數值分析 MA-3021

Chapter 3. Interpolation and Polynomial Approximation

§3.1 Interpolation and the Lagrange Polynomial

1. **Interpolation:** find a function that fits the given data

$$(x_0, y_0), (x_1, y_1), \cdots, (x_n, y_n).$$

2. **Why polynomials?**

   The Weierstrass Approximation Theorem:

   Suppose that $f \in C([a, b])$. Then for every $\varepsilon > 0$, there exists a polynomial p defined on $[a, b]$ such that $|f(x) - p(x)| < \varepsilon$ for all $x \in [a, b]$. In other words, every continuous function f on $[a, b]$ is the **uniform** limit of polynomials.

3. **Why not Taylor polynomials?**
   - need to calculate $f'(x), f''(x), \cdots$
   - accurate near at a specific point, not on entire interval.

§3.1 Interpolation and the Lagrange Polynomial

1. We solve the following problem: given a table of $n+1$ data points (x_i, y_i),

x	x_0	x_1	x_2	$\cdots$	x_n
y	y_0	y_1	y_2	$\cdots$	y_n

we seek a polynomial p of **lowest possible degree** for which

$$p(x_i) = y_i \quad (0 \leqslant i \leqslant n).$$

2. *Such a polynomial is said to "interpolate" the data.*

§3.1 Interpolation and the Lagrange Polynomial

Theorem

Given $n+1$ distinct real (or complex) numbers $x_0, x_1, \cdots, x_n$ and their function values $f(x_0), f(x_1), \cdots, f(x_n)$, there exists a unique polynomial $p(x)$, degree $p(x) \leqslant n$, such that

$$p(x_k) = f(x_k), \quad k = 0, 1, \cdots, n.$$

Definition

The polynomial p given in the theorem above is called the n-th Lagrange interpolating polynomial for the data $(x_0, f(x_0))$, $(x_1, f(x_1))$, $\cdots$, $(x_n, f(x_n))$.

§3.1 Interpolation and the Lagrange Polynomial

Theorem

Given $n+1$ distinct real (or complex) numbers $x_0, x_1, \cdots, x_n$ and their function values $f(x_0), f(x_1), \cdots, f(x_n)$, there exists a unique polynomial $p(x)$, degree $p(x) \leqslant n$, such that

$$p(x_k) = f(x_k), \quad k = 0, 1, \cdots, n.$$

Definition

The polynomial p given in the theorem above is called the n-th Lagrange interpolating polynomial ~~for the data $(x_0, f(x_0)), (x_1, f(x_1)), \cdots, (x_n, f(x_n))$~~.

§3.1 Interpolation and the Lagrange Polynomial

Proof.

Assume that

$$p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n.$$

The interpolation conditions, $p(x_k) = f(x_k)$ for $0 \leqslant k \leqslant n$, lead to the following system of $n+1$ linear equations for determining $a_0, a_1, \cdots, a_n$:

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{bmatrix}.$$

The coefficient matrix X is called the Vandermonde matrix. It is nonsingular with $\det X = \prod_{0 \leqslant i < j \leqslant n} (x_j - x_i) \neq 0$, (but it is often ill-conditioned). □

§3.1 Interpolation and the Lagrange Polynomial

1. Given $(x_0, f(x_0))$ and $(x_1, f(x_1))$, $x_0 \neq x_1$, we consider

$$p(x) = \frac{x - x_1}{x_0 - x_1} f(x_0) + \frac{x - x_0}{x_1 - x_0} f(x_1) \equiv L_{1,0}(x) f(x_0) + L_{1,1}(x) f(x_1) \, .$$

Then degree $p(x) \leqslant 1$ and $p(x_0) = f(x_0)$, $p(x_1) = f(x_1)$.

2. Given $n+1$ distinct numbers $x_0, x_1, \cdots, x_n$, then for each $k = 0, 1, \cdots, n$, how to construct a quotient $L_{n,k}(x)$ such that

$$L_{n,k}(x_i) = \begin{cases} 1 & \text{if } i = k, \\ 0 & \text{if } i \neq k. \end{cases}$$

Answer: for $k = 0, 1, \cdots, n$,

$$\begin{aligned} L_{n,k}(x) &= \frac{(x - x_0)}{(x_k - x_0)} \frac{\cdots}{\cdots} \frac{(x - x_{k-1})}{(x_k - x_{k-1})} \frac{(x - x_{k+1})}{(x_k - x_{k+1})} \frac{\cdots}{\cdots} \frac{(x - x_n)}{(x_k - x_n)} \\ &= \prod_{i=0, i \neq k}^{n} \frac{x - x_i}{x_k - x_i}. \end{aligned}$$

§3.1 Interpolation and the Lagrange Polynomial

1. Given $(x_0, f(x_0))$ and $(x_1, f(x_1))$, $x_0 \neq x_1$, we consider

$$p(x) = \frac{x - x_1}{x_0 - x_1} f(x_0) + \frac{x - x_0}{x_1 - x_0} f(x_1) \equiv L_{1,0}(x) f(x_0) + L_{1,1}(x) f(x_1).$$

Then degree $p(x) \leqslant 1$ and $p(x_0) = f(x_0)$, $p(x_1) = f(x_1)$.

2. Given $n+1$ distinct numbers $x_0, x_1, \cdots, x_n$, then for each $k = 0, 1, \cdots, n$, how to construct a quotient $L_{n,k}(x)$ such that

$$L_{n,k}(x_i) = \begin{cases} 1 & \text{if } i = k, \\ 0 & \text{if } i \neq k. \end{cases}$$

Answer: for $k = 0, 1, \cdots, n$,

$$L_{n,k}(x) = \frac{(x - x_0)}{(x_k - x_0)} \frac{\cdots}{\cdots} \frac{(x - x_{k-1})}{(x_k - x_{k-1})} \frac{(x - x_{k+1})}{(x_k - x_{k+1})} \frac{\cdots}{\cdots} \frac{(x - x_n)}{(x_k - x_n)}$$
$$= \prod_{i=0, i \neq k}^{n} \frac{x - x_i}{x_k - x_i}.$$

§3.1 Interpolation and the Lagrange Polynomial

1. Given $(x_0, f(x_0))$ and $(x_1, f(x_1))$, $x_0 \neq x_1$, we consider

$$p(x) = \frac{x - x_1}{x_0 - x_1} f(x_0) + \frac{x - x_0}{x_1 - x_0} f(x_1) \equiv L_{1,0}(x) f(x_0) + L_{1,1}(x) f(x_1).$$

Then degree $p(x) \leqslant 1$ and $p(x_0) = f(x_0)$, $p(x_1) = f(x_1)$.

2. Given $n+1$ distinct numbers $x_0, x_1, \cdots, x_n$, then for each $k = 0, 1, \cdots, n$, how to construct a quotient $L_{n,k}(x)$ such that

$$L_{n,k}(x_i) = \begin{cases} 1 & \text{if } i = k, \\ 0 & \text{if } i \neq k. \end{cases}$$

Answer: for $k = 0, 1, \cdots, n$,

$$L_{n,k}(x) = \frac{(x - x_0)}{(x_k - x_0)} \frac{\cdots}{\cdots} \frac{(x - x_{k-1})}{(x_k - x_{k-1})} \frac{(x - x_{k+1})}{(x_k - x_{k+1})} \frac{\cdots}{\cdots} \frac{(x - x_n)}{(x_k - x_n)}$$
$$= \prod_{i=0, i \neq k}^{n} \frac{x - x_i}{x_k - x_i}.$$

§3.1 Interpolation and the Lagrange Polynomial

Theorem

Given $n+1$ distinct real (or complex) numbers $x_0, x_1, \cdots, x_n$ and their function values $f(x_0), f(x_1), \cdots, f(x_n)$, there exists a unique polynomial $p(x)$, degree $p(x) \leqslant n$, such that

$$p(x_k) = f(x_k), \quad k = 0, 1, \cdots, n.$$

In fact, this polynomial is given by

$$\begin{aligned} p(x) &= f(x_0)L_{n,0}(x) + f(x_1)L_{n,1}(x) + \cdots + f(x_n)L_{n,n}(x) \\ &= \sum_{k=0}^{n} f(x_k)L_{n,k}(x). \end{aligned}$$

§3.1 Interpolation and the Lagrange Polynomial

Example

Let $x_0 = 2$, $f(x_0) = 0.5$, $x_1 = 2.5$, $f(x_1) = 0.4$, $x_2 = 4$, $f(x_2) = 0.25$ (in fact, $f(x) = 1/x$). Find the second ($n = 2$) Lagrange interpolating polynomial.

$$L_{2,0}(x) = \frac{(x-2.5)(x-4)}{(2-2.5)(2-4)} = \frac{x^2 - 6.5x + 10}{1},$$

$$L_{2,1}(x) = \frac{(x-2)(x-4)}{(2.5-2)(2.5-4)} = \frac{x^2 - 6x + 8}{-0.75},$$

$$L_{2,2}(x) = \frac{(x-2)(x-2.5)}{(4-2)(4-2.5)} = \frac{x^2 - 4.5x + 5}{3}.$$

Therefore,

$$p(x) = 0.5\left(\frac{x^2 - 6.5x + 10}{1}\right) + 0.4\left(\frac{x^2 - 6x + 8}{-0.75}\right) + 0.25\left(\frac{x^2 - 4.5x + 5}{3}\right) = 0.05x^2 - 0.425x + 1.15.$$

Note that $p(3) = 0.325$ and $f(3) \approx 0.333$.

§3.1 Interpolation and the Lagrange Polynomial

Theorem

Let f be a given real-valued function in $C^{n+1}([a,b])$, and $x_0, x_1, \cdots, x_n \in [a,b]$ be $n+1$ distinct numbers. Then for each x in $[a,b]$, there exists $\xi(x) \in (a,b)$ such that

$$f(x) = p(x) + \frac{1}{(n+1)!} f^{(n+1)}(\xi(x)) \prod_{i=0}^{n} (x - x_i),$$

where $p(x)$ is the n-th Lagrange interpolating polynomial for the data $(x_0, f(x_0)), (x_1, f(x_1)), \cdots, (x_n, f(x_n))$.

Proof.

Let $x \in [a,b]$. If $x = x_k$ for some $0 \leqslant k \leqslant n$, then the assertion holds; thus W.L.O.G., we assume that $x \neq x_k$ for any $k = 0, 1, \cdots, n$. □

§3.1 Interpolation and the Lagrange Polynomial

Proof (Cont.) x is given such that $x \neq x_k$ for all $k = 0, 1, \cdots, n$.

Let $w(t) = \prod_{i=0}^{n} (t - x_i)$ and define $g : [a, b] \to \mathbb{R}$ by

$$g(t) = f(t) - p(t) - \frac{f(x) - p(x)}{w(x)} w(t) .$$

Then $g \in C^{n+1}([a, b])$ and g vanishes at (distinct) $n + 2$ points $x, x_0, x_1, \cdots, x_n$. By generalized Rolle's Theorem, $g^{(n+1)}$ has at least one zero $\xi(x) \in (a, b)$. Since

$$g^{(n+1)}(t) = f^{(n+1)}(t) - p^{(n+1)}(t) - \frac{f(x) - p(x)}{w(x)} w^{(n+1)}(t)$$

we have

$$0 = g^{(n+1)}(\xi(x)) = f^{(n+1)}(\xi(x)) - (n + 1)! \frac{f(x) - p(x)}{w(x)} .$$

□

§3.1 Interpolation and the Lagrange Polynomial

Proof (Cont.) x is given such that $x \neq x_k$ for all $k = 0, 1, \cdots, n$.

