Numerical Methods
Polynomial Interpolation

Martijn Boussé, Pieter Collins & Başak Sakçak

Department of Advanced Computing Sciences
Maastricht University
m.bousse,pieter.collins,basak.sakcak@maastrichtuniversity.nl

KEN1540 & KEN2540, Block 5, April-May 2025

Data Analysis

Data and Models A key task of data science is the construction of *models* from *data*.

Often, the data consists of pairs (x_i, y_i), and the model is a function g describing y in terms of x.

The function g is usually restricted to lie in some *model class*, such as linear functions.

Data Interpolation Given data points $(x_0, y_0), \ldots, (x_n, y_n)$, find a function g such that $g(x_i) = y_i$ for $i = 0, \ldots, n$.

Data Approximation Given data points $(x_0, y_0), \ldots, (x_n, y_n)$, find a function g such that $g(x_i) \approx y_i$ for $i = 0, \ldots, n$.

e.g. Minimise the *sum-of-squares error* $\sum_{i=0}^{n}(g(x_i) - y_i)^2$.

Taylor polynomial

Problem Find a polynomial p approximating a function f in a neighbourhood of a point x_0.

$$p(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)^2 + a_3(x - x_0)^3 + a_4(x - x_0)^4 + \cdots$$

Derivatives

$$\begin{aligned} p'(x) &= a_1 + 2a_2(x - x_0) + 3a_3(x - x_0)^2 + 4a_4(x - x_0)^3 + \cdots \\ p''(x) &= 2a_2 + 6a_3(x - x_0) + 12a_4(x - x_0)^2 + \cdots \\ p'''(x) &= 6a_3 + 24a_4(x - x_0) + \cdots \end{aligned}$$

Match derivatives $p^{(k)}(x_0) = k!\, a_k = f^{(k)}(x_0)$, so $a_k = f^{(k)}(x_0)/k!$.

Taylor polynomial

$$\begin{aligned} p_n(x) = f(x_0) + f'(x_0)(x - x_0) + \tfrac{1}{2} f''(x_0)(x - x_0)^2 \\ + \cdots \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k + \cdots \frac{f^{(n)}(x_0)}{n}(x - x_0)^n. \end{aligned}$$

Taylor polynomial

Example Approximate $f(x) = e^x \cos(2x)$ in a neighbourhood of $x_0 = 0$ by a cubic polynomial p.

Match derivatives, $p^{(k)}(x_0) = f^{(k)}(x_0)$:

$$p(x) = f(x_0) + f'(x_0)(x - x_0) + \tfrac{1}{2} f''(x_0)(x - x_0)^2 + \tfrac{1}{3!} f'''(x_0)(x - x_0)^3.$$

Compute derivatives:

$$f'(x) = e^x\big(\cos(2x) - 2\sin(2x)\big),\ f''(x) = e^x\big(-3\cos(2x) - 4\sin(2x)\big),$$
$$f'''(x) = e^x\big(-11\cos(2x) + 2\sin(2x)\big).$$

Evaluate derivatives:

$$f(0) = 1,\ f'(0) = 1,\ f''(0) = -3,\ f'''(0) = -11.$$

Approximating polynomial:

$$f(x) \approx p_3(x) = 1 + x - \tfrac{3}{2}x^2 - \tfrac{11}{6}x^3.$$

Draw in Matlab:

```
f=@(x)exp(x).*cos(2*x), p3=@(x)1+x-3/2*x.^2-11/6*x.^3,
fplot(f,[-2,2]); hold on; fplot(p3,[-2,2]); hold off;
fplot(@(x)abs(f(x)-p3(x)),[-2,+2]);
```

Taylor series

Taylor series Infinite series

$$f(x) = \sum_{k=0}^{\infty} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k.$$

Example $f(x) = \exp(x)$ gives $f^{(k)}(x) = \exp(x)$, $f^{(k)}(0) = 1$, so the Taylor series at $x_0 = 0$ is

$$\exp(x) = \sum_{k=0}^{\infty} \frac{1}{k!}\, x^k.$$

Example $f(x) = 1/(1-x) = (1-x)^{-1}$ gives $f^{(k)}(x) = k!(1-x)^{-(k+1)}$, so

$$\frac{1}{1-x} = \sum_{k=0}^{\infty} x^k. \quad (\textit{Only} \text{ for } |x| < 1.)$$

Taylor Series

Taylor series Standard functions

$$\frac{1}{1+x} = \sum_{k=0}^{\infty}(-x)^k = \sum_{k=0}^{\infty}(-1)^k x^k = 1 - x + x^2 - x^3 + x^4 - \cdots$$

$$\log(1+x) = \sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{k}\, x^k = x - \tfrac{1}{2}x^2 + \tfrac{1}{3}x^3 - \tfrac{1}{4}x^4 + \cdots.$$

$$\exp(x) = \sum_{k=0}^{\infty} \frac{1}{k!}\, x^k = 1 + x + \tfrac{1}{2}x^2 + \tfrac{1}{6}x^3 + \tfrac{1}{24}x^4 + \cdots.$$

$$\sin(x) = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)!}\, x^{2k+1} = x - \tfrac{1}{6}x^3 + \tfrac{1}{120}x^5 - \tfrac{1}{5040}x^7 + \cdots.$$

$$\cos(x) = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k)!}\, x^{2k} = 1 - \tfrac{1}{2}x^2 + \tfrac{1}{24}x^4 - \tfrac{1}{720}x^6 + \cdots.$$

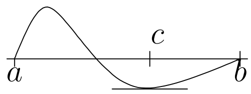
Error in Taylor polynomial

Question: Error in Taylor polynomial What is the error in the approximation

$$f(x) \approx \sum_{k=0}^{n} \frac{1}{k!} f^{(k)}(x_0)(x - x_0)^k\ ?$$

Rolle's theorem (Advanced)

Rolle's Theorem If f is differentiable on $[a,b]$, and $f(a) = f(b) = 0$, then there exists $c \in (a,b)$ such that $f'(c) = 0$.



Multiplicity The *multiplicity* of a root x of f is n if $f^{(k)}(x) = 0$ for $k < n$ and $f^{(n)}(x) \neq 0$.

Theorem (Generalised Rolle's theorem) If f is n-times differentiable on $[a,b]$, and f has roots in $[a,b]$ of total multiplicity at least $n+1$, then there exists $c \in (a,b)$ such that $f^{(n)}(c) = 0$.

Example $f(x) = x(x-1)^3(x-2)^2(x-4)$ has roots at $x = 0, 1, 2, 4$. The root at $x = 1$ has multiplicity 3, and that at $x = 2$ has multiplicity 2 for a total multiplicity of 7. So $f^{(6)}(c) = 0$ for some $c \in (0,4)$; can show $c = 1\frac{4}{7}$.

Error in linear Taylor polynomial (Non-examinable)

Error in (linear) Taylor polynomial
Let $p(x) = f(x_0) + f'(x_0)(x - x_0)$, the linear approximation to f at x_0.

Fix x_* and let $E_* = f(x_*) - p(x_*)$.

Define $q(x) = p(x) + E_*(x - x_0)^2/(x_* - x_0)^2$ and $g(x) = q(x) - f(x)$,
so $g(x) = f(x_0) + f'(x_0)(x - x_0) + E_*(x - x_0)^2/(x_* - x_0)^2 - f(x)$
with $g(x_0) = 0$ and $g(x_*) = 0$; also $g'(x_0) = 0$.

By Rolle's theorem, $g'(\xi_1) = 0$ for some ξ_1 between x_0 and x_*.
Derivative $g'(x) = f'(x_0) + 2E_*(x - x_0)/(x_* - x_0)^2 - f'(x)$.

By Rolle's theorem again, $g''(\xi_2) = 0$ for some ξ_2 between x_0 and ξ_1.
Second derivative $g''(x) = 2E_*/(x_* - x_0)^2 - f''(x)$.

Thus $E_* = \frac{1}{2}f''(\xi_2)(x_* - x_0)^2$ for ξ_2 between x_0 and x_*.

Hence (dropping the $_*$), for some ξ between x_0 and x:

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \tfrac{1}{2}f''(\xi)(x - x_0)^2.$$

Taylor's theorem (Advanced)

Taylor's theorem If f is $(n+1)$-times differentiable. Then there exists $\xi(x)$ between x_0 and x such that

$$f(x) = \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k + \frac{f^{(n+1)}(\xi(x))}{(n+1)!}(x - x_0)^{n+1}.$$

Error bound If $x, x_0 \in [a,b]$, then

$$\left| f(x) - \sum_{k=0}^{n} \frac{1}{k!} f^{(k)}(x_0)\,(x - x_0)^k \right| \leq \frac{1}{(n+1)!} \max_{\xi \in [a,b]} |f^{(n+1)}(\xi)|\,|x - x_0|^{n+1}.$$

Taylor's theorem

Example (Advanced) Approximate $\exp(\frac{1}{2})$ using the Taylor series with $n = 3$, $x_0 = 0$.

