CME 108/MATH 114
Introduction to Scientific Computing
Summer 2020

Spline Interpolation

We've approached the interpolation problem by choosing (high-degree) polynomials for our basis functions $\phi_i : f(x) = \sum_{j=0}^{n} c_j \phi_j(x)$. This approach can be efficient (recall the barycentric form of the Lagrange interpolant), but using high degree polynomials can lead to large errors due to erratic oscillations, especially near the interval endpoints.

To mediate this, we'll try a different approach. We'll break up the interval over which the data is defined into small pieces, and we'll use a low-degree polynomial interpolant over each piece!

Piecewise polynomial interpolation To begin, we'll consider the simplest case: piecewise linear interpolants (used by MATLAB when plotting).

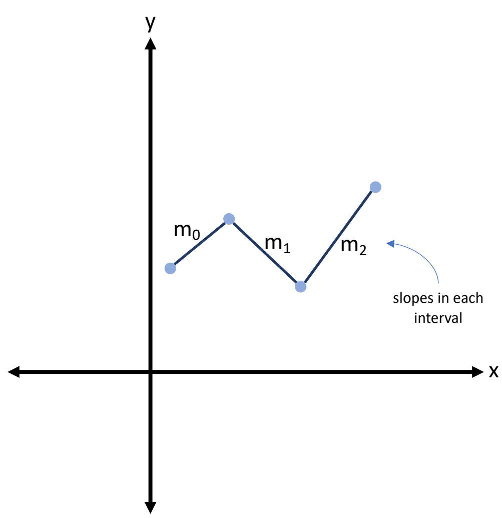

Figure 1: Piecewise linear interpolation

To find this interpolant we need only find the line between each pair of adjacent points on each interval,

$$
\begin{aligned}
s_0(x) &= f(x_0) + m_0(x - x_0), && x_0 \le x \le x_1 && (1)\\
s_1(x) &= f(x_1) + m_1(x - x_1), && x_1 \le x \le x_2 && (2)\\
&\vdots && && (3)\\
s_{n-1}(x) &= f(x_{n-1}) + m_{n-1}(x - x_{n-1}), && x_{n-1} \le x \le x_n && (4)
\end{aligned}
$$

and note that the slope in $[x_i, x_{i+1}]$ is given by $m_i = \frac{f(x_{i+1})-f(x_i)}{x_{i+1}-x_i}$

In this linear case the formula for each piece follows simply from the point-slope form of a line. However, we can gain insight from its structure.

Remark 0.0.1. On each subinterval $[x_i, x_{i+1}]$, for $i = 0, 1, \ldots, n-1$, the piecewise polynomial interpolant s coincides with the linear polynomial

$$s(x) = s_i(x) = a_i + b_i(x - x_i),$$

with

$$a_i = f(x_i), \qquad b_i = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}$$

The values of a_i follow from the interpolation requirement ($s(x_i) = s_i(x_i) = f(x_i)$), and the values of b_i follow from the (so far implicit) requirement that s be continuous, which we can express as $s_i(x_{i+1}) = s_{i+1}(x_{i+1}), i = 0, 1, \ldots, n-2$

Let's use this insight and consider the popular cubic case (quadratic case is developed in *HW5*).

Cubic Spline Mimicking the form of the piecewise linear interpolant, in this case we require that on each subinterval $[x_i, x_{i+1}]$ the piecewise interpolant s satisfies

$$s(x) = s_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3,$$

where a_i, b_i, c_i, and d_i are coefficients to be determined. Since for $n+1$ given points there are n pieces, there are $4n$ coefficients in total. Our best bet then is to impose $4n$ constraints and hope the resulting system is linear (in this case it's possible that the solution to the system is unique, if it exists).

Conditions

1. Interpolation: $s(x_i) = s_i(x_i) = f(x_i), i = 0, 1, \ldots, n-1$, AND $s_{n-1}(x_n) = f(x_n)$.

   ($n+1$ conditions here)

2. Continuity: $s_i(x_{i+1}) = s_{i+1}(x_{i+1}), i = 0, 1, \ldots, n-2$ (holds at interior points, gives $n-1$ conditions).

   These are the same as in the linear case. We need more conditions so we can ask for more! A drawback of piecewise linear interpolation is that it is not differentiable, so here we ask for smoothness:

3. Continuity of s' at interior points:

   $$s_i'(x_{i+1}) = s_{i+1}'(x_{i+1}), i = 0, 1, \ldots, n-2$$

   (n-1) conditions

4. Continuity of s'' at interior points:

   $$s_i''(x_{i+1}) = s_{i+1}''(x_{i+1}), i = 0, 1, \ldots, n-2$$

   (n-1) condition

This gives $3(n-1) + (n+1) = 4n-2$ condition in total, so we need two more. These are boundary conditions, and there are different kinds (see below).

