Interpolation with splines

↳ Alternative approach for fitting

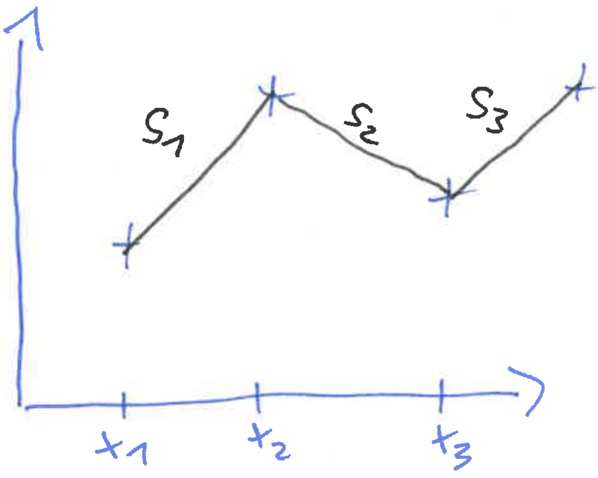
<u>Def</u>: A <u>spline</u> is a piecewise defined function which fulfills certain constraints w.r.t. the data points and the connection of its pieces

1 Linear Splines

$$\underbrace{S(t)}_{\text{spline}} = \sum_{i=1}^{N-1} s_i(t)$$

$s_i(t)$: pieces, $s_i(t) = 0$ if $t \notin [t_i, t_{i+1}]$

Linear spline: $s_i(t) = a_1^{(i)} t + a_0^{(i)}$



N data points

N-1 sections between data points

Constraints:

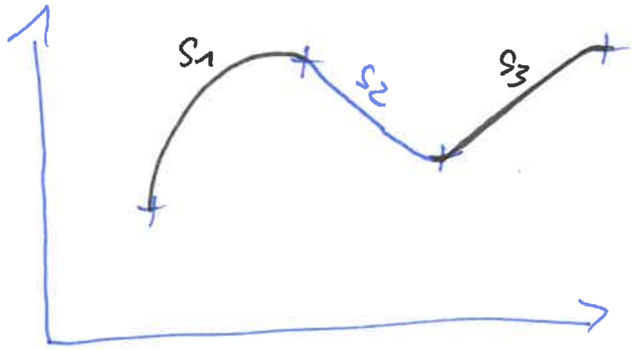
left point: $S_i(x_i) = y_i \Rightarrow a_1^{(i)} x_i + a_0^{(i)} = y_i$

right " : $S_i(x_{i+1}) = y_{i+1} \Rightarrow a_1^{(i)} x_{i+1} + a_0^{(i)} = y_{i+1}$

$$\begin{pmatrix} x_i & 1 \\ x_{i+1} & 1 \end{pmatrix} \begin{pmatrix} a_1^{(i)} \\ a_0^{(i)} \end{pmatrix} = \begin{pmatrix} y_i \\ y_{i+1} \end{pmatrix}$$

(2) Quadratic splines:

Ansatz: $S_i(x) = a_2^{(i)} \cdot x^2 + a_1^{(i)} \cdot x + a_0^{(i)}$



- parabolic pieces which are smothly linked
- derivatives should be equal left and right to the touch-point (intermediate points)

Constraints:

$S_i(x_i) = y_i$ → $(N-1)$ constraints

$S_i(x_{i+1}) = y_{i+1}$ → $(N-1)$ "

$S_i'(x_{i+1}) = S_{i+1}'(x_{i+1})$ → $(N-2)$ constraints at intermediate points

we need $3 \cdot (N-1)$ constraints to fit all parameters of the spline

→ choose a boundary condition:

$S_1''(x_1) = 0 \Rightarrow a_2^{(1)} = 0$

I $\alpha_2^{(i)} x_i^2 + \alpha_1^{(i)} x_i + \alpha_0^{(i)} = y_i$

II $\alpha_2^{(i)} x_{i+1}^2 + \alpha_1^{(i)} x_{i+1} + \alpha_0^{(i)} = y_{i+1}$

III $2\alpha_2^{(i)} x_{i+1} + \alpha_1^{(i)} = 2\alpha_2^{(i+1)} x_{i+1} + \alpha_1^{(i+1)}$

$$2\alpha_2^{(i)} x_{i+1} + \alpha_1^{(i)} - 2\alpha_2^{(i+1)} - \alpha_1^{(i+1)} = 0$$

(IV) $\alpha_2^{(1)} = 0$

Note: In equ III there is a mixing of parameters of different pieces.
$\Rightarrow$ it cannot be solved for every piece seperately!

$i=1$

$$\underbrace{\begin{pmatrix} \boxed{\begin{matrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \end{matrix}} & 0 \ldots & & & 0 \\ \boxed{\begin{matrix} 2x_2 & 1 & 0 & -2x_2 & -1 & 0 \end{matrix}} & \ldots & & & 0 \\ & & & & \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} \alpha_2^{(1)} \\ \alpha_1^{(1)} \\ \alpha_0^{(1)} \\ \alpha_2^{(2)} \\ \alpha_1^{(2)} \\ \alpha_0^{(2)} \\ \vdots \\ \alpha_0^{(n-1)} \end{pmatrix}}_{\vec{\alpha}} = \underbrace{\begin{pmatrix} y_1 \\ y_2 \\ 0 \\ \vdots \\ \vdots \end{pmatrix}}_{\vec{b}}$$

Coding: Fill in entries of A systematically