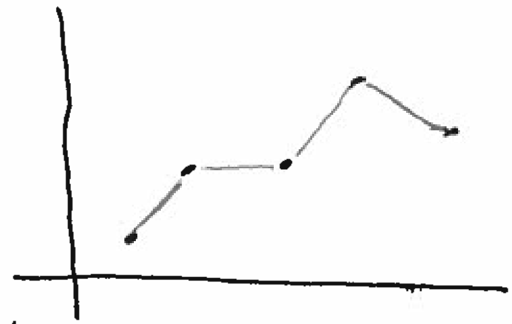
§3.5 Cubic Splines

Piece-wise polynomial interpolation

- Piece-wise Linear
  - not differentiable at the nodes (in general).
  - not "smooth"
  - fairly easy to impliment.



- If f and f' are known, then Hermite polynomials can be used on each $[x_k, x_{k+1}]$ (cubic)
  - The interpolation would be $C^1(x_0, x_n)$
- If f' is not known?
  - quadratic polynomials
    - 3 degrees of freedom
    - On $\{(x_k, y_k), (x_{k+1}, y_{k+1})\}$ gives 2 constraints, so...
    - One degree of freedom to match derivatives at (two) end points
    - Not good if additional info is given at end points (like $f'(x_0), f'(x_n)$).

Cubic spline interpolation

Given f defined on $[a,b]$ and $a = x_0 < x_1 < \cdots < x_n = b$, then S, the cubic spline interpolation for f, is a piecewise defined function satisfying

(a) S_j is a cubic poly. on $[x_j, x_{j+1}]$ $\forall j = 0, 1, 2, \dots, n-1$.

(b) $S_j(x_j) = f(x_j)$ $\forall j$

(c) $S_{j+1}(x_{j+1}) = S_j(x_{j+1})$ $\forall j$

(d) $S'_{j+1}(x_{j+1}) = S'_j(x_{j+1})$ $\forall j$

(e) $S''_{j+1}(x_{j+1}) = S''_j(x_{j+1})$ $\forall j$

(f) One of the following is satisfied:

(i) $S''(x_0) = S''(x_n) = 0$ (free or natural boundaries)

(ii) $\cancel{S'(x_0) = S'(x_n) = 0}$ $S'(x_0) = f'(x_0)$ and $S'(x_n) = f'(x_n)$ (clamped boundaries)

$$S_j(x) = \begin{cases} a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3, & x \in I_j \\ 0, & x \notin I_j \end{cases} \quad \text{where } I_j = [x_j, x_{j+1}]$$

$$S(x) = \sum_{j=0}^{n-1} S_j(x)$$

Note $\quad S_j(x_j) = a_j = f(x_j)$

$S_{j+1}(x_{j+1}) = S_j(x_{j+1}) \Rightarrow$

$$a_{j+1} = a_j + b_j(x_{j+1} - x_j) + c_j(x_{j+1} - x_j)^2 + d_j(x_{j+1} - x_j)^3$$

Let $\quad h_j = x_{j+1} - x_j$, then

$$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3 \qquad (1)$$

$S_j'(x_j) = b_j$, so $\quad S_{j+1}'(x_{j+1}) = S_j'(x_{j+1}) \Rightarrow$

$$b_{j+1} = b_j + 2c_j h_j + 3d_j h_j^2 \qquad (2)$$

Similarly

$$c_{j+1} = c_j + 3d_j h_j \qquad (3)$$

So from (3) we have

$$d_j = \frac{c_{j+1} - c_j}{3h_j}$$

Substitute this into (1) to get

$$a_{j+1} = a_j + b_j h_j + c_j h_j^2 + \frac{h_j^2}{3}(c_{j+1} - c_j)$$

or $$a_{j+1} = a_j + b_j h_j + \frac{h_j^2}{3}(2c_j + c_{j+1}) \qquad (4)$$

Also by substituting into (2) we have

$$b_{j+1} = b_j + 2c_j h_j + h_j(c_{j+1} - c_j)$$

or $$b_{j+1} = b_j + h_j(c_j + c_{j+1}) \qquad (5)$$

Now solve (4) for b_j to get

$$b_j = \frac{1}{h_j}(a_{j+1} - a_j) - \frac{h_j}{3}(2c_j + c_{j+1}) \qquad (6)$$

