Interpolation

Interpolation Error

- Lagrange form of interpolating polynomial.
  (Has a simple form and useful for the error estimation.)

Derive an interpolating polynomial for points, $(x_i, f_i), \; i = 0, \cdots n, \quad f_i := f(x_i)$

Defining the Lagrange polynomial by

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

Lagrange form of interpolating polynomial is written

$$p_n(x) = \sum_{i=0}^{n} L_{n,i}(x) \, f_i$$

Theorem: (Interpolation Error)
If a function f is continuous on [a,b] and has n+1 continuous derivatives on (a,b), then for 8 x2[a,b], 9 ξ(x)2(a,b), such that

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

- Newton form of interpolating polynomial is written

$$p_{0,\cdots,n}(x) = f[x_0] + \sum_{k=1}^{n} f[x_0, x_1, \cdots, x_k] \prod_{i=0}^{k-1} (x - x_i).$$

namely, $a_k = f[x_0, x_1, \cdots, x_k]$

- Interpolation error in Newton form can be derived as follows:

If $f(x) \in C^{n+1}(a,b)$, $\exists p_{0,\cdots,n+1}(x)$ with abscissas $x_0, \cdots, x_n, t$

$$p_{0,\cdots,n+1}(x) = p_{0,\cdots,n}(x) + f[x_0, x_1, \cdots, x_n, t] \prod_{i=0}^{n} (x - x_i).$$

At the point t, $f(t) = p_{0,\cdots,n+1}(t)$. Writing $t = x$,

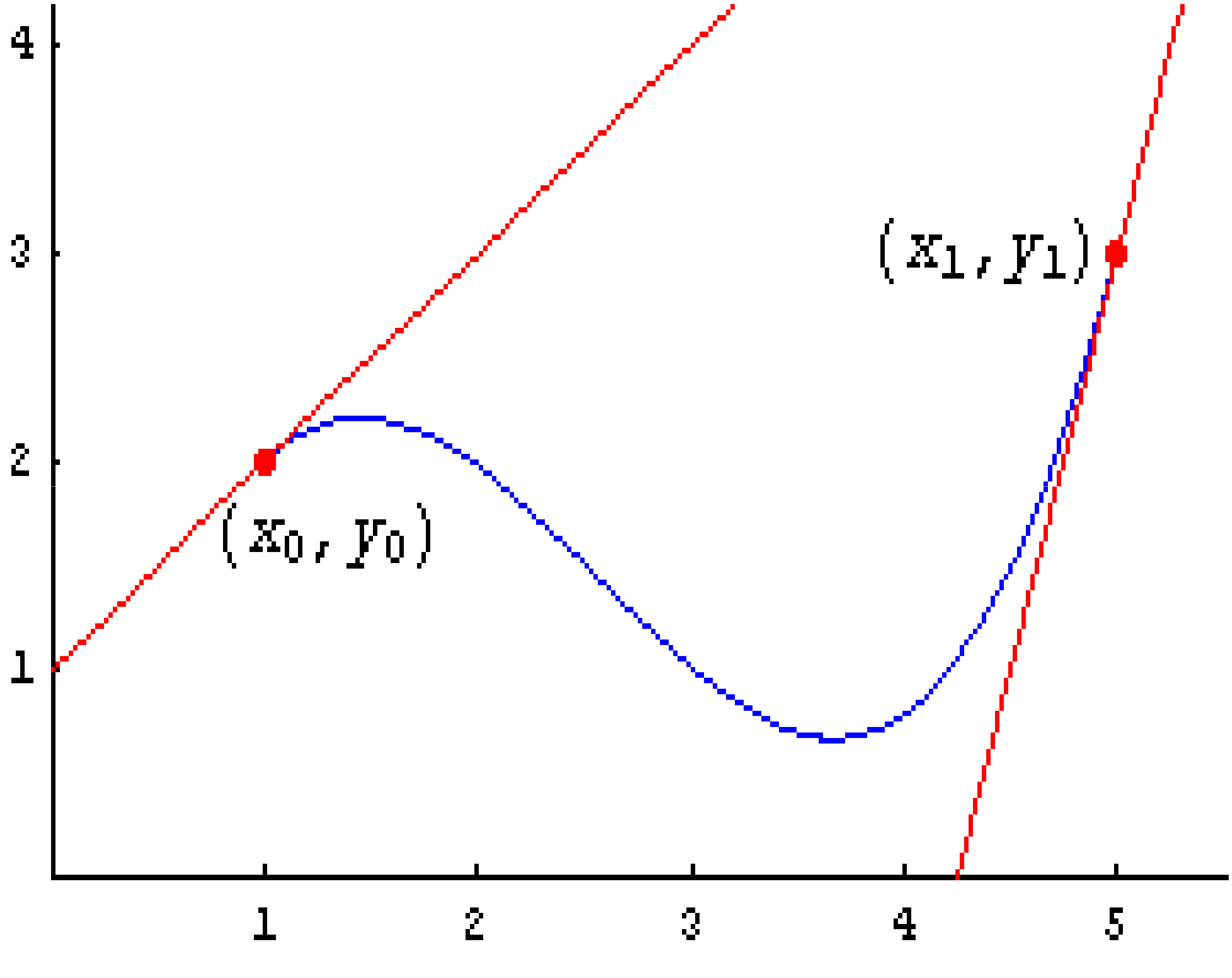
$$f(x) - p_{0,\cdots,n}(x) = f[x_0, \cdots, x_n, x] \prod_{i=0}^{n} (x - x_i).$$

Important Theorem

Theorem: Let f be n times continuously differentiable on $[a, b]$, and let $x_0, x_1, \ldots\ldots\ldots \ldots\ldots\ldots x_n$ be distinct points in $[a, b]$ .then there exists a number $c_x \in [a, b]$ such that

$$f[x_0, x_1, \ldots\ldots\ldots\ldots\ldots\ldots x_n] = \frac{f^n(c_x)}{n!}$$

French Mathematician
Charles Hermite
1822 - 1901


4
3
2
1
1
2
3
4
5
(x_0, y_0)
(x_1, y_1)

Defining Hermite polynomials

The theorem that states the existence of a Hermite polynomial is also the theorem that precisely constructs the polynomial.

Theorem If $f \in C^1[a,b]$ and $x_0,\ldots,x_n \in [a,b]$ are distinct, the unique polynomial of at least degree agreeing f and f' at $x_0,\ldots,x_n$ is the Hermite polynomial of degree at most $2n+1$ given by

$$H_{2n+1}(x) \;=\; \sum_{j=0}^{n} \underset{\substack{\uparrow \\ \text{matching } f(x)}}{f(x_j)} H_{n,j}(x) \;+\; \sum_{j=0}^{n} \underset{\substack{\uparrow \\ \text{matching } f'(x)}}{f'(x_j)} \hat{H}_{n,j}(x)$$

where $H_{n,j}(x) = [1-2(x-x_j)L'_{n,j}(x_j)]L^2_{n,j}(x)$

and $\hat{H}_{n,j}(x) = (x-x_j)L^2_{n,j}(x)$.

Finding Hermite polynomials

The theorem we just looked at defined Hermite polynomials. While the definition enables us to quickly analyze the behavior of the polynomial for theoretical purposes, it is difficult to construct the polynomials this way.

An alternative method has as its basis the Newton interpolatory divided-difference formula for the Lagrange polynomial.

An adaptation of the divided difference method that we looked at earlier allows for quick computation of the coefficients of Hermite polynomials. It is this method that you would code and use by hand.

We take our points $x_0, x_1, \ldots, x_n$ and create a new sequence $z0, z_1, \ldots z_{2n+1}$ by

$$z_{2i} = z_{2i+1} = x_i \text{ for each } i \in [0, n]$$

Note that $f[z_{2i}, z_{2i+1}]$ is not defined. In its place we use $f'(x_i)$. Let us look at this with the divided-difference table.

