ and $x = x_{n-1}$. This comes down to the equality of the leading coefficients $c_{1,3} \;=\; c_{2,3}$ and $c_{n-2,3} \;=\; c_{n-1,3}$. Thus, we get

$$\begin{array}{ccccccc} h_2^2\, m_1 & + & \big(h_2^2 - h_1^2\big) m_2 & - & h_1^2\, m_3 & = & \beta_1, \\ h_{n-1}^2\, m_{n-2} & + & \big(h_{n-1}^2 - h_{n-2}^2\big) m_{n-1} & - & h_{n-2}^2\, m_n & = & \beta_2, \end{array} \tag{2.18}$$

where

$$\begin{aligned} \beta_1 &= 2\Big(h_2^2 f[x_1, x_2] - h_1^2 f[x_2, x_3]\Big), \\ \beta_2 &= 2\Big(h_{n-1}^2 f[x_{n-2}, x_{n-1}] - h_{n-2}^2 f[x_{n-1}, x_n]\Big). \end{aligned}$$

Again, we place the first equation at the beginning of the system (2.13) and the second at the end of it. Even so, the resulting system is *no longer* tridiagonal, but it can be transformed into a tridiagonal one, by combining equations 1 and 2, and $n-1$ and n, respectively. Consequently, the first and the last equations become

$$
\begin{array}{lllllll}
h_2\, m_1 & + & \big(h_2+h_1\big)m_2 & & & = & \gamma_1, \\
& & & \big(h_{n-1}-h_{n-2}\big)m_{n-1} & + \quad h_{n-2}\, m_n & = & \gamma_2,
\end{array} \tag{2.19}
$$

where

$$
\begin{aligned}
\gamma_1 &= \frac{1}{h_2+h_1}\Big[f[x_1,x_2]h_2\Big(h_1+2\big(h_1+h_2\big)\Big)+h_1^2 f[x_2,x_3]\Big], \\
\gamma_2 &= \frac{1}{h_{n-1}+h_{n-2}}\Big[h_{n-1}^2 f[x_{n-2},x_{n-1}]+\Big(2\big(h_{n-1}+h_{n-2}\big)+h_{n-1}\Big)h_{n-2}f[x_{n-1},x_n]\Big].
\end{aligned}
$$

Finding cubic splines using the second derivatives

Computational formulas for finding cubic splines $s \in \mathbb{S}_3^2(\Delta)$ can be derived (in a similar way) when the arbitrary numbers $M_1, M_2, \ldots, M_n$ are given and forced to satisfy the conditions

$$
\begin{aligned}
s_3\big|_{[x_i,x_i+1]} &= p_i(x) \in \mathbb{P}_3,\ i=1,2,\ldots,n-1, \\
s_3(f;x_i) &= f_i,\ i=1,2,\ldots,n, \\
s_3''(f;x_i) &= M_i,\ i=1,2,\ldots,n.
\end{aligned} \tag{2.20}
$$

Since s_3 is a cubic polynomial, its second derivative is linear. Hence, on $[x_i, x_{i+1}]$, we have

$$
s_3''(f;x) = ax+b,
$$

satisfying the conditions

$$
s_3''(f;x_i) = M_i,\ s_3''(f;x_{i+1}) = M_{i+1},\ \ i=1,2,\ldots,n-1.
$$

The values a and b are determined from the system

$$
\left\{
\begin{array}{lllll}
ax_i & + & b & = & M_i \\
ax_{i+1} & + & b & = & M_{i+1}
\end{array}
\right. .
$$

Integrating successively, then imposing (2.20) and the continuity conditions at the nodes, $s_3'(f;x_i) = s_3'(f;x_{i+1}),\ i=\overline{1,n-1}$, we get the linear system

$$
h_{i-1}\, M_{i-1} \;+\; 2\left(h_{i-1}+h_i\right)M_i \;+\; h_i\, M_{i+1} \;=\; 6\left(f[x_i,x_{i+1}]-f[x_{i-1},x_i]\right), \tag{2.21}
$$

for $i=\overline{2,n-1}$.
The two extra conditions needed for a closed system can be imposed, e.g., on M_1 and M_n. If $M_1=M_n=0$, we get the natural cubic spline.
Other conditions can be enforced, such as the continuity of $s_3'''(f;x)$ at $x=x_2$ and $x=x_{n-1}$, which lead to deBoor cubic splines.
If the first and last equations are

$$\begin{array}{ccccccc} 2M_1 & + & M_2 & & & = & 6\left(f[x_1,x_2]-f_1'\right), \\ & & M_{n-1} & + & 2M_n & = & 6\left(f_n'-f[x_{n-1},x_n]\right), \end{array} \tag{2.22}$$

where $f_1' = f'(a), f_n' = f'(b)$, then the resulting function is the complete cubic spline.