By Taylor's theorem,

$$\exp(x) = 1 + x + \tfrac{1}{2}x^2 + \tfrac{1}{6}x^3 + \tfrac{1}{24}\exp(\xi)x^4 \text{ for some } \xi \in (0, x).$$

Since $e < 4$, $\exp(\frac{1}{2}) = e^{1/2} < 4^{1/2} = 2$.

Since $\exp$ is monotonic, if $\xi \in (0, \frac{1}{2})$, $\exp(\xi) \in (\exp(0), \exp(\frac{1}{2})) \subset [1, 2]$.

So for $x \in [0, \frac{1}{2}]$,

$$\exp(x) \in 1 + x + \tfrac{1}{2}x^2 + \tfrac{1}{6}x^3 + \tfrac{1}{24}[1, 2]x^4.$$

Then

$$\begin{aligned}\exp(\tfrac{1}{2}) &\in 1 + \tfrac{1}{2} + \tfrac{1}{2\cdot 4} + \tfrac{1}{6\cdot 8} + \tfrac{1}{24\cdot 16}[1, 2] = [1.6484375, 1.651041\dot{6}] \\ &= 1\tfrac{499}{768} \pm \tfrac{1}{768} = 1.6497 \pm 0.0014.\end{aligned}$$

The exact value is 1.64872127 (8dp), within the computed bounds.

Taylor Series

Exercise (Advanced) Estimate $\cos(\frac{1}{2})$ using the fact that $|\cos(x)| \leq 1$ and the Taylor approximation

$$\cos(x) = 1 - \tfrac{1}{2}x^2 + \tfrac{1}{24}\cos(\xi)x^4.$$

Answer

$$\cos(\tfrac{1}{2}) = 1 - \tfrac{1}{2\cdot 4} + \tfrac{1}{24\cdot 16}[-1, +1] = \tfrac{7}{8} \pm \tfrac{1}{384} \in [0.872395, 0.877605].$$

Exact answer 0.87758256 (8dp).

Error bounds (Advanced)

Bounds of a sum $\max_{x\in[a,b]} |f(x) \pm g(x)| \leq \max_{x\in[a,b]} |f(x)| + \max_{x\in[a,b]} |g(x)|$.

Bounds of a product $\max_{x\in[a,b]} |f(x) \cdot g(x)| \leq \max_{x\in[a,b]} |f(x)| \cdot \max_{x\in[a,b]} |g(x)|$.

Bounds of a monotone function
If f is increasing/decreasing, $\max_{x\in[a,b]} |f(x)| = \max\{|f(a)|, |f(b)|\}$

Bounds of a differentiable function If f has critical points c_i with $f'(c_i) = 0$,then
$\max_{x\in[a,b]} |f(x)| = \max\{|f(a)|, |f(c_1)|, \ldots, |f(c_k)|, |f(b)|\}$

Example Let $f(x) = x(e^x + x)$ on $[-2, +1]$.

$e^x + x$ is increasing, so $|e^x + x| \leq \max(|e^{-2} - 2|, |e^1 + 1|) = |e + 1| \leq 4$.

$\max_{x\in[-2,+1]} |f(x)| \leq \max_{x\in[-2,+1]} |x| \cdot \max_{x\in[-2,+1]} |e^x + x| \leq 2\times 4 = 8$.

A more careful analysis shows $\max_{x\in[-2,+1]} |f(x)| \leq 5$.

Polynomial interpolation

Problem Let $x = (x_0, x_1, \ldots, x_n)$ and $y = (y_0, y_1, \ldots, y_n)$.

Find a polynomial p such that $p(x_i) = y_i$ for $i = 0, \ldots, n$.

Polynomial interpolation in Matlab

Computing polynomials The command

```
cs=polyfit(xs,ys,d)
```

computes coefficients of the polynomial p of degree $d = n - 1$ interpolating

$$\mathtt{xs} = [x_1, x_2, \ldots, x_n], \ \mathtt{ys} = [y_1, y_2, \ldots, y_n].$$

The result cs is a row vector of the coefficients of p in descending order

$$\mathtt{cs} = [c_d, c_{d-1}, \ldots, c_1, c_0].$$

Evaluating polynomials To evaluate the polynomial with coefficients cs at x, use

```
polyval(cs,x)
```

Polynomial function To construct a function p with the coefficients, use

```
p=@(x)polyval(cs,x)
```


Polynomial interpolation in Matlab

Example Find a polynomial p such that $p(x_i) = y_i$ for $i = 0, \ldots, n$.

x_i	0.0	0.5	1.0	2.0	3.0
y_i	1.0	0.8	0.5	0.2	0.1

```
d=4
xs=[0.0,0.5,1.0,2.0,3.0]
ys=[1.0,0.8,0.5,0.2,0.1]
cs = polyfit(xs,ys,d)
polyval(cs,xs) polyval(cs,1.5)
p = @(x)polyval(cs,x)
fplot(p,[-0.5,3.5])
```

Quadratic interpolation

Exercise Interpolate f at $x-h, x, x+h$ by a quadratic polynomial p_2.

Answer

$$\begin{aligned} p_2(y) = f(x) &+ \frac{f(x+h)-f(x-h)}{2h}(y-x) \\ &+ \frac{f(x-y)-2f(x)+f(x+h)}{2h^2}(y-x)^2. \end{aligned}$$

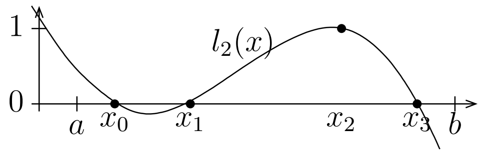
Check interpolation at $y = x+h$:

$$\begin{aligned} p_2(x+h) &= f(x) + \frac{f(x+h)-f(x-h)}{2\cancel{h}}\cancel{h} + \frac{f(x-y)-2f(x)+f(x+h)}{2\cancel{h^2}}\cancel{h^2} \\ &= \cancel{f(x)} + \big(\tfrac{1}{2}f(x+h) \cancel{-\tfrac{1}{2}f(x-h)}\big) + \big(\cancel{\tfrac{1}{2}f(x-y)} \cancel{-f(x)} + \tfrac{1}{2}f(x+h)\big) \\ &= f(x+h) \ \checkmark \end{aligned}$$

Lagrange polynomials

Lagrange basis Fix $x_0, \dots, x_n$. Define Lagrange basis polynomial $l_i(x)$ so that

$$l_i(x_j) = 1 \text{ if } i = j \text{ and } l_i(x_j) = 0 \text{ if } i \neq j.$$



Lagrange form The interpolating polynomial is then

$$p(x) = \sum_{i=0}^{n} y_i\, l_i(x),$$

since

$$p(x_j) = \sum_{i=0}^{n} y_i\, l_i(x_j) = y_j l_j(x_j) + \sum_{i \neq j} \cancel{y_i\, l_i(x_j)} = y_j \cancel{l_j(x_j)} = y_j.$$

Lagrange polynomials

Lagrange basis element Derive

$$\begin{aligned} l_i(x) &= \frac{(x-x_0)}{(x_i-x_0)} \cdots \frac{(x-x_{i-1})}{(x_i-x_{i-1})} \frac{(x-x_{i+1})}{(x_i-x_{i+1})} \cdots \frac{(x-x_n)}{(x_i-x_n)} \\ &= \prod_{\substack{j \neq i \\ j=0}}^{n} \left(\frac{x-x_j}{x_i-x_j} \right). \end{aligned}$$

Example For $n = 2$,

$$l_1(x) = \prod_{\substack{j \neq 1 \\ j=0}}^{2} \left(\frac{x-x_j}{x_i-x_j} \right) = \prod_{j=0,2} \left(\frac{x-x_j}{x_i-x_j} \right) = \frac{(x-x_0)}{(x_1-x_0)} \frac{(x-x_2)}{(x_1-x_2)}$$

$$l_1(x_0) = \frac{\cancel{(x_0-x_0)}}{(x_1-x_0)} \frac{(x_0-x_2)}{(x_1-x_2)} = 0; \quad l_1(x_1) = \frac{\cancel{(x_1-x_0)}}{\cancel{(x_1-x_0)}} \frac{\cancel{(x_1-x_2)}}{\cancel{(x_1-x_2)}} = 1;$$

Lagrange polynomials

Example Interpolate the following data by a polynomial of degree 2.