The Hermite interpolation polynomial is given by

$$
\begin{aligned}
P(x) \quad = \quad & f(x_0) + f[x_0, x_0](x - x_0) + f[x_0, x_0, x_1](x - x_0)^2 + \\
& + f[x_0, x_0, x_1, x1](x - x_0)^2(x - x_1) + \\
& + f[x_0, x_0, x_1, x_1, x_2](x - x_0)^2(x - x_1)^2 \\
& \cdots \\
& + f[x_0, x_0, x_1, x_1, \ldots, x_n, x_n](x - x_0)^2(x - x_1)^2 \cdots (x - x_n)
\end{aligned}
$$

Divided Differences for Hermite interpolation problems

- Question: what happens in the divided difference scheme, if you have twice the same point?

$$\begin{array}{cc|l} x_0 & f[x_0] & ? \\ x_0 & f[x_0] & \end{array}$$

$$f[x_0, x_1] = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \quad \rightarrow \quad f[x_0, x_0] = \frac{f(x_0) - f(x_0)}{x_0 - x_0}$$

However:

$$\lim_{x \to x_0} f[x_0, x] = \frac{f(x) - f(x_0)}{x - x_0} = f'(x_0)$$

Divided Differences for Hermite interpolation problems (II)

- Thus

$$\begin{array}{cc|c} x_0 & f[x_0] & f'(x_0) \\ x_0 & f[x_0] & \end{array}$$

- Example: $p(x_0) = c_{00}$ $p'(x_0) = c_{01}$ $p(x_1) = c_{10}$ $p'(x_1) = c_{11}$

$$\begin{array}{cc|cccc} x_0 & c_{00} & c_{01} & f[x_0, x_0, x_1] & f[x_0, x_0, x_1, x_1] \\ x_0 & c_{00} & f[x_0, x_1] & f[x_0, x_1, x_1] & \\ x_1 & c_{10} & c_{11} & & \\ x_1 & c_{10} & & & \end{array}$$

Example

- Use the extended divided difference algorithm to determine a polynomial that takes the following values:

$$p(1)=2 \quad p'(1)=3 \quad p(2)=6 \quad p'(2)=7 \quad p''(2)=8$$

$$\begin{array}{cc|cccc} 1 & 2 & 3 & ? & ? & ? \\ 1 & 2 & ? & ? & ? & \\ 2 & 6 & 7 & 4 & & \\ 2 & 6 & & & & \\ 2 & 6 & & & & \end{array} \quad \rightarrow \quad \begin{array}{cccccc} 1 & 2 & 3 & 1 & 2 & -1 \\ 1 & 2 & 4 & 3 & 1 & \\ 2 & 6 & 7 & 4 & & \\ 2 & 6 & & & & \\ 2 & 6 & & & & \end{array}$$

$$p(x)=2+3(x-1)+(x-1)^2+2(x-1)^2(x-2)-(x-1)^2(x-2)^2$$

Hermite Interpolation

- Hermite interpolation allows us to find a ploynomial that matched both <u>function value</u> and some of the <u>derivative values</u>

The following table illustrates the interpolation method:

x_i	$f(x_i)$	$f[x_i, x_{i+1}]$	$D^2 f[x_i]$	$D^3 f[x_i]$	$D^4 f[x_i]$	$\cdots$
x_0	y_0					
		$f'(x_0)$				
x_0	y_0		$f[x_0, x_0, x_1]$			
		$f[x_0, x_1]$		$f[x_0, x_0, x_1, x_1]$		
x_1	y_1		$f[x_0, x_1, x_1]$		$f[x_0, x_0, x_1, x_1, x_2]$	$\cdots$
		$f'(x_1)$		$f[x_0, x_1, x_1, x_2]$		
x_1	y_1		$f[x_1, x_1, x_2]$		$f[x_0, x_1, x_1, x_2, x_2]$	$\cdots$
		$f[x_1, x_2]$		$f[x_0, x_1, x_1, x_2]$		
x_2	y_2		$f[x_1, x_2, x_2]$		$f[x_1, x_1, x_2, x_2, x_3]$	$\cdots$
		$f'(x_2)$		$f[x_1, x_1, x_2, x_2]$		
x_2	y_2		$f[x_2, x_2, x_3]$		$f[x_1, x_2, x_2, x_3, x_3]$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	

Example Use the modified divided difference basis to construct the Hermite interpolant to set $S = \{(1,2,1),(3,1,-1),(4,2,0)\}$.

The corresponding divided difference table is

x	0^{th}DD	1^{st}DD	2^{nd}DD	3^{rd}DD	4^{th}DD	5^{th}DD
1	2					
		[1]				
1	2		$-3/4$			
		$-1/2$		$1/4$		
3	1		$-1/4$		$1/6$	
		[-1]		$3/4$		$-34/72$
3	1		2		$-5/4$	
		1		-3		
4	2		-1			
		[0]				
4	2					

Entries in the boxes are derivative data values.

It follows that the Hermite interpolating polynomial is

$$H_5(x) = 2 + 1(x-1) - \frac{3}{4}(x-1)^2 + \frac{1}{4}(x-1)^2(x-3) + \frac{1}{6}(x-1)^2(x-3)^2 - \frac{34}{72}(x-1)^2(x-3)^2(x-4)$$

++

Hermite Interpolation Exercises[1]

In the following use the generalization of divided differences to compute any Hermite interpolation polynomial. Show the all divided difference tables for each problem.

1. The following sample of function f is given.

x	f(x)	f′(x)
-1	-.79	-.15
0	-1.0	-1.0
1	-14.6	-49.0

a) Approximate f(0.5) using a polynomial interpolant.

b) Approximate f(0.5) using a Hermite interpolant.

c) The function sample is from $f(x) = 2xe^{x} - e^{3x}$. Compute the absolute error in each of the results from parts a and b.

d) Graph the absolute error expressions for parts a and b over [-1,1]. Comment on the accuracy of part a vs that in part b.

The Error In Hermite Interpolation

Theorem Let f(x) be in C^{2n+2}[a, b] and let all the x-coordinates of t he data be in [a, b]. For Hermite interpolant $\mathbf{H_{2n+1}(x)}$ to the data set

$$\mathbf{S} = \{(\mathbf{x_i}, \mathbf{f(x_i)}, \mathbf{f'(x_i)}) \mid \mathbf{i} = 0, 1, 2, \ldots, \mathbf{n}\}$$

of 2n + 1 distinct points for each x in [a, b] there exist an $\alpha_{\mathbf{x}}$ in [a, b] such that

$$\mathbf{f(x)} - \mathbf{H_{2n+1}(x)} = \frac{\prod_{k=0}^{n}(\mathbf{x} - \mathbf{x_k})^2}{(2\mathbf{n}+2)!}\mathbf{f}^{(2n+2)}(\alpha_{\mathbf{x}}).$$

Example Let $f(x)=\sin(x)$. Construct the Hermite data set for $x=0,\pi/2$, the Hermite interpolant to the data and find a least upper bound on the error.

