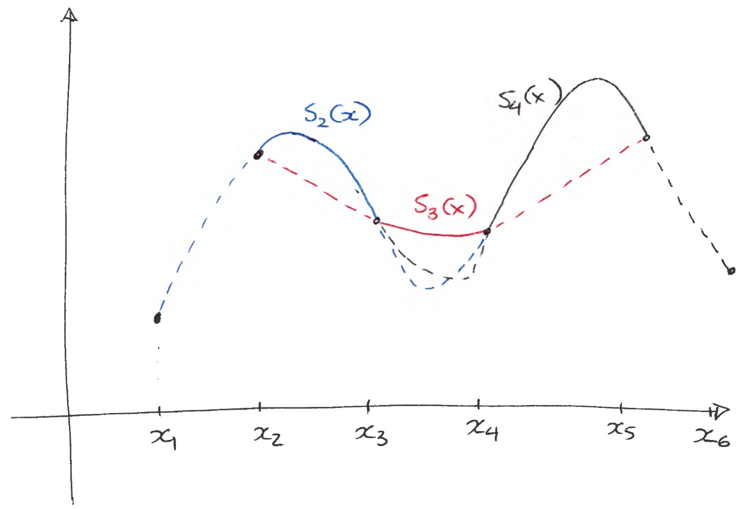


In this approach, each $S_k(x)$ is a *cubic* polynomial designed such that it interpolates the 4 data points:

$$(x_{k-1}, y_{k-1}), (x_k, y_k), (x_{k+1}, y_{k+1}), (x_{k+2}, y_{k+2})$$

As we will see, the benefit is that the error can be made even smaller than with piecewise linear curves; the drawback is that (as seen in the last figure) $S(x)$ can develop kinks (or corners) where 2 pieces $S_k(x)$ and $S_{k+1}(x)$ are joined.

The error of piecewise cubic interpolation is:

$$f(x) - S_k(x) = \frac{f''''(\theta_k)}{4!}(x - x_{k-1})(x - x_k)(x - x_{k+1})(x - x_{k+2})$$

An analysis similar to the linear case can show that $|q(x)| \leq \frac{9}{16} \max^4\{h_{k-1}, h_k, h_{k+1}\}$. If we again assume $h_1 = h_2 = \cdots = h_k = h$, the error bound becomes:

$$|f(x) - S(x)| \leq \frac{1}{24}\|f''''\|_\infty \cdot \frac{9}{16}h^4, \quad \text{or:}$$

$$f(x) - S(x) \leq \frac{9}{384}\|f''''\|_\infty \cdot h^4$$

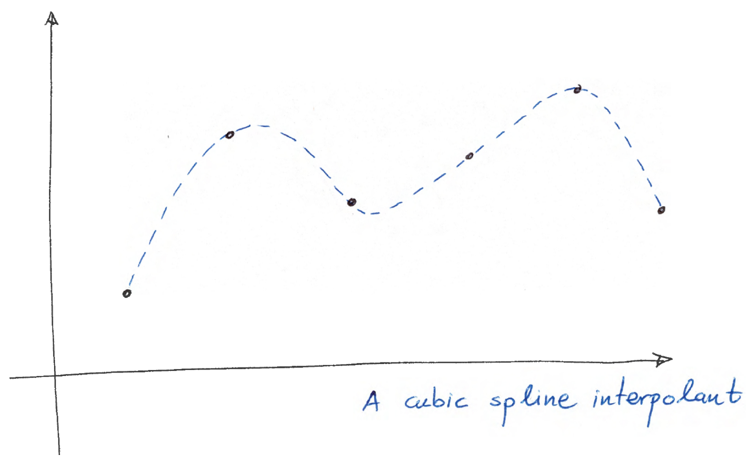
The next possibility we shall consider is a piecewise cubic curve:

$$S(x) = \begin{cases} S_1(x), & x \in I_1 \\ \vdots & \\ S_{n-1}(x), & x \in I_{n-1} \end{cases}$$

where each $S_k(x) = a_3^{(k)} x^3 + a_2^{(k)} x^2 + a_1^{(k)} x + a_0^{(k)}$, and the coefficients $a_i^{(j)}$ are chosen as to *force* that the curve has continuous values, first and second derivatives:

$$\begin{aligned} S_k(x_{k+1}) &= S_{k+1}(x_{k+1}) \\ S'_k(x_{k+1}) &= S'_{k+1}(x_{k+1}) \\ S''_k(x_{k+1}) &= S''_{k+1}(x_{k+1}) \end{aligned}$$

The curve constructed this way is called a *cubic spline* interpolant.