i	0	1	2
x_i	2.0	2.5	4.0
y_i	0.50	0.40	0.25

Lagrange basis

$$\begin{aligned} l_0(x) &= \tfrac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} = \tfrac{(x-2.5)(x-4.0)}{(2.0-2.5)(2.0-4.0)} = \tfrac{x^2-6.5x+10.0}{1.0} \\ l_1(x) &= \tfrac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} = \tfrac{(x-2.0)(x-4.0)}{(2.5-2.0)(2.5-4.0)} = \tfrac{x^2-6.0x+8.0}{-0.75} \\ l_2(x) &= \tfrac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} = \tfrac{(x-2.0)(x-2.5)}{(2.4-2.0)(4.0-2.5)} = \tfrac{x^2-4.5x+5.0}{3.0} \end{aligned}$$

So

$$\begin{aligned} p(x) &= y_0 l_0(x) + y_1 l_1(x) + y_2 l_2(x) \\ &= 0.50 \times \tfrac{x^2-6.5x+10.0}{1.0} + 0.40 \times \tfrac{x^2-6.0x+8.0}{-0.75} + 0.25 \times \tfrac{x^2-4.5x+5.0}{3.0} \end{aligned}$$

Simplify

$$p(x) = 0.05x^2 - 0.425x + 1.15 = \tfrac{1}{20}x^2 - \tfrac{17}{40}x + \tfrac{23}{20}.$$

23 / 73

Lagrange polynomials

Example Interpolate the following data by a polynomial of degree 2.

i	0	1	2
x_i	2.0	2.5	4.0
y_i	0.50	0.40	0.25

Find

$$p(x) = 0.05x^2 - 0.425x + 1.15 = \tfrac{1}{20}x^2 - \tfrac{17}{40}x + \tfrac{23}{20}.$$

Check by substitution:

$$\begin{aligned} &p(2.0) = 0.05 \times 2.0^2 - 0.85 \times 2.0 + 1.15 = 0.2 - 0.85 + 1.15 = 0.5; \\ &p(2.5) = \cdots . \end{aligned}$$

If $y_i = f(x_i)$, approximate f at other points by p:

$$\begin{aligned} f(3.0) \approx p(3.0) &= 0.05 \times 3.0^2 - 0.425 \times 3.0 + 1.15 \\ &= 0.45 - 1.275 + 1.15 = 0.325. \end{aligned}$$

24 / 73

Existence and uniqeness

Existence and Uniqueness Theorem *There exists a unique polynomial p of degree at most n such that $p(x_i) = y_i$ for $i = 0, \ldots, n$.*

Proof. Existence is proved by the construction of the Lagrange form.

To show uniqueness, note that for fixed $x_0, x_1, \ldots, x_n$, the $n+1$ coefficents c_i in the expansion $p(x) = \sum_{i=0}^{n} c_i x^i$ satisfy the $n+1$ linear equations,

$$c_0 + c_1 x_i + c_2 x_i^2 + \cdots + c_n x_i^n = y_i, \quad i = 0, \ldots, n.$$

Write as a matrix equation

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}.$$

Since there is a solution for any $(y_0, \ldots, y_n)$, the rank of the matrix is $n+1$.
Since the matrix has $n+1$ columns, the nullity is 0.
Hence for any $(y_0, \ldots, y_n)$ the solution $(c_0, \ldots, c_n)$ is unique. □

25 / 73

Polynomial Interpolation by Divided Differences

26 / 73

Nested form

Example Interpolate data

x_i	1	3	-2	4
$f(x_i)$	5.0	1.0	-4.0	9.5

Interpolating at the data point (x_0, y_0) gives

$$p_0(x) = a_0 = y_0 = 5.0\,.$$

To also interpolate at (x_1, y_1) we can add a constant multiple of $(x - x_0)$ so as not to change the value at x_0:

$$p_1(x) = p_0(x) + (x - x_0)a_1 = a_0 + (x - x_0)a_1$$

Substituting $p_1(x_1) = y_1$ gives

$$p_1(x_1) = 5.0 + (3 - 1)a_1 = y_1 = 1.0$$

So $2a_1 = -4.0$, $a_1 = -2.0$ and hence

$$p_1(x) = a_0 + (x - x_0)a_1 = 5.0 - (x - 1) \times 2.0\,.$$

27 / 73

Nested form

Example Interpolate data

x_i	1	3	-2	4
$f(x_i)$	5.0	1.0	-4.0	9.5

The interpolant at x_0, x_1 is

$$p_1(x) = a_0 + (x - x_0)a_1 = 5.0 - (x - 1) \times 2.0\,.$$

To also interpolate at (x_2, y_2) we can add a constant multiple of $(x - x_0)(x - x_1)$ so as not to change the value at x_0 or x_1:

$$\begin{aligned} p_2(x) &= p_1(x) + (x - x_0)(x - x_1)a_2 \\ &= a_0 + (x - x_0)a_1 + (x - x_0)(x - x_1)a_2 \\ &= a_0 + (x - x_0)\big(a_1 + (x - x_1)a_2\big). \end{aligned}$$

Substituting $p_2(x_2) = y_2$ gives

$$p_2(x_2) = p_1(x_2) + (x_2 - x_0)(x_2 - x_1)a_2 = 11.0 + 15a_2 = y_2 = -4.0$$

So $a_2 = -1.0$ and hence

$$p_2(x) = 5.0 + (x - 1) \times (-2.0 + (x - 3) \times (-1.0))$$

28 / 73

Nested form

Example Interpolate data

x_i	1	3	-2	4
$f(x_i)$	5.0	1.0	-4.0	9.5

The interpolant at x_0, x_1, x_2 is

$$p_2(x) = a_0 + (x - x_0)\big(a_1 + (x - x_1)a_2\big) = 5.0 + (x - 1) \times \big(-2.0 + (x - 3) \times (-1.0)\big).$$

To also interpolate at (x_3, y_3) add a constant multiple of $(x - x_0)(x - x_1)(x - x_2)$:

$$\begin{aligned} p_3(x) &= p_2(x) + (x - x_0)(x - x_1)(x - x_2)a_3 \\ &= a_0 + (x - x_0)a_1 + (x - x_0)(x - x_1)a_2 + (x - x_0)(x - x_1)(x - x_2)a_3 \\ &= a_0 + (x - x_0)\big(a_1 + (x - x_1)a_2 + (x - x_1)(x - x_2)a_3)\big) \\ &= a_0 + (x - x_0)\big(a_1 + (x - x_1)\big(a_2 + (x - x_2)a_3\big)\big). \end{aligned}$$

Substituting $p_3(x_3) = p_3(4) = -4.0 + 18a_3 = 9.5 = y_3$ gives $a_3 = 0.75$, so

$$p_3(x) = 5.0 + (x - 1) \times \big(-2.0 + (x - 3) \times \big(-1.0 + (x + 2) \times 0.75\big)\big).$$

29 / 73

Nested form

Example Interpolate data

x_i	1	3	-2	4
$f(x_i)$	5.0	1.0	-4.0	9.5

The interpolant is

$$p_3(x) = 5.0 + (x-1) \times \big(-2.0 + (x-3) \times \big(-1.0 + (x+2) \times 0.75\big)\big).$$

We can expand to the standard basis:

$$p_3(x) = 0.75x^3 - 2.5x^2 - 1.75x + 8.5.$$

However, it is usually more accurate to leave the polynomial in nested form!

e.g. For $x = 1.4$, the error using the nested form and single precision is 5.5×10^{-8}, and for the expanded version is 1.8×10^{-6}.

Nested form

Nested Form This *(Newton) nested form* is a very useful way of writing polynomials:

$$\begin{aligned}
p_0(x) &= a_0\\
p_1(x) &= a_0 + (x-x_0)a_1\\
p_2(x) &= a_0 + (x-x_0)a_1 + (x-x_0)(x-x_1)a_2\\
&= a_0 + (x-x_0)\big(a_1 + (x-x_1)a_2\big)\\
p_3(x) &= a_0 + (x-x_0)a_1 + (x-x_0)(x-x_1)a_2 + (x-x_0)(x-x_1)(x-x_2)a_3\\
&= a_0 + (x-x_0)\big(a_1 + (x-x_1)\big(a_2 + (x-x_2)a_3\big)\big)\\
p_4(x) &= a_0 + (x-x_0)\big(a_1 + (x-x_1)\big(a_2 + (x-x_2)\big(a_3 + (x-x_3)a_4\big)\big)\big).
\end{aligned}$$

The general formula is

$$\begin{aligned}
p(x) &= a_0 + (x-x_0)\big(a_1 + \cdots + (x-x_{n-2})\big(a_{n-1} + (x-x_{n-1})a_n\big)\cdots\big)\\
&= \big(\cdots\big(a_n(x-x_{n-1}) + a_{n-1}\big)(x-x_{n-2}) + \cdots + a_1\big)(x-x_0) + a_0\\
&= \textstyle\sum_{i=0}^n a_i \prod_{j=0}^{i-1}(x-x_j)
\end{aligned}$$

Recursively:

$$p_n(x) = z_0 \text{ where } z_n = a_n; \quad z_k = a_k + (x-x_k)z_{k+1} \text{ for } k = 0, \ldots, n-1.$$

Nested form

Notation Write $p_{[f;x_0,\ldots,x_k]}$ for the polynomial interpolating f at $x_0, \ldots, x_k$.