Example 2.2. Find the natural cubic spline that interpolates the data

x_i	1	2	4	5
f_i	3	5	9	10

Solution.
We have $n=4$ nodes and $h_1 = 1, h_2 = 2, h_3 = 1$.
From (2.13)–(2.17), the linear system for the unknowns m_i, also called *slopes*, is

$$\begin{cases} 2m_1 + m_2 = 6 \\ 2m_1 + 6m_2 + m_3 = 18 \\ 2m_2 + 6m_3 + 2m_4 = 12 \\ m_3 + 2m_4 = 3 \end{cases}$$

with solution

$$m_1 = \frac{87}{46},\ m_2 = \frac{51}{23},\ m_3 = \frac{21}{23},\ m_4 = \frac{24}{23}.$$

The system (from (2.21) together with the conditions $M_1=M_4=0$) for the *moments* M_i becomes

$$\begin{cases} M_1 = 0 \\ M_1 + 6M_2 + 2M_3 = 0 \\ 2M_2 + 6M_3 + M_4 = -6 \\ M_4 = 0 \end{cases}$$

whose solution is

$$M_1 = 0,\ M_2 = \frac{3}{8},\ M_3 = -\frac{9}{8},\ M_4 = 0.$$

Hence, both ways, we get the natural cubic spline function

$$s_3(x) = \begin{cases} \dfrac{x^3}{16} - \dfrac{3x^2}{16} + \dfrac{17x}{8} + 1, & x \in [1,2] \\[2ex] -\dfrac{x^3}{8} + \dfrac{15x^2}{16} - \dfrac{x}{8} + \dfrac{5}{2}, & x \in [2,4] \\[2ex] -\dfrac{3x^3}{16} - \dfrac{45x^2}{16} + \dfrac{119x}{8} - \dfrac{35}{2}, & x \in [4,5] \end{cases} .$$

■

Minimality properties of cubic spline interpolants

Natural and complete splines have interesting optimality properties. Henceforth, we denote them by $s_{nat}(f;\cdot)$ and $s_{compl}(f;\cdot)$, respectively.

Theorem 2.3. *Let $g \in C^2[a,b]$ be any function that interpolates f on Δ. Then*

$$\int_a^b |s''_{nat}(f;x)|^2 dx \quad \leq \quad \int_a^b |g''(x)|^2 dx, \tag{2.23}$$

with equality if and only if $g(\cdot) \; = \; s_{nat}(f;\cdot)$.

For the next minimality result, we slightly change the subdivision Δ. Consider the grid

$$\Delta' \quad : \quad a = x_0 = x_1 < x_2 < \cdots < x_{n-1} < x_n = x_{n+1} = b, \tag{2.24}$$

where the endpoints are double nodes. That means that when we use Δ', we interpolate the function values at all interior points, and, both the functional and the derivative values, at the endpoints.

Theorem 2.4. *Let $g \in C^2[a,b]$ be any function that interpolates f on Δ'. Then*

$$\int_a^b |s''_{compl}(f;x)|^2 dx \quad \leq \quad \int_a^b |g''(x)|^2 dx, \tag{2.25}$$

with equality if and only if $g(\cdot) \; = \; s_{compl}(f;\cdot)$.

Remark 2.5. Taking $g(\cdot) = s_{compl}(f;\cdot)$ in Theorem 2.3, we get

$$\int_a^b |s''_{nat}(f;x)|^2 dx \quad \leq \quad \int_a^b |s''_{compl}(f;x)|^2 dx. \tag{2.26}$$

So, in a sense, the natural cubic spline is the "smoothest" interpolant.

Example 2.6. Consider the function $f(x) = \arctan x,\ x \in [-2,2]$ and the nodes $\{-2,-1,0,1,2\}$. Figure 3 shows the graphs of the function f, the nodes and the complete, natural, deBoor and piecewise Hermite cubic splines interpolating f. In Figure 4 we have the interpolation errors.

Remark 2.7. These minimality properties are at the origin of the name "spline". A *spline* is a flexible strip of wood used in drawing curves (or a musical instrument in that shape).

