Appendix: Derivation of Cubic Spline Algorithm

Recap Piecewise cubic polynomial interpolation — Hermite cubic / cubic spline

$\boxed{\text{Set-up}}$ Given (x_i, y_i) for $i = 1, \cdots, n$,

find a piecewise cubic interpolant $\mathbb{P}(x)$ of the form

$$\mathbb{P}(x) = \begin{cases} p_1(x), & [x_1, x_2) \\ p_2(x), & [x_2, x_3) \\ \vdots & \vdots \\ p_{n-1}(x), & [x_{n-1}, x_n] \end{cases}$$

← half-open, half-closed

← closed interval

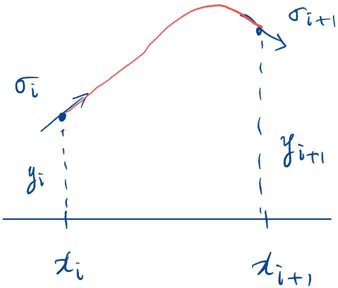
where the i^{th} local cubic $p_i(x)$ is written as

$$p_i(x) = c_{i,1} + c_{i,2}(x - x_i) + c_{i,3}(x - x_i)^2 + c_{i,4}(x - x_i)^3$$

Hermite Cubic Interpolation

- In addition to x_i & y_i, $\underline{\underline{\sigma_i}}$ is given.
  ↳ slope

- For each $i = 1, \dots, n-1$, we require



i^{th} interval

$\underline{\underline{4}}$ eqn.
$$\begin{cases} p_i(x_i) = y_i, & p_i(x_{i+1}) = y_{i+1} \\ p_i'(x_i) = \sigma_i, & p_i'(x_{i+1}) = \sigma_{i+1} \end{cases}$$

$\underline{\underline{4}}$ unk.
$$p_i(x) = c_{i,1} + c_{i,2}(x-x_i) + c_{i,3}(x-x_i)^2 + c_{i,4}(x-x_i)^3$$

Cubic Spline: Problem Set-Up

Given $\{(x_i, y_i) \,|\, i = 1, 2, \ldots, n\}$, find a piecewise polynomial $\mathbb{p}(x) = p_i(x)$ on $[x_i, x_{i+1}]$ with

$$p_i(x) = c_{i,1} + c_{i,2}(x - x_i) + c_{i,3}(x - x_i)^2 + c_{i,4}(x - x_i)^3,$$

Note: σ_i not given.

Goal: With only (x_i, y_i), find a piecewise cubic interpolant $\mathbb{P}$ which is in $C^2[x_1, x_n]$.

$\mathbb{P}$ is twice continuously differentiable. (smooth)

satisfying[1]

n ❶ $\mathbb{p}(x_i) = y_i$ for $i = 1, \ldots, n$; (interpolation)

$n-2$ ❷ $p_i(x_{i+1}) = p_{i+1}(x_{i+1})$ for $i = 1, \ldots, n-2$; (matching values)

$n-2$ ❸ $p_i'(x_{i+1}) = p_{i+1}'(x_{i+1})$ for $i = 1, \ldots, n-2$; (matching deriv.)

$n-2$ ❹ $p_i''(x_{i+1}) = p_{i+1}''(x_{i+1})$ for $i = 1, \ldots, n-2$. (matching 2^{nd} deriv.)

$(4n-6)$ eqns

Unknowns: $c_{i,1}, c_{i,2}, c_{i,3}, c_{i,4}$ for each $i = 1, \cdots, n-1$

$\Rightarrow 4(n-1) = (4n-4)$ unknowns

Need 2 more eqns

[1]Let us not worry about the boundary conditions yet.

Connection to Hermite Cubic Interpolation

Key Observation. If $c_{i,j}$'s are set to be

$$
\begin{aligned}
c_{i,1} &= y_i \,, \qquad & c_{i,3} &= \frac{3y[x_i, x_{i+1}] - 2\sigma_i - \sigma_{i+1}}{\Delta x_i} \,, \\
c_{i,2} &= \sigma_i \,, \qquad & c_{i,4} &= \frac{\sigma_i + \sigma_{i+1} - 2y[x_i, x_{i+1}]}{(\Delta x_i)^2} \,,
\end{aligned}
\tag{$\star$}
$$

as in the Hermite cubic interpolation for some constants $\sigma_1, \sigma_2, \ldots, \sigma_n$ to be determined, then $\mathbb{p}(x)$ satisfies the first three requirements from the previous slide.

Reduction. Determine $\sigma_1, \sigma_2, \ldots, \sigma_n$ so that the fourth requirement is satisfied.

Derivation of a Linear System for Cubic Splines (1)

Using ($\star$), write out the fourth requirement $p''_{i-1}(x_i) = p''_i(x_i)$ in terms of $\sigma_{i-1}, \sigma_i, \sigma_{i+1}$, where $i \in \mathbb{N}[2, n-1]$.