Subpolynomials Suppose the nested form of the interpolating polynomial at $x_0, \ldots, x_n$ is

$$p_{[f;x_0,\ldots,x_n]} = \textstyle\sum_{i=0}^n a_i \prod_{j=0}^{i-1}(x-x_j),$$

Then for $k < n$, the interpolating polynomial at $x_0, \ldots, x_k$ has the same coefficients a_i!
Hence

$$p_{[f;x_0,\ldots,x_k]} = \textstyle\sum_{i=0}^k a_i \prod_{j=0}^{i-1}(x-x_j).$$

Neville's method

Neville's Method Recursively use the formula:

$$p_{[f;x_0,\ldots,x_k]}(x) = \frac{(x-x_0)p_{[f;x_1,\ldots,x_k]}(x) - (x-x_k)p_{[f;x_0,\ldots,x_{k-1}]}(x)}{x_k - x_0}.$$

How do we know this is correct? Compare values at interpolation points!

$$\begin{aligned}
p_{[f;x_0,\ldots,x_k]}(x_0) &= \frac{(x_0-x_0)p_{[f;x_1,\ldots,x_k]}(x_0) - (x_0-x_k)p_{[f;x_0,\ldots,x_{k-1}]}(x_0)}{x_k - x_0} \\
&= \frac{-(x_0-x_k)f(x_0)}{x_k-x_0} = f(x_0) \\
p_{[f;x_0,\ldots,x_k]}(x_1) &= \frac{(x_1-x_0)p_{[f;x_1,\ldots,x_k]}(x_1) - (x_1-x_k)p_{[f;x_0,\ldots,x_{k-1}]}(x_1)}{x_k - x_0} \\
&= \frac{(x_1-x_0)f(x_1)-(x_1-x_k)f(x_1)}{x_k-x_0} = \frac{x_1-x_0-x_1+x_k}{x_k-x_0}f(x_1) = f(x_1) \\
p_{[f;x_0,\ldots,x_k]}(x_2) &= \cdots
\end{aligned}$$

Neville's method

Example Use Neville's method to interpolate data

x_i	1	-4	0
$y_i = f(x_i)$	0.3	1.3	-2.3

.

$$p_{[f;x_0]} = f(x_0) = 0.3; \quad p_{[f;x_1]} = f(x_1) = 1.3; \quad p_{[f;x_2]} = f(x_2) = -2.3.$$

$$\begin{aligned}
p_{[f;x_0,x_1]} &= \frac{(x-x_0)p_{[f;x_1]} - (x-x_1)p_{[f;x_0]}}{x_1-x_0} \\
&= \big((x-1)\times 1.3 - (x+4)\times 0.3\big)/(-4-1) = -0.2x+5.0 \\
p_{[f;x_1,x_2]} &= \frac{(x-x_1)p_{[f;x_2]} - (x-x_2)p_{[f;x_1]}}{x_2-x_1} \\
&= \big((x+4)\times(-2.3) - x\times 1.3\big)/(0-(-4)) = -0.9x-2.3 \\
p_{[f;x_0,x_1,x_2]} &= \frac{(x-x_0)p_{[f;x_1,x_2]} - (x-x_2)p_{[f;x_0,x_1]}}{x_2-x_0} \\
&= \big((x-1)(-0.9x-2.3) - x(-0.2x+5.0)\big)/(0-1) \\
&= 0.7x^2+1.9x-2.3
\end{aligned}$$

Divided differences

Highest-order coefficient Denote the coefficient of x^k in $p_{[f;x_0,\dots,x_k]}$ by $f[x_0,\dots,x_k]$.

Coefficients In the nested form

$$p_{[f;x_0,\dots,x_k]}(x) = a_0 + (x-x_0)\big(a_1 + \cdots + (x-x_{k-2})\big(a_{k-1} + (x-x_{k-1})a_k\big)\cdots\big),$$

the coefficient of x^k is a_k, so by definition, $a_k = f[x_0,\dots,x_k]$.

Divided differences From Neville's method:

$$p_{[f;x_0,\dots,x_k]}(x) = \frac{(x-x_0)p_{[f;x_1,\dots,x_k]}(x) - (x-x_k)p_{[f;x_0,\dots,x_{k-1}]}(x)}{x_k - x_0},$$

by considering the coefficient of x^k, we obtain:

$$f[x_0,\dots,x_k] = \frac{f[x_1,\dots,x_k] - f[x_0,\dots,x_{k-1}]}{x_k - x_0}.$$

The $f[x_0,\dots,x_k]$ are therefore called *divided differences*.

Divided differences

Divided difference formula The divided differences satisfy

$$f[x_i] = f(x_i)\ ;$$

$$f[x_m,\dots,x_n] = \frac{f[x_m,\dots,x_{i-1},x_{i+1},\dots,x_n] - f[x_m,\dots,x_{j-1},x_{j+1},\dots,x_n]}{x_j - x_i}\ .$$

In particular

$$f[x_i,\dots,x_j] = \frac{f[x_{i+1},\dots,x_j] - f[x_i,\dots,x_{j-1}]}{x_j - x_i}\ .$$

Newton formula Since the divided difference $f[x_0,\dots,x_k]$ is the coefficient a_k in the nested form

$$p_{[f;x_0,\dots,x_n]}(x) = \textstyle\sum_{i=0}^n a_i \prod_{j=0}^{i-1}(x-x_j)$$

we can write the nested form of the interpolating polynomial as:

$$p_{[f;x_0,\dots,x_n]}(x) = \textstyle\sum_{i=0}^n f[x_0,\dots,x_i] \prod_{j=0}^{i-1}(x-x_j)$$

Divided differences

Divided differences table ($n = 3$)

x_0	$f[x_0]$	$f[x_0, x_1]$	$f[x_0, x_1, x_2]$	$f[x_0, x_1, x_2, x_3]$
x_1	$f[x_1]$	$f[x_1, x_2]$	$f[x_1, x_2, x_3]$	
x_2	$f[x_2]$	$f[x_2, x_3]$		
x_3	$f[x_3]$			

Divided differences formulae

x_0	$f(x_0)$	$\frac{f[x_1]-f[x_0]}{x_1-x_0}$	$\frac{f[x_1,x_2]-f[x_0,x_1]}{x_2-x_0}$	$\frac{f[x_1,x_2,x_3]-f[x_0,x_1,x_2]}{x_3-x_0}$
x_1	$f(x_1)$	$\frac{f[x_2]-f[x_1]}{x_2-x_1}$	$\frac{f[x_2,x_3]-f[x_1,x_2]}{x_3-x_1}$	
x_2	$f(x_2)$	$\frac{f[x_3]-f[x_2]}{x_3-x_2}$		
x_3	$f(x_3)$			

Nested form

$$p(x) = f[x_0] + (x - x_0)\big(f[x_0, x_1] + (x - x_1)\big(f[x_0, x_1, x_2] + (x - x_2)f[x_0, x_1, x_2, x_3]\big)\big).$$

Divided differences

Example Interpolate data

x_i	1	-4	0
$f(x_i)$	0.3	1.3	-2.3

.

Compute divided differences

$$f[x_0, x_1] = \frac{f[x_1] - f[x_0]}{x_1 - x_0} = \frac{1.3 - 0.3}{-4 - 1} = -0.2$$

$$f[x_1, x_2] = \frac{f[x_2] - f[x_1]}{x_2 - x_1} = \frac{-2.3 - 1.3}{0 - (-4)} = -0.9$$

$$f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} = \frac{-0.9 - (-0.2)}{0 - 1} = 0.7$$

Divided difference table

$x_0 = 1$	$f(x_0) = 0.3$	$f[x_0, x_1] = -0.2$	$f[x_0, x_1, x_2] = 0.7$
$x_1 = -4$	$f(x_1) = 1.3$	$f[x_1, x_2] = -0.9$	
$x_2 = 0$	$f(x_2) = -2.3$		

Nested form: $p(x) = 0.3 + (x - 1) \times (-0.2 + (x + 4) \times 0.7)$.

Divided differences

Example Interpolate data

x_i	1	-4	0
$f(x_i)$	0.3	1.3	-2.3

Nested form: $p(x) = 0.3 + (x - 1) \times (-0.2 + (x + 4) \times 0.7)$.