Let $w(t) = \prod_{i=0}^{n}(t - x_i)$ and define $g : [a, b] \to \mathbb{R}$ by

$$g(t) = f(t) - p(t) - \frac{f(x) - p(x)}{w(x)} w(t) .$$

Then $g \in C^{n+1}([a, b])$ and g vanishes at (distinct) $n + 2$ points $x, x_0, x_1, \cdots, x_n$. By generalized Rolle's Theorem, $g^{(n+1)}$ has at least one zero $\xi(x) \in (a, b)$. Since

$$g^{(n+1)}(t) = f^{(n+1)}(t) - p^{(n+1)}(t) - \frac{f(x) - p(x)}{w(x)} w^{(n+1)}(t)$$

we have

$$0 = g^{(n+1)}(\xi(x)) = f^{(n+1)}(\xi(x)) - (n + 1)! \frac{f(x) - p(x)}{w(x)} .$$

□

§3.1 Interpolation and the Lagrange Polynomial

Proof (Cont.) x is given such that $x \neq x_k$ for all $k = 0, 1, \cdots, n$.

Let $w(t) = \prod_{i=0}^{n}(t - x_i)$ and define $g : [a, b] \to \mathbb{R}$ by

$$g(t) = f(t) - p(t) - \frac{f(x) - p(x)}{w(x)} w(t)\,.$$

Then $g \in C^{n+1}([a, b])$ and g vanishes at (distinct) $n + 2$ points $x, x_0, x_1, \cdots, x_n$. By generalized Rolle's Theorem, $g^{(n+1)}$ has at least one zero $\xi(x) \in (a, b)$. Since

$$g^{(n+1)}(t) = f^{(n+1)}(t) - p^{(n+1)}(t) - \frac{f(x) - p(x)}{w(x)} w^{(n+1)}(t)$$

we have

$$0 = g^{(n+1)}(\xi(x)) = f^{(n+1)}(\xi(x)) - (n+1)!\frac{f(x) - p(x)}{w(x)}\,.$$

□

§3.1 Interpolation and the Lagrange Polynomial

Proof (Cont.) x is given such that $x \neq x_k$ for all $k = 0, 1, \cdots, n$.

Let $w(t) = \prod_{i=0}^{n} (t - x_i)$ and define $g : [a, b] \to \mathbb{R}$ by

$$g(t) = f(t) - p(t) - \frac{f(x) - p(x)}{w(x)} w(t) .$$

Then $g \in C^{n+1}([a, b])$ and g vanishes at (distinct) $n + 2$ points $x, x_0, x_1, \cdots, x_n$. By generalized Rolle's Theorem, $g^{(n+1)}$ has at least one zero $\xi(x) \in (a, b)$. Since

$$g^{(n+1)}(t) = f^{(n+1)}(t) - p^{(n+1)}(t) - \frac{f(x) - p(x)}{w(x)} w^{(n+1)}(t)$$

we have

$$0 = g^{(n+1)}(\xi(x)) = f^{(n+1)}(\xi(x)) - (n+1)! \frac{f(x) - p(x)}{w(x)} .$$

□

§3.1 Interpolation and the Lagrange Polynomial

Example

$f(x) = e^x, x \in [0,1]$. Let $x_0, x_1, \cdots, x_n$ be a uniform partition of $[0,1]$ with step size $h = 1/n$.

Consider $[x_j, x_{j+1}]$ for some j. Let $p(x)$ be the first Lagrange interpolating polynomial on $[x_j, x_{j+1}]$. Then for $x \in [x_j, x_{j+1}]$,

$$\begin{aligned}\left|f(x) - p(x)\right| &= \left|\frac{f''(\xi)}{2!}(x - x_j)(x - x_{j+1})\right| \\ &\leqslant \frac{e^\xi}{2}\left|(x - jh)(x - (j+1)h)\right| \qquad \xi \in (x_j, x_{j+1}) \\ &\leqslant \frac{1}{2}\max_{\xi\in[0,1]} e^\xi \max_{x\in[x_j,x_{j+1}]}\left|(x - jh)(x - (j+1)h)\right| \\ &\leqslant \frac{1}{2}e\frac{h^2}{4} = \frac{eh^2}{8}.\end{aligned}$$

If $\left|f(x) - p(x)\right| \leqslant (eh^2)/8 \leqslant 10^{-6}$ then $h < 1.72 \times 10^{-3}$. We can choose $h = 0.001$.

§3.2 Divided Differences

1. Let f be a function whose values are known at points (nodes) $x_0, x_1, \cdots, x_n$.
2. We assume that these nodes are distinct, but they need not be ordered.
3. We know there is a unique polynomial p of degree at most n such that

$$p(x_i) = f(x_i) \quad \text{for } 0 \leqslant i \leqslant n.$$

4. p can be constructed as a linear combination of $1, x, x^2, \cdots, x^n$.

§3.2 Divided Differences

Instead of expressing p as the linear combination of monomials $1, x, x^2, \cdots, x^n$, we should use the Newton form of the interpolating polynomials:

$$
\begin{aligned}
q_0(x) &= 1\,,\\
q_1(x) &= (x - x_0)\,,\\
q_2(x) &= (x - x_0)(x - x_1)\,,\\
q_3(x) &= (x - x_0)(x - x_1)(x - x_2)\,,\\
&\vdots\\
q_n(x) &= (x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{n-1})\,.
\end{aligned}
$$

The n-th Lagrange interpolating polynomial p can be expressed as

$$p(x) = \sum_{j=0}^{n} c_j q_j(x).$$

§3.2 Divided Differences

1. The interpolation conditions give rise to a linear system of equations for the unknown coefficients:

$$\sum_{j=0}^{n} c_j q_j(x_i) = f(x_i) \quad \text{for } 0 \leqslant i \leqslant n\,.$$

2. The elements of the coefficient matrix are

$$a_{ij} = q_j(x_i) \quad \text{for } 0 \leqslant i, j \leqslant n\,.$$

3. *The* $(n+1) \times (n+1)$ *matrix* $A = [a_{ij}]_{(n+1)\times(n+1)}$ *is a lower triangular matrix* because

$$q_j(x) = \prod_{k=0}^{j-1} (x - x_k)$$

which implies that

$$a_{ij} = q_j(x_i) = \prod_{k=0}^{j-1} (x_i - x_k) = 0 \quad \text{if } i \leqslant j-1\,.$$

§3.2 Divided Differences

Example

Consider the case of three nodes with

$$\begin{aligned} p(x) &= c_0 q_0(x) + c_1 q_1(x) + c_2 q_2(x) \\ &= c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1). \end{aligned}$$

Setting $x = x_0, x = x_1$, and $x = x_2$, we have a lower triangular system

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & (x_1 - x_0) & 0 \\ 1 & (x_2 - x_0) & (x_2 - x_0)(x_2 - x_1) \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \end{bmatrix}.$$

Thus, c_n depends on f at $x_0, x_1, \cdots, x_n$, and define the notation

$$c_n := f[x_0, x_1, \cdots, x_n].$$

We call $f[x_0, x_1, \cdots, x_n]$ a divided difference of f.

§3.2 Divided Differences

Note that $f[x_0, x_1, \cdots, x_n]$ is the **leading coefficient** of polynomial p of degree $\leqslant n$ which interpolates f at $x_0, x_1, \cdots, x_n$.

Theorem (Theorem on Higher-Order Divided Differences)

The divided differences satisfy the equation:

$$f[x_0, x_1, \cdots, x_n] = \frac{f[x_1, x_2, \cdots, x_n] - f[x_0, x_1, \cdots, x_{n-1}]}{x_n - x_0}.$$

Proof.

Let p_k be the polynomial of degree $\leqslant k$ that interpolates f at $x_0, x_1, \cdots, x_k$. Let q denote the polynomial of degree $\leqslant n-1$ that interpolates f at $x_1, x_2, \cdots, x_n$. Then

$$p_n(x) = q(x) + \frac{x - x_n}{x_n - x_0}\left[q(x) - p_{n-1}(x)\right].$$

since both sides have the same values at $x_0, x_1, \cdots, x_n$ and same degree $\leqslant n$. Examining the coefficient of x^n on the both sides, we arrive at the assertion. □

§3.2 Divided Differences

Note that $f[x_0, x_1, \cdots, x_n]$ is the **leading coefficient** of polynomial p of degree $\leqslant n$ which interpolates f at $x_0, x_1, \cdots, x_n$.

Theorem (Theorem on Higher-Order Divided Differences)

The divided differences satisfy the equation:

$$f[x_0, x_1, \cdots, x_n] = \frac{f[x_1, x_2, \cdots, x_n] - f[x_0, x_1, \cdots, x_{n-1}]}{x_n - x_0}.$$

Proof.

Let p_k be the polynomial of degree $\leqslant k$ that interpolates f at $x_0, x_1, \cdots, x_k$. Let q denote the polynomial of degree $\leqslant n-1$ that interpolates f at $x_1, x_2, \cdots, x_n$. Then

$$p_n(x) = q(x) + \frac{x - x_n}{x_n - x_0}\left[q(x) - p_{n-1}(x)\right].$$

since both sides have the same values at $x_0, x_1, \cdots, x_n$ and same degree $\leqslant n$. Examining the coefficient of x^n on the both sides, we arrive at the assertion. □

§3.2 Divided Differences

Note that $f[x_0, x_1, \cdots, x_n]$ is the **leading coefficient** of polynomial p of degree $\leqslant n$ which interpolates f at $x_0, x_1, \cdots, x_n$.

Theorem (Theorem on Higher-Order Divided Differences)

The divided differences satisfy the equation:

$$f[x_0, x_1, \cdots, x_n] = \frac{f[x_1, x_2, \cdots, x_n] - f[x_0, x_1, \cdots, x_{n-1}]}{x_n - x_0}.$$

Proof.

Let p_k be the polynomial of degree $\leqslant k$ that interpolates f at $x_0, x_1, \cdots, x_k$. Let q denote the polynomial of degree $\leqslant n-1$ that interpolates f at $x_1, x_2, \cdots, x_n$. Then

$$p_n(x) = q(x) + \frac{x - x_n}{x_n - x_0}\left[q(x) - p_{n-1}(x)\right].$$

since both sides have the same values at $x_0, x_1, \cdots, x_n$ and same degree $\leqslant n$. Examining the coefficient of x^n on the both sides, we arrive at the assertion. □

§3.2 Divided Differences

Note that $f[x_0, x_1, \cdots, x_n]$ is the **leading coefficient** of polynomial p of degree $\leqslant n$ which interpolates f at $x_0, x_1, \cdots, x_n$.

Theorem (Theorem on Higher-Order Divided Differences)

The divided differences satisfy the equation:

$$f[x_0, x_1, \cdots, x_n] = \frac{f[x_1, x_2, \cdots, x_n] - f[x_0, x_1, \cdots, x_{n-1}]}{x_n - x_0}.$$

Proof.