The data set is
$\{(0,0,1),(\pi/2,1,0)\}$.
The divided difference table is ➔

x	0^{th}DD	1^{st}DD	2^{nd}DD	3^{rd}DD
0	0			
		1		
0	0		−0.2313	
		$2/\pi$		−0.1107
$\pi/2$	1		−0.4053	
		0		
$\pi/2$	1			

The Hermite interpolant is

$$H_3(x)=0+1(x-0)-0.2313(x-0)^2-0.1107(x-0)^2(x-\pi/2).$$

The error is given by $E(x) = \frac{(x-0)^2 (x-\pi/2)^2}{4!} f^{(4)}(\alpha_x)$. The fourth derivative of f(x) is sin(x) so we have

$$|E(x)| = \left|\frac{(x-0)^2 (x-\pi/2)^2}{4!}\right| \left|f^{(4)}(\alpha_x)\right| \le \frac{1}{24} \max_{x \in [0,\pi/2]} \left|(x-0)^2 (x-\pi/2)^2\right| \max_{x \in [0,\pi/2]} |\sin(x)|$$

$$\le \frac{1}{24} \max_{x \in [0,\pi/2]} \left|(x-0)^2 (x-\pi/2)^2\right|$$

One way to proceed is to plot $(x-0)^2 (x-\pi/2)^2$ over [0, π/2] and estimate its max in absolute value. Alternatively we could use calculus with max/min techniques. From the

graph it is rather easy to observe that the max is at x = π/4. It follows that

$$|E(x)| \le \left(\frac{\pi}{4}\right)^4 \frac{1}{24} \approx 0.01585 .$$

Runge's phenomenon

Carle David Tolmé Runge (August 30, 1856 – January 3, 1927) was a German mathematician, physicist, and spectroscopist

Nonconvergence

Polynomial interpolating an underlying continuous function at equally spaced points may not converge to function as number of data points (and hence **polynomial degree) increases**, as illustrated by **Runge's function.** $f(x) = \frac{1}{1+25x^2}$.

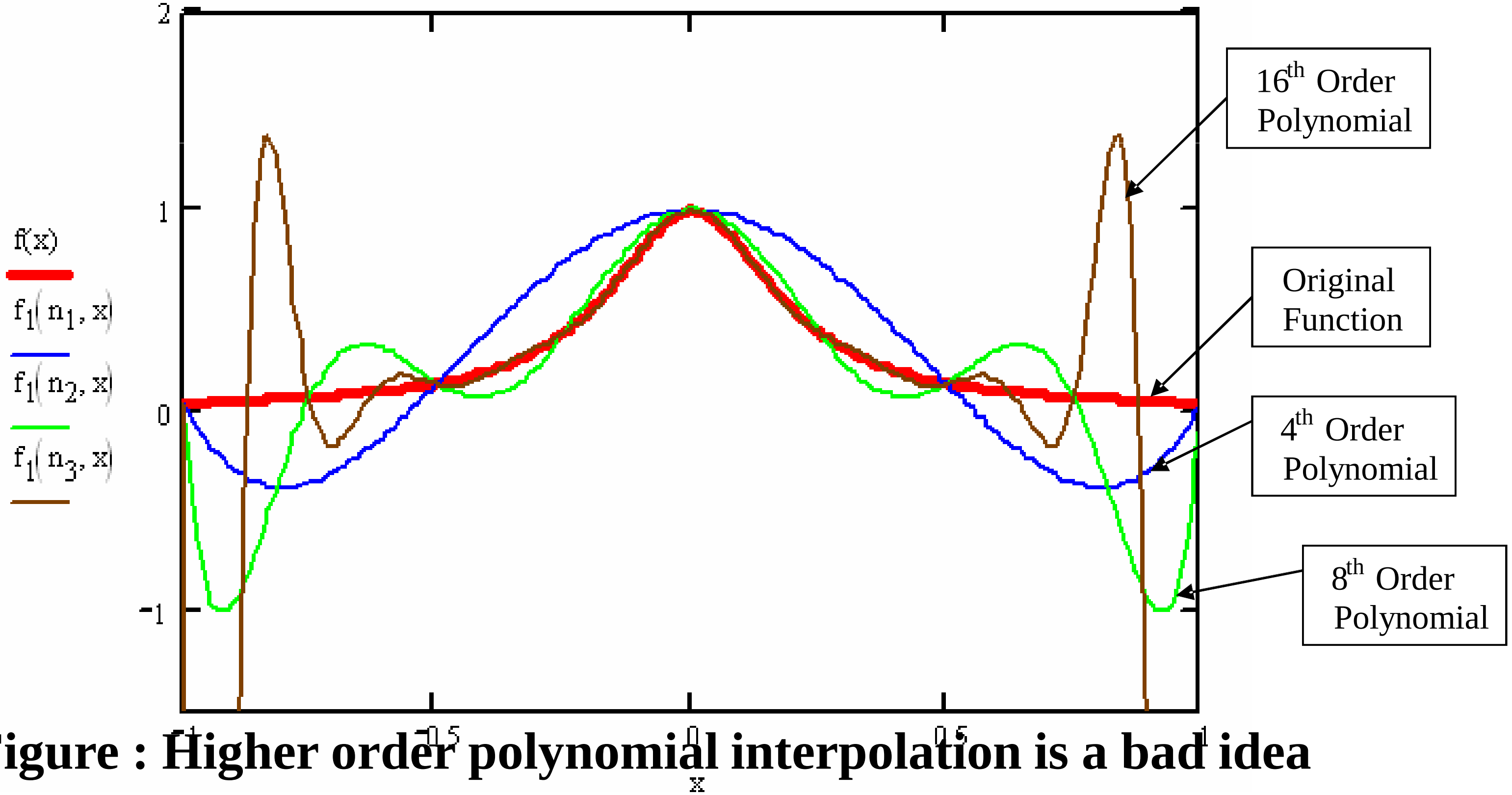
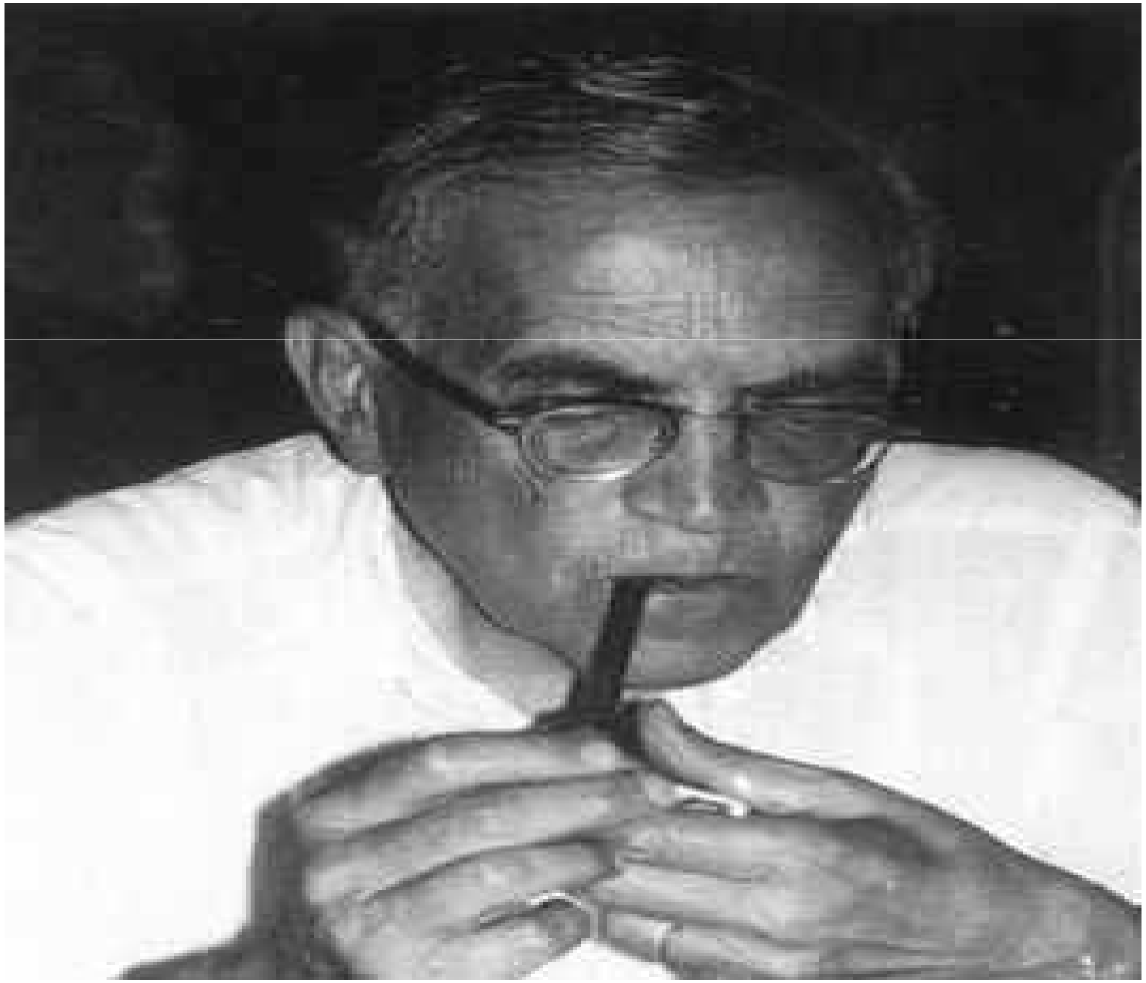