Check:

$$\begin{aligned} p(1) &= 0.3 + (1{-}1) \times (-0.2 + (1 + 4) \times 0.7) = 0.3{+}0{\times}(\cdots) = 0.3. \\ p(-4) &= 0.3 + (-4{-}1) \times (-0.2 + (-4{+}4) \times 0.7) \\ &= 0.3{-}5{\times}(-0.2{+}0{\times}0.7) = 1.3. \\ p(0) &= 0.3 + (0{-}1) \times (-0.2 + (0{+}4) \times 0.7) \\ &= 0.3 - (-0.2 + 4 \times 0.7) = 0.3 - 2.6 = -2.3. \end{aligned}$$

Note: Only the check $p(x_n) = y_n$ at the final data point tests all calculated coefficients!

Note: The expanded form is $p(x) = 0.7x^2 + 1.9x - 2.3$, but nested form is usually more accurate to evaluate, so it is better to leave your answer in nested form.

Divided differences

Example Interpolate data

x_i	1	2	4	5
$y_i = f(x_i)$	3.0	1.0	2.0	6.0

Divided differences

$x_0 = 1$	$f[x_0] = 3.0$	$f[x_0, x_1] = -2.0$	$f[x_0, x_1, x_2]{=}0.83$	$f[x_0, x_1, x_2, x_3]{=}0.083$
$x_1 = 2$	$f[x_1] = 1.0$	$f[x_1, x_2] = 0.5$	$f[x_1, x_2, x_3]{=}1.17$	
$x_2 = 4$	$f[x_2] = 2.0$	$f[x_2, x_3] = 4.0$		
$x_3 = 5$	$f[x_3] = 6.0$			

Interpolating polynomial

$$p(x) = 3.0 + (x - 1) \times \big(-2.0 + (x - 2) \times \big(0.83 + (x - 4) \times 0.083\big)\big).$$

Check by computing $p(x)$ at interpolation point x_3.

$$\begin{aligned} p(x_3) &= 3.0 + (5 - 1) \times \big(-2.0 + (5 - 2) \times \big(0.83 + (5 - 4) \times 0.083\big)\big) \\ &= 3.0 + 4{\times}\big(-2.0 + 3{\times}\big(0.83 + 1{\times}0.083\big)\big) = 3.0 + 4{\times}\big(-2.0 + 3{\times}0.917\big) \\ &= 3.0 + 4 \times \big(-2.0 + 2.75\big) = 3.0 + 4 \times 0.75 = 3.0 + 3.0 = 6.0 = f(x_3). \end{aligned}$$

Properties of divided differences

Symmetry The divided difference $f[x_0,\ldots,x_k]$ is independent of the order of the variables:

$$f[x_0,\ldots,x_i,\ldots,x_j,\ldots,x_k] = f[x_0,\ldots,x_j,\ldots,x_i,\ldots,x_k].$$

e.g. $f[x_0,x_1,x_2,x_3] = f[x_0,x_3,x_2,x_1];\ f[x_1,x_2,x_4] = f[x_4,x_1,x_2]$.

Order of computation The divided differences can be computed in many ways.

$$f[x_0,x_1,x_2,x_3] := \frac{f[x_1,x_2,x_3]-f[x_0,x_1,x_2]}{x_3-x_0} = \frac{f[x_0,x_2,x_3]-f[x_1,x_2,x_3]}{x_0-x_1}.$$

Explicit formula From the Lagrange form of the interpolating polynomial:

$$f[x_0,\ldots,x_k] = \sum_{i=0}^{k} \frac{f(x_i)}{\prod_{j=0;\,j\neq i}^{k}(x_i-x_j)}.$$

Extended divided differences (Non-Examinable)

Theorem (Divided differences and derivatives) If $f^{(n)}$ is continous on $[a,b]$ and $x_0,\ldots,x_n$ are distinct points in $[a,b]$, then there exists $\xi \in [a,b]$ such that $f[x_0,\ldots,x_n] = f^{(n)}(\xi)/n!$.

i.e. the n^{th} divided differences are approximations to $f^{(n)}(x)/n!$

Extended divided differences Extend divided differences to the case some of the x_i are equal by defining

$$f[x,x] = f'(x); \quad f[x,x,x] = f''(x)/2; \qquad f[x,x,\ldots,x] = f^{(n)}(x)/n!$$

Then we can compute e.g.

$$f[x,x,y] = \frac{f[x,y]-f[x,x]}{y-x} = \frac{\frac{f(y)-f(x)}{y-x} - f'(x)}{y-x}.$$

Newton's nested form extends naturally to this case!

Function Approximation by Polynomial Interpolation

Function approximation

Function Approximation Given a function $f : [a,b] \to \mathbb{R}$, find a function g such that $g(x) \approx f(x)$

Approximation error Minimise the *uniform/supremum norm*

$$||f-g||_\infty := \sup_{x\in[a,b]} |f(x)-g(x)|.$$

or (easier) the *two-norm*

$$||f-g||_2 := \left(\int_a^b |f(x)-g(x)|^2\,dx\right)^{1/2}.$$

Approximation by interpolation Often compute g as a function interpolating f at points $x_0,x_1,\ldots,x_n$.

Applications

- Often used for computer arithmetic, by approximating a transcendental function (such as $\exp(x)$) by polynomial or rational function.
- May also be used to *simplify* a model by replacing a slow-to-evaluate function by a faster-to-evaluate approximation.

Error of polynomial interpolation

Lagrange basis example Interpolate $y(0) = 1, y(i) = 0$ for $i = -m, \ldots, +m$.

```
m=4, n=2*m; xs=[-m:+m], ys=zeros(1,n+1); ys(m+1)=1,
cs = polyfit(xs,ys,n), p = @(x)polyval(cs,x),
p(xs),
plot([-m,+m],[0,0]); hold; fplot(p,[-m,+m]); hold;
```

Runge example Interpolate $f(x) = 1/(1 + x^2)$ using $n + 1$ equally spaced nodes on $[-4, +4]$.

```
f=@(x)1./(1+x.^2); a=4;
n=8, xs=linspace(-a,+a,n+1), ys=f(xs),
cs = polyfit(xs,ys,n); p = @(x)polyval(cs,x),
fplot(f,[-a,+a]); hold; fplot(p,[-a,+a]); hold;
```

Errors $e_8 = 0.73$, $e_{16} = 5.9$, $e_{32} = 7.1 \cdot 10^2$, $e_{64} = 2.8 \cdot 10^8$.

Approximation accuracy worsens as n increases!!

Error of interpolating polynomial

Theorem (Error of interpolating polynomial) If p is the polynomial of degree at most n that interpolates f at the $n + 1$ distinct nodes $x_0, x_1, \ldots, x_n$ belonging to an interval $[a, b]$, and if $f^{(n+1)}$ is continuous, then for each x in $[a, b]$, there is a ξ in (a, b) for which

$$f(x) - p(x) = \frac{1}{(n+1)!} f^{(n+1)}(\xi) \prod_{i=0}^{n} (x - x_i)$$

Hence

$$|f(x) - p(x)| \leq \frac{1}{(n+1)!} \max_{\xi \in [a,b]} \left| f^{(n+1)}(\xi) \right| \prod_{i=0}^{n} |x - x_i|$$

Error of interpolating polynomial

Proof of interpolating polynomial error (Non-Examinable)
Let p interpolate f at $x_0, \ldots, x_n$. Fix x_*. Let $E_* = f(x_*) - p(x_*)$.

Let l_* be the Lagrange polynomial $l_*(x_*) = 1$ and $l_*(x_i) = 0$ for $i = 0, \ldots, n$.

Then $p(x_i) + E_* l_*(x_i) = f(x_i)$ and $p(x_*) + E_* l_*(x_*) = f(x_*)$.

Let $g(x) = p(x) + E_* l_*(x) - f(x)$ which has zeros at $x_0, \ldots, x_n, x_*$.

By Rolle's theorem, there exists ξ such that $g^{(n+1)}(\xi) = 0$.

Note for all x, $p^{(n+1)}(x) = 0$ and $l_*^{(n+1)}(x) = (n+1)!/\prod_{i=0}^{n}(x_* - x_i)$.

Hence $E_*(n+1)!/\prod_{i=0}^{n}(x_* - x_i) - f^{(n+1)}(\xi) = 0$.

Rearranging gives $f(x_*) - p(x_*) = \frac{1}{(n+1)!} f^{(n+1)}(\xi) \prod_{i=0}^{n}(x_* - x_i)$.