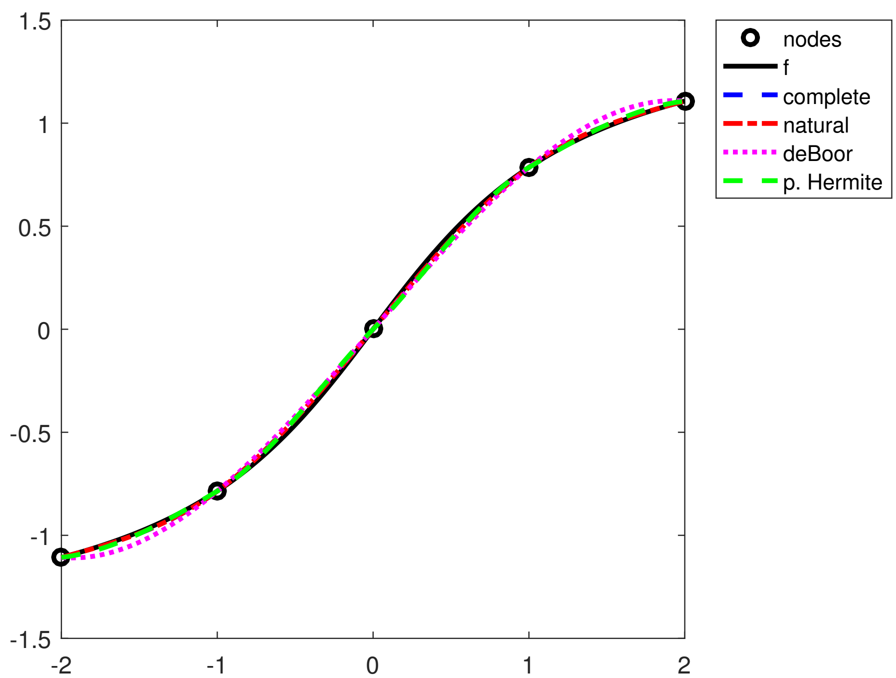

Fig. 3: Interpolation with cubic splines, $f(x) = \arctan x$

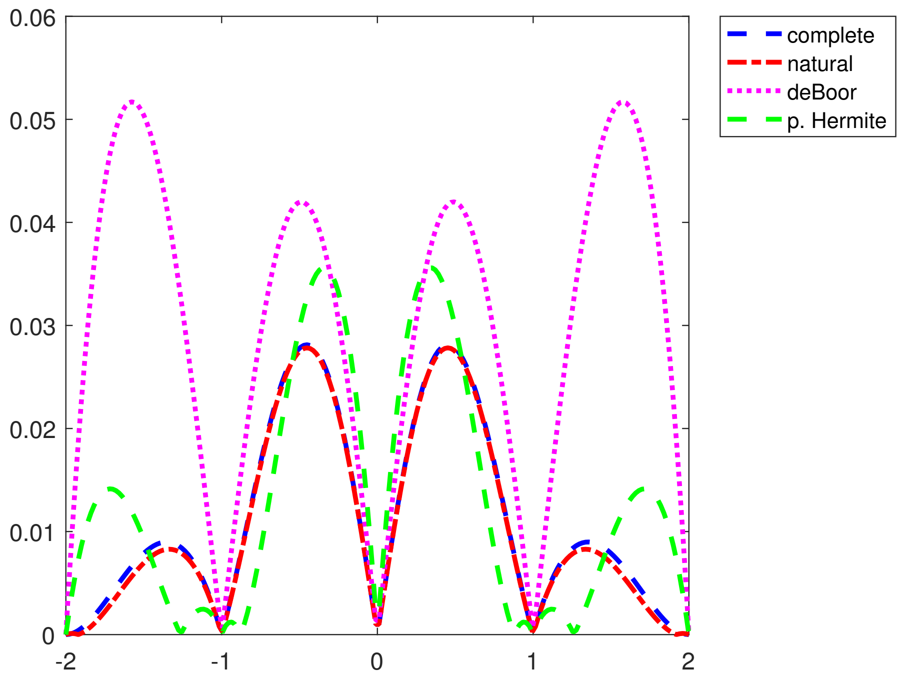

Fig. 4: Errors in cubic spline interpolation, $f(x) = \arctan x$

3 Least Squares Approximation

3.1 Best approximation problem

In general, an approximation problem can be described as follows: Let $f \in X$ be a function, Φ, a family of approximants and $||\cdot||$ a norm on X. We seek an approximation $\hat{\varphi} \in \Phi$ of f that approximates the given function "as well as possible".