Figure : Higher order polynomial interpolation is a bad idea

Isaac Jacob Schoenberg

(April 21, 1903, Galați—February 21, 1990)



Spline Interpolation

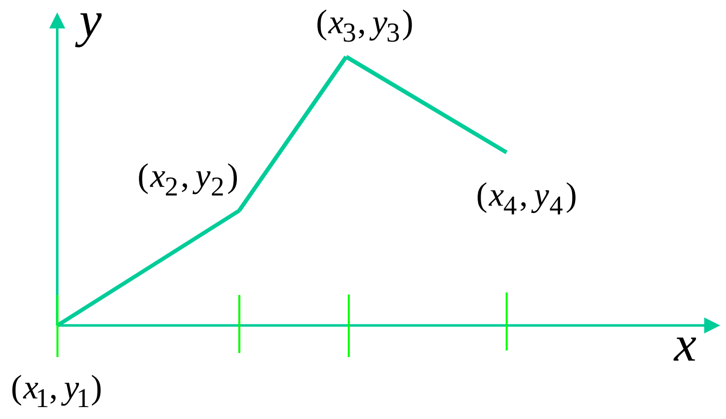
- Piecewise Linear Interpolation
  - Simplest form of piecewise polynomial interpolation

Set of data points : $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$ with $x_1 < x_2 < x_3 < x_4$

Define three subintervals, $I_1 = [x_1, x_2]$, $I_2 = [x_2, x_3]$, $I_3 = [x_3, x_4]$

  - Interpolate the data with piecewise linear function

$$P(x) = \begin{cases} \dfrac{x - x_2}{x_1 - x_2} y_1 + \dfrac{x - x_1}{x_2 - x_1} y_2, & x_1 \le x \le x_2 \\ \dfrac{x - x_3}{x_2 - x_3} y_2 + \dfrac{x - x_2}{x_3 - x_2} y_3, & x_2 \le x \le x_3 \\ \dfrac{x - x_4}{x_3 - x_4} y_3 + \dfrac{x - x_3}{x_4 - x_3} y_4, & x_3 \le x \le x_4 \end{cases}$$



Piecewise Linear Interpolation

- Example 8.14 Piecewise Linear Interpolation

Using, $x = [0 \quad 1 \quad 2 \quad 3]$, $y = [0 \quad 1 \quad 4 \quad 3]$

$$P(x) = \begin{cases} x, & 0 \le x \le 1 \\ 3x - 2, & 1 \le x \le 2 \\ -x + 6, & 2 \le x \le 3 \end{cases}$$

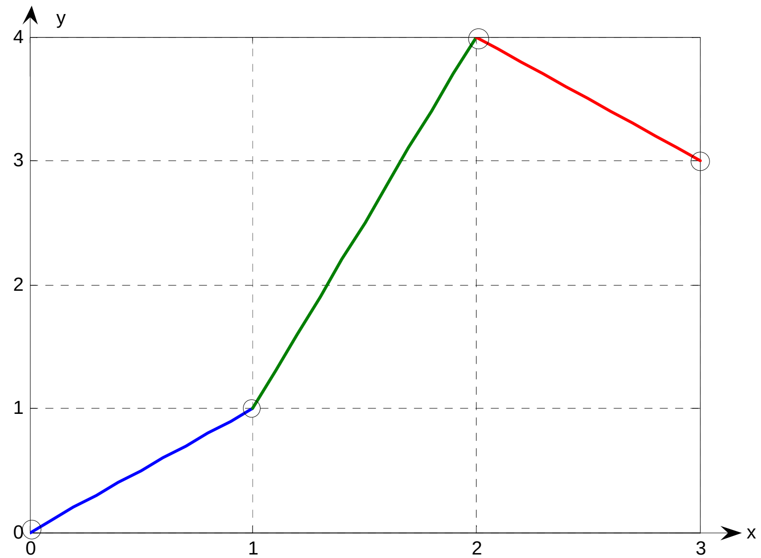


Figure 8.18 Piecewise linear interpolation

Piecewise Polynomial (Spline) Interpolation

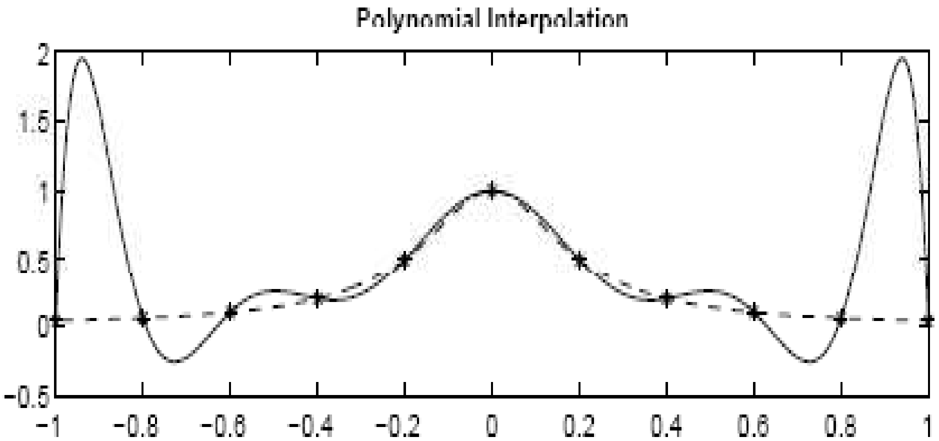
Let's return to interpolating a set of data points (x_0, y_0), (x_1, y_1), ..., (x_n, y_n).
Passing a single polynomial through many data points can sometimes lead to oscillations in the interpolant. Instead, we could use (linear) interpolation between successive points:

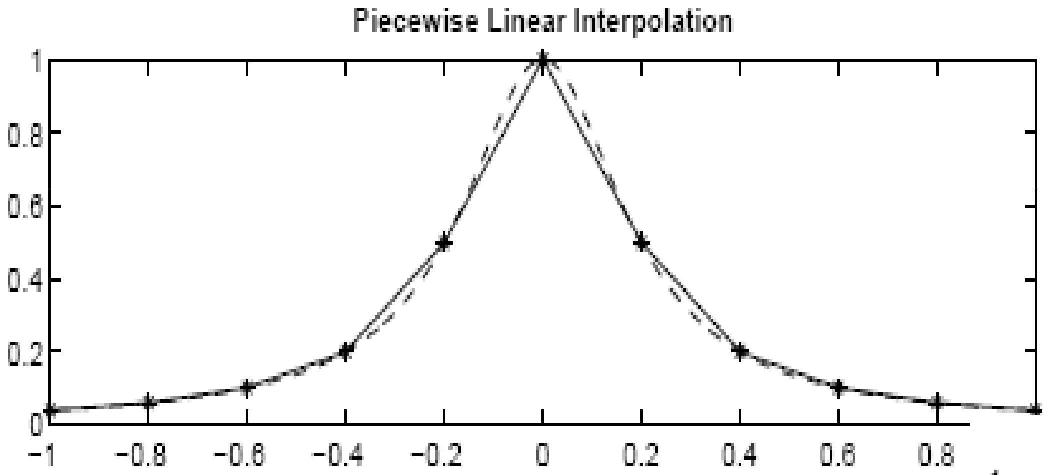
$$s_k(x) = y_k + d_k(x - x_k) \qquad \text{where} \qquad d_k = \frac{y_{k+1} - y_k}{x_{k+1} - x_k}$$