Let p_k be the polynomial of degree $\leqslant k$ that interpolates f at $x_0, x_1, \cdots, x_k$. Let q denote the polynomial of degree $\leqslant n-1$ that interpolates f at $x_1, x_2, \cdots, x_n$. Then

$$p_n(x) = q(x) + \frac{x - x_n}{x_n - x_0}\left[q(x) - p_{n-1}(x)\right].$$

since both sides have the same values at $x_0, x_1, \cdots, x_n$ and same degree $\leqslant n$. Examining the coefficient of x^n on the both sides, we arrive at the assertion. □

§3.2 Divided Differences

Note that $f[x_0, x_1, \cdots, x_n]$ is the **leading coefficient** of polynomial p of degree $\leqslant n$ which interpolates f at $x_0, x_1, \cdots, x_n$.

Theorem (Theorem on Higher-Order Divided Differences)

The divided differences satisfy the equation:

$$f[x_0, x_1, \cdots, x_n] = \frac{f[x_1, x_2, \cdots, x_n] - f[x_0, x_1, \cdots, x_{n-1}]}{x_n - x_0}.$$

Proof.

Let p_k be the polynomial of degree $\leqslant k$ that interpolates f at $x_0, x_1, \cdots, x_k$. Let q denote the polynomial of degree $\leqslant n-1$ that interpolates f at $x_1, x_2, \cdots, x_n$. Then

$$p_n(x) = q(x) + \frac{x - x_n}{x_n - x_0}\left[q(x) - p_{n-1}(x)\right].$$

since both sides have the same values at $x_0, x_1, \cdots, x_n$ and same degree $\leqslant n$. Examining the coefficient of x^n on the both sides, we arrive at the assertion. □

§3.2 Divided Differences

1. If a table of function values $(x_i, f(x_i))$ is given, we can construct from it a table of divided differences as follows:

x_0	$f[x_0]$	$f[x_0, x_1]$	$f[x_0, x_1, x_2]$	$f[x_0, x_1, x_2, x_3]$
x_1	$f[x_1]$	$f[x_1, x_2]$	$f[x_1, x_2, x_3]$	
x_2	$f[x_2]$	$f[x_2, x_3]$		
x_3	$f[x_3]$			

2. The following formula is called **Newton's interpolatory divided-difference formula**:

$$p_n(x) = f[x_0] + \sum_{k=1}^{n} f[x_0, x_1, \cdots, x_k](x - x_0)\cdots(x - x_{k-1}).$$

3. The coefficients required in the Newton interpolatory divided-difference formula occupy the top row in the divided difference table.

§3.2 Divided Differences

Example

Compute a divided difference table from

x_i	1.0	1.3	1.6	1.9	2.2
$f(x_i)$	0.7651977	0.6200860	0.4554022	0.2818186	0.1103623

Solution.

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, \cdots, x_{i+2}]$	$f[x_i, \cdots, x_{i+3}]$	$f[x_i, \cdots, x_{i+4}]$
$x_0 = 1.0$	0.7651977	−0.4837057	−0.1087339	0.0658784	0.0018251
$x_1 = 1.3$	0.6200860	−0.5489460	−0.0494433	0.0680685	
$x_2 = 1.6$	0.4454022	−0.5786120	0.0118183		
$x_3 = 1.9$	0.2818186	−0.5715210			
$x_4 = 2.2$	0.1103623				

Then Newton's interpolatory divided-difference formula provides

$$
\begin{aligned}
p_4(x) = {} & 0.7651977 - 0.4837057(x - 1.0) \\
& -0.1087339(x - 1.0)(x - 1.3) \\
& +0.0658784(x - 1.0)(x - 1.3)(x - 1.6) \\
& +0.0018251(x - 1.0)(x - 1.3)(x - 1.6)(x - 1.9).
\end{aligned}
$$

□

§3.2 Divided Differences

Theorem

If $(z_0, z_1, \cdots z_n)$ is a permutation of $(x_0, x_1, \cdots x_n)$, then

$$f[z_0, z_1, \cdots, z_n] = f[x_0, x_1, \cdots, x_n].$$

Proof.

1. $f[z_0, z_1, \cdots, z_n]$ is the coefficient of x^n in the polynomial of degree $\leqslant n$ that interpolates f at the nodes $z_0, z_1, \cdots, z_n$.
2. $f[x_0, x_1, \cdots, x_n]$ is the coefficient of x^n in the polynomial of degree $\leqslant n$ that interpolates f at the nodes $x_0, x_1, \cdots, x_n$.
3. These two polynomials are identical. This leads to the conclusion. □

§3.3 Hermite Interpolation

Definition

Let $x_0, x_1, \cdots, x_n \in [a, b]$ be $n + 1$ distinct numbers. Let $m_0, m_1, \cdots, m_n \geqslant 0$ integers and $m = \max\{m_0, m_1, \cdots, m_n\}$. Suppose that $f \in C^m([a, b])$. Then the **osculating polynomial** approximating f is the polynomial $p(x)$ of least degree such that

$$\frac{d^k p}{dx^k}(x_i) = \frac{d^k f}{dx^k}(x_i) \quad \text{for } i = 0, 1, \cdots, n \text{ and } k = 0, 1, \cdots, m_i.$$

Remark:

1. The degree of $p(x) \leqslant \left(\sum_{i=0}^{n} m_i\right) + n := M$.
2. If $n = 0$, then $p(x) = m_0$-th Taylor polynomial of $f(x)$ at x_0.
3. If $m_i = 0$ for $i = 0, 1, \cdots, n$, then $p(x) = n$-th Lagrange interpolating polynomial of $f(x)$ at $x_0, x_1, \cdots, x_n$.
4. If $m_i = 1$ for $i = 0, 1, \cdots, n$, then $p(x)$ is the **Hermite interpolating polynomial**.

§3.3 Hermite Interpolation

Definition

Let $x_0, x_1, \cdots, x_n \in [a, b]$ be $n+1$ distinct numbers. Let $m_0, m_1, \cdots, m_n \geqslant 0$ integers and $m = \max\{m_0, m_1, \cdots, m_n\}$. Suppose that $f \in C^m([a, b])$. Then the **osculating polynomial** approximating f is the polynomial $p(x)$ of least degree such that

$$\frac{d^k p}{dx^k}(x_i) = \frac{d^k f}{dx^k}(x_i) \quad \text{for } i = 0, 1, \cdots, n \text{ and } k = 0, 1, \cdots, m_i.$$

Remark:

1. The degree of $p(x) \leqslant \left(\sum_{i=0}^{n} m_i\right) + n := M$.
2. If $n = 0$, then $p(x) = m_0$-th Taylor polynomial of $f(x)$ at x_0.
3. If $m_i = 0$ for $i = 0, 1, \cdots, n$, then $p(x) = n$-th Lagrange interpolating polynomial of $f(x)$ at $x_0, x_1, \cdots, x_n$.
4. If $m_i = 1$ for $i = 0, 1, \cdots, n$, then $p(x)$ is the **Hermite interpolating polynomial**.

§3.3 Hermite Interpolation

Definition

Let $x_0, x_1, \cdots, x_n \in [a, b]$ be $n+1$ distinct numbers. Let $m_0, m_1, \cdots, m_n \geqslant 0$ integers and $m = \max\{m_0, m_1, \cdots, m_n\}$. Suppose that $f \in C^m([a, b])$. Then the **osculating polynomial** approximating f is the polynomial $p(x)$ of least degree such that

$$\frac{d^k p}{dx^k}(x_i) = \frac{d^k f}{dx^k}(x_i) \quad \text{for } i = 0, 1, \cdots, n \text{ and } k = 0, 1, \cdots, m_i.$$

Remark:

1. The degree of $p(x) \leqslant \left(\sum_{i=0}^n m_i\right) + n := M$.
2. If $n = 0$, then $p(x) = m_0$-th Taylor polynomial of $f(x)$ at x_0.
3. If $m_i = 0$ for $i = 0, 1, \cdots, n$, then $p(x) = n$-th Lagrange interpolating polynomial of $f(x)$ at $x_0, x_1, \cdots, x_n$.
4. If $m_i = 1$ for $i = 0, 1, \cdots, n$, then $p(x)$ is the **Hermite interpolating polynomial**.

§3.3 Hermite Interpolation

Definition

Let $x_0, x_1, \cdots, x_n \in [a, b]$ be $n+1$ distinct numbers. Let $m_0, m_1, \cdots, m_n \geqslant 0$ integers and $m = \max\{m_0, m_1, \cdots, m_n\}$. Suppose that $f \in C^m([a, b])$. Then the **osculating polynomial** approximating f is the polynomial $p(x)$ of least degree such that

$$\frac{d^k p}{dx^k}(x_i) = \frac{d^k f}{dx^k}(x_i) \quad \text{for } i = 0, 1, \cdots, n \text{ and } k = 0, 1, \cdots, m_i.$$

Remark:

1. The degree of $p(x) \leqslant \left(\sum_{i=0}^{n} m_i\right) + n := M$.
2. If $n = 0$, then $p(x) = m_0$-th Taylor polynomial of $f(x)$ at x_0.
3. If $m_i = 0$ for $i = 0, 1, \cdots, n$, then $p(x) = n$-th Lagrange interpolating polynomial of $f(x)$ at $x_0, x_1, \cdots, x_n$.
4. If $m_i = 1$ for $i = 0, 1, \cdots, n$, then $p(x)$ is the **Hermite interpolating polynomial**.

§3.3 Hermite Interpolation

Definition

Let $x_0, x_1, \cdots, x_n \in [a, b]$ be $n+1$ distinct numbers. Let $m_0, m_1, \cdots, m_n \geqslant 0$ integers and $m = \max\{m_0, m_1, \cdots, m_n\}$. Suppose that $f \in C^m([a, b])$. Then the **osculating polynomial** approximating f is the polynomial $p(x)$ of least degree such that

$$\frac{d^k p}{dx^k}(x_i) = \frac{d^k f}{dx^k}(x_i) \quad \text{for } i = 0, 1, \cdots, n \text{ and } k = 0, 1, \cdots, m_i.$$

Remark:

1. The degree of $p(x) \leqslant \left(\sum_{i=0}^{n} m_i\right) + n := M$.
2. If $n = 0$, then $p(x) = m_0$-th Taylor polynomial of $f(x)$ at x_0.
3. If $m_i = 0$ for $i = 0, 1, \cdots, n$, then $p(x) = n$-th Lagrange interpolating polynomial of $f(x)$ at $x_0, x_1, \cdots, x_n$.
4. If $m_i = 1$ for $i = 0, 1, \cdots, n$, then $p(x)$ is the **Hermite interpolating polynomial**.

§3.3 Hermite Interpolation

Theorem

Let $x_0, x_1, \cdots, x_n \in [a, b]$ be $n+1$ distinct numbers and $f \in C^1([a, b])$. The unique polynomial of least degree agreeing with $f(x)$ and $f'(x)$ at $x_0, x_1, \cdots, x_n$ is the Hermite polynomial of degree at most $2n+1$ and is given by

$$H_{2n+1}(x) = \sum_{j=0}^{n} f(x_j) H_{n,j}(x) + \sum_{j=0}^{n} f'(x_j) \widehat{H}_{n,j}(x) ,$$

where, with $L_{n,k}(x) = \prod_{i=0, i \neq k}^{n} \frac{x - x_i}{x_k - x_i}$, $H_{n,j}$ and $\widehat{H}_{n,j}$ are given by

$$H_{n,j}(x) = \Big[1 - 2(x - x_j) L'_{n,j}(x_j)\Big] L^2_{n,j}(x) ,$$
$$\widehat{H}_{n,j}(x) = (x - x_j) L^2_{n,j}(x) .$$

Moreover, if $f \in C^{2n+2}([a, b])$, then there exists $\xi \in (a, b)$ such that

$$f(x) = H_{2n+1}(x) + \frac{f^{(2n+2)}(\xi)}{(2n+2)!} (x - x_0)^2 (x - x_1)^2 \cdots (x - x_n)^2 .$$

§3.3 Hermite Interpolation

Proof. $H_{n,j}(x)=\left[1-2(x-x_j)L'_{n,j}(x_j)\right]L^2_{n,j}(x)$, $\widehat{H}_{n,j}(x)=(x-x_j)L^2_{n,j}(x)$.