Before we specify the additional conditions, let's write out our system and manipulate it to obtain formulas for a_i, b_i, c_i, d_i:

1. (Interpolation)

   $$\begin{aligned} a_i &= a_i + b_i(x_i - x_i) + c_i(x_i - x_i)^2 + d_i(x_i - x_i)^3 && (5)\\ &= s_i(x_i) && (6)\\ &= f(x_i), i = 0, 1, \ldots, n && (7) \end{aligned}$$

   Remark 0.0.2. Here we extend our previous notation and use $a_n = f(x_n)$, to simplify our derivation below

2. (Continuity)

   $$a_i + b_i h_i + c_i h_i^2 + d_i h_i^3 = s_i(x_{i+1}) = s_{i+1}(x_{i+1}) = a_{i+1}, \quad (8)$$

   for $i = 0, 1, \ldots, n-2$ with $h_i = x_{i+1} - x_i$.

3. (Continuity of s')

$$b_i h_i + 2c_i h_i + 3d_i h_i^2 = s_i'(x_{i+1}) = s_{i+1}'(x_{i+1}) = b_{i+1}, \tag{9}$$

for $i = 0, 1, \ldots, n-2$.

4. (Continuity of s'')

$$2c_i h_i + 3 \cdot 2d_i h_i = s_i''(x_{i+1}) = s_{i+1}''(x_{i+1}) = 2c_{i+1}, \tag{10}$$

for $i = 0, 1, \ldots, n-2$.

Now we solve some of these. The interpolation conditions imply that

$$a_i = f(x_i).$$

From the continuity of s'' it follows that

$$d_i = \frac{c_{i+1} - c_i}{3h_i}. \tag{11}$$

Substituting (11) into (9) gives

$$\begin{aligned} b_{i+1} &= b_i + 2c_i h_i + (c_{i+1} - c_i) h_i \\ &= b_i + (c_{i+1} + c_i) h_i. \end{aligned} \tag{12}$$

Similarly, substituting (11) into (8) gives

$$\begin{aligned} a_{i+1} &= a_i + b_i h_i + c_i h_i^2 + \frac{c_{i+1} - c_i}{3} h_i^2 \\ &= a_i + b_i h_i + \frac{c_{i+1} + 2c_i}{3} h_i^2. \end{aligned} \tag{13}$$

Now we solve (13) for b_i to obtain

$$b_i = \frac{a_{i+1} - a_i}{h_i} - \frac{2c_i + c_{i+1}}{3} h_i \tag{14}$$

At this point, a_i, b_i, d_i are either known or given in terms of c_i!

Now we substitute (14) into (12) and simplify to obtain

$$\begin{aligned} b_{i+1} &= \frac{a_{i+1} - a_i}{h_i} - \frac{2c_i + c_{i+1}}{3} h_i + (c_{i+1} + c_i) h_i \\ &= \frac{a_{i+1} - a_i}{h_i} + \frac{c_i + 2c_{i+1}}{3} h_i. \end{aligned} \tag{15}$$

To find a relation for the $c_1's$, we use (14) in place of the LHS:

$$\frac{a_{i+1}-a_i}{h_{i+1}} - \frac{2c_i + c_{i+2}}{3}h_{i+1} = b_{i+1} = \frac{a_{i+1}-a_i}{h_{i+1}} + \frac{c_i + 2c_{i+1}}{3}$$

Now we move the known values ($a_i's$) to one side:

$$(2c_{i+1} + c_{i+2})h_{i+1} + (c_i + 2c_{i+1})h_i = \frac{3(a_{i+2} - a_{i+1})}{h_{i+1}} - \frac{3(a_{i+1} - a_i)}{h_i} \tag{16}$$

or equivalently,

$$h_i c_i + 2(h_i + h_{i+1})c_{i+1}h_{i+1}c_{i+2} = \frac{3}{h_{i+1}}(a_{i+2} - a_{i+1}) - \frac{3}{h_{i+1}}(a_{i+1} - a_i) \tag{17}$$

For convenience, we replace i by $i-1$ (shifting every index down by 1) to obtain

$$h_{i-1}c_{i-1} + 2(h_{i-1} + h_i)c_i + h_i c_{i-+1} = \frac{3}{h_i}(a_{i+2} - a_i) - \frac{3}{h_{i-1}}(a_i - a_{i-1}). \tag{18}$$

This equation holds for $i = 1, 2, \ldots, n-1$, and in this form it is clear that the resulting system for the c_i's is tridiagonal (hence it can be solved efficiently in $O(n)$). The equations for $i = 0$ and $i = n$ depend on the type of boundary conditions used.