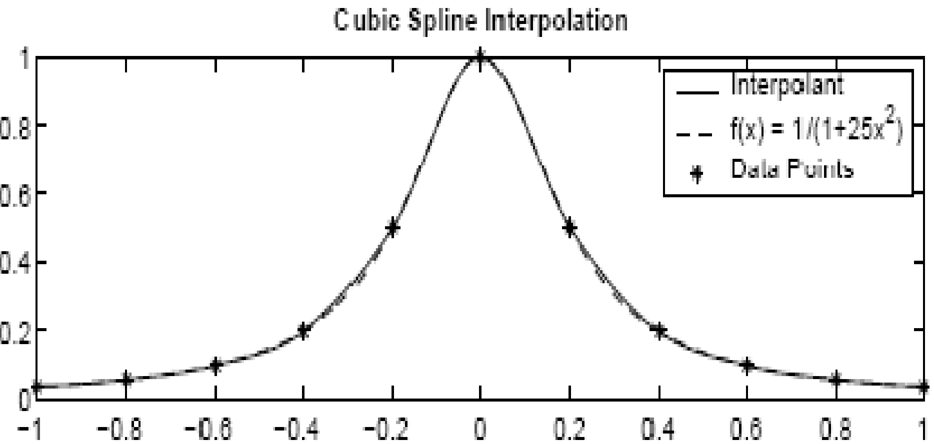
and in this way build a piecewise linear interpolant of the data points (x_k, y_k):

$$s(x) = \begin{cases} y_0 + d_0(x - x_0) & x \in [x_0, x_1] \\ y_1 + d_1(x - x_1) & x \in [x_1, x_2] \\ \vdots & \vdots \\ y_{n-1} + d_{n-1}(x - x_{n-1}) & x \in [x_{n-1}, x_n] \end{cases} \qquad \text{Linear Spline}$$

Note the derivatives are discontinuous at all the $x_k, k = 1, \ldots, n-1$.
We can do better...


Polynomial Interpolation
2
1.5
1
0.5
0
-0.5
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1


Piecewise Linear Interpolation
1
0.8
0.6
0.4
0.2
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8


Cubic Spline Interpolation
Interpolant
f(x) = 1/(1+25x^2)
Data Points
1
0.8
0.6
0.4
0.2
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

Piecewise Quadratic Interpolation

- "Knots"
  - Where the intervals meet to be the midpoints between the data points where the function values are given
- Processing
  - 4 data points $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$ with $x_1 < x_2 < x_3 < x_4$
  - Define node points

$$z_1 = x_1, \quad z_2 = (x_1 + x_2)/2, \quad z_3 = (x_2 + x_3)/2, \quad z_4 = (x_3 + x_4)/2, \quad z_5 = x_4$$

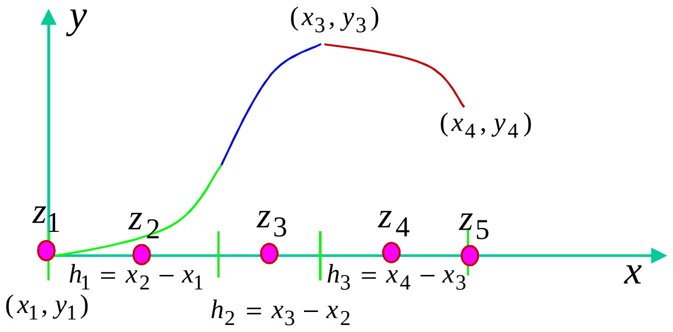
  - Spacing between consecutive data points

$$h_1 = x_2 - x_1, \quad h_2 = x_3 - x_2, \quad h_3 = x_4 - x_3$$

– Relationships:

$$z_2 - x_1 = h_1/2, \quad z_3 - x_2 = h_2/2, \quad z_4 - x_3 = h_3/2$$

$$z_2 - x_2 = -h_1/2, \quad z_3 - x_3 = -h_2/2, \quad z_4 - x_4 = -h_3/2$$



Piecewise Quadratic Interpolation – Example

- Example : 8: 15

Data points : (0,0), (1,1), (2,4), (3,3)

$$A = \begin{bmatrix} 1 & -1 & 0 & 0 & 2 & 0 \\ 0 & 1 & -1 & 0 & 2 & 2 \\ 0 & 0 & 1 & -1 & 0 & 2 \\ 1 & 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 1 & 0 & 1 & -1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

$$r = [4 \quad 12 \quad -4 \quad 0 \quad 0 \quad 0]'.$$

Solution : coefficients $a_1, a_2, a_3, a_4, b_2, b_3$:

Using Gaussian ellimination

$$x = [0.7429 \quad 1.7714 \quad -3.3714 \quad 2.4571 \quad 2.5143 \quad 0.9143]$$

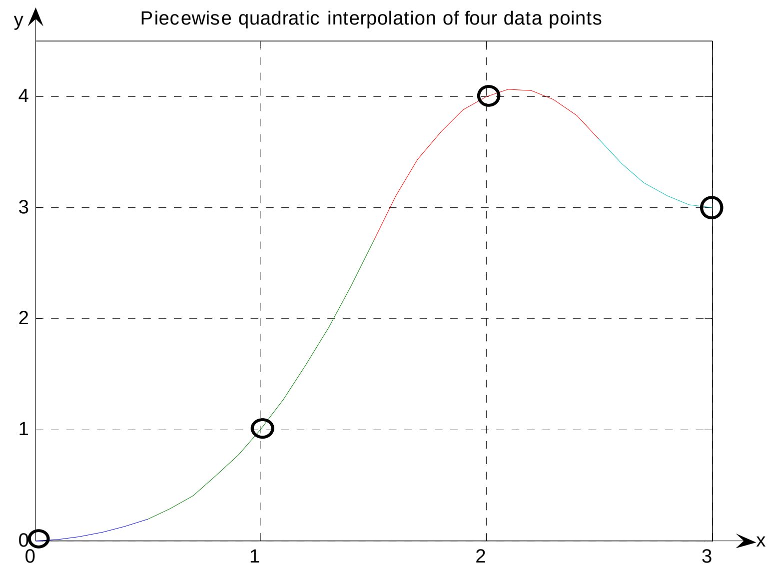
Piecewise interpolating polynomial

$P_1(x) = 0.7429(x-0)^2$ on $[0.0, 0.5]$

$P_2(x) = 1.7714(x-1)^2 + 2.5143(x-1) + 1$ on $[0.5, 1.5]$

$P_3(x) = -3.3714(x-2)^2 + 0.9143(x-2) + 4$ on $[1.5, 2.5]$

$P_4(x) = 2.4571(x-3)^2 + 3$ on $[2.5, 3.0]$



Spline Interpolation Definition

- Given n+1 distinct **knots** x_i such that:

$$x_0 < x_1 < \ldots < x_{n-1} < x_n,$$

with n+1 **knot values** y_i find a spline function

$$S(x) := \begin{cases} S_0(x) & x \in [x_0, x_1] \\ S_1(x) & x \in [x_1, x_2] \\ \vdots & \vdots \\ S_{n-1}(x) & x \in [x_{n-1}, x_n] \end{cases}$$

with each $S_i(x)$ a polynomial of degree at most n.