Note the increased smoothness (continuity of values and derivatives) at the endpoints of each interval I_k.

Cubic spline interpolants

As always our goal in this interpolation task is to define a curve $S(x)$ which interpolates the n data points:

$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \text{ (where } x_1 < x_2 < \cdots < x_n)$$

In the fashion of piecewise polynomials, we will define $S(x)$ as a *different* cubic polynomial $S_k(x)$ at each sub-interval $I_k = [x_k, x_{k+1}]$, i.e.:

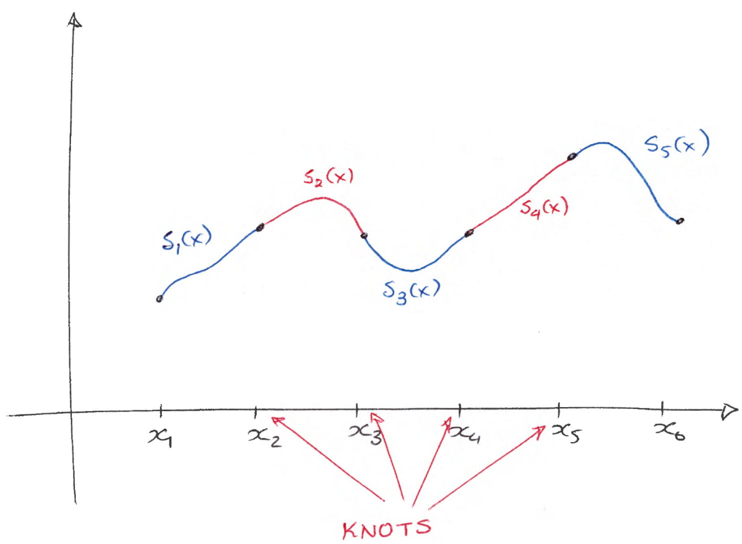
$$S(x) = \begin{cases} S_1(x), & x \in I_1 \\ S_2(x), & x \in I_2 \\ \quad\vdots & \\ S_k(x), & x \in I_k \\ \quad\vdots & \\ S_{n-1}(x), & x \in I_{n-1} \end{cases}$$

Each of the S_k's is a cubic polynomial

$$S_k(x) = a_0^{(k)} + a_1^{(k)}x + a_2^{(k)}x^2 + a_3^{(k)}x^3$$

This polynomial is defined by the four undetermined coefficients $a_0^{(k)}, \ldots, a_3^{(k)}$. Since we have $n-1$ piecewise polynomials, in total we shall have to determine $4(n-1) = 4n-4$ unknown coefficients.

The points $(x_2, x_3, \ldots, x_{n-1})$ where the formula for $S(x)$ changes from one cubic polynomial (S_k) to another (S_{k+1}) are called *knots*.



Note: In some textbooks, the extreme points x_1 and x_n are also included in the definition of what a knot is. We however retain the definition we stated previously.

The piecewise polynomial interpolation method described as *cubic spline* also requires the neighboring polynomials S_k and S_{k+1} to be joined at x_{k+1} with a certain degree of smoothness. In detail:

The curve should be continuous: $S_k(x_{k+1}) = S_{k+1}(x_{k+1})$

The derivative (slope) should be continuous: $S'_k(x_{k+1}) = S'_{k+1}(x_{k+1})$

The 2nd derivatives, as well: $S''_k(x_{k+1}) = S''_{k+1}(x_{k+1})$

(*Note:* If we force the next (3rd) derivative to match, this will force S_k and S_{k+1} to be exactly identical.)

When determining the unknown coefficients $a_i^{(j)}$, each of these 3 smoothness constraints (for knots $k = 2, 3, \ldots, n-1$) needs to be satisfied for a total of $3(n-2) =$