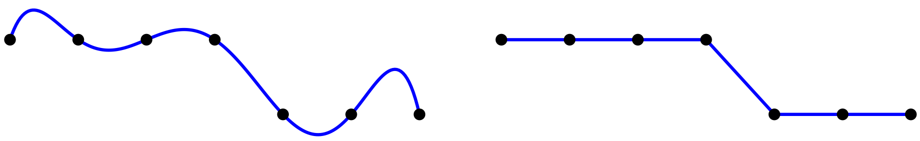
Numerical Analysis Hilary Term 2018
Lectures 14–15: Piecewise Polynomial Interpolation: Splines

Sometimes a 'global' approximation like Lagrange interpolation is not appropriate, e.g., for 'rough' data.



On the left the Lagrange interpolant p_6 'wiggles' through the points, while on the right a **piecewise** linear interpolant ('join the dots'), or linear **spline** interpolant, s appears to represent the data better.

Remark: for any given data s clearly exists and is unique.
Suppose that $a = x_0 < x_1 < \cdots < x_n = b$. Then, s is linear on each interval $[x_{i-1}, x_i]$ for $i = 1, \ldots, n$ and continuous on $[a, b]$. The x_i, $i = 0, 1, \ldots, n$, are called the **knots** of the **linear spline**.

Notation: $f \in \mathrm{C}^k[a, b]$ if $f, f', \ldots, f^k$ exist and are continuous on $[a, b]$.

Theorem. Suppose that $f \in \mathrm{C}^2[a, b]$. Then,

$$\|f - s\|_\infty \leq \frac{1}{8} h^2 \|f''\|_\infty$$

where $h = \max_{1 \leq i \leq n} (x_i - x_{i-1})$ and $\|f''\|_\infty = \max_{x \in [a,b]} |f''(x)|$.

Proof. For $x \in [x_{i-1}, x_i]$, the error from linear interpolation is

$$f(x) - s(x) = \frac{1}{2} f''(\eta)(x - x_{i-1})(x - x_i)$$

where $\eta = \eta(x) \in (x_{i-1}, x_i)$. However, $|(x - x_{i-1})(x - x_i)| = (x - x_{i-1})(x_i - x) = -x^2 + x(x_{i-1} + x_i) - x_{i-1}x_i$, which has its maximum value when $2x = x_i + x_{i-1}$, i.e., when $x - x_{i-1} = x_i - x = \frac{1}{2}(x_i - x_{i-1})$. Thus, for any $x \in [x_{i-1}, x_i]$, $i = 1, 2, \ldots, n$, we have

$$|f(x) - s(x)| \leq \frac{1}{2} \|f''\|_\infty \max_{x \in [x_{i-1}, x_i]} |(x - x_{i-1})(x - x_i)| = \frac{1}{8} h^2 \|f''\|_\infty. \qquad \square$$

Note that s may have discontinuous derivatives, but is a locally defined approximation, since changing the value of one data point affects the approximation in only two intervals.
To get greater smoothness but retain some 'locality', we can define **cubic splines** $s \in \mathrm{C}^2[a, b]$. For a given 'partition', $a = x_0 < x_1 < \cdots < x_n = b$, these are (generally different!) cubic polynomials in each interval (x_{i-1}, x_i), $i = 1, \ldots, n$, which are 'joined' at each knot to have continuity and continuity of s' and s''. **Interpolating cubic splines** also satisfy $s(x_i) = f_i$ for given data f_i, $i = 0, 1, \ldots, n$.