Cubic Splines

To build a better interpolant, we allow curvature between the points and require that the first and second derivatives are continuous throughout the interval $[x_0, x_n]$.
Assume that the interpolant linking the data points (x_k, y_k) and (x_{k+1}, y_{k+1}) is a cubic polynomial of the form:

$$s_k(x) = s_{k,0} + s_{k,1}(x - x_k) + s_{k,2}(x - x_k)^2 + s_{k,3}(x - x_k)^3 \qquad x \in [x_k, x_{k+1}]$$

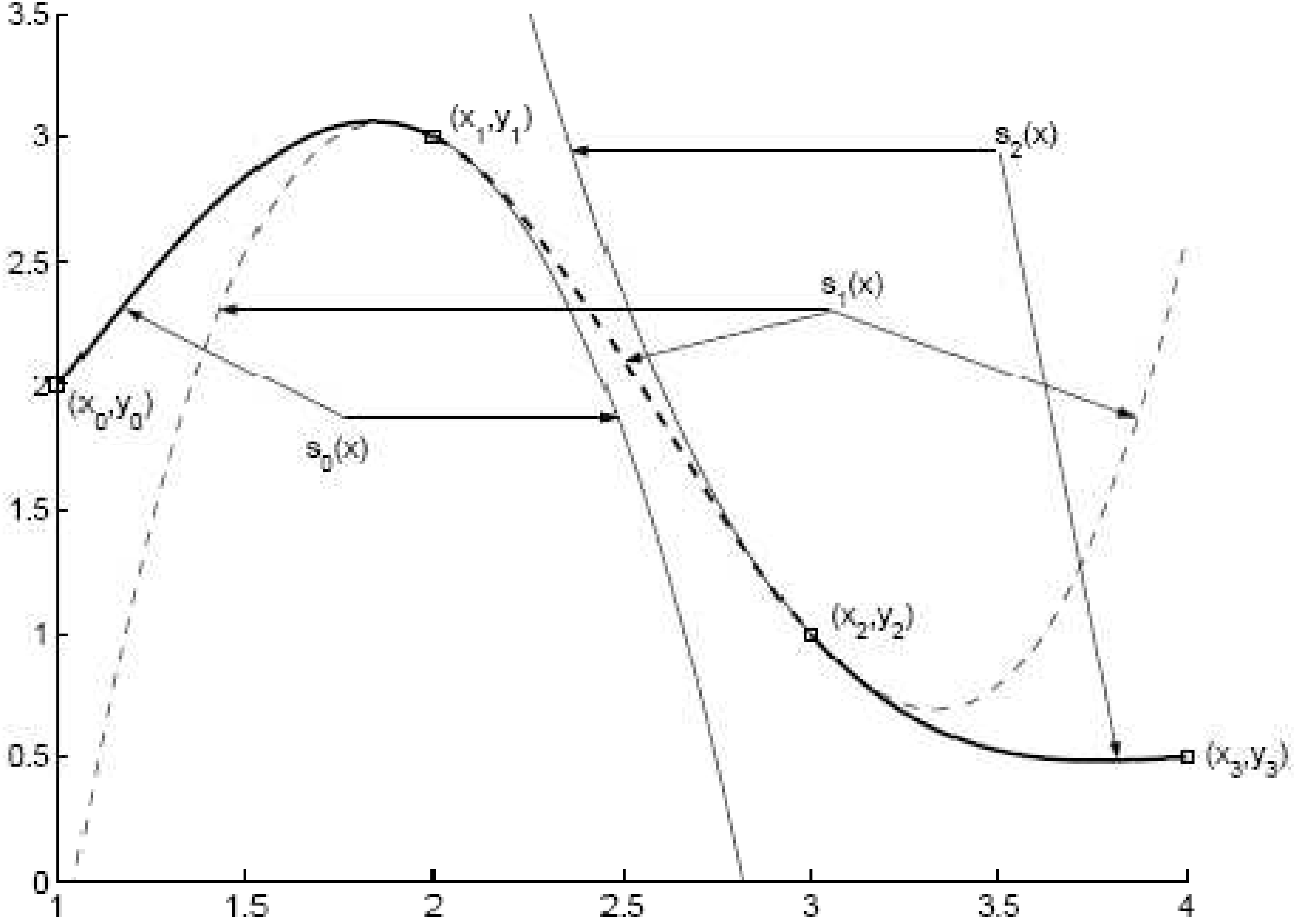
subject to the following constraints:

$$\begin{aligned} s_k(x_k) &= y_k && \text{Interpolation of } (x_k, y_k) \\ s_k(x_{k+1}) &= s_{k+1}(x_{k+1}) && \text{Continuity of interpolant} \\ s'_k(x_{k+1}) &= s'_{k+1}(x_{k+1}) && \text{Continuity of first derivatives} \\ s''_k(x_{k+1}) &= s''_{k+1}(x_{k+1}) && \text{Continuity of second derivatives} \end{aligned}$$

Note that for each spline $s_k(x)$ (except for the last one), there are four equations and four unknowns (the $s_{k,i}$'s). On the last spline, you don't have the last two conditions so that we end up with two more variables (the $s_{k,i}$) than equations/constraints.
To complete the system of equations, we will impose conditions at the end points.

There are two approaches for determining the end conditions for m_0 and m_n:

- Clamped spline: $s'(a) = f'(a)$ and $s'(b) = f'(b)$.
- Natural spline: $s''(a) = s''(b) = 0$.



3.5
3
2.5
2
1.5
1
0.5
0
1
1.5
2
2.5
3
3.5
4
(x1,y1)
s2(x)
s1(x)
(x0,y0)
s0(x)
(x2,y2)
(x3,y3)

Construction of Cubic Splines

Before we start, define three quantities:

$$\begin{aligned} h_k &= x_{k+1} - x_k && \text{spacing between } x_k \text{ and } x_{k+1} \\ d_k &= (y_{k+1} - y_k)/(x_{k+1} - x_k) && \text{slope between } (x_k, y_k) \text{ and } (x_{k+1}, y_{k+1}) \\ m_k &= s''(x_k) && \text{Second derivative of spline interpolant at } x_k \end{aligned}$$

We require that $s(x)$ be piecewise cubic.
This implies that $s'(x)$ is piecewise quadratic and $s''(x)$ is piecewise linear.
Let's start with the easiest one and perform piecewise linear (Lagrange) interpolation of the second derivative:

$$s_k''(x) = s''(x_k)\frac{x - x_{k+1}}{x_k - x_{k+1}} + s''(x_{k+1})\frac{x - x_k}{x_{k+1} - x_k}$$

Substituting in the h_k and the m_k:

$$s_k''(x) = \frac{m_k}{h_k}(x_{k+1} - x) + \frac{m_{k+1}}{h_k}(x - x_k) \qquad \begin{array}{l} x \in [x_k, x_{k+1}] \\ k = 0, 1, 2, \ldots, n-1 \end{array}$$

Integrating up to get an equation for the first derivative and the interpolant itself:

$$\begin{aligned} s_k'(x) &= -\frac{m_k}{2h_k}(x_{k+1} - x)^2 + \frac{m_{k+1}}{2h_k}(x - x_k)^2 - p_k + q_k \\ s_k(x) &= \frac{m_k}{6h_k}(x_{k+1} - x)^3 + \frac{m_{k+1}}{6h_k}(x - x_k)^3 + p_k(x_{k+1} - x) + q_k(x - x_k) \end{aligned}$$