Substitute (6) into (5) to get

$$\frac{1}{h_{j+1}}(a_{j+2} - a_{j+1}) - \frac{h_{j+1}}{3}(2c_{j+1} + c_{j+2}) =$$
$$\frac{1}{h_j}(a_{j+1} - a_j) - \frac{h_j}{3}(2c_j + c_{j+1}) + h_j(c_j + c_{j+1})$$

$\Longrightarrow$

$$h_j c_j + 2(h_j + h_{j+1})c_{j+1} + h_{j+1} c_{j+2} =$$
$$\frac{3}{h_{j+1}}(a_{j+2} - a_{j+1}) - \frac{3}{h_j}(a_{j+1} - a_j)$$

In order to "center" the subscripts, reduce each by 1.

$$h_{j-1} c_{j-1} + 2(h_{j-1} + h_j)c_j + h_j c_{j+1} = \beta_j \qquad (\bigstar)$$

where $\beta_j = \frac{3}{h_j}(a_{j+1} - a_j) - \frac{3}{h_{j-1}}(a_j - a_{j-1})$

Note that h_k and a_i are known for $k = 0, 1, 2, \ldots, n-1$ and $i = 0, 1, 2, \ldots, n$ from the data. Thus $(\bigstar)$ has only the c_k's as unknowns and is valid for $j = 1, 2, 3, \ldots, n-1$

In matrix form (★) becomes

$$
\begin{bmatrix}
 & & ? & & & & \\
h_0 & 2(h_0+h_1) & h_1 & 0 & 0 & \cdots & 0 \\
0 & h_1 & 2(h_1+h_2) & h_2 & 0 & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
 & & & & \ddots & h_{n-2} & 0 \\
0 & \cdots & 0 & h_{n-2} & 2(h_{n-2}-h_{n-1}) & h_{n-1} & \\
 & & ? & & & &
\end{bmatrix}
\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_{n-1} \\ c_n \end{bmatrix}
=
\begin{bmatrix} \beta_0 = ? \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{n-1} \\ \beta_n = ? \end{bmatrix}
$$

For free boundary conditions $S''(x_0) = S''(x_n) = 0$ $\Rightarrow$ $c_0 = 0$ and $c_n = 0$, so the first and last rows are $1\ 0\ 0 \ldots 0$ and $0\ 0 \ldots 0\ 1$ and $\beta_0 = \beta_n = 0$.

For clamped boundary conditions $f'(x_0) = S'(x_0) = b_0$ and $f'(x_n) = S'(x_n) = b_n$ so (6) $\Rightarrow$

$$2h_0 c_0 + h_0 c_1 = \frac{3}{h_0}(a_1 - a_0) + 3f'(x_0)$$

and

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

This gives the first and last rows and β_0, β_n values.

Look at Algorithm 3.4 on page 147 and Algorithm 3.5 on page 152.

Example Construct the cubic spline with free boundaries that interpolates

$$\text{Data} = \{(0,1),(1,-1),(3,1),(4,2),(7,1)\}$$

$$\vec{h} = \langle 1,2,1,3 \rangle$$

$$\vec{a} = \langle 1,-1,1,2,1 \rangle$$

$$\vec{\beta} = \langle 0, \tfrac{3}{2}(1-(-1)) - \tfrac{3}{1}(-1-1), \tfrac{3}{1}(2-1) - \tfrac{3}{2}(1-(-1)), \tfrac{3}{3}(1-2) - \tfrac{3}{1}(2-1), 0 \rangle$$

$$\Rightarrow$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 6 & 2 & 0 & 0 \\ 0 & 2 & 6 & 1 & 0 \\ 0 & 0 & 1 & 8 & 3 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 9 \\ 0 \\ -4 \\ 0 \end{bmatrix}$$

$$\xRightarrow{\text{rref}}$$

$$\vec{c} = \langle 0, \tfrac{83}{50}, -\tfrac{12}{25}, -\tfrac{11}{25}, 0 \rangle$$

Then use (6) to get $\vec{b}$ and (3) to get $\vec{d}$.