Cubic spline interpolation

Interpolating linear splines are easy to construct, but they are not smooth enough–they do not have cts derivatives. To get a smoother interpolant try higher degree splines.

Let $S(x)$ be a cubic spline over the set of nodes $\{x_0, \ldots, x_n\}$

$$S(x) = \begin{cases} S_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i \\ \text{for } x_i \le x \le x_{i+1}, \forall i = 0, \ldots, n-1 \end{cases}$$

And $\forall i = 1, \ldots, n-1$

$$\begin{aligned} S_i(x_{i+1}) &= S_{i+1}(x_{i+1}), \\ S_i'(x_{i+1}) &= S_{i+1}'(x_{i+1}), \\ S_i''(x_{i+1}) &= S_{i+1}''(x_{i+1}) \end{aligned}$$

$S(x)$ has $4n$ coefs but we have $3(n-1)$ conditions from continuity.

A cubic spline $S(x)$ on $\{x_0, x_1, \ldots, x_n\}$ has $n+3$ free parameters. Therefore we need $n+3$ linearly independent equations to determine the free parameters. Given data $\{(x_i, y_i), i = 0, \ldots, n\}$ we can fill $n+1$ conditions

$$S(x_i) = y_i, i = 0, \ldots, n.$$

But we still need 2 extra conditions. Some commonly used conditions are

• **Clamped** If y_0', y_n' are given, require $S'(x_0) = y_0', S'(x_n) = y_n'$.

• **Natural** Set $S''(x_0) = 0, S''(x_n) = 0$.

• **Not-a-knot** Force $S^{(3)}$ to be cts at x_1, x_{n-1}. This results in $S_0(x) = S_1(x)$ and $S_{n-1}(x) = S_{n-2}(x)$.

Construction of the cubic spline

Computing the coefs of a cubic spline reduces to solving a tri-diagonal system of $n+1$ equations, by which the $n+1$ coefs are computed and then the rest of the coefs are computed by applying some simple relations.

A convenient representation of $S(x)$ is the following:

$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$

$\Rightarrow S_i'(x) = b_i + 2c_i(x - x_i) + 3d_i(x - x_i)^2$

$\Rightarrow S_i''(x) = 2c_i + 6d_i(x - x_i)$

Interpolating conditions:

$$S_i(x_i) = y_i \Rightarrow a_i = y_i \tag{1}$$

Cont:

$$\begin{aligned} &S_{i+1}(x_{i+1}) = S_i(x_{i+1}) \\ &\Rightarrow a_{i+1} = a_i + b_i h_i + c_i h_i^2 + d_i h_i^3 \end{aligned} \tag{2}$$

1st derivative continuity:

$$\begin{aligned} &S_{i+1}'(x_{i+1}) = S_i'(x_{i+1}) \\ &\Rightarrow b_{i+1} = b_i + 2c_i h_i + 3d_i h_i^2 \end{aligned} \tag{3}$$

2nd derivative continuity:

$$\begin{aligned} &S_{i+1}''(x_{i+1}) = S_i''(x_{i+1}) \\ &\Rightarrow 2c_{i+1} = 2c_i + 6d_i h_i \end{aligned} \tag{4}$$

Here, $h_i = x_{i+1} - x_i$. With some manipulation

$$\begin{aligned} &h_i c_i + 2(h_i + h_{i+1})c_{i+1} + h_{i+1}c_{i+2} \\ &= \tfrac{3}{h_{i+1}}(y_{i+2} - y_{i+1}) - \tfrac{3}{h_i}(y_{i+1} - y_i) \\ &\text{for } i = 0, \ldots, n-2 \end{aligned}$$

This results in a linear system with unknowns c_i's. Once the c_i's are computed the d_i's are recovered from (4) and the b_i's from (2).

The above procedure requires $O(n)$ flops. To evaluate S at an arbitrary point x, we choose the branch where x belongs to and do a constant amount work.