Now, evaluate the interpolant at $x = x_k$ and $x = x_{k+1}$ and require that $s_k(x_k) = y_k$ and $s_k(x_{k+1}) = y_{k+1}$:

$$s_k(x_k) = y_k = \frac{m_k}{6h_k}(x_{k+1} - x_k)^3 + p_k(x_{k+1} - x_k) \quad \Rightarrow \quad p_k = \frac{y_k}{h_k} - \frac{m_k h_k}{6}$$

$$s_k(x_{k+1}) = y_{k+1} = \frac{m_{k+1}}{6h_k}(x_{k+1} - x_k)^3 + q_k(x_{k+1} - x_k) \quad \Rightarrow \quad q_k = \frac{y_{k+1}}{h_k} - \frac{m_{k+1} h_k}{6}$$

So then:

$$\begin{aligned} s_k(x) &= \frac{m_k}{6h_k}(x_{k+1} - x)^3 + \frac{m_{k+1}}{6h_k}(x - x_k)^3 \\ &\quad + \left(\frac{y_k}{h_k} - \frac{m_k h_k}{6}\right)(x_{k+1} - x) + \left(\frac{y_{k+1}}{h_k} - \frac{m_{k+1} h_k}{6}\right)(x - x_k) \end{aligned}$$

The only unknowns left in the equations for the splines are the m_k's. The only constraint that we haven't used is the continuity of the first derivatives $s'_{k-1}(x_k) = s'_k(x_k)$. We can determine these by matching the first derivatives:

$$s'_k(x) = -\frac{m_k}{2h_k}(x_{k+1} - x)^2 + \frac{m_{k+1}}{2h_k}(x - x_k)^2 - \left(\frac{y_k}{h_k} - \frac{m_k h_k}{6}\right) + \left(\frac{y_{k+1}}{h_k} - \frac{m_{k+1} h_k}{6}\right)$$

Evaluate $s'_k(x_k)$ and $s'_{k-1}(x_k)$:

$$\begin{aligned} s'_k(x_k) &= -\frac{m_k h_k}{2} - \left(\frac{y_k}{h_k} - \frac{m_k h_k}{6}\right) + \left(\frac{y_{k+1}}{h_k} - \frac{m_{k+1} h_k}{6}\right) \\ &= -\frac{1}{3} m_k h_k - \frac{1}{6} m_{k+1} h_k + d_k \quad \text{where} \quad d_k = \frac{y_{k+1} - y_k}{h_k} \\ s'_{k-1}(x_k) &= \frac{m_k h_{k-1}}{2} - \left(\frac{y_{k-1}}{h_{k-1}} - \frac{1}{6} m_{k-1} h_{k-1}\right) + \left(\frac{y_k}{h_{k-1}} - \frac{1}{6} m_k h_{k-1}\right) \\ &= \frac{1}{3} m_k h_{k-1} + \frac{1}{6} m_{k-1} h_{k-1} + d_{k-1} \end{aligned}$$

Then require continuity of the first derivatives, $s'_k(x_k) = s'_{k-1}(x_k)$:

$$-\frac{1}{3} m_k h_k - \frac{1}{6} m_{k+1} h_k + d_k = \frac{1}{3} m_k h_{k-1} + \frac{1}{6} m_{k-1} h_{k-1} + d_{k-1}$$

Simplifying yields (this is the important equation):

$$h_{k-1} m_{k-1} + 2(h_{k-1} + h_k) m_k + h_k m_{k+1} = 6(d_k - d_{k-1})$$

which yields a linear system (with $n-1$ equations) that can be solved for the m_k ($n+1$ unknowns) that (once end conditions are specified) uniquely determines the spline coefficients.

Divide through by $h_{k-1}+h_k$ and let $\lambda_k = h_k/(h_{k-1}+h_k)$ and $\rho_k = 1-\lambda_k$.
Note that $\rho_k = h_{k-1}/(h_{k-1}+h_k)$:

$$\rho_k m_{k-1} + 2m_k + \lambda_k m_{k+1} = 6\frac{d_k - d_{k-1}}{h_{k-1}+h_k}$$

Note that there are $n+1$ unknowns $(m_k, k = 0,1,\ldots,n)$ and only $n-1$ equations. We need to specify m_0 and m_n. This leads to the linear system $\mathbf{Mm} = \mathbf{b}$:

$$\begin{pmatrix} 2 & \lambda_1 & 0 & 0 & \ldots & 0 \\ \rho_2 & 2 & \lambda_2 & 0 & \ldots & 0 \\ 0 & \rho_3 & 2 & \lambda_3 & \ldots & 0 \\ \vdots & \vdots & & \ddots & \ddots & \vdots \\ 0 & \ldots & 0 & \rho_{n-2} & 2 & \lambda_{n-2} \\ 0 & \ldots & 0 & 0 & \rho_{n-1} & 2 \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \\ \vdots \\ m_{n-3} \\ m_{n-2} \\ m_{n-1} \end{pmatrix} = \begin{pmatrix} 6\frac{d_1-d_0}{h_0+h_1} - \rho_1 m_0 \\ 6\frac{d_2-d_1}{h_1+h_2} \\ \vdots \\ 6\frac{d_{n-2}-d_{n-3}}{h_{n-3}+h_{n-2}} \\ 6\frac{d_{n-1}-d_{n-2}}{h_{n-2}+h_{n-1}} - \lambda_{n-1} m_n \end{pmatrix}$$

The matrix $\mathbf{M}$ is called the mass matrix. Now, remembering that:

$$s_k(x) = s_{k,0} + s_{k,1}(x-x_k) + s_{k,2}(x-x_k)^2 + s_{k,3}(x-x_k)^3 \qquad x \in [x_k, x_{k+1}]$$

Once you have solved the system for the m_k's, you can construct the spline using the coefficients:

$$s_{k,0} = y_k \quad s_{k,1} = d_k - \frac{h_k}{6}(2m_k + m_{k+1}) \quad s_{k,2} = \frac{m_k}{2} \quad s_{k,3} = \frac{m_{k+1}-m_k}{6h_k}$$

Error Bound (Burden & Faires, p. 154)

Suppose f and its first four derivatives are continuous on $[a, b]$ and $\max_{a \leq x \leq b} |f^{(4)}(x)| = M$. If $S(x)$ is the unique clamped spline interpolant of $f(x)$ on the points $x_k, k = 0, 1, \ldots, n$ with $a = x_0 < x_1 < \ldots < x_{n-1} < x_n = b$, then:

$$\max_{a \leq x \leq b} |f(x) - S(x)| \leq \frac{5M}{384} \max_{0 \leq j \leq n-1} (x_{j+1} - x_j)^4$$

A similar bound holds for the natural spline and the not-a-knot spline.

B-splines, continued

To start recursion, define B-splines of degree 0 by

$$B_i^0(t) = \begin{cases} 1 & \text{if } t_i \le t < t_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