We remark that these formulae simplify when evenly spaced nodes, knots, or abscissas are given (so $h_0 = h_1 = \cdots = h_{n-1} = h$):

FIRST:

$$c_{i-1} + 4c_i + c_{i+1} = \frac{3}{h^2}(a_{i+1} - 2a_i + a_{i-1}) \tag{19}$$

(along with boundary conditions)

THEN

$$b_i = \frac{a_{i+1} - a_i}{h} - \frac{2c_i + c_{i+1}}{3}h \tag{20}$$

$$d_i = \frac{c_{i+1} - c_i}{3h} \tag{21}$$

Solving for coefficients is a two-step process! We first solve a linear system for the c_i's, then evaluate formulae for b_i, d_i. Likewise, evaluating the resulting spline is a two-step process: given x, we first determine the index i for which $x \in [x_i, x_{i+1}]$(to what piece does x belong?), then we evaluate $s(x) = s_i(x)$ using the calculated coefficients.

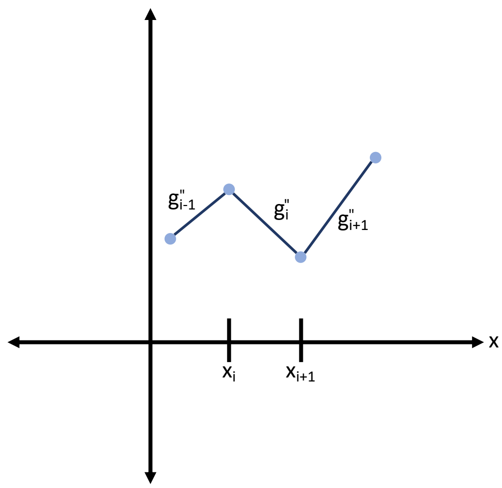

Figure 2: Piecewise linear interpolation

Before we introduce the different kinds of Boundary Conditions, we remark there is another approach for obtaining the coefficients, based on Lagrange interpolation!

Let g_i denote the interpolating cubic on $[x_i, x_{i+1}]$ and note g_i'' is linear.

Using Lagrange interpolation, continuity of the second derivative implies.

$$g_i''(x) = g''(x_i)\frac{x - x_{i+1}}{x_i - x_{x+1}} + g''(x_{i+1})\frac{x - x_i}{x_{i+1} - x_i} \tag{22}$$

This is Lagrange form of degree-1 interpolant.

Integrate the above expression twice to obtain

$$g_i(x) = g''(x_i)\frac{1}{6}\frac{(x - x_{i+1})^3}{x_i - x_{x+1}} + g''(x_{i+1}) \cdot \frac{1}{6} \cdot \frac{(x - x_i)^3}{x_{i+1} - x_i} + k_1 x + k_2, \tag{23}$$

for some constants k_1, k_2. We find k_1, k_2 by enforcing interpolation:(evaluate the above at $x = x_i, x - x_{i+1}$)

$$f(x_i) = g_i(x_i) = g''(x_i) \cdot \frac{1}{6} \cdot (x_i - x_{i+1})^2 + k_i x_i + k_2, \tag{24}$$

$$f(x_{i+1}) = g_i(x_i) = g''(x_{i+1}) \cdot \frac{1}{6} \cdot (x_{i+1} - x_i)^2 + k_i x_{i+1} + k_2 \tag{25}$$

Here we are using g to denote the full piecewise interpolant.

Subtracting the first equation from the second implies

$$f(x_{i+1}) - f(x_i) = \frac{g''(x_{i+1} - g''(x_i)}{6} h_i^2 + k_1 h_i, \tag{26}$$

so

$$K_1 = \frac{f(x_{i+1}) - f(x_i)}{h_i} - \frac{g''(x_{i+1}) - g''(x_i)}{6} h_i \tag{27}$$

Substitution then gives

$$K_2 = f(x_i) - \frac{g''(x_i)}{6} h_i^2 - k_1 x_i^2 \tag{28}$$

so that

$$g_i(x) = \frac{g''(x_i)}{6}\left(\frac{(x_{i+1} - x)^3}{h_i} - h_i(x_{i+1} - x)\right) + \frac{g''(x_{i+1})}{6}\left(\frac{x - x_i)^3}{h_i} - hi(x - x_i)\right) \tag{29}$$

$$+ f(x_i)\frac{x_{i+1} - x}{h_i} + f(x - x_i)\frac{x - x_i}{h_i} \tag{30}$$

Here g'' is still unknown, but we can use the continuity of g':

$$g_i'(x_{i+1}) = g_{i+1}'(x_i) \tag{31}$$

Using

$$g_i'(x) = \frac{(g''(x_i)}{6}\frac{(x_{i+1} - x)^3}{h_i} + \frac{g''(x_{i+1})}{6}\frac{(x - x_i)^3}{h_i} + \frac{f(x_{i+1}) - f(x_i)}{h_i} \tag{32}$$

$$- \frac{g'' x_{i+1} - g''(x_i)}{6} h_i \tag{33}$$

we obtain, after some algebraic manipulation, the same system as before!