Let δ_{ij} be the Kronecker delta defined by $\delta_{ij}=0$ if $i\neq j$ and $\delta_{ij}=1$ if $i=j$. Then

$$H'_{n,j}(x) = -2L'_{n,j}(x_j)L^2_{n,j}(x) + 2\left[1-2(x-x_j)L'_{n,j}(x_j)\right]L_{n,j}(x)L'_{n,j}(x),$$

$$\widehat{H}'_{n,j}(x) = L^2_{n,j}(x) + 2(x-x_j)L_{n,j}(x)L'_{n,j}(x);$$

thus using that $L_{n,j}(x_i)=\delta_{ij}$ we find that

$$\left.\begin{array}{lll} H_{n,j}(x_i)=\delta_{ij} & \text{and} & \widehat{H}_{n,j}(x_i)=0 \\ H'_{n,j}(x_i)=0 & \text{and} & \widehat{H}'_{n,j}(x_i)=\delta_{ij} \end{array}\right. \quad \text{for all } 0\leqslant i,j\leqslant n\,.$$

Therefore, by the fact that $H_{n,j}$ and $\widehat{H}_{n,j}$ are polynomials of degree $2n+1$ we conclude that $\{H_{n,j},\widehat{H}_{n,j}\}_{j=0}^{n}$ is a basis in the space of polynomials of degree $2n+1$. □

§3.3 Hermite Interpolation

Proof (Cont.) $H_{2n+1}(x) = \sum_{j=0}^{n} f(x_j)H_{n,j}(x) + \sum_{j=0}^{n} f'(x_j)\widehat{H}_{n,j}(x)$.

In other words, every polynomial of degree $\leqslant 2n+1$ can be expressed as a **unique** linear combination of $\{H_{n,j}, \widehat{H}_{n,j}\}_{j=0}^{n}$.

If g is a polynomial of degree $\leqslant 2n+1$ satisfying $g(x_j) = f(x_j)$ and $g'(x_j) = f'(x_j)$ for $j = 0, 1, \cdots, n$, then

1. $g(x) = \sum_{j=0}^{n} c_j H_{n,j}(x) + \sum_{j=0}^{n} d_j \widehat{H}_{n,j}(x)$ for some coefficients c_j, d_j.
2. $g(x_j) = f(x_j)$ for all j implies that $c_j = f(x_j)$.
3. $g'(x_j) = f'(x_j)$ for all j implies that $d_j = f'(x_j)$ for all j.

In other words, the function

$$H_{2n+1}(x) = \sum_{j=0}^{n} f(x_j)H_{n,j}(x) + \sum_{j=0}^{n} f'(x_j)\widehat{H}_{n,j}(x)$$

is the only function agreeing with $f(x)$ and $f'(x)$ at $x_0, x_1, \cdots, x_n$.

□

§3.3 Hermite Interpolation

Proof (Cont.) $H_{2n+1}(x) = \sum_{j=0}^{n} f(x_j)H_{n,j}(x) + \sum_{j=0}^{n} f'(x_j)\widehat{H}_{n,j}(x)$.

In other words, every polynomial of degree $\leqslant 2n+1$ can be expressed as a **unique** linear combination of $\{H_{n,j}, \widehat{H}_{n,j}\}_{j=0}^{n}$.

If g is a polynomial of degree $\leqslant 2n+1$ satisfying $g(x_j) = f(x_j)$ and $g'(x_j) = f'(x_j)$ for $j = 0, 1, \cdots, n$, then

1. $g(x) = \sum_{j=0}^{n} c_j H_{n,j}(x) + \sum_{j=0}^{n} d_j \widehat{H}_{n,j}(x)$ for some coefficients c_j, d_j.
2. $g(x_j) = f(x_j)$ for all j implies that $c_j = f(x_j)$.
3. $g'(x_j) = f'(x_j)$ for all j implies that $d_j = f'(x_j)$ for all j.

In other words, the function

$$H_{2n+1}(x) = \sum_{j=0}^{n} f(x_j)H_{n,j}(x) + \sum_{j=0}^{n} f'(x_j)\widehat{H}_{n,j}(x)$$

is the only function agreeing with $f(x)$ and $f'(x)$ at $x_0, x_1, \cdots, x_n$.

□

§3.3 Hermite Interpolation

Proof (Cont.) $H_{2n+1}(x) = \sum_{j=0}^{n} f(x_j)H_{n,j}(x) + \sum_{j=0}^{n} f'(x_j)\widehat{H}_{n,j}(x)$.

In other words, every polynomial of degree $\leqslant 2n+1$ can be expressed as a **unique** linear combination of $\{H_{n,j}, \widehat{H}_{n,j}\}_{j=0}^{n}$.

If g is a polynomial of degree $\leqslant 2n+1$ satisfying $g(x_j) = f(x_j)$ and $g'(x_j) = f'(x_j)$ for $j = 0, 1, \cdots, n$, then

1. $g(x) = \sum_{j=0}^{n} c_j H_{n,j}(x) + \sum_{j=0}^{n} d_j \widehat{H}_{n,j}(x)$ for some coefficients c_j, d_j.
2. $g(x_j) = f(x_j)$ for all j implies that $c_j = f(x_j)$.
3. $g'(x_j) = f'(x_j)$ for all j implies that $d_j = f'(x_j)$ for all j.

In other words, the function

$$H_{2n+1}(x) = \sum_{j=0}^{n} f(x_j)H_{n,j}(x) + \sum_{j=0}^{n} f'(x_j)\widehat{H}_{n,j}(x)$$

is the only function agreeing with $f(x)$ and $f'(x)$ at $x_0, x_1, \cdots, x_n$.

□

§3.3 Hermite Interpolation

Proof (Cont.) $f(x) = H_{2n+1}(x) + \frac{f^{(2n+2)}(\xi)}{(2n+2)!}(x-x_0)^2(x-x_1)^2\cdots(x-x_n)^2.$

Error estimate: We mimic the proof of the error estimate for Lagrange polynomials. For each $x \neq x_0, x_1 \cdots, x_n$, define

$$g(t) = f(t) - H_{2n+1}(t) - \frac{(t-x_0)^2(t-x_1)^2\cdots(t-x_n)^2}{(x-x_0)^2(x-x_1)^2\cdots(x-x_n)^2}\left[f(x) - H_{2n+1}(x)\right].$$

Then $g \in C^{2n+2}([a,b])$ and g vanishes at $n+2$ points x, x_0, x_1, $\cdots$, x_n. By Rolle's Theorem, there exist distinct $n+1$ points $c_0, c_1, \cdots, c_n$ such that $g'(c_j) = 0$ for all $0 \leqslant j \leqslant n$. Moreover, all c_j's are different from $x, x_0, x_1, \cdots, x_n$; that is, $x_0, x_1, \cdots, x_n, c_0, c_1, \cdots, c_n$ are distinct $2n+2$ points. Since g' vanishes at these $2n+2$ points, the generalized Rolle's Theorem implies that there exists ξ such that $(g')^{(2n+1)}(\xi) = 0$. This fact implies the error estimate above. □

§3.3 Hermite Interpolation

Proof (Cont.) $f(x) = H_{2n+1}(x) + \frac{f^{(2n+2)}(\xi)}{(2n+2)!}(x-x_0)^2(x-x_1)^2 \cdots (x-x_n)^2$.

Error estimate: We mimic the proof of the error estimate for Lagrange polynomials. For each $x \neq x_0, x_1 \cdots, x_n$, define

$$g(t) = f(t) - H_{2n+1}(t) - \frac{(t-x_0)^2(t-x_1)^2 \cdots (t-x_n)^2}{(x-x_0)^2(x-x_1)^2 \cdots (x-x_n)^2}\left[f(x) - H_{2n+1}(x)\right].$$

Then $g \in C^{2n+2}([a,b])$ and g vanishes at $n+2$ points x, x_0, x_1, $\cdots$, x_n. By Rolle's Theorem, there exist distinct $n+1$ points $c_0, c_1, \cdots, c_n$ such that $g'(c_j) = 0$ for all $0 \leqslant j \leqslant n$. Moreover, all c_j's are different from $x, x_0, x_1, \cdots, x_n$; that is, $x_0, x_1, \cdots, x_n, c_0, c_1, \cdots, c_n$ are distinct $2n+2$ points. Since g' vanishes at these $2n+2$ points, the generalized Rolle's Theorem implies that there exists ξ such that $(g')^{(2n+1)}(\xi) = 0$. This fact implies the error estimate above. □

§3.3 Hermite Interpolation

Proof (Cont.) $f(x) = H_{2n+1}(x) + \frac{f^{(2n+2)}(\xi)}{(2n+2)!}(x-x_0)^2(x-x_1)^2\cdots(x-x_n)^2$.

Error estimate: We mimic the proof of the error estimate for Lagrange polynomials. For each $x \neq x_0, x_1 \cdots, x_n$, define

$$g(t) = f(t) - H_{2n+1}(t) - \frac{(t-x_0)^2(t-x_1)^2\cdots(t-x_n)^2}{(x-x_0)^2(x-x_1)^2\cdots(x-x_n)^2}\left[f(x) - H_{2n+1}(x)\right].$$

Then $g \in C^{2n+2}([a,b])$ and g vanishes at $n+2$ points x, x_0, x_1, $\cdots$, x_n. By Rolle's Theorem, there exist distinct $n+1$ points $c_0, c_1, \cdots, c_n$ such that $g'(c_j) = 0$ for all $0 \leqslant j \leqslant n$. Moreover, all c_j's are different from $x, x_0, x_1, \cdots, x_n$; that is, $x_0, x_1, \cdots, x_n, c_0, c_1, \cdots, c_n$ are distinct $2n+2$ points. Since g' vanishes at these $2n+2$ points, the generalized Rolle's Theorem implies that there exists ξ such that $(g')^{(2n+1)}(\xi) = 0$. This fact implies the error estimate above. □

§3.3 Hermite Interpolation

Proof (Cont.) $f(x) = H_{2n+1}(x) + \frac{f^{(2n+2)}(\xi)}{(2n+2)!}(x-x_0)^2(x-x_1)^2\cdots(x-x_n)^2$.