Remark: if there are n intervals, there are $4n$ free coefficients (four for each cubic 'piece'), *but* $2n$ interpolation conditions (one each at the ends of each interval), $n - 1$ derivative continuity conditions (at $x_1, \ldots, x_{n-1}$) and $n - 1$ second derivative continuity conditions (at the same points), giving a total of $4n-2$ conditions (which are linear in the free coefficients). Thus the spline is not unique. So we need to add two extra conditions to generate a spline that might be unique. There are three common ways of doings this:

(a) specify $s'(x_0) = f'(x_0)$ and $s'(x_n) = f'(x_n)$; or

(b) specify $s''(x_0) = 0 = s''(x_n)$ — this gives a **natural** cubic spline; or

(c) enforce continuity of s''' at x_1 and x_{n-1} (which implies that the first two pieces are the same cubic spline, i.e., on $[x_0, x_2]$, and similarly for the last two pieces, i.e., on $[x_{n-2}, x_n]$, from which it follows that x_1 and x_{n-1} are not knots! — this is usually described as the 'not a knot' end-conditions).

We may describe a cubic spline within the i-th interval as

$$s_i(x) = \begin{cases} a_i x^3 + b_i x^2 + c_i x + d_i & \text{for } x \in (x_{i-1}, x_i) \\ 0 & \text{otherwise} \end{cases}$$

and overall, to ensure interpolation (of f), as

$$s(x) = \begin{cases} \displaystyle\sum_{i=1}^{n} s_i(x) & \text{for } \ x \in [x_0, x_n] \setminus \{x_0, x_1, \ldots, x_n\} \\ f(x_i) & \text{for } \ x = x_i, \ \ i = 0, 1, \ldots, n. \end{cases}$$

The $4n$ linear conditions for an interpolating cubic spline s are:

$$\begin{array}{ccc} & s_i(x_i^-) = f(x_i) & \\ s_1(x_0) = f(x_0) & s_{i+1}(x_i^+) = f(x_i) & s_n(x_n) = f(x_n) \\ s_1'(x_0) = f'(x_0) \ \ \text{(a)} & s_i'(x_i^-) - s_{i+1}'(x_i^+) = 0 & s_n'(x_n) = f'(x_n) \ \ \text{(a)} \\ \text{or} \ \ s_1''(x_0) = 0 \ \ \text{(b)} & s_i''(x_i^-) - s_{i+1}''(x_i^+) = 0 & \text{or} \ \ s_n''(x_n) = 0 \ \ \text{(b)} \\ & i = 1, \ldots, n-1. & \end{array} \tag{1}$$

We may write this as $Ay = g$, with

$$y = (a_1, b_1, c_1, d_1, a_2, \ldots, d_{n-1}, a_n, b_n, c_n, d_n)^{\mathrm{T}}$$

and the various entries of g are $f(x_i)$, $i = 0, 1, \ldots, n$, and $f'(x_0)$, $f'(x_n)$ and zeros for (a) and zeros for (b).
So if A is nonsingular, this implies that $y = A^{-1}g$, that is there is a unique set of coefficients $\{a_1, b_1, c_1, d_1, a_2, \ldots, d_{n-1}, a_n, b_n, c_n, d_n\}$. We now prove that if $Ay = 0$ then $y = 0$, and thus that A is nonsingular for cases (a) and (b) — it is also possible, but more complicated, to show this for case (c).

Theorem. If $f(x_i) = 0$ at the knots x_i, $i = 0, 1, \ldots, n$, and additionally $f'(x_0) = 0 = f'(x_n)$ for case (a), then $s(x) = 0$ for all $x \in [x_0, x_n]$.

Proof. Consider

$$\begin{aligned} \int_{x_0}^{x_n} (s''(x))^2 \, \mathrm{d}x &= \sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} (s_i''(x))^2 \, \mathrm{d}x \\ &= \sum_{i=1}^{n} [s_i'(x) s_i''(x)]_{x_{i-1}}^{x_i} - \sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} s_i'(x) s_i'''(x) \, \mathrm{d}x \end{aligned}$$

using integration by parts. However,

$$\int_{x_{i-1}}^{x_i} s_i'(x)s_i'''(x)\,\mathrm{d}x = s_i'''(x)\int_{x_{i-1}}^{x_i} s_i'(x)\,\mathrm{d}x = s_i'''(x)\,[s_i(x)]_{x_{i-1}}^{x_i} = 0$$