For the interpolant cubic spline S of $g \in \mathcal{C}^4[a,b]$ satisfying the clamped or the not-a-knot conditions it can be shown that

$$\begin{aligned} \max_{a \le x \le b} |g(x) - S(x)| &\le K \max_{a \le x \le b} g^{(4)}(x) h^4 \\ &= O(h^4), \end{aligned}$$

where K is a constant and $h = \max h_i$.

B-splines

Just as a polynomial of degree n can be written as a linear combination of $n+1$ basis functions with $n+1$ parameters that determine the poly, so can a spline of degree k defined wrt $P = \{x_0, \ldots, x_n\}$ be written as a linear combination of $n+k$ basis functions with $n+k$ parameters that uniquely determine the spline. A very common set of basis functions for the splines are **B-splines**.

A B-spline of degree k defined on some partition P is a
- pp of degree k defined on P
- cts and with cts derivatives up to order $k-1$ on all nodes
- positive on $k+1$ consecutive subintervals and zero elsewhere
- the sum of its values on all nodes is 1.

Linear spline interpolation using B-splines

Model linear B-spline

$$B(x) = \begin{cases} \alpha_1 x + \beta_1 & 0 \le x \le 1 \\ \alpha_2 x + \beta_2 & 1 \le x \le 2 \\ 0 & \text{else} \end{cases}$$

Requiring continuity on the nodes and sum of its values on all nodes to be 1, we have

$$B(x) = \begin{cases} x & 0 \le x \le 1 \\ 2 - x & 1 \le x \le 2 \\ 0 & \text{else} \end{cases}$$

Map $B(x)$ from (0,2) to (x_{i-1}, x_{i+1}). Define

$$B_i(x) = \begin{cases} B(\frac{x - x_{i-1}}{x_i - x_{i-1}}) & x_{i-1} \le x \le x_i \\ B(2 - \frac{x_{i+1} - x}{x_{i+1} - x_i}) & x_i \le x \le x_{i+1} \\ 0 & \text{else} \end{cases}$$

That is

$$B_i(x) = \begin{cases} \frac{x - x_{i-1}}{x_i - x_{i-1}} & x_{i-1} \le x \le x_i \\ \frac{x_{i+1} - x}{x_{i+1} - x_i} & x_i \le x \le x_{i+1} \\ 0 & \text{else} \end{cases}$$

$\{B_i(x), i = 0, \ldots, n\}$ is the set of linear B-spline basis functions in $P = \{x_0, \ldots, x_n\}$.

Then $L(x) = \sum_{i=0}^{n} y_i B_i(x)$ is another form of writing the unique linear spline defined on $\{x_0, \ldots, x_n\}$ and interpolating $\{(x_i, y_i), i = 0, \ldots, n\}$.

The definition of $L(x)$ is the same as the earlier definition of the linear spline. To see this consider any x and wolog assume it belongs to $[x_{i-1}, x_i]$. Only $B_{i-1}(x)$ and $B_i(x)$ are non-zero. Therefore

$$\begin{aligned} L(x) &= y_{i-1}B_{i-1}(x) + y_i B_i(x) \\ &= y_{i-1}\frac{x_i - x}{x_i - x_{i-1}} + y_i \frac{x - x_{i-1}}{x_i - x_{i-1}} \end{aligned}$$

since

$$B_{i-1}(x) = \begin{cases} \frac{x-x_{i-2}}{x_{i-1}-x_{i-2}} & x_{i-2} \le x \le x_{i-1} \\ \frac{x_i - x}{x_i - x_{i-1}} & x_{i-1} \le x \le x_i \\ 0 & \text{else} \end{cases}$$

Notes:

- $B_i(x_j) = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$
- $B_i(x)$ is nonzero in 2 subintervals, (x_{i-1}, x_i) and (x_i, x_{i+1})
- In each subinterval (x_{i-1}, x_i) there are 2 nonzero B_j's, $B_{i-1}(x)$ and $B_i(x)$