Error estimate: We mimic the proof of the error estimate for Lagrange polynomials. For each $x \neq x_0, x_1 \cdots, x_n$, define

$$g(t) = f(t) - H_{2n+1}(t) - \frac{(t-x_0)^2(t-x_1)^2\cdots(t-x_n)^2}{(x-x_0)^2(x-x_1)^2\cdots(x-x_n)^2}\big[f(x) - H_{2n+1}(x)\big].$$

Then $g \in C^{2n+2}([a,b])$ and g vanishes at $n+2$ points x, x_0, x_1, $\cdots$, x_n. By Rolle's Theorem, there exist distinct $n+1$ points $c_0, c_1, \cdots, c_n$ such that $g'(c_j) = 0$ for all $0 \leqslant j \leqslant n$. Moreover, all c_j's are different from $x, x_0, x_1, \cdots, x_n$; that is, $x_0, x_1, \cdots, x_n, c_0, c_1, \cdots, c_n$ are distinct $2n+2$ points. Since g' vanishes at these $2n+2$ points, the generalized Rolle's Theorem implies that there exists ξ such that $(g')^{(2n+1)}(\xi) = 0$. This fact implies the error estimate above. □

§3.3 Hermite Interpolation

Proof (Cont.) $f(x) = H_{2n+1}(x) + \frac{f^{(2n+2)}(\xi)}{(2n+2)!}(x-x_0)^2(x-x_1)^2\cdots(x-x_n)^2$.

Error estimate: We mimic the proof of the error estimate for Lagrange polynomials. For each $x \neq x_0, x_1 \cdots, x_n$, define

$$g(t) = f(t) - H_{2n+1}(t) - \frac{(t-x_0)^2(t-x_1)^2\cdots(t-x_n)^2}{(x-x_0)^2(x-x_1)^2\cdots(x-x_n)^2}\left[f(x) - H_{2n+1}(x)\right].$$

Then $g \in C^{2n+2}([a,b])$ and g vanishes at $n+2$ points x, x_0, x_1, $\cdots$, x_n. By Rolle's Theorem, there exist distinct $n+1$ points $c_0, c_1, \cdots, c_n$ such that $g'(c_j) = 0$ for all $0 \leqslant j \leqslant n$. Moreover, all c_j's are different from $x, x_0, x_1, \cdots, x_n$; that is, $x_0, x_1, \cdots, x_n, c_0, c_1, \cdots, c_n$ are distinct $2n+2$ points. Since g' vanishes at these $2n+2$ points, the generalized Rolle's Theorem implies that there exists ξ such that $(g')^{(2n+1)}(\xi) = 0$. This fact implies the error estimate above. □

§3.3 Hermite Interpolation

Example

k	x_k	$f(x_k)$	$f'(x_k)$
0	1.3	0.6200860	−0.5220232
1	1.6	0.4554022	−0.5698959
2	1.9	0.2818186	−0.5811571

$$
\begin{aligned}
L_{2,0}(x) &= \frac{(x-1.6)(x-1.9)}{(-0.3)(-0.6)} = \frac{50}{9}x^2 - \frac{175}{9}x + \frac{152}{9},\\
L'_{2,0}(x) &= \frac{100}{9}x - \frac{175}{9},\\
L_{2,1}(x) &= \frac{(x-1.3)(x-1.9)}{(0.3)(-0.3)} = -\frac{100}{9}x^2 + \frac{320}{9}x - \frac{247}{9},\\
L'_{2,1}(x) &= -\frac{200}{9}x + \frac{320}{9},\\
L_{2,2}(x) &= \frac{(x-1.3)(x-1.6)}{(0.6)(0.3)} = \frac{50}{9}x^2 - \frac{145}{9}x + \frac{104}{9},\\
L'_{2,2}(x) &= \frac{100}{9}x - \frac{145}{9}.
\end{aligned}
$$

§3.3 Hermite Interpolation

Example (Cont.)

Therefore,

$$H_{2,0}(x) = (10x-12)(\frac{50}{9}x^2 - \frac{175}{9}x + \frac{152}{9})^2,$$
$$H_{2,1}(x) = 1(-\frac{100}{9}x^2 + \frac{320}{9}x - \frac{247}{9})^2,$$
$$H_{2,2}(x) = 10(2-x)(\frac{50}{9}x^2 - \frac{145}{9}x + \frac{104}{9})^2,$$
$$\widehat{H}_{2,0}(x) = (x-1.3)(\frac{50}{9}x^2 - \frac{175}{9}x + \frac{152}{9})^2,$$
$$\widehat{H}_{2,1}(x) = (x-1.6)(-\frac{100}{9}x^2 + \frac{320}{9}x - \frac{247}{9})^2,$$
$$\widehat{H}_{2,2}(x) = (x-1.9)(\frac{50}{9}x^2 - \frac{145}{9}x + \frac{104}{9})^2;$$

thus

$$H_5(x) = 0.6200860H_{2,0}(x) + 0.4554022H_{2,1}(x) + 0.2818186H_{2,2}(x)$$
$$-0.5220232\widehat{H}_{2,0}(x) - 0.5698959\widehat{H}_{2,1}(x) - 0.5811571\widehat{H}_{2,2}(x).$$

§3.3 Hermite Interpolation

1. The Newton interpolatory divided-difference formula for the n-th Lagrange polynomial at distinct numbers $x_0, x_1, \cdots, x_n$ is given by

$$p_n(x) = f[x_0] + \sum_{k=1}^{n} f[x_0, x_1, \cdots, x_k](x - x_0)\cdots(x - x_{k-1}).$$

2. Define $z_0, z_1, \cdots, z_{2n+1}$ by $z_{2i} = z_{2i+1} = x_i$, for $i = 0, 1, \cdots, n$. Then the Newton interpolatory divided-difference formula for the Hermite interpolating polynomial at distinct numbers x_0, x_1, $\cdots$, x_n is given by

$$H_{2n+1}(x) = f[z_0] + \sum_{k=1}^{2n+1} f[z_0, z_1, \cdots, z_k](x - z_0)\cdots(x - z_{k-1}),$$

where $f[x_i, x_i] := f'(x_i)$, since

$$\lim_{x\to x_i} f[x_i, x] = \lim_{x\to x_i} \frac{f(x) - f(x_i)}{x - x_i} = f'(x_i).$$

§3.3 Hermite Interpolation

z_0	$f[z_0]$	$f[z_0, z_1]$	$f[z_0, z_1, z_2]$	$f[z_0, z_1, z_2, z_3]$	$f[z_0, z_1, z_2, z_3, z_4]$
z_1	$f[z_1]$	$f[z_1, z_2]$	$f[z_1, z_2, z_3]$	$f[z_1, z_2, z_3, z_4]$	$f[z_1, z_2, z_3, z_4, z_5]$
z_2	$f[z_2]$	$f[z_2, z_3]$	$f[z_2, z_3, z_4]$	$f[z_2, z_3, z_4, z_5]$	
z_3	$f[z_3]$	$f[z_3, z_4]$	$f[z_3, z_4, z_5]$		
z_4	$f[z_4]$	$f[z_4, z_5]$			

$$\Updownarrow$$

z_0	$f[z_0]$	$f'(x_0)$	$f[z_0, z_1, z_2]$	$f[z_0, z_1, z_2, z_3]$	$f[z_0, z_1, z_2, z_3, z_4]$
z_1	$f[z_1]$	$f[z_1, z_2]$	$f[z_1, z_2, z_3]$	$f[z_1, z_2, z_3, z_4]$	$f[z_1, z_2, z_3, z_4, z_5]$
z_2	$f[z_2]$	$f'(x_1)$	$f[z_2, z_3, z_4]$	$f[z_2, z_3, z_4, z_5]$	
z_3	$f[z_3]$	$f[z_3, z_4]$	$f[z_3, z_4, z_5]$		
z_4	$f[z_4]$	$f'(x_2)$			

§3.3 Hermite Interpolation

z_0	$f[z_0]$	$f[z_0, z_1]$	$f[z_0, z_1, z_2]$	$f[z_0, z_1, z_2, z_3]$	$f[z_0, z_1, z_2, z_3, z_4]$
z_1	$f[z_1]$	$f[z_1, z_2]$	$f[z_1, z_2, z_3]$	$f[z_1, z_2, z_3, z_4]$	$f[z_1, z_2, z_3, z_4, z_5]$
z_2	$f[z_2]$	$f[z_2, z_3]$	$f[z_2, z_3, z_4]$	$f[z_2, z_3, z_4, z_5]$	
z_3	$f[z_3]$	$f[z_3, z_4]$	$f[z_3, z_4, z_5]$		
z_4	$f[z_4]$	$f[z_4, z_5]$			

$$\Updownarrow$$

z_0	$f[z_0]$	$f'(x_0)$	$f[z_0, z_1, z_2]$	$f[z_0, z_1, z_2, z_3]$	$f[z_0, z_1, z_2, z_3, z_4]$
z_1	$f[z_1]$	$f[z_1, z_2]$	$f[z_1, z_2, z_3]$	$f[z_1, z_2, z_3, z_4]$	$f[z_1, z_2, z_3, z_4, z_5]$
z_2	$f[z_2]$	$f'(x_1)$	$f[z_2, z_3, z_4]$	$f[z_2, z_3, z_4, z_5]$	
z_3	$f[z_3]$	$f[z_3, z_4]$	$f[z_3, z_4, z_5]$		
z_4	$f[z_4]$	$f'(x_2)$			

§3.4 Spline Interpolation

Disadvantages of

- *Lagrange interpolating polynomial:* oscillation of high-degree polynomial.
- *Piecewise linear approximation:* no assurance of differentiability at each endpoints of the subintervals.
- *Piecewise Hermite interpolating polynomial $H_3(x)$ of degree 3:* $f'(x_0), f'(x_1), \cdots, f'(x_n)$ are usually not available.

Goals:

- piecewise polynomial;
- no derivative information is required, except perhaps at $x_0(= a)$ and $x_n(= b)$;
- continuously differentiable in the whole domain $[a, b]$.

$\Longrightarrow$ **spline interpolation**

§3.4 Spline Interpolation

Let f be defined on $[x_0, x_2]$, and $f(x_0)$, $f(x_1)$ and $f(x_2)$ are given.

A quadratic spline function S consists of the quadratic polynomials:

$$S_0(x) = a_0 + b_0(x - x_0) + c_0(x - x_0)^2 \quad \text{on } [x_0, x_1],$$
$$S_1(x) = a_1 + b_1(x - x_1) + c_1(x - x_1)^2 \quad \text{on } [x_1, x_2]$$

such that

1. $S(x_0) = f(x_0), S(x_1) = f(x_1) \text{ and } S(x_2) = f(x_2)$;
2. $S \in C^1([x_0, x_2])$.

§3.4 Spline Interpolation

- From condition ①, we must have

$$
\begin{aligned}
a_0 &= f(x_0), \\
a_0 + b_0(x_1 - x_0) + c_0(x_1 - x_0)^2 &= f(x_1), \\
a_1 &= f(x_1), \\
a_1 + b_1(x_2 - x_1) + c_1(x_2 - x_1)^2 &= f(x_2).
\end{aligned}
$$

- From condition ②, we must have $S_0'(x_1) = S_1'(x_1)$. By the fact that $S_0'(x) = b_0 + 2c_0(x - x_0)$ and $S_1'(x) = b_1 + 2c_1(x - x_1)$, we conclude that

$$b_0 + 2c_0(x_1 - x_0) = b_1 \,.$$

- 6 unknowns, 5 equations $\Rightarrow$ flexibility exists.
- If we require $S \in C^2([x_0, x_2])$, then $S_0''(x_1) = 2c_0, S_1''(x_1) = 2c_1$
  $\Rightarrow c_0 = c_1$
  $\Rightarrow$ 5 unknowns and 5 equations $\Rightarrow$ a solution may not exist!