since $s_i'''(x)$ is constant on the interval (x_{i-1}, x_i) and $s_i(x_{i-1}) = 0 = s_i(x_i)$. Thus, matching first and second derivatives at the knots, telescopic cancellation gives

$$\begin{aligned}\int_{x_0}^{x_n} (s''(x))^2\,\mathrm{d}x = &\ \sum_{i=1}^{n} [s_i'(x)s_i''(x)]_{x_{i-1}}^{x_i} \\ = &\ s_1'(x_1)s_1''(x_1) - s_1'(x_0)s_1''(x_0) \\ &+ s_2'(x_2)s_2''(x_2) - s_2'(x_1)s_2''(x_1) + \cdots \\ &+ s_{n-1}'(x_{n-1})s_{n-1}''(x_{n-1}) - s_{n-1}'(x_{n-2})s_{n-1}''(x_{n-2}) \\ &+ s_n'(x_n)s_n''(x_n) - s_n'(x_{n-1})s_n''(x_{n-1}) \\ = &\ s_n'(x_n)s_n''(x_n) - s_1'(x_0)s_1''(x_0).\end{aligned}$$

However, in case (a), $f'(x_0) = f'(x_n) \implies s_1'(x_0) = 0 = s_n'(x_n)$, while in case (b) $s_1''(x_0) = 0 = s_n''(x_n)$. Thus, either way,

$$\int_{x_0}^{x_n} (s''(x))^2\,\mathrm{d}x = 0,$$

which implies that $s_i''(x) = 0$ and thus $s_i(x) = c_i x + d_i$. Since $s(x_{i-1}) = 0 = s(x_i)$, $s(x)$ is identically zero on $[x_0, x_n]$. □

Constructing cubic splines. Note that (1) provides a constructive method for finding an interpolating spline, but generally this is not used. Motivated by the next result, it is better to find a good basis.

Proposition. The set of natural cubic splines on a given set of knots $x_0 < x_1 < \cdots < x_n$ is a vector space.

Proof. If $p, q \in \mathrm{C}^2[a,b] \implies \alpha p + \beta q \in \mathrm{C}^2[a,b]$ and $p, q \in \Pi_3 \implies \alpha p + \beta q \in \Pi_3$ for every α, $\beta \in \mathbb{R}$. Finally, the natural end-conditions (b) $\implies (\alpha p + \beta q)''(x_0) = 0 = (\alpha p + \beta q)''(x_n)$ whenever p'' and q'' are zero at x_0 and x_n. □

Best spline bases: the **Cardinal splines**, C_i, $i = 0, 1, \ldots, n$, defined as the interpolatory natural cubic splines satisfying

$$C_i(x_j) = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j, \end{cases}$$

are a basis for which

$$s(x) = \sum_{i=0}^{n} f(x_i)C_i(x)$$

is the interpolatory natural cubic spline to f.

Preferred are the **B-splines** (locally) defined by $B_i(x_i) = 1$ for $i = 2, 3, \ldots, n-2$, $B_i(x) \equiv 0$ for $x \notin (x_{i-2}, x_{i+2})$, B_i a cubic spline with knots x_j, $j = 0, 1, \ldots, n$, with special definitions for B_0, B_1, B_{n-1} and B_n.

Example/construction: Cubic B-spline with knots $0, 1, 2, 3, 4$. On $[0, 1]$,

$$B(x) = ax^3$$

for some a in order that B, B' and B'' are continuous at $x = 0$ (recall that $B(x)$ is required to be identically zero for $x < 0$). So

$$B(1) = a, \quad B'(1) = 3a, \quad \text{and} \quad B''(1) = 6a.$$

On $[1, 2]$, since B is a cubic polynomial, using Taylor's Theorem,

$$\begin{aligned} B(x) = & \quad B(1) + B'(1)(x-1) + \frac{B''(1)}{2}(x-1)^2 + \beta(x-1)^3 \\ = & \quad a + 3a(x-1) + 3a(x-1)^2 + \beta(x-1)^3 \end{aligned}$$

for some β, and since we require $B(2) = 1$, then $\beta = 1 - 7a$. Now, in order to continue, by symmetry, we must have $B'(2) = 0$, i.e.,