Chebyshev nodes

Chebyshev nodes Interpolation over $[a,b]$ often best using nodes

$$x_k = \frac{a+b}{2} - \frac{b-a}{2}\cos\Bigl(\frac{2k+1}{2(n+1)}\pi\Bigr) \text{ for } k=0,\ldots,n.$$

Runge example Interpolate $f(x)=1/(1+x^2)$ using $n+1$ Chebyshev nodes on $[-4,+4]$.

```
f=@(x)1./(1+x.^2); a=4;
n=8, xs=-a*cos((2*[0:n]+1)*pi/(2*n+2)), ys=f(xs),
cs = polyfit(xs,ys,n); p = @(x)polyval(cs,x);
fplot(f,[-a,+a]); hold on; fplot(p,[-a,+a]); hold off;
```

Errors $e_8 = 0.10$, $e_{16} = 0.015$, $e_{32} = 2.8\cdot 10^{-4}$.

Approximation accuracy improves as n increases.

Error of interpolating polynomials

Theorem (Interpolation error with equally-spaced nodes) If $p(x)$ is the interpolating polynomial of f with $n+1$ equally-spaced nodes on $[a,b]$, then

$$|f(x)-p(x)| \leq \frac{(b-a)^{n+1}}{4n^{n+1}(n+1)} \max_{\xi\in[a,b]} |f^{(n+1)}(\xi)|$$

Theorem (Interpolation error with Chebyshev nodes) If $p(x)$ is the interpolating polynomial of f with $n+1$ Chebyshev nodes, then

$$|f(x)-p(x)| \leq \frac{(b-a)^{n+1}}{2^{2n+1}(n+1)!} \max_{\xi\in[a,b]} |f^{(n+1)}(\xi)|$$

Example If $f(x)=\sin(x)$ on $[-1,+1]$, then $\max_{x\in[a,b]}|f^{(n+1)}(\xi)| \leq 1$.
Hence with $4+1$ equally-spaced nodes have error

$$\epsilon \leq 2^{n+1}/4n^{n+1}(n+1) = 2^5/(4\cdot 4^5\cdot 5) = 1/640$$

and with $4+1$ Chebyshev nodes,

$$\epsilon \leq 2^{n+1}/2^{2n+1}(n+1)! = 1/(2^4\cdot 5!) = 1/1920.$$

Approximation theorems (Non-examinable)

Theorem (Weierstrass) Let f be continuous on $[a,b]$. Then for every $\epsilon>0$, there exists a polynomial p such that

$$||f-p||_\infty := \sup_{x\in[a,b]} |f(x)-p(x)| < \epsilon.$$

Theorem (Chebyshev alternation) Let f be continuous on $[a,b]$. Then there is a unique best approximating polynomial p_m of degree m.

Further, there exist $m+2$ points $w_0,\ldots,w_{m+1}$ such that

$$f(w_i)-p_m(w_i) = \pm(-1)^i||f-p_m||_\infty,$$

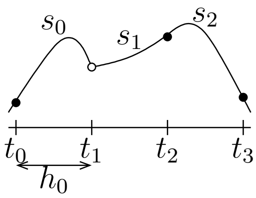
and $m+1$ points $x_0,\ldots,x_m$ such that $p_m(x_i)=f(x_i)$.

If f is n-times continuously differentiable, there is a constant C such that $||f-p_m|| \leq C/m^n$, and if f is smooth (*analytic*), then there is are constants C and $R>1$ such that $||f-p_m|| \leq C/R^m$.

The best approximating polynomial can be computed using the *Remez exchange algorithm*.

Splines

Definition A *spline* of degree n on $[a,b]$ with *knots* at $a = t_0 < t_1 < \cdots < t_{n+1} = b$ is an function s such that s is equal to a degree-n polynomial s_i on $[t_i, t_{i+1}]$ and s is $n-1$ times differentiable (with continuous derivative) at each t_i.



The function above is not a cubic spline since it is not differentiable at t_1.

$$\lim_{x \nearrow t_1} s'(x) = s_0'(t_1) \neq s_1'(t_1) = \lim_{x \searrow t_1} s'(x)$$

52 / 73

Spline interpolation

(Cubic) Spline Interpolation Compute a (cubic) spline s such that $s(x_i) = y_i$ for $i = 0, \ldots, n$.

Knots at interpolation points For cubic splines, take knots at interpolation points, so $t_i = x_i$ for $i = 0, \ldots, n$.

53 / 73

Spline Interpolation in Matlab

Computing splines The command

```
S=spline(X,Y)
```

computes the spline interpolating data

$$X = [x_1, x_2, \ldots, x_n],\ Y = [y_1, y_2, \ldots, y_n]$$

with *not-a-knot* end conditions $s'''(x_2) = s'''(x_{n-1}) = 0$.

The command

```
S=spline(X,[b1,Y,bn])
```

computes the spline interpolating data X, Y with *clamped* end conditions $s'(x_1) = b_1$ and $s'(x_n) = b_n$.

Evaluating splines To evaluate the spline S at x, use the command

```
ppval(S,x)
```

54 / 73

Spline Interpolation in Matlab

Example Standard basis $s(0) = 1$, otherwise $s(i) = 0$, for $i = -n, \dots, n$.

```
n=4; X=[-n:+n], Y=0*X; Y(n+1)=1,
S=spline(X,Y), s=@(x) ppval(S,x),
plot([-n:+n],[0,0]); hold; fplot(s,[-n,+n]); hold
```

Example Interpolate $f(x) = 1/(1 + x^2)$ on $[-4, 4]$ with clamped ends.

```
f=@(x)1./(1+x.^2), df=@(x) 2*x./(1+x.^2).^2, a=-4, b=+4,
n=8; X=linspace(a,b,n+1), Y=f(X), wa=df(a), wb=df(b),
S=spline(X,[wa,Y,wb]), s=@(x) ppval(S,x),
fplot(f,[-4,+4]); hold on; fplot(s,[-4,+4]); hold off;
```

Spline interpolation does not suffer from the extreme oscillations which may occur in polynomial interpolation!

55 / 73

Spline interpolation conditions (Non-examinable)

Spline formulae For $x \in [x_j, x_{j+1}]$, write $s(x) = s_j(x)$ given as

$$s_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3.$$

The derivatives of s_j are

$$s_j'(x) = b_j + 2c_j(x - x_j) + 3d_j(x - x_j)^2,$$

$$s_j''(x) = 2c_j + 6d_j(x - x_j).$$

Knot points At knot points, $s(x_j) = s_j(x_j) = a_j$, $s'(x_j) = s_j'(x_j) = b_j$, $s''(x_j) = s_j''(x_j) = 2c_j$. Note $c_j = s''(x_j)/2$.

Interpolation conditions The interpolation conditions at x_j imply $a_j = y_j$ for $j = 0, \dots, n$.

Continuity conditions The continuity of s, s', s'' imply for $j = 1, \dots, n - 1$

$$s_{j-1}(x_j) = s_j(x_j), \quad s_{j-1}'(x_j) = s_j'(x_j), \quad s_{j-1}''(x_j) = s_j''(x_j).$$

Alternatively, reindexing gives for $j = 0, \dots, n - 2$.

$$s_j(x_{j+1}) = s_{j+1}(x_{j+1}) \quad s_j'(x_{j+1}) = s_{j+1}'(x_{j+1}), \quad s_j''(x_{j+1}) = s_{j+1}''(x_{j+1}).$$

56 / 73

Spline interpolation conditions (Non-examinable)

Continuity conditions Let $h_j = x_{j+1} - x_j$.
The continuity conditions at x_{j+1} for $j = 0, \dots, n-2$ are:

$$s_j(x_{j+1}) = a_j + b_j h_j + c_j h_j^2 + d_j h_j^3 = a_{j+1} = s_{j+1}(x_{j+1}).$$
$$s'_j(x_{j+1}) = b_j + 2c_j h_j + 3d_j h_j^2 = b_{j+1} = s'_{j+1}(x_{j+1}).$$
$$s''_j(x_{j+1}) = 2c_j + 6d_j h_j = 2c_{j+1} = s''_j(x_{j+1}).$$

The second derivative condition gives

$$d_j = (c_{j+1} - c_j)/3h_j.$$

The zeroth derivative condition then becomes

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

From this we find

$$b_j = (a_{j+1} - a_j)/h_j - (h_j/3)(2c_j + c_{j+1}).$$

and also

$$h_j(2c_j + c_{j+1}) = 3\big((a_{j+1} - a_j)/h_j - b_j\big).$$

Spline interpolation conditions (Non-examinable)

Continuity conditions
The first derivative conditions at x_j for $j = 1, \dots, n-1$ are

$$s'_{j-1}(x_j) = b_{j-1} + 2c_{j-1}h_{j-1} + 3d_{j-1}h_{j-1}^2 = b_j = s'_j(x_j).$$

The coefficients b_j, d_j are given by

$$b_j = (a_{j+1} - a_j)/h_j - h_j(2c_j + c_{j+1})/3; \qquad d_j = (c_{j+1} - c_j)/3h_j.$$

Substituting for b_{j-1}, b_j, d_{j-1} above gives

$$\begin{aligned} &\tfrac{a_j - a_{j-1}}{h_{j-1}} - \tfrac{h_{j-1}}{3}(2c_{j-1} + c_j) + 2c_{j-1}h_{j-1} + 3\tfrac{c_j - c_{j-1}}{3h_{j-1}} h_{j-1}^2 \\ &\qquad = \tfrac{a_{j+1} - a_j}{h_j} - \tfrac{h_j}{3}(2c_j + c_{j+1}) \end{aligned}$$

Rearranging gives

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

Note that the equation for $j = n-1$ involves a_n and c_n, even though they are not needed for any s_j!
However, $a_n = s(x_n) = y_n$ and $c_n = \frac{1}{2}s''(x_n)$.