$$\frac{h_{i-1}}{6} g''(x_{i-1}) + \frac{h_{i-1} + h_i}{3} g''(x_i) + \frac{h_i}{6} g''(x_{i+1}) \tag{34}$$

$$= \frac{f(x_{i+1}) - f(x_i)}{h_i} - \frac{f(x_1) - f(_{x_i - 1})}{h_{i-1}} \tag{35}$$

(the unknowns here are the (constants) $g''(x_i)$).
Now on to the Boundary Conditions.

Not-a-knot When no additional information is known about function f, this choice is recommended. We require continuity of s''' at $x = x_1, x = x_{n-1}$, which means $d_0 = d = 1, d_{n-2} = d_{n-1}$. These give conditions for c_j using $d_i = \frac{c_{i+1}-c_i}{3h_i}$:

$$h_1 c_0 - (h_0 + h_1)c_1 + h_0 c_2 = 0, \tag{36}$$

$$h_{n-1}c_{n-2} - (h_{n-2} + h_{n-1})c_{n-1} + h_{n-2}c_n = 0 \tag{37}$$

- Bad news: these ruin the tridiagonal structure of our system (why?)
- Good news: we can solve for c_0 and c_n and recover the tridiagonal structure, by substituting into the equations for c_j when $j = 1$ and $j = n - 1$:
  - $j = 1$:

$$\left(3h_0 + 2h_1 + \frac{h_0^2}{h_1}\right)c_1 + \left(h_1 - \frac{h_0^2}{h_1}\right)c_2 = \frac{3}{h_i}(a_2 - a_1) - \frac{3}{h_0}(a_1 - a_0) \tag{38}$$

  - $j = 2, \ldots, n-2$:

$$h_{i-1}c_{i-1} + 2(h_{i-1} + h_i)c_i + h_i c_{i+1} = \frac{3}{h_i}(a_{i+1} - a_i) - \frac{3}{h_{i-1}}(a_i - a_{i-1}) \tag{39}$$

  - $j = n - 1$:

$$\left(h_{n-2} - \frac{h_{n-1}^2}{h_{n-2}}\right)c_{n-2} + (3h_{n-1} + 2h_{n-2} + \frac{h_{n-1}^2}{h_{n-2}}\Big)c_{n-1} \tag{40}$$

$$= \frac{3}{h_{n-1}}(a_n - a_{n-1}) - \frac{3}{h_{n-2}}(a_{n-1} - a_{n-2}) \tag{41}$$

When the nodes are equidistant, the boundary conditions are

$$6c_1 = \frac{3}{h^2}(a_2 - 2a_1 + a_0) \tag{42}$$

$$6c_{n-1} = \frac{3}{h^2}(a_n - 2a_{n-1} + a_{n-2}) \tag{43}$$

Clamped or Complete Spline If $f'(a)$ and $f'(b)$ are known, it is better to apply the *clamped* boundary conditions

$$s'(a) = f'(a), \quad s'(b) = f'(b).$$

We use the relation

$$b_j = \frac{a_{j+1} - a_j}{h_j} - \frac{2c_j + c_{j+1}}{3} h_j \tag{44}$$

with $j = 0$ and $j = n-1$ to re-write the boundary conditions in terms of the c_j.

Note that $f'(a) = s'(a) = s_0'(a) = b_0$ and $f'(b) = s'(b) = s_{n-1}'(b) = b_n$, so

$$f'(a) = b_0 = \frac{a_1 - a_0}{h_0} - \frac{2c_0 + c_1}{3} h_0 \tag{45}$$

and

$$2h_0c_0 + h_0c_1 = \frac{3}{h_0}(a_1 - a_0) - 3f'(a). \tag{46}$$

For the right boundary, we first use $b_n = b_{n-1} + (c_n + c_{n-1})h_{n-1}$ and then (44) at $j = n-1$ to obtain

$$f'(b) = b_n = \frac{a_n - a_{n-1}}{h_{n-1}} - \frac{2c_{n-1} + c_n}{3} h_{n-1} + (c_n + c_{n-1})h_{n-1} \tag{47}$$

or equivalently

$$h_{n-1}c_{n-1} + 2h_{n-1}c_n = \frac{-3}{h_{n-1}}(a_n - a_{n-1}) + 3f'(b). \tag{48}$$