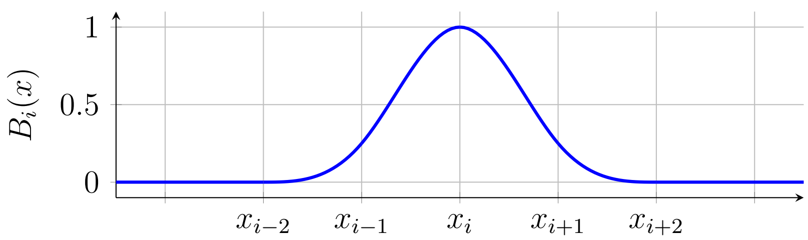
$$3a + 6a(x-1)_{x=2} + 3(1-7a)(x-1)^2_{x=2} = 3 - 12a = 0$$

and hence $a = \frac{1}{4}$. So

$$B(x) = \begin{cases} 0 & \text{for } \ x < 0 \\ \frac{1}{4}x^3 & \text{for } \ x \in [0, 1] \\ -\frac{3}{4}(x-1)^3 + \frac{3}{4}(x-1)^2 + \frac{3}{4}(x-1) + \frac{1}{4} & \text{for } \ x \in [1, 2] \\ -\frac{3}{4}(3-x)^3 + \frac{3}{4}(3-x)^2 + \frac{3}{4}(3-x) + \frac{1}{4} & \text{for } \ x \in [2, 3] \\ \frac{1}{4}(4-x)^3 & \text{for } \ x \in [3, 4] \\ 0 & \text{for } \ x > 4. \end{cases}$$

More generally: B-spline on $x_i = a + hi$, where $h = (b-a)/n$.

$$B_i(x) = \begin{cases} 0 & \text{for } \ x < x_{i-2} \\ \dfrac{(x - x_{i-2})^3}{4h^3} & \text{for } \ x \in [x_{i-2}, x_{i-1}] \\ -\dfrac{3(x-x_{i-1})^3}{4h^3} + \dfrac{3(x-x_{i-1})^2}{4h^2} + \dfrac{3(x-x_{i-1})}{4h} + \dfrac{1}{4} & \text{for } \ x \in [x_{i-1}, x_i] \\ -\dfrac{3(x_{i+1}-x)^3}{4h^3} + \dfrac{3(x_{i+1}-x)^2}{4h^2} + \dfrac{3(x_{i+1}-x)}{4h} + \dfrac{1}{4} & \text{for } \ x \in [x_i, x_{i+1}] \\ \dfrac{(x_{i+2}-x)^3}{4h^3} & \text{for } \ x \in [x_{i+1}, x_{i+2}] \\ 0 & \text{for } \ x > x_{i+2}. \end{cases}$$



The 'end' B-splines B_0, B_1, B_{n-1} and B_n are defined analogously by introducing 'phantom' knots x_{-2}, x_{-1}, x_{n+1} and x_{n+2}. The (cubic) B-spline *basis* is only locally affected if some x_i is changed. But note this is not true of the resulting spline itself.

Spline interpolation: find the cubic spline

$$s(x) = \sum_{j=0}^{n} c_j B_j(x),$$

which interpolates f_i at x_i for $i = 0, 1, \ldots, n$. Require

$$f_i = \sum_{j=0}^{n} c_j B_j(x_i) = c_{i-1}B_{i-1}(x_i) + c_i B_i(x_i) + c_{i+1}B_{i+1}(x_i).$$