Spline interpolation conditions (Non-examinable)

End conditions The continuity conditions for the c_j given $n-1$ linear equations (at knots $x_1, \dots, x_{n-1}$) for the $n+1$ unknowns $c_0, \dots, c_n$.

Impose *end conditions* to determine c_0 and c_n.

Clamped boundary $s'(x_0) = b_0$ and $s'(x_n) = b_n$ are given.

$$\begin{aligned} 2h_0c_0 + h_0c_1 &= 3\big((a_1 - a_0)/h_0 - b_0\big), \\ h_{n-1}c_{n-1} + 2h_{n-1}c_n &= 3\big(b_n - (a_n - a_{n-1})/h_{n-1}\big) \\ &= 3\big((a_{n-1} - a_n)/h_{n-1} + b_n\big). \end{aligned}$$

Second derivative at boundary $s''(x_0) = 2c_0$ and $s''(x_n) = 2c_n$ given.

$$c_0 = s''(x_0)/2; \quad c_n = s''(x_n)/2.$$

Natural spline $s''(x_0) = s''(x_n) = 0$, so $c_0 = 0;\ c_n = 0.$

Not a knot (Used by Matlab.) s''' is continuous at x_1 and x_{n-1}.

$$\begin{aligned} h_1c_0 \quad - \quad (h_0 + h_1)c_1 \quad + \ h_0c_2 \ &= 0, \\ h_{n-1}c_{n-2} - (h_{n-2} + h_{n-2})c_{n-1} + h_{n-2}c_n &= 0. \end{aligned}$$

Spline interpolation formulae (Non-examinable)

Pieces For $j = 0, \dots, n-1,\ h_j = x_{j+1} - x_j$. For $x \in [x_j, x_{j+1}]$, $s(x) = s_j(x)$.

Polynomials $s_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3$.

Interpolation conditions For $j = 0, \dots, n,\ a_j = y_j$.

Continuity conditions at knots For $j = 1, \dots, n-1$,

$$h_{j-1}c_{j-1} + 2(h_{j-1} + h_j)c_j + h_jc_{j+1} = 3\big((a_{j+1} - a_j)/h_j - (a_j - a_{j-1})/h_{j-1}\big).$$

Coefficients For $j = 0, \dots, n-1$,

$$b_j = (a_{j+1} - a_j)/h_j - h_j(2c_j + c_{j+1})/3; \qquad d_j = (c_{j+1} - c_j)/3h_j.$$

Clamped boundary

$$\begin{aligned} s'(x_0) = b_0 &\implies 2h_0c_0 + h_0c_1 = 3\big((a_1 - a_0)/h_0 - b_0\big), \\ s'(x_n) = b_n &\implies h_{n-1}c_{n-1} + 2h_{n-1}c_n = 3\big(b_n - (a_n - a_{n-1})/h_{n-1}\big). \end{aligned}$$

Natural spline $s''(x_0) = 0 \implies c_0 = 0; \qquad s''(x_n) = 0 \implies c_n = 0.$

Spline interpolation alternative formulae (Non-examinable)

Symmetric spline formula

$$\begin{aligned} s_j(x) = \frac{c_{j+1}}{3h_j}(x - x_j)^3 + \frac{c_j}{3h_j}(x_{j+1} - x)^3 \\ + \Big(\frac{a_{j+1}}{h_j} - \frac{c_{j+1}h_j}{3}\Big)(x - x_j) + \Big(\frac{a_j}{h_j} - \frac{c_jh_j}{3}\Big)(x_{j+1} - x) \end{aligned}$$

Spline interpolation example (Non-examinable)

Example Compute the cubic spline interpolating the data

i	0	1	2
x_i	1	2	4
y_i	5	3	2

with natural end conditions $s''(1) = s''(4) = 0$.

We have

$$h_0 = x_1 - x_0 = 2 - 1 = 1; \qquad h_1 = x_2 - x_1 = 4 - 2 = 2.$$

The end conditions give $c_0 = 0$ and $c_2 = 0$.

Taking $j = 1$ gives

$$h_0c_0 + 2(h_0 + h_1)c_1 + h_1c_2 = 3\big((a_2 - a_1)/h_1 - (a_1 - a_0)/h_0\big).$$

$$2(1 + 2)c_1 = 3\times\big((2 - 3)/2 - (3 - 5)/1\big) = 3\times(-\tfrac{1}{2} + 2) = \tfrac{9}{2}$$

Hence $6c_1 = \frac{9}{2}$, so $c_1 = \frac{3}{4}$.

Spline interpolation example (Non-examinable)

Example Compute the cubic spline interpolating the data $(i = 0, 1, 2)$

x_i	1	2	4
y_i	5	3	2

with natural end conditions $s''(1) = s''(4) = 0$.

Given $h_0 = 1$, $h_1 = 2$, $a_i = y_i$, $c_0 = 0$, $c_1 = \frac{3}{4}$ and $c_2 = 0$, compute

$$\begin{aligned}
d_0 &= (c_1 - c_0)/3h_0 = (\tfrac{3}{4} - 0)/(3\times 1) = \tfrac{1}{4} \\
d_1 &= (c_2 - c_1)/3h_1 = (0 - \tfrac{3}{4})/(3\times 2) = -\tfrac{1}{8} \\
b_0 &= (a_1 - a_0)/h_0 - h_0(2c_0 + c_1)/3 = (3 - 5)/1 - 1\times(2\times 0 + \tfrac{3}{4})/3 = -\tfrac{9}{4} \\
b_1 &= (a_2 - a_1)/h_1 - h_1(2c_1 + c_2)/3 = (2 - 3)/2 - 2\times(2\times\tfrac{3}{4} + 0)/3 = -\tfrac{3}{2}.
\end{aligned}$$

$$s(x) = \begin{cases} s_0(x) = \ \ \tfrac{1}{4}(x - 1)^3 + 0(x - 1)^2 - \tfrac{9}{4}(x - 1) + 5 \text{ for } x \in [1, 2]; \\ s_1(x) = -\tfrac{1}{8}(x - 2)^3 + \tfrac{3}{4}(x - 2)^2 - \tfrac{3}{2}(x - 2) + 3 \text{ for } x \in [2, 4]. \end{cases}$$

Equally-spaced knots (Non-examinable)

Equally-spaced knots $h_j = h$ for all j.

Continuity conditions at knots

$$c_{j-1} + 4c_j + c_{j+1} = 3(a_{j+1} - 2a_j + a_{j-1})/h^2.$$

Coefficients

$$b_j = (a_{j+1} - a_j)/h - h(2c_j + c_{j+1})/3; \qquad d_j = (c_{j+1} - c_j)/3h.$$

Clamped boundary s' is given at x_0 and/or x_n.

$$s'(x_0) = b_0 \Rightarrow 2c_0 + c_1 = 3\big((a_1 - a_0)/h - b_0\big)/h = 3\big(a_1 - a_0 - b_0 h\big)/h^2,$$
$$s'(x_n) = b_n \Rightarrow c_{n-1} + 2c_n = 3\big(b_n - (a_n - a_{n-1})/h\big)/h = 3\big(b_n h - a_n + a_{n-1}\big)/h^2.$$

Not a knot s''' is continuous at x_1 and/or x_{n-1}.

$$s_0'''(x_1) = s_1'''(x_1) \implies c_0 - 2c_1 + c_2 = 0,$$
$$s_{n-1}'''(x_{n-1}) = s_n'''(x_{n-1}) \implies c_{n-2} - 2c_{n-1} + c_n = 0.$$

64 / 73

Tridiagonal system (Non-examinable)

Tridiagonal matrix The equations for c_j with clamped boundary conditions $s'(x_0) = b_0$, $s'(x_n) = b_n$ can be written in matrix form as:

$$\begin{pmatrix} 2 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{n-2} \\ c_{n-1} \\ c_n \end{pmatrix} = \frac{3}{h^2} \begin{pmatrix} -b_0 h - a_0 + a_1 \\ a_0 - 2a_1 + a_2 \\ a_1 - 2a_2 + a_3 \\ \vdots \\ a_{n-2} - 2a_{n-1} + a_n \\ a_{n-1} - a_n + b_n h \end{pmatrix}$$

The matrix is *tridiagonal*, and the system easy to solve with $\sim 3n$ operations.