§3.4 Spline Interpolation

Definition (Cubic spline)

Given $a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b$ and a set of function values $f(x_0), f(x_1), \cdots, f(x_n)$, a cubic spline interpolant S for f is a function that satisfies

1. $S|_{[x_j,x_{j+1}]}$ is a cubic polynomial for $j = 0, 1, \cdots, n-1$, denoted by $S|_{[x_j,x_{j+1}]}(x) = S_j(x)$;
2. $S(x_j) = f(x_j)$, $j = 0, 1, \cdots, n$;
3. $S_{j+1}(x_{j+1}) = S_j(x_{j+1})$, $j = 0, 1, \cdots, n-2$;
4. $S'_{j+1}(x_{j+1}) = S'_j(x_{j+1})$, $j = 0, 1, \cdots, n-2$;
5. $S''_{j+1}(x_{j+1}) = S''_j(x_{j+1})$, $j = 0, 1, \cdots, n-2$;
6. one of the following is satisfied:
   - $S''(x_0) = S''(x_n) = 0$, free or natural boundary conditions $\Rightarrow$ natural spline;
   - $S'(x_0) = f'(x_0), S'(x_n) = f'(x_n)$, clamped boundary conditions $\Rightarrow$ clamped spline.

§3.4 Spline Interpolation

Definition (Cubic spline)

Given $a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b$ and a set of function values $f(x_0), f(x_1), \cdots, f(x_n)$, a cubic spline interpolant S for f is a function that satisfies

1. $S|_{[x_j,x_{j+1}]}$ is a cubic polynomial for $j = 0, 1, \cdots, n-1$, denoted by $S|_{[x_j,x_{j+1}]}(x) = S_j(x)$;
2. $S(x_j) = f(x_j)$, $j = 0, 1, \cdots, n$;
3. $S_{j+1}(x_{j+1}) = S_j(x_{j+1})$, $j = 0, 1, \cdots, n-2$;
4. $S'_{j+1}(x_{j+1}) = S'_j(x_{j+1})$, $j = 0, 1, \cdots, n-2$;
5. $S''_{j+1}(x_{j+1}) = S''_j(x_{j+1})$, $j = 0, 1, \cdots, n-2$;
6. one of the following is satisfied:
   - $S''(x_0) = S''(x_n) = 0$, free or natural boundary conditions $\Rightarrow$ natural spline;
   - $S'(x_0) = f'(x_0), S'(x_n) = f'(x_n)$, clamped boundary conditions $\Rightarrow$ clamped spline.

§3.4 Spline Interpolation

- Condition (1) implies that

  $S_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3, j = 0, 1, \cdots, n-1.$

- Condition (2) $\Rightarrow S_j(x_j) = a_j = f(x_j)$ (given), $j = 0, 1, \cdots, n-1$.

  Define $a_n := S_{n-1}(x_n) = f(x_n)$ (given).

- Condition (3) implies that

$$\begin{aligned} a_{j+1} &= S_{j+1}(x_{j+1}) = S_j(x_{j+1}) \\ &= a_j + b_j(x_{j+1} - x_j) + c_j(x_{j+1} - x_j)^2 + d_j(x_{j+1} - x_j)^3 \end{aligned}$$

for $j = 0, 1, \cdots, n-2$. Define $h_j = x_{j+1} - x_j$, $j = 0, 1, \cdots, n-1$.

$$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3, \quad j = 0, 1, \cdots, n-2.$$

Moreover,

$$a_n = f(x_n) = S_{n-1}(x_n) = a_{n-1} + b_{n-1} h_{n-1} + c_{n-1} h_{n-1}^2 + d_{n-1} h_{n-1}^3.$$

§3.4 Spline Interpolation

- Condition (1) implies that

  $S_j(x) = a_j + b_j(x-x_j) + c_j(x-x_j)^2 + d_j(x-x_j)^3, j = 0, 1, \cdots, n-1.$

- Condition (2) $\Rightarrow S_j(x_j) = a_j = f(x_j)$ (given), $j = 0, 1, \cdots, n-1$.

  Define $a_n := S_{n-1}(x_n) = f(x_n)$ (given).

- Condition (3) implies that

$$\begin{aligned} a_{j+1} &= S_{j+1}(x_{j+1}) = S_j(x_{j+1}) \\ &= a_j + b_j(x_{j+1} - x_j) + c_j(x_{j+1} - x_j)^2 + d_j(x_{j+1} - x_j)^3 \end{aligned}$$

for $j = 0, 1, \cdots, n-2$. Define $h_j = x_{j+1} - x_j$, $j = 0, 1, \cdots, n-1$.

$$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3, \quad j = 0, 1, \cdots, n-2.$$

Moreover,

$$a_n = f(x_n) = S_{n-1}(x_n) = a_{n-1} + b_{n-1} h_{n-1} + c_{n-1} h_{n-1}^2 + d_{n-1} h_{n-1}^3 .$$

§3.4 Spline Interpolation

- Condition (1) implies that

  $S_j(x) = a_j + b_j(x-x_j) + c_j(x-x_j)^2 + d_j(x-x_j)^3, j = 0, 1, \cdots, n-1.$

- Condition (2) $\Rightarrow S_j(x_j) = a_j = f(x_j)$ (given), $j = 0, 1, \cdots, n-1.$

  Define $a_n := S_{n-1}(x_n) = f(x_n)$ (given).

- Condition (3) implies that

$$\begin{aligned} a_{j+1} &= S_{j+1}(x_{j+1}) = S_j(x_{j+1}) \\ &= a_j + b_j(x_{j+1}-x_j) + c_j(x_{j+1}-x_j)^2 + d_j(x_{j+1}-x_j)^3 \end{aligned}$$

for $j = 0, 1, \cdots, n-2$. Define $h_j = x_{j+1} - x_j$, $j = 0, 1, \cdots, n-1$.

$$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3, \quad j = 0, 1, \cdots, n-2.$$

Moreover,

$$a_n = f(x_n) = S_{n-1}(x_n) = a_{n-1} + b_{n-1}h_{n-1} + c_{n-1}h_{n-1}^2 + d_{n-1}h_{n-1}^3.$$

§3.4 Spline Interpolation

- Condition (1) implies that

  $$S_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3, j = 0, 1, \cdots, n-1.$$

- Condition (2) $\Rightarrow S_j(x_j) = a_j = f(x_j)$ (given), $j = 0, 1, \cdots, n-1$.
  Define $a_n := S_{n-1}(x_n) = f(x_n)$ (given).

- Condition (3) implies that

  $$\begin{aligned} a_{j+1} &= S_{j+1}(x_{j+1}) = S_j(x_{j+1}) \\ &= a_j + b_j(x_{j+1} - x_j) + c_j(x_{j+1} - x_j)^2 + d_j(x_{j+1} - x_j)^3 \end{aligned}$$

  for $j = 0, 1, \cdots, n-2$. Define $h_j = x_{j+1} - x_j$, $j = 0, 1, \cdots, n-1$.

  $$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3, \quad j = 0, 1, \cdots, n-2.$$

  Moreover,

  $$a_n = f(x_n) = S_{n-1}(x_n) = a_{n-1} + b_{n-1} h_{n-1} + c_{n-1} h_{n-1}^2 + d_{n-1} h_{n-1}^3 .$$

§3.4 Spline Interpolation

- Condition (1) implies that

  $$S_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3, j = 0, 1, \cdots, n-1.$$

- Condition (2) $\Rightarrow S_j(x_j) = a_j = f(x_j)$ (given), $j = 0, 1, \cdots, n-1$.

  Define $a_n := S_{n-1}(x_n) = f(x_n)$ (given).

- Condition (3) implies that

  $$\begin{aligned} a_{j+1} &= S_{j+1}(x_{j+1}) = S_j(x_{j+1}) \\ &= a_j + b_j(x_{j+1} - x_j) + c_j(x_{j+1} - x_j)^2 + d_j(x_{j+1} - x_j)^3 \end{aligned}$$

  for $j = 0, 1, \cdots, n-2$. Define $h_j = x_{j+1} - x_j$, $j = 0, 1, \cdots, n-1$.

  $$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3, \quad j = 0, 1, \cdots, n-2.$$

  Moreover,

  $$a_n = f(x_n) = S_{n-1}(x_n) = a_{n-1} + b_{n-1} h_{n-1} + c_{n-1} h_{n-1}^2 + d_{n-1} h_{n-1}^3 .$$

§3.4 Spline Interpolation

- Condition (1) implies that

  $S_j(x) = a_j + b_j(x-x_j) + c_j(x-x_j)^2 + d_j(x-x_j)^3, j = 0, 1, \cdots, n-1.$

- Condition (2) $\Rightarrow S_j(x_j) = a_j = f(x_j)$ (given), $j = 0, 1, \cdots, n-1$.

  Define $a_n := S_{n-1}(x_n) = f(x_n)$ (given).

- Condition (3) implies that

$$\begin{aligned} a_{j+1} &= S_{j+1}(x_{j+1}) = S_j(x_{j+1}) \\ &= a_j + b_j(x_{j+1} - x_j) + c_j(x_{j+1} - x_j)^2 + d_j(x_{j+1} - x_j)^3 \end{aligned}$$

for $j = 0, 1, \cdots, n-2$. Define $h_j = x_{j+1} - x_j$, $j = 0, 1, \cdots, n-1$.

$$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3, \quad j = 0, 1, \cdots, n-1.$$

Moreover, $\Uparrow$

$$a_n = f(x_n) = S_{n-1}(x_n) = a_{n-1} + b_{n-1} h_{n-1} + c_{n-1} h_{n-1}^2 + d_{n-1} h_{n-1}^3 .$$

§3.4 Spline Interpolation

- Condition (1) implies that

  $S_j(x) = a_j + b_j(x-x_j) + c_j(x-x_j)^2 + d_j(x-x_j)^3, j = 0, 1, \cdots, n-1.$

- Condition (2) $\Rightarrow S_j(x_j) = a_j = f(x_j)$ (given), $j = 0, 1, \cdots, n-1$.

  Define $a_n := S_{n-1}(x_n) = f(x_n)$ (given).

