4 Interpolation

4.1 Lagrange interpolation

4.2 Lagrange interpolation

One of the simplest methods for constructing a polynomial approximation to a given function $f(x)$ is to require that the error vanishes at an appropriate number of points, i.e. that the approximation is exact at a certain number of points. In particular, to get a polynomial approximation of degree n to $f(x)$ in $[a,b]$ we look for a polynomial $L_n(x)$ that is exact at $(n+1)$ points inside (a,b). That is

$$L_n(x_i) = a_0 + a_i x_i + a_2 x_i^2 + \cdots + a_n x_i^n = f(x_i) \text{ for } i = 0, 1, \cdots, n.$$

Using matrix notation, we could try to obtain the polynomial coefficients by solving,

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_n) \end{bmatrix}.$$

There are $(n+1)$ equations in $(n+1)$ unknowns. To guarantee the existence and uniqueness of a solution, it is necessary to show that the matrix above is non-singular. Although this is not difficult to do, the matrix is ill-conditioned and solving this system is not a good method of constructing $L_n(x)$. It is more convenient to construct the polynomial $L_n(x)$ explicitly and, by construction, establish the existence of a solution to the above set of equations. The uniqueness of the solution is then established by a separate argument.

The case $\mathbf{n = 1}$
It is not difficult to see that

$$L_n(x) = \frac{x - x_1}{x_0 - x_1} f(x_0) + \frac{x - x_0}{x_1 - x_0} f(x_1)$$

is a linear function that satisfies $L_1(x_i) = f(x_i)$ for $i = 0, 1$. Introducing the notation $l_0(x)$ and $l_1(x)$ as linear functions satisfying

$$l_0(x) = \begin{cases} 1 & \text{at} \quad x = x_0 \\ 0 & \text{at} \quad x = x_1 \end{cases} \text{ and } l_1(x) = \begin{cases} 0 & \text{at} \quad x = x_0 \\ 1 & \text{at} \quad x = x_1 \end{cases}$$

then we write

$$L_n(x) = l_0(x) f(x_0) + l_1(x) f(x_1).$$

Hence we have constructed the linear interpolant between two function values. It is easy to visualize that this is the unique linear function equal to $f(x_0)$ at x_0 and $f(x_1)$ at x_1, in that there is only one line connecting two points in a plane.

The case for general n
We write

$$L_n(x) = l_0(x) f(x_0) + l_1(x) f(x_1) + \cdots + l_n(x) f(x_n),$$

where $l_j(x)$ is a polynomial of degree n in x satisfying

$$l_j(x) = \begin{cases} 1 & \text{at} \quad x = x_j \\ 0 & \text{at} \quad x = x_i \end{cases}, \ i = 0, 1, \cdots, n \text{ with } i \neq j.$$

We introduce the notation

$$\Pi(x) = \prod_{i=0}^{n} (x - x_i),$$
$$\Pi_j(x) = \prod_{\substack{i=0 \\ i \neq j}}^{n} (x - x_i).$$

Then we construct $l_i(x)$ as,

$$l_i(x) = \frac{(x - x_0)(x - x_1) \cdots (x - x_{j-1})(x - x_{j+1}) \cdots (x - x_n)}{(x_j - x_0)(x_j - x_1) \cdots (x_j - x_{j-1})(x_j - x_{j+1}) \cdots (x_j - x_n)}$$

$$= \frac{\Pi_j(x)}{\Pi_j(x_j).}$$

Hence, the linear interpolant is

$$L_n(x) = \sum_{j=0}^{n} \frac{f_j \, \Pi_j(x)}{\Pi_j \ (x_j)} = \sum_{j=0}^{n} f_j \, l_j(x) \ \text{ where } f_j = f(x_j).$$

Theorem I1
Let $\{x_0, x_1, \cdots, x_n\}$ be distinct points, there **exists** a **unique** polynomial $L_n(x)$ of degree n or less that satisfies

$$L_n(x_i) = f(x_i) \ \text{ for } i = 0, 1, \cdots, n.$$

Proof
We have already established the existence of the polynomial $L_n(x)$ by construction. It remains to show that the constructed polynomial it is unique. This is done by assuming that it is not unique and establishing a contradiction. We assume that there exists a second polynomial $q_n(x)$ of degree n or less such that:

$$q_n(x) \neq L_n(x)$$
$$q_n(x_j) = f(x_j) \ \text{ for } j = 0, 1, \cdots, n.$$

If this is true, there exists another polynomial $Q_n(x)$ of degree n or less such that:

$$Q_n(x) = L_n(x) - q_n(x) \neq 0$$
$$Q_n(x) = 0 \text{ at } x_j \text{ for } j = 0, 1, \cdots, n.$$