For equally-spaced data

$$f_i = \tfrac{1}{4}c_{i-1} + c_i + \tfrac{1}{4}c_{i+1},$$

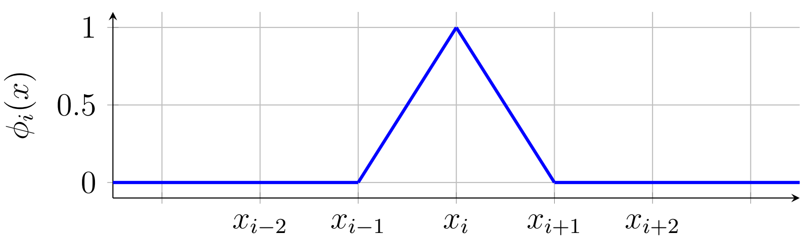
i.e.,

$$\begin{bmatrix} \ddots & \ddots & \ddots & & & & \\ & \frac{1}{4} & 1 & \frac{1}{4} & & & \\ & & \frac{1}{4} & 1 & \frac{1}{4} & & \\ & & & \frac{1}{4} & 1 & \frac{1}{4} & \\ & & & & \ddots & \ddots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ c_{i-2} \\ c_{i-1} \\ c_i \\ c_{i+1} \\ c_{i+2} \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ f_{i-1} \\ f_i \\ f_{i+1} \\ \vdots \end{bmatrix}.$$

The first few and last few rows of this system depend on the type of spline under consideration. For natural cubic splines, see Problem Sheet 7, Question 11.

For linear splines, a similar local basis of 'hat functions' or **Linear B-splines** $\phi_i(x)$ exist:

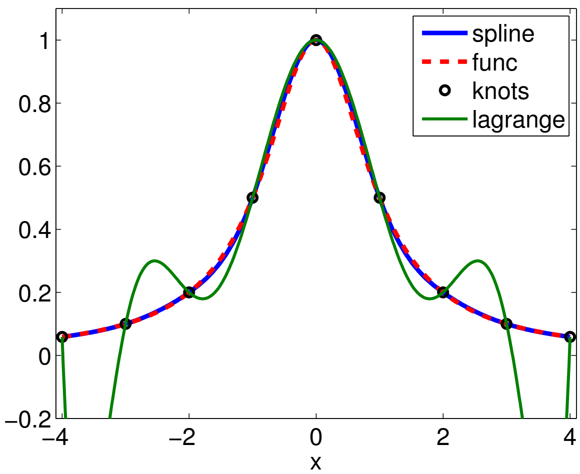
$$\phi_i(x) = \begin{cases} \dfrac{x - x_{i-1}}{x_i - x_{i-1}} & x \in (x_{i-1}, x_i) \\ \dfrac{x - x_{i+1}}{x_i - x_{i+1}} & x \in (x_i, x_{i+1}) \\ 0 & x \notin (x_{i-1}, x_{i+1}) \end{cases}$$



and provide a C^0 piecewise basis.

Listing 1: demo_lec14_spline_vs_lagrange.m

```
N = 9;  % number of interpolation points
x = linspace(-4, 4, N);  % the knots

% values at knots
f = @(x) 1 ./ (1 + x.^2);
fp = @(x) -2*x ./ (1 + x.^2)^2;
ypoints = f(x);

% an extended vector padded with the slope at the first and last
% interpolation points, see "help spline": end-point choices available
% with the matlab command spline (called option (a) in lecture notes).
y = [fp(x(1))  ypoints  fp(x(end))];

% a data structure containing the pieces of the spline
s = spline(x, y);

fine = linspace(-4, 4, 500);
h = figure(1); clf; lw = 'linewidth';
plot(fine, ppval(s, fine), lw,2); % see "help ppval"
ff = f(fine);

% Plot function
hold on
plot(fine, f(fine), 'r--', lw,2);
plot(x, ypoints, 'ko', lw,2)
% Compare to Lagrange interpolating polynomial
p = lagrange_poly(x, ypoints);
plot(fine, polyval(p, fine), 'g-', 'color',[0 0.5 0], lw,2);

set(get(h, 'children'), 'fontsize', 16)
legend('spline', 'func', 'knots', 'lagrange')
ylim([-0.2 1.1]); xlim([-4.1 4.1])
xlabel('x')
```