- Condition (3) implies that

$$\begin{aligned} a_{j+1} &= S_{j+1}(x_{j+1}) = S_j(x_{j+1}) \\ &= a_j + b_j(x_{j+1} - x_j) + c_j(x_{j+1} - x_j)^2 + d_j(x_{j+1} - x_j)^3 \end{aligned}$$

  for $j = 0, 1, \cdots, n-2$. Define $h_j = x_{j+1} - x_j$, $j = 0, 1, \cdots, n-1$.

$$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3, \quad j = 0, 1, \cdots, n-1.$$

§3.4 Spline Interpolation

- Note that

$$S_j'(x) = b_j + 2c_j(x - x_j) + 3d_j(x - x_j)^2, \quad j = 0, 1, \cdots, n-1\,;$$

thus $S_j'(x_j) = b_j$ for $j = 0, 1, \cdots, n-1$. Condition (4) implies that

$$b_{j+1} = S_{j+1}'(x_{j+1}) = S_j'(x_{j+1}) = b_j + 2c_jh_j + 3d_jh_j^2$$

for $j = 0, 1, \cdots, n-2$. Define $b_n := S'(x_n) = S_{n-1}'(x_n)$. Then

$$\begin{aligned} b_n = S_{n-1}'(x_n) &= b_{n-1} + 2c_{n-1}(x_n - x_{n-1}) + 3d_{n-1}(x_n - x_{n-1})^2 \\ &= b_{n-1} + 2c_{n-1}h_{n-1} + 3d_{n-1}h_{n-1}^2 . \end{aligned}$$

Therefore,

$$b_{j+1} = b_j + 2c_jh_j + 3d_jh_j^2, \quad j = 0, 1, \cdots, n-1.$$

§3.4 Spline Interpolation

- Note that

$$S_j'(x) = b_j + 2c_j(x - x_j) + 3d_j(x - x_j)^2, \quad j = 0, 1, \cdots, n-1\,;$$

thus $S_j'(x_j) = b_j$ for $j = 0, 1, \cdots, n-1$. Condition (4) implies that

$$b_{j+1} = S_{j+1}'(x_{j+1}) = S_j'(x_{j+1}) = b_j + 2c_jh_j + 3d_jh_j^2$$

for $j = 0, 1, \cdots, n-2$. Define $b_n := S'(x_n) = S_{n-1}'(x_n)$. Then

$$\begin{aligned} b_n = S_{n-1}'(x_n) &= b_{n-1} + 2c_{n-1}(x_n - x_{n-1}) + 3d_{n-1}(x_n - x_{n-1})^2 \\ &= b_{n-1} + 2c_{n-1}h_{n-1} + 3d_{n-1}h_{n-1}^2. \end{aligned}$$

Therefore,

$$b_{j+1} = b_j + 2c_jh_j + 3d_jh_j^2, \quad j = 0, 1, \cdots, n-1.$$

§3.4 Spline Interpolation

- Note that

$$S_j'(x) = b_j + 2c_j(x - x_j) + 3d_j(x - x_j)^2, \quad j = 0, 1, \cdots, n-1;$$

thus $S_j'(x_j) = b_j$ for $j = 0, 1, \cdots, n-1$. Condition (4) implies that

$$b_{j+1} = S_{j+1}'(x_{j+1}) = S_j'(x_{j+1}) = b_j + 2c_jh_j + 3d_jh_j^2$$

for $j = 0, 1, \cdots, n-2$. Define $b_n := S'(x_n) = S_{n-1}'(x_n)$. Then

$$\begin{aligned} b_n = S_{n-1}'(x_n) &= b_{n-1} + 2c_{n-1}(x_n - x_{n-1}) + 3d_{n-1}(x_n - x_{n-1})^2 \\ &= b_{n-1} + 2c_{n-1}h_{n-1} + 3d_{n-1}h_{n-1}^2. \end{aligned}$$

Therefore,

$$b_{j+1} = b_j + 2c_jh_j + 3d_jh_j^2, \quad j = 0, 1, \cdots, n-1.$$

§3.4 Spline Interpolation

- Note that

$$S_j''(x) = 2c_j + 6d_j(x - x_j), \quad j = 0, 1, \cdots, n-1;$$

thus $S_j''(x_j) = 2c_j$ for $j = 0, 1, \cdots, n-1$. Condition (5) implies that

$$2c_{j+1} = S_{j+1}''(x_{j+1}) = S_j''(x_{j+1}) = 2c_j + 6d_jh_j$$

for $j = 0, 1, \cdots, n-2$. Therefore,

$$c_{j+1} = c_j + 3d_jh_j, \quad j = 0, 1, \cdots, n-2.$$

Define $c_n := \frac{1}{2}S_{n-1}''(x_n)$. Then

$$c_n = \frac{1}{2}(2c_{n-1} + 6d_{n-1}(x_n - x_{n-1})) = c_{n-1} + 3d_{n-1}h_{n-1}.$$

Therefore,

$$c_{j+1} = c_j + 3d_jh_j, \quad j = 0, 1, \cdots, n-1.$$

§3.4 Spline Interpolation

- Note that

$$S_j''(x) = 2c_j + 6d_j(x - x_j), \quad j = 0, 1, \cdots, n-1;$$

thus $S_j''(x_j) = 2c_j$ for $j = 0, 1, \cdots, n-1$. Condition (5) implies that

$$2c_{j+1} = S_{j+1}''(x_{j+1}) = S_j''(x_{j+1}) = 2c_j + 6d_jh_j$$

for $j = 0, 1, \cdots, n-2$. Therefore,

$$c_{j+1} = c_j + 3d_jh_j, \quad j = 0, 1, \cdots, n-2.$$

Define $c_n := \frac{1}{2}S_{n-1}''(x_n)$. Then

$$c_n = \frac{1}{2}(2c_{n-1} + 6d_{n-1}(x_n - x_{n-1})) = c_{n-1} + 3d_{n-1}h_{n-1}.$$

Therefore,

$$c_{j+1} = c_j + 3d_jh_j, \quad j = 0, 1, \cdots, n-1.$$

§3.4 Spline Interpolation

- Note that

$$S_j''(x) = 2c_j + 6d_j(x - x_j), \quad j = 0, 1, \cdots, n-1;$$

thus $S_j''(x_j) = 2c_j$ for $j = 0, 1, \cdots, n-1$. Condition (5) implies that

$$2c_{j+1} = S_{j+1}''(x_{j+1}) = S_j''(x_{j+1}) = 2c_j + 6d_jh_j$$

for $j = 0, 1, \cdots, n-2$. Therefore,

$$c_{j+1} = c_j + 3d_jh_j, \quad j = 0, 1, \cdots, n-2.$$

Define $c_n := \frac{1}{2}S_{n-1}''(x_n)$. Then

$$c_n = \frac{1}{2}(2c_{n-1} + 6d_{n-1}(x_n - x_{n-1})) = c_{n-1} + 3d_{n-1}h_{n-1}.$$

Therefore,

$$c_{j+1} = c_j + 3d_jh_j, \quad j = 0, 1, \cdots, n-1.$$

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3 \qquad (\star)$$

$$b_{j+1} = b_j + 2c_j h_j + 3d_j h_j^2 \qquad (\star\star)$$

$$c_{j+1} = c_j + 3d_j h_j \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using $(\star\star\star)$ in $(\star)$ and $(\star\star)$, we find that for $j = 0, 1, \cdots, n-1$,

$$\begin{aligned} a_{j+1} &= a_j + b_j h_j + c_j h_j^2 + \frac{c_{j+1} - c_j}{3h_j} h_j^3 \\ &= a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2, \\ b_{j+1} &= b_j + 2c_j h_j + 3\frac{c_{j+1} - c_j}{3h_j} h_j^2 \\ &= b_j + (c_j + c_{j+1}) h_j. \end{aligned}$$

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3 \qquad (\star)$$

$$b_{j+1} = b_j + 2c_j h_j + 3d_j h_j^2 \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using $(\star\star\star)$ in $(\star)$ and $(\star\star)$, we find that for $j = 0, 1, \cdots, n-1$,

$$\begin{aligned} a_{j+1} &= a_j + b_j h_j + c_j h_j^2 + \frac{c_{j+1} - c_j}{3h_j} h_j^3 \\ &= a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 , \\ b_{j+1} &= b_j + 2c_j h_j + 3\frac{c_{j+1} - c_j}{3h_j} h_j^2 \\ &= b_j + (c_j + c_{j+1}) h_j . \end{aligned}$$

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3 \qquad (\star)$$

$$b_{j+1} = b_j + 2c_j h_j + 3d_j h_j^2 \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using $(\star\star\star)$ in $(\star)$ and $(\star\star)$, we find that for $j = 0, 1, \cdots, n-1$,

$$\begin{aligned} a_{j+1} &= a_j + b_j h_j + c_j h_j^2 + \frac{c_{j+1} - c_j}{3h_j} h_j^3 \\ &= a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2, \\ b_{j+1} &= b_j + 2c_j h_j + 3\frac{c_{j+1} - c_j}{3h_j} h_j^2 \\ &= b_j + (c_j + c_{j+1}) h_j. \end{aligned}$$

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using $(\star\star\star)$ in $(\star)$ and $(\star\star)$, we find that for $j = 0, 1, \cdots, n-1$,

$$\begin{aligned} a_{j+1} &= a_j + b_j h_j + c_j h_j^2 + \frac{c_{j+1} - c_j}{3h_j} h_j^3 \\ &= a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 , \\ b_{j+1} &= b_j + 2c_j h_j + 3\frac{c_{j+1} - c_j}{3h_j} h_j^2 \\ &= b_j + (c_j + c_{j+1}) h_j . \end{aligned}$$

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$.

We note that b_j, d_j for $j = 0, 1, \cdots, n-1$ can be computed using $(\star\star)$ and $(\star\star\star)$ as long as all c_j's are obtained. Therefore, we focus on solving for c_j's.

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Rearranging terms in $(\star)$, we find that

$$b_j h_j = (a_{j+1} - a_j) - \frac{2c_j + c_{j+1}}{3} h_j^2 \quad \text{for } j = 0, 1, \cdots, n-1$$

thus

$$b_j = \frac{a_{j+1} - a_j}{h_j} - \frac{(2c_j + c_{j+1}) h_j}{3} \qquad \text{for } j = 0, 1, \cdots, n-1, \qquad (\square)$$

$$b_{j+1} = \frac{a_{j+2} - a_{j+1}}{h_{j+1}} - \frac{(2c_{j+1} + c_{j+2}) h_{j+1}}{3} \qquad \text{for } j = -1, 0, \cdots, n-2. \qquad (\circ)$$

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Rearranging terms in $(\star)$, we find that

$$b_j h_j = (a_{j+1} - a_j) - \frac{2c_j + c_{j+1}}{3} h_j^2 \quad \text{for } j = 0, 1, \cdots, n-1$$

thus

$$b_j = \frac{a_{j+1} - a_j}{h_j} - \frac{(2c_j + c_{j+1}) h_j}{3} \qquad \text{for } j = 0, 1, \cdots, n-1\,, \qquad (\square)$$

$$b_{j+1} = \frac{a_{j+2} - a_{j+1}}{h_{j+1}} - \frac{(2c_{j+1} + c_{j+2}) h_{j+1}}{3} \qquad \text{for } j = -1, 0, \cdots, n-2. \qquad (\circ)$$

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using $(\square)$ and $(\circ)$ in $(\star\star)$, we find that

$$\frac{a_{j+2} - a_{j+1}}{h_{j+1}} - \frac{(2c_{j+1} + c_{j+2})h_{j+1}}{3} = \frac{a_{j+1} - a_j}{h_j} - \frac{(2c_j + c_{j+1})h_j}{3} + (c_j + c_{j+1})h_j$$

for $j = 0, 1, \cdots, n-2$. $\Uparrow$

$$b_j = \frac{a_{j+1} - a_j}{h_j} - \frac{(2c_j + c_{j+1})h_j}{3} \qquad \text{for } j = 0, 1, \cdots, n-1, \qquad (\square)$$

$$b_{j+1} = \frac{a_{j+2} - a_{j+1}}{h_{j+1}} - \frac{(2c_{j+1} + c_{j+2})h_{j+1}}{3} \qquad \text{for } j = -1, 0, \cdots, n-2. \qquad (\circ)$$

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using $(\square)$ and $(\circ)$ in $(\star\star)$, we find that

$$\frac{a_{j+2} - a_{j+1}}{h_{j+1}} - \frac{(2c_{j+1} + c_{j+2})h_{j+1}}{3} = \frac{a_{j+1} - a_j}{h_j} - \frac{(2c_j + c_{j+1})h_j}{3} + \frac{3(c_j + c_{j+1})h_j}{3}$$

for $j = 0, 1, \cdots, n-2$.