65 / 73

Spline interpolation example (Non-examinable)

Example Compute the cubic spline interpolating the data

i	0	1	2	3
x_i	2	3	4	5
y_i	1.0	2.0	5.0	10.0

with $s''(2) = 0$ and $s'(5) = 6$ over the interval $[4, 5]$. Estimate y at $x = 4.5$.

Equally-spaced knots $h = 1$. Boundary conditions $c_0 = 0$, $b_3 = 6$.

$$\begin{aligned} 4c_1 + \ c_2 \qquad &= 3(a_2 - 2a_1 + a_0)/h^2 = 3\times(5 - 2\times2 + 1)/1^2 = 6; \\ c_1 + 4c_2 + \ c_3 &= 3(a_3 - 2a_2 + a_1)/h^2 = 3\times(10 - 2\times5 + 2)/1^2 = 6; \\ c_2 + 2c_3 &= 3(b_3 - a_3 + a_2)/h^2 = 3\times(6 - 10 + 5)/1^2 = 3. \end{aligned}$$

Solve linear equations

$$\begin{pmatrix} 4 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} 6 \\ 6 \\ 3 \end{pmatrix}$$

to obtain

$$c_1 = \tfrac{33}{26},\ c_2 = \tfrac{24}{26},\ c_3 = \tfrac{27}{26}.$$

66 / 73

Spline interpolation example (Non-examinable)

Example Compute the cubic spline interpolating the data ($i = 0, 1, 2, 3$)

x_i	2	3	4	5
y_i	1.0	2.0	5.0	10.0

with $s''(2) = 0$ and $s'(5) = 6$ over the interval $[4, 5]$. Estimate y at $x = 4.5$.

Interpolation and continuity conditions give

$$c_0 = 0,\ c_1 = \tfrac{33}{26},\ c_2 = \tfrac{24}{26},\ c_3 = \tfrac{27}{26}.$$

For $x \in [4, 5] = [x_2, x_3]$, $s(x) = s_2(x)$.

$$\begin{aligned} b_2 &= (a_3 - a_2)/h - h(2c_2 + c_3)/3 = (10-5)/1 - 1\times(2\times\tfrac{24}{26} - \tfrac{27}{26})/3 = \tfrac{105}{26}; \\ d_2 &= (c_3 - c_2)/3h = (\tfrac{27}{26} - \tfrac{24}{26})/(3\times1) = \tfrac{1}{26}. \end{aligned}$$

Polynomial piece

$$\begin{aligned} s_2(x) &= \tfrac{1}{26}(x-4)^3 + \tfrac{24}{13}(x-4)^2 + \tfrac{105}{26}(x-4) + 5 \\ &= 0.0385(x-4)^3 + 0.923(x-4)^2 + 4.04(x-4) + 5.00. \end{aligned}$$

Evaluate at $x = 4.5$ using s_2. $x - 4 = 4.5 - 4 = 0.5$, so

$$s_2(4.5) = 0.0385\times0.5^3 + 0.923\times0.5^2 + 4.04\times0.5 + 5.00 = 7.3 \ (1\,\text{dp}).$$

67 / 73

Spline interpolation example (Non-examinable)

Exercise Compute the cubic spline interpolating the data

i	0	1	2	3
x_i	2	3	4	5
y_i	1	2	5	10

with $s''(2) = s''(5) = 0$. Evaluate your result at $x = 4.5$.

Answer:

$$c_0 + 4c_1 + c_2 \qquad = 3\times(5 - 2\times2 + 1)/1^2 = 6;$$
$$c_1 + 4c_2 + c_3 = 3\times(10 - 2\times5 + 2)/1^2 = 6.$$

End conditions give $c_0 = 0$ and $c_3 = 0$. Find $c_1 = 1.2$ and $c_2 = 1.2$. Compute $b_2 = 4.2$, $d_2 = -0.4$.

$$s_2(x) = -0.4(x-4)^3 + 1.2(x-4)^2 + 4.2(x-4) + 5.0.$$

Find $s_2(4.5) = 7.35 = 7.4\ (1\,\text{dp})$.

68 / 73

B-Splines (Non-examinable)

69 / 73

B-splines (Non-examinable)

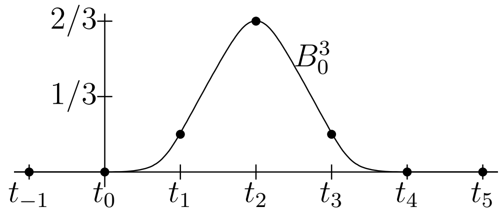
Idea The B-splines B_i^k form a *basis* for splines of degree k and have *compact support* $[t_i, t_{i+k+1})$ i.e. $B_i^k = 0$ unless $t_i \leq x < t_{i+k+1}$.

Constant B-spline Support $[t_i, t_{i+1})$

$$B_i^0(x) = \begin{cases} 1 \text{ if } x \in [t_i, t_{i+1}). \\ 0 \text{ otherwise}. \end{cases}$$

Higher-order B-splines

$$B_i^k(x) = \frac{x - t_i}{t_{i+k} - t_i} B_i^{k-1}(x) + \frac{t_{i+k+1} - x}{t_{i+k+1} - t_{i+1}} B_{i+1}^{k-1}(x) \text{ for } k > 0.$$



70 / 73

B-splines (Non-examinable)

B-splines with integer knots

$$B_i^k(x) = \frac{1}{k}\big((x-i)B_i^{k-1}(x) + (i+k+1-x)B_{i+1}^{k-1}(x)\big).$$

Linear B-spline $B_0^1(x) = \begin{cases} x \text{ if } x \in [0,1), \\ 2-x \text{ if } x \in [1,2). \end{cases}$

Quadratic B-spline $B_0^2(x) = \begin{cases} x^2/2 \text{ if } x \in [0,1), \\ 3/4 - (x-3/2)^2 \text{ if } x \in [1,2), \\ (3-x)^2 \text{ if } x \in [2,3). \end{cases}$

Cubic B-spline $B_0^3(x) = \begin{cases} x^3/6 \text{ if } x \in [0,1), \\ 2/3 - x(2-x)^2/2 \text{ if } x \in [1,2), \\ 2/3 - (4-x)(x-2)^2/2 \text{ if } x \in [2,3), \\ (4-x)^3/6 \text{ if } x \in [3,4). \end{cases}$

71 / 73

B-splines (Non-examinable)

Properties of B-splines

Finite support $B_i^k(x) = 0$ for $x \notin [i, i+k]$.

Sum to unity $\sum_{i=-\infty}^{+\infty} B_i^k(x) = 1$ for all k, x.

Basis Any k-spline can be written as $\sum_{i=-\infty}^{\infty} c_i B_i^k$.

Derivatives $\frac{d}{dx} B_i^k(x) = \frac{k}{t_{i+k}-t_i} B_i^{k-1}(x) - \frac{k}{t_{i+k+1}-t_{i+1}} B_{i+1}^{k-1}(x)$.

Evaluation Use recurrence relation!

$$B_i^k(x) = \frac{x-t_i}{t_{i+k}-t_i} B_i^{k-1}(x) + \frac{t_{i+k+1}-x}{t_{i+k+1}-t_{i+1}} B_{i+1}^{k-1}(x).$$

72 / 73

B-splines example (Non-examinable)

Compute the natural cubic B-spline s interpolating the data

x_i	2	3	4	5
y_i	1	2	5	10

.

For integer knots, we have

$$B_0^3(1) = B_0^3(3) = \tfrac{1}{6},\ B_0^3(2) = \tfrac{2}{3},\ \ [B_0^3]'(1) = -[B_0^3]'(3) = \tfrac{1}{2},\ [B_0^3]'(2) = 0.$$

Hence the coefficients c_i of B_i satisfy

$$\begin{pmatrix} \frac{-1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & 0 & 0 \\ 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 & 0 \\ 0 & 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} & 0 \\ 0 & 0 & 0 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ 0 & 0 & 0 & \frac{1}{2} & 0 & \frac{-1}{2} \end{pmatrix} \begin{pmatrix} c_{-1} \\ c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \\ 3 \\ 5 \\ 10 \\ 0 \end{pmatrix}$$

The solution of the coefficients is $c = \begin{pmatrix} 3\frac{1}{3} & 1\frac{1}{3} & 3\frac{1}{3} & 3\frac{1}{3} & 13\frac{1}{3} & 3\frac{1}{3} \end{pmatrix}^T$, so

$$s(x) = \tfrac{1}{3}\big(10B_{-1}^3(x) + 4B_0^3(x) + 10B_1^3(x) + 10B_2^3(x) + 40B_3^3(x) + 10B_4^3(x)\big).$$