Error analysis for cubic splines

Theorem. Amongst all functions $t \in \mathrm{C}^2[x_0, x_n]$ that interpolate f at x_i, $i = 0, 1, \ldots, n$, the unique function that minimizes

$$\int_{x_0}^{x_n} [t''(x)]^2 \, \mathrm{d}x$$

is the natural cubic spline s. Moreover, for any such t,

$$\int_{x_0}^{x_n} [t''(x)]^2 \, \mathrm{d}x - \int_{x_0}^{x_n} [s''(x)]^2 \, \mathrm{d}x = \int_{x_0}^{x_n} [t''(x) - s''(x)]^2 \, \mathrm{d}x.$$

Proof. See exercises (uses integration by parts and telescopic cancellation, and is similar to the proof of existence above). □

We will also need:

Lemma. (Cauchy–Schwarz inequality): if $f, g \in \mathrm{C}[a, b]$, then

$$\left[\int_a^b f(x)g(x) \, \mathrm{d}x\right]^2 \leq \int_a^b [f(x)]^2 \, \mathrm{d}x \int_a^b [g(x)]^2 \, \mathrm{d}x.$$

Proof. For any $\lambda \in \mathbb{R}$,

$$0 \leq \int_a^b [f(x) - \lambda g(x)]^2 \, \mathrm{d}x = \int_a^b [f(x)]^2 \, \mathrm{d}x - 2\lambda \int_a^b [f(x)g(x)] \, \mathrm{d}x + \lambda^2 \int_a^b [g(x)]^2 \, \mathrm{d}x.$$

The result then follows directly since the discriminant of this quadratic must be nonpositive. □

Theorem. For the natural cubic spline interpolant s of $f \in \mathrm{C}^2[x_0, x_n]$ at $x_0 < x_1 < \cdots < x_n$ with $h = \max_{1 \leq i \leq n}(x_i - x_{i-1})$, we have that

$$\|f' - s'\|_\infty \leq h^{\frac{1}{2}} \left[\int_{x_0}^{x_n} [f''(x)]^2 \, \mathrm{d}x\right]^{\frac{1}{2}} \quad \text{and} \quad \|f - s\|_\infty \leq h^{\frac{3}{2}} \left[\int_{x_0}^{x_n} [f''(x)]^2 \, \mathrm{d}x\right]^{\frac{1}{2}}.$$

Proof. Let $e := f - s$. Take any $x \in [x_0, x_n]$, in which case $x \in [x_{j-1}, x_j]$ for some $j \in \{1, \ldots, n\}$. Then $e(x_{j-1}) = 0 = e(x_j)$ as s interpolates f. So by the Mean Value Theorem, there is a $c \in (x_{j-1}, x_j)$ with $e'(c) = 0$. Hence $e'(x) = \int_c^x e''(t) \, \mathrm{d}t$. Then the Cauchy–Schwarz inequality gives that

$$|e'(x)|^2 \leq \left|\int_c^x \mathrm{d}t\right| \left|\int_c^x [e''(t)]^2 \, \mathrm{d}t\right|.$$

However, the first required inequality then follows since for $x \in [x_{j-1}, x_j]$, $\left|\int_c^x \mathrm{d}t\right| \leq h$ and because the previous theorem gives that

$$\left|\int_c^x [e''(t)]^2 \, \mathrm{d}t\right| \leq \left|\int_c^x [f''(t)]^2 \, \mathrm{d}t\right| \leq \int_{x_0}^{x_n} [f''(x)]^2 \, \mathrm{d}x.$$

The remaining result follows from Taylor's Theorem. □

Theorem. Suppose that $f \in \mathrm{C}^4[a,b]$ and s satisfies end-conditions (a). Then,

$$\|f - s\|_\infty \leq \frac{5}{384} h^4 \|f^{(4)}\|_\infty$$

and

$$\|f' - s'\|_\infty \leq \frac{9 + \sqrt{3}}{216} h^3 \|f^{(4)}\|_\infty,$$

where $h = \max_{1 \leq i \leq n}(x_i - x_{i-1})$.

Proof. Beyond the scope of this course. □

Similar bounds exist for natural cubic splines and splines satisfying end-condition (c).