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using ($\square$) and ($\circ$) in ($\star\star$), we find that

$$\frac{a_{j+2} - a_{j+1}}{h_{j+1}} - \frac{(2c_{j+1} + c_{j+2}) h_{j+1}}{3} = \frac{a_{j+1} - a_j}{h_j} + \frac{(c_j + 2c_{j+1}) h_j}{3}$$

for $j = 0, 1, \cdots, n-2$.

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using $(\square)$ and $(\circ)$ in $(\star\star)$, we find that

$$\frac{a_{j+2} - a_{j+1}}{h_{j+1}} - \frac{a_{j+1} - a_j}{h_j} = \frac{(2c_{j+1} + c_{j+2}) h_{j+1}}{3} + \frac{(c_j \quad + 2c_{j+1}) h_j}{3}$$

for $j = 0, 1, \cdots, n-2$.

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using $(\square)$ and $(\circ)$ in $(\star\star)$, we find that

$$\frac{a_{j+1} - a_j}{h_j} - \frac{a_j - a_{j-1}}{h_{j-1}} = \frac{(2c_j + c_{j+1})h_j}{3} + \frac{(c_{j-1} + 2c_j)h_{j-1}}{3}$$

for $j = 1, 2, \cdots, n-1$.

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using ($\square$) and ($\circ$) in ($\star\star$), we find that

$$\frac{(2c_j + c_{j+1})h_j}{3} + \frac{(c_{j-1} + 2c_j)h_{j-1}}{3} = \frac{a_{j+1} - a_j}{h_j} - \frac{a_j - a_{j-1}}{h_{j-1}}$$

for $j = 1, 2, \cdots, n-1$.

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using $(\square)$ and $(\circ)$ in $(\star\star)$, we find that

$$h_{j-1} c_{j-1} + 2(h_j + h_{j-1}) c_j + h_j c_{j+1} = \frac{3(a_{j+1} - a_j)}{h_j} - \frac{3(a_j - a_{j-1})}{h_{j-1}}$$

for $j = 1, 2, \cdots, n-1$. Hence, we have now $(n+1)$ unknowns with $(n-1)$ equations.

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using $(\square)$ and $(\circ)$ in $(\star\star)$, we find that

$$h_{j-1} c_{j-1} + 2(h_j + h_{j-1}) c_j + h_j c_{j+1} = \frac{3(a_{j+1} - a_j)}{h_j} - \frac{3(a_j - a_{j-1})}{h_{j-1}} =: g_j$$

for $j = 1, 2, \cdots, n-1$. Hence, we have now $(n+1)$ unknowns with $(n-1)$ equations.

§3.4 Spline Interpolation

- As a summary, the unknowns b_j, c_j, d_j for $j = 0, 1, \cdots, n-1$ and b_n, c_n satisfy

$$a_{j+1} = a_j + b_j h_j + \frac{2c_j + c_{j+1}}{3} h_j^2 \qquad (\star)$$

$$b_{j+1} = b_j + (c_j + c_{j+1}) h_j \qquad (\star\star)$$

$$d_j = \frac{c_{j+1} - c_j}{3h_j} \qquad (\star\star\star)$$

for $j = 0, 1, \cdots, n-1$. Using ($\square$) and ($\circ$) in ($\star\star$), we find that

$$h_{j-1} c_{j-1} + 2(h_j + h_{j-1}) c_j + h_j c_{j+1} = \frac{3(a_{j+1} - a_j)}{h_j} - \frac{3(a_j - a_{j-1})}{h_{j-1}} =: g_j$$

for $j = 1, 2, \cdots, n-1$. Hence, we have now $(n+1)$ unknowns with $(n-1)$ equations.

§3.4 Spline Interpolation

- **Natural boundary condition**:

  Condition ⑥(i) ($S''(x_0) = S''(x_n) = 0$) implies that $c_0 = c_n = 0$; thus $(n-1)$ unknowns with $(n-1)$ equations. The resulting linear system is strictly diagonally dominant; thus **a unique natural spline exists**.

$$\begin{bmatrix} 2(h_1+h_0) & h_1 & 0 & \cdots\cdots & \cdots\cdots & \cdots\cdots & 0 \\ h_1 & 2(h_2+h_1) & h_2 & 0 & \cdots\cdots & \cdots\cdots & 0 \\ 0 & h_2 & 2(h_3+h_2) & h_3 & 0 & \cdots\cdots & 0 \\ \vdots & & & & & & \vdots \\ \vdots & & 0 & h_{n-4} & 2(h_{n-3}+h_{n-4}) & h_{n-3} & 0 \\ \vdots & & & 0 & h_{n-3} & 2(h_{n-2}+h_{n-3}) & h_{n-2} \\ 0 & \cdots\cdots & \cdots\cdots & \cdots\cdots & 0 & h_{n-2} & 2(h_{n-1}+h_{n-2}) \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ \vdots \\ \vdots \\ c_{n-2} \\ c_{n-1} \end{bmatrix} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ \vdots \\ \vdots \\ g_{n-2} \\ g_{n-1} \end{bmatrix}.$$

§3.4 Spline Interpolation

- **Natural boundary condition**:

  Condition ⑥(i) ($S''(x_0) = S''(x_n) = 0$) implies that $c_0 = c_n = 0$; thus $(n-1)$ unknowns with $(n-1)$ equations. The resulting linear system is strictly diagonally dominant; thus **a unique natural spline exists**.

$$\begin{bmatrix} 2(h_1+h_0) & h_1 & 0 & \cdots\cdots & \cdots\cdots & \cdots\cdots & 0 \\ h_1 & 2(h_2+h_1) & h_2 & 0 & \cdots\cdots & \cdots\cdots & 0 \\ 0 & h_2 & 2(h_3+h_2) & h_3 & 0 & \cdots\cdots & 0 \\ \vdots & & & & & & \vdots \\ \vdots & & 0 & h_{n-4} & 2(h_{n-3}+h_{n-4}) & h_{n-3} & 0 \\ \vdots & & & 0 & h_{n-3} & 2(h_{n-2}+h_{n-3}) & h_{n-2} \\ 0 & \cdots\cdots & \cdots\cdots & \cdots\cdots & 0 & h_{n-2} & 2(h_{n-1}+h_{n-2}) \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ \vdots \\ \vdots \\ c_{n-2} \\ c_{n-1} \end{bmatrix} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ \vdots \\ \vdots \\ g_{n-2} \\ g_{n-1} \end{bmatrix}.$$

§3.4 Spline Interpolation

- **Clamped boundary condition**:

Consider the clamped boundary condition ⑥(ii) ($S'(x_0) = f'(x_0)$ and $S'(x_n) = f'(x_n)$). Note that $b_0 = S'(x_0)$ and $b_n = S'(x_n)$ are now treated as given. Recall that for $j = 0, 1, \cdots, n-1$,

$$b_{j+1} = b_j + (c_j + c_{j+1})h_j \qquad (\star\star)$$

$$b_j = \frac{a_{j+1} - a_j}{h_j} - \frac{(2c_j + c_{j+1})h_j}{3} \qquad (\square)$$

1. For $j = 0$, $(\square)$ provides an additional equation

$$2h_0 c_0 + h_0 c_1 = \frac{3(a_1 - a_0)}{h_0} - 3b_0 =: g_0 .$$

2. For $j = n-1$, $(\square)$ and $(\star\star)$ imply that

$$2h_{n-1}c_{n-1} + h_{n-1}c_n = \frac{3(a_n - a_{n-1})}{h_{n-1}} - 3b_{n-1}$$

which provides another addition equation

$$h_{n-1}c_{n-1} + 2h_{n-1}c_n = \frac{3(a_n - a_{n-1})}{h_{n-1}} - 3b_n =: g_n .$$

§3.4 Spline Interpolation

- **Clamped boundary condition**:

Consider the clamped boundary condition ⑥(ii) ($S'(x_0) = f'(x_0)$ and $S'(x_n) = f'(x_n)$). Note that $b_0 = S'(x_0)$ and $b_n = S'(x_n)$ are now treated as given. Recall that for $j = 0, 1, \cdots, n-1$,

$$b_{j+1} = b_j + (c_j + c_{j+1})h_j \qquad (\star\star)$$

$$b_j = \frac{a_{j+1} - a_j}{h_j} - \frac{(2c_j + c_{j+1})h_j}{3} \qquad (\square)$$

1. For $j = 0$, ($\square$) provides an additional equation

$$2h_0c_0 + h_0c_1 = \frac{3(a_1 - a_0)}{h_0} - 3b_0 =: g_0\,.$$

2. For $j = n-1$, ($\square$) and ($\star\star$) imply that

$$2h_{n-1}c_{n-1} + h_{n-1}c_n = \frac{3(a_n - a_{n-1})}{h_{n-1}} - 3\big[b_n - (c_{n-1} + c_n)h_{n-1}\big]$$

which provides another addition equation

$$h_{n-1}c_{n-1} + 2h_{n-1}c_n = \frac{3(a_n - a_{n-1})}{h_{n-1}} - 3b_n =: g_n\,.$$

§3.4 Spline Interpolation

- **Clamped boundary condition**:

Consider the clamped boundary condition ⑥(ii) ($S'(x_0) = f'(x_0)$ and $S'(x_n) = f'(x_n)$). Note that $b_0 = S'(x_0)$ and $b_n = S'(x_n)$ are now treated as given. Recall that for $j = 0, 1, \cdots, n-1$,

$$b_{j+1} = b_j + (c_j + c_{j+1})h_j \qquad (\star\star)$$

$$b_j = \frac{a_{j+1} - a_j}{h_j} - \frac{(2c_j + c_{j+1})h_j}{3} \qquad (\square)$$

1. For $j = 0$, ($\square$) provides an additional equation

$$2h_0 c_0 + h_0 c_1 = \frac{3(a_1 - a_0)}{h_0} - 3b_0 =: g_0 .$$

2. For $j = n-1$, ($\square$) and ($\star\star$) imply that

$$2h_{n-1}c_{n-1} + h_{n-1}c_n = \frac{3(a_n - a_{n-1})}{h_{n-1}} - 3\left[b_n - (c_{n-1} + c_n)h_{n-1}\right]$$

which provides another addition equation

$$h_{n-1}c_{n-1} + 2h_{n-1}c_n = \frac{3(a_n - a_{n-1})}{h_{n-1}} - 3b_n =: g_n .$$

§3.4 Spline Interpolation

Therefore, we have $(n+1)$ unknowns with $(n+1)$ equations. The resulting linear system is again strictly diagonally dominant; thus **a unique clamped spline exists**.

$$\begin{bmatrix} 2h_0 & h_0 & 0 & \cdots & \cdots & \cdots & 0 \\ h_0 & 2(h_1+h_0) & h_1 & 0 & \cdots & \cdots & 0 \\ 0 & h_1 & 2(h_2+h_1) & h_2 & 0 & \cdots & 0 \\ \vdots & & & & & & \vdots \\ \vdots & & 0 & h_{n-3} & 2(h_{n-2}+h_{n-3}) & h_{n-2} & 0 \\ \vdots & & & 0 & h_{n-2} & 2(h_{n-1}+h_{n-2}) & h_{n-1} \\ 0 & \cdots & \cdots & \cdots & 0 & h_{n-1} & 2h_{n-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ \vdots \\ c_{n-1} \\ c_n \end{bmatrix} = \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ \vdots \\ g_{n-1} \\ g_n \end{bmatrix}.$$