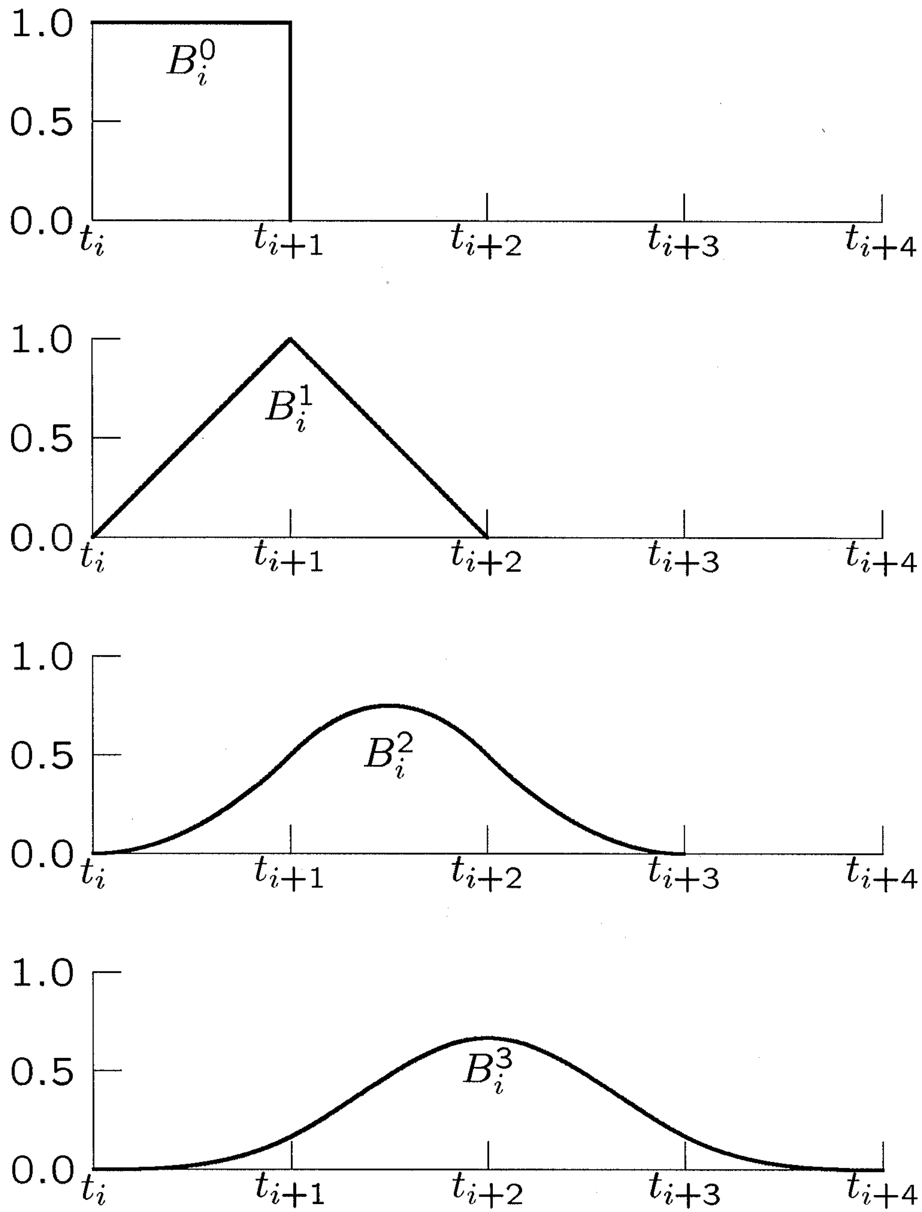
and then for $k > 0$ define B-splines of degree k by

$$B_i^k(t) = v_i^k(t) B_i^{k-1}(t) + (1 - v_{i+1}^k(t)) B_{i+1}^{k-1}(t).$$

Since B_i^0 is piecewise constant and v_i^k is linear, B_i^1 is piecewise linear.

Similarly, B_i^2 is in turn piecewise quadratic, and in general, B_i^k is piecewise polynomial of degree k.

B-splines



1.0
0.5
0.0
B_i^0
t_i
t_{i+1}
t_{i+2}
t_{i+3}
t_{i+4}
1.0
0.5
0.0
B_i^1
t_i
t_{i+1}
t_{i+2}
t_{i+3}
t_{i+4}
1.0
0.5
0.0
B_i^2
t_i
t_{i+1}
t_{i+2}
t_{i+3}
t_{i+4}
1.0
0.5
0.0
B_i^3
t_i
t_{i+1}
t_{i+2}
t_{i+3}
t_{i+4}

B-splines, continued

Important properties of B-spline functions B_i^k :

1. For $t < t_i$ or $t > t_{i+k+1}$, B_i^k (t) = 0.

2. For $t_i < t < t_{i+k+1}$, B_i^k (t) > 0.

3. For all t, $$\sum_{i=-\infty}^{\infty} B_i^k(t) = 1.$$

4. For $k \geq 1$, B_i^k has $k - 1$ continuous derivatives.

5. Set of functions $\{B_{i-1}^k, \ldots, B_{n-1}^k\}$ is linearly independent on interval $[t_1, t_n]$ and spans set of all splines of degree k having knots t_i.

B-splines, continued

Properties 1 and 2 together say that B-spline functions have local support.

Property 3 indicates how functions are normalized.

Property 4 says that they are indeed splines.

Property 5 says that for given k these functions form basis for set of all splines of degree k.

B-splines, continued

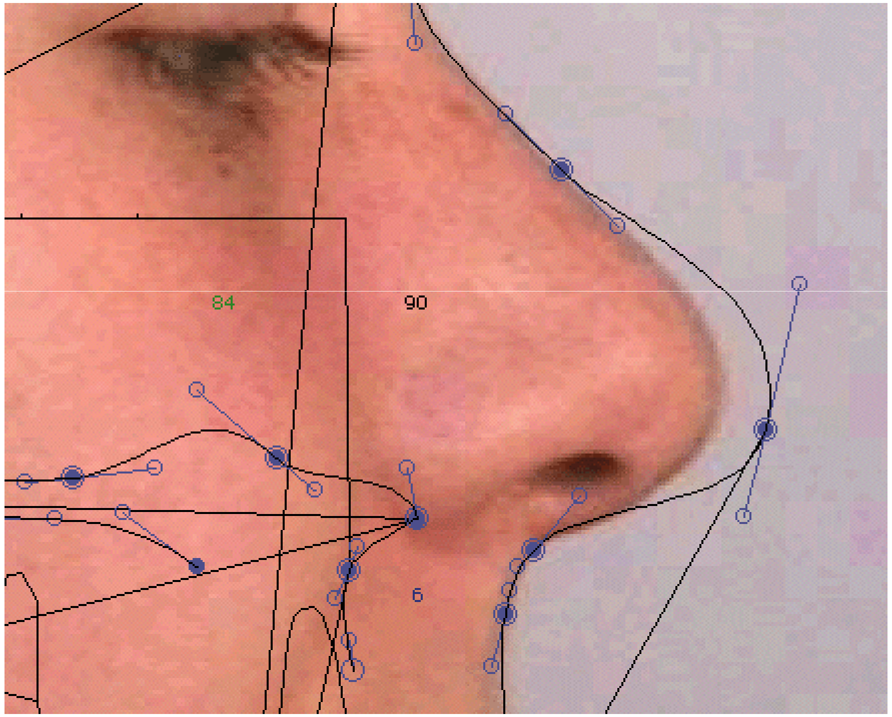
If we use B-spline basis, linear system to be solved for spline coefficients will be nonsingular and banded.

Use of B-spline basis yields efficient and stable methods for determining and evaluating spline interpolants, and many library routines for spline interpolation are based on this approach.

B-splines are also useful in many other contexts, such as numerical solution of differential equations, as we will see later.

Bezier Spline Interpolation

Practical Application



Problem 37. If $s(x) = 0$ for $x < 2$ and $s(x) = (x-2)^3$ for $x \geq 2$ is it true that s is a cubic spline? Justify your answer.

Problem 38. Let

$$s(x) = \begin{cases} 1 - x + ax^2 + x^3 & \text{if } 0 \leq x \leq 1 \\ 3 + bx + cx^2 - x^3 & \text{if } 1 < x \leq 2 \end{cases}$$

Determine a, b and c so that $s(x)$ is a *natural* cubic spline on the interval $[0, 2]$.

Problem 39. Let

$$s(x) = \begin{cases} x^3 + ax^2 & \text{if } 0 \leq x \leq 1 \\ x^2 - bx + 1 & \text{if } 1 < x \leq 2 \end{cases}$$

Determine a and b so that $s(x)$ is a cubic spline on the interval $[0, 2]$.

Problem 40. Let

$$s(x) = \begin{cases} 0 & \text{if } x \leq 2 \\ (x-2)^3 & \text{if } 2 < x \end{cases}$$

Is $s(x)$ a cubic spline? Justify your answer.