$$||f - \hat{\varphi}|| \quad \leq \quad ||f - \varphi||, \; \forall \varphi \in \Phi. \tag{3.1}$$

This is called a *best approximation problem* of f with elements of Φ. The function $\hat{\varphi}$ is called a *best approximation* of f relative to the norm $||\cdot||$. Given a basis $\{\pi_j\}_1^m$ of Φ, we can write

$$\Phi \;=\; \Phi_m \quad = \quad \Big\{ \varphi \mid \varphi(t) = \sum_{j=1}^{m} c_j \pi_j(t), \; c_j \in \mathbb{R} \Big\}. \tag{3.2}$$

Φ is a finite dimensional linear space or a subset of one. We have already seen examples of such spaces, $\Phi = \mathbb{P}_n$ or $\Phi = S_m^k(\Delta)$, when we discussed interpolation problems. The norms can be

$$\begin{aligned} ||u||_\infty &= \max_{t\in[a,b]} |u(t)|, \\ ||u||_2 &= \Big(\int_a^b |u(t)|^2\, dt\Big)^{1/2}, \\ ||u||_{2,w} &= \Big(\int_a^b w(t)\, |u(t)|^2\, dt\Big)^{1/2}, \end{aligned}$$

or their discrete versions,

$$\begin{aligned} ||u||_\infty &= \max_{i=\overline{1,m}} |u(t_i)|, \\ ||u||_2 &= \Big(\sum_{i=1}^m |u(t_i)|^2\Big)^{1/2}, \\ ||u||_{2,w} &= \Big(\sum_{i=1}^m w_i\, |u(t_i)|^2\Big)^{1/2}, \end{aligned}$$

where w is a *weight* function. An intuitive, physical justification for a *weighted norm* would be that some observations are more important than others, or they are more common, so they "weigh more".

We will consider a particular case for problem (3.1) by choosing the (discrete or continuous) $||\cdot||_2$ norm. This is then called a *least squares approximation problem* or *square mean approximation problem*. Its solution was given by Gauss and Legendre at the beginning of the 19th century.

In what follows, we will only consider the *discrete* least squares approximation problem, or the problem of *curve (data) fitting*, as this is most common in various applications.

3.2 Normal Equations

To simplify the writing, let us recall the *inner (scalar) product* on $\mathbb{R}^m$:

$$(u, v) = \sum_{j=1}^m u(t_i)v(t_i).$$

Let x and y be two variables. For distinct values $x_1, x_2, \ldots, x_n$, we have (measure) the corresponding values of y, denoted by $y_1, y_2, \ldots, y_n$. We seek a relation of the form

$$y \;=\; f(x) \;=\; c_1\pi_1(x) + \ldots c_m\pi_m(x), \tag{3.3}$$

so that the *residues* (errors)

$$r_i \;=\; y_i - f(x_i),\; i = 1, \ldots, n, \tag{3.4}$$

have a minimum norm. In other words, we want to find the values c_j that minimize the error

$$E \;=\; \sum_{i=1}^{n} r_i^2 = \sum_{i=1}^{n} \big(y_i - \sum_{j=1}^{m} c_j\pi_j(x_i)\big)^2. \tag{3.5}$$

We solve this problem (finding the minimum of a function) by taking partial derivatives (with respect to each unknown) and setting them equal to 0. We get

$$\frac{\partial E}{\partial c_i} \;=\; -2\sum_{i=1}^{n} \pi_i(x_i)\big(y_i - \sum_{j=1}^{m} c_j\pi_j(x)\big) \;=\; 0,\; i = 1, \ldots, n,$$

or

$$\sum_{j=1}^{m} (\pi_i, \pi_j)c_j \;=\; (\pi_i, y),\; i = 1, \ldots, n. \tag{3.6}$$

The equations in (3.6) are called *normal equations*. The determinant of the system is the Gram determinant of the vectors $\{\pi_1, \ldots, \pi_m\}$, which is *not* 0, as these vectors are linearly independent.

Let us see some examples.