Derivation of a Linear System for Cubic Splines (2)

Express the system of $n-2$ equations for $\sigma_1, \ldots, \sigma_n$ as a matrix equation $X\boldsymbol{\sigma} = \mathbf{r}$, where $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)^{\mathrm{T}}$ and $X \in \mathbb{R}^{n\times n}$ and $\mathbf{r} \in \mathbb{R}^n$ are to be found[2].

[2] Since two equations are still missing, leave the first and last rows of X and $\mathbf{r}$ empty for now.

Tridiagonal System for Cubic Splines

Notation. $\Delta x_i = x_{i+1} - x_i$ and $\nabla x_i = \Delta x_{i-1} + \Delta x_i = x_{i+1} - x_{i-1}$.

$$\boxed{X \vec{\sigma} = \vec{r}}$$

$$X = \begin{bmatrix} * & * & * & \cdots & * & * & * \\ \Delta x_2 & 2\nabla x_2 & \Delta x_1 & & & & \\ & \Delta x_3 & 2\nabla x_3 & \Delta x_2 & & & \mathbf{0} \\ & & \ddots & \ddots & \ddots & & \\ \mathbf{0} & & & \Delta x_{n-2} & 2\nabla x_{n-2} & \Delta x_{n-3} & \\ & & & & \Delta x_{n-1} & 2\nabla x_{n-1} & \Delta x_{n-2} \\ * & * & * & \cdots & * & * & * \end{bmatrix},$$

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \vdots \\ \sigma_{n-2} \\ \sigma_{n-1} \\ \sigma_n \end{bmatrix}, \quad \text{and} \quad \mathbf{r} = \begin{bmatrix} * \\ 3\left(y[x_1, x_2]\Delta x_2 + y[x_2, x_3]\Delta x_1\right) \\ 3\left(y[x_2, x_3]\Delta x_3 + y[x_3, x_4]\Delta x_2\right) \\ \vdots \\ 3\left(y[x_{n-3}, x_{n-2}]\Delta x_{n-2} + y[x_{n-2}, x_{n-1}]\Delta x_{n-3}\right) \\ 3\left(y[x_{n-2}, x_{n-1}]\Delta x_{n-1} + y[x_{n-1}, x_n]\Delta x_{n-2}\right) \\ * \end{bmatrix}.$$

Implementation of Boundary Conditions (1)

- (**clamped cubic spline**) If slopes at each end are known, fill in the first and the last equation of $X\boldsymbol{\sigma} = \mathbf{r}$ with

$$\sigma_1 = y_1', \quad \sigma_n = y_n'.$$

- (**natural cubic spline**) If the second derivatives at the endpoints are known, then use

$$\begin{aligned} 2\sigma_1 + \sigma_2 &= 3y[x_1, x_2] - \frac{1}{2}\Delta x_1 y_1'' \\ \sigma_{n-1} + 2\sigma_n &= 3y[x_{n-1}, x_n] + \frac{1}{2}\Delta x_{n-1} y_n''. \end{aligned}$$

Implementation of Boundary Conditions (2)

- (**periodic boundary condition**) If the data points come from a periodic function with period $P = x_n - x_1$ so that $\sigma_1 = \sigma_n$, then use

$$\begin{aligned}\Delta x_1 \sigma_{n-1} + 2\nabla x_1 \sigma_1 + \Delta x_{n-1}\sigma_2 &= 3\left(y[x_{n-1}, x_n]\Delta x_1 + y[x_1, x_2]\Delta x_{n-1}\right)\\ \sigma_1 - \sigma_n &= 0.\end{aligned}$$

  Here, take $\nabla x_1 = x_2 - x_0 = x_2 - (x_{n-1} - P)$.

Implementation of Boundary Conditions (3)

- (***not-a-knot* boundary condition**) If nothing is known about the endpoints, require $p_1(x) \equiv p_2(x)$ and $p_{n-2}(x) = p_{n-1}(x)$:

$$\begin{aligned}
(\Delta x_2)^2\,\sigma_1 + \left((\Delta x_2)^2 - (\Delta x_1)^2\right)\sigma_2 - (\Delta x_1)^2\,\sigma_3 \\
= 2\left(y[x_1,x_2]\,(\Delta x_2)^2 - y[x_2,x_3]\,(\Delta x_1)^2\right),
\end{aligned}$$

$$\begin{aligned}
-(\Delta x_{n-1})^2\,\sigma_{n-2} + \left((\Delta x_{n-2})^2 - (\Delta x_{n-1})^2\right)\sigma_{n-1} + (\Delta x_{n-2})^2\,\sigma_n \\
= 2\left(y[x_{n-1},x_n]\,(\Delta x_{n-2})^2 - y[x_{n-2},x_{n-1}]\,(\Delta x_{n-1})^2\right).
\end{aligned}$$

Exercise. Derive the equations shown above.