1. If we seek a linear function f, i.e.

$$f(x) = ax + b, \tag{3.7}$$

then we want the minimum of the function

$$E(a, b) = \sum_{i=1}^{n} (ax_i + b - y_i)^2$$

This is a function of two variables, whose minimum we want to find. The normal equations are:

$$\begin{aligned} E'_a &= 2\sum_{i=1}^{n}(ax_i + b - y_i)x_i, & \sum_{i=1}^{n}(ax_i + b - y_i)x_i &= 0, \\ E'_b &= 2\sum_{i=1}^{n}(ax_i + b - y_i), & \sum_{i=1}^{n}(ax_i + b - y_i) &= 0, \end{aligned}$$

so, we have

$$\begin{array}{rcrcl} a\sum_{i=i}^{n} x_i^2 & + & b\sum_{i=i}^{n} x_i & = & \sum_{i=i}^{n} x_i y_i \\ a\sum_{i=i}^{n} x_i & + & nb & = & \sum_{i=i}^{n} y_i. \end{array} \tag{3.8}$$

This 2×2 linear system has a unique solution, because its determinant is

$$n\sum_{i=i}^{n} x_i^2 - \Big(\sum_{i=i}^{n} x_i\Big)^2 \;>\; 0$$

(a consequence of Cauchy's inequality).

2. If we want a quadratic function f, of the form

$$f(x) \;=\; ax^2 + bx + c, \tag{3.9}$$

then we seek the minimum of the error function

$$E(a,b,c) \;=\; \sum_{i=1}^{n}(ax_i^2 + bx_i + c - y_i)^2.$$

The parameters a, b and c are determined from the normal system

$$\begin{aligned} a\sum_{i=1}^{n} x_i^4 + b\sum_{i=1}^{n} x_i^3 + c\sum_{i=1}^{n} x_i^2 &= \sum_{i=1}^{n} x_i^2 y_i, \\ a\sum_{i=1}^{n} x_i^3 + b\sum_{i=1}^{n} x_i^2 + c\sum_{i=1}^{n} x_i &= \sum_{i=1}^{n} x_i y_i, \\ a\sum_{i=1}^{n} x_i^2 + b\sum_{i=1}^{n} x_i + \; nc &= \sum_{i=1}^{n} y_i, \end{aligned} \tag{3.10}$$

obtained by setting the partial derivatives of E (with respect to a, b and c) equal to zero.

Example 3.1. Find the least squares polynomial approximation that fits the following data:

x_i	0.5	1.5	2	3	3.5	4.5	5	6	7	8
y_i	5	5.8	5.8	6.8	6.9	7.6	7.8	8.2	9.2	9.9

Solution. The scatterplot is shown in Figure 5. We see from the graph that the best fit is given by a linear function,

$$f\,(x; a, b) \;=\; ax + b.$$

The solution of the normal equations (3.8) is

$$a = 0.63, \quad b = 4.71.$$

So, the least squares polynomial best fit is given by

$$f(x) \;=\; 0.63x + 4.71.$$

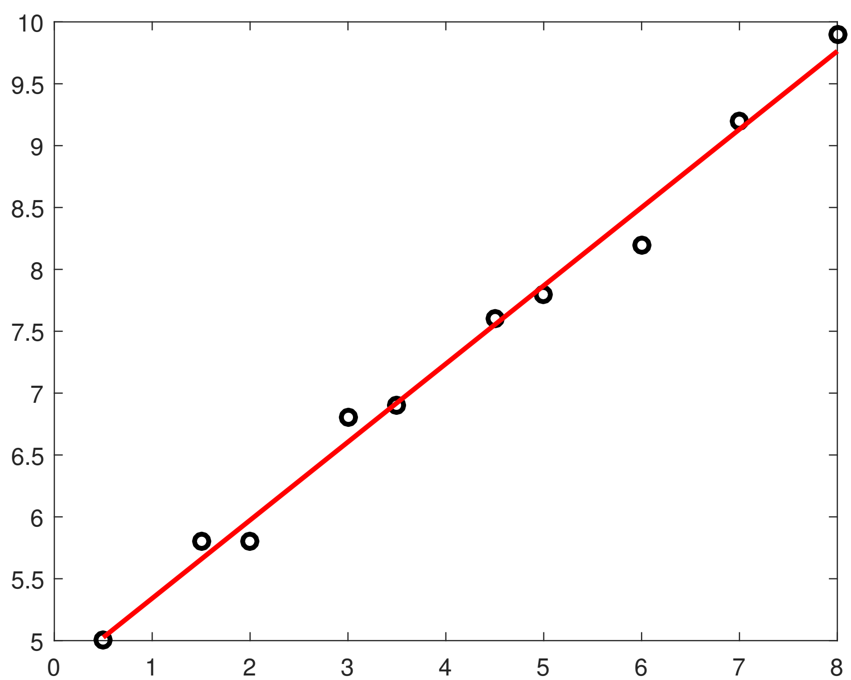

Fig. 5: Example 3.1

■