Thus $Q_n(x)$ is a polynomial of degree n (or less) with $(n+1)$ zeros. Hence $Q_n(x) = 0$ and we have established a contradiction. We can now conclude that the Lagrange polynomial is unique.

The following theorem gives us the error in Lagrange interpolation.

Theorem I2   Error term in Lagrange Interpolation

If the $(n+1)^{th}$ derivative of $f(x)$ is continuous in $[a,b]$ and all the points $\{x_i\}_{i=0}^{n}$ lie in $[a,b]$, then for each value of $x \in [a,b]$ there exists an η such that

1) $a < \eta < b$
2) $\varepsilon_n(x) = f(x) - L_n(x) = \dfrac{\Pi(x) f^{(n+1)}(\eta)}{(n+1)!}$.

The proof of this theorem is not part of the material of this course. It is proved by using multiple applications of Rolle's theorem. (See section 6 on Standard Theorems and Results for a statement of Rolle's theorem and see Numerical Analysis by Burden and Faires for the proof of the theorem.)

4.3 Interpolation using divided differences

The Lagrange formula for interpolation is not particularly convenient for numerical work. Other ways of expressing it which are easier to handle also exist. For instance, Newton's form of Lagrange interpolation expresses,

$$L_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \cdots + a_n(x - x_0)(x - x_1)\cdots(x - x_{n-1}).$$

If we then impose the interpolation constraints,

$$L_n(x_i) = f(x_i) \text{ for } i = 0 : n,$$

then we can solve for the a_i. For example

$$L_n(x_0) = f(x_0) = f_0 \Rightarrow a_0 = f_0$$
$$L_n(x_1) = f(x_1) = f_1 \Rightarrow a_0 + a_1(x_1 - x_0) = f_1$$
$$\therefore\ a_1 = \frac{f_1 - f_0}{x_1 - x_0}$$

$$L_n(x_2) = f(x_2) = f_2 \Rightarrow a_0 + a_1(x_2 - x_0) + a_2(x_2 - x_0)(x_2 - x_1) = f_2$$

$$\therefore\ a_2 = \frac{\frac{f_2 - f_0}{x_2 - x_0} - \frac{f_1 - f_0}{x_1 - x_0}}{x_2 - x_1}$$

etc.

The process of determining these coefficients can be mechanized in a way that is suitable for automatic computation by introducing the idea of **divided differences.**

The first divided differences are defined by,

$$f[x_i, x_{i+1}] = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}.$$

The k^{th} divided differences are defined by

$$f[x_i, x_{i+1}, \cdots, x_{i+k}] = \frac{f[x_{i+1}, x_{i+2}, \cdots, x_{i+k}] - f[x_i, x_{i+1}, \cdots, x_{i+k-1}]}{x_{i+k} - x_i}$$

i.e. recursively in terms of the $(k-1)^{th}$ divided differences. In this notation we write

$$f[x_i] \equiv f(x_i) = f_i.$$

For example,

$$\begin{aligned} f[x_2, x_3] &= \frac{f[x_3] - f[x_2]}{x_3 - x_2} \\ f[x_3, x_4, x_5] &= \frac{f[x_4, x_5] - f[x_3, x_4]}{x_5 - x_3} \\ f[x_1, x_2, x_3, x_4] &= \frac{f[x_2, x_3, x_4] - f[x_1, x_2, x_3]}{x_4 - x_1}. \end{aligned}$$

Theorem I3
Suppose that $f(x)$ is defined on $[a,b]$ and suppose that $\{x_0, x_1, \cdots, x_n\}$ are a set of distinct points in $[a,b]$. The k^{th} degree polynomial interpolating $f(x)$ at $\{x_i, x_{i+1}, \cdots, x_{i+k}\}$ is given by

$$\begin{aligned} P_{i,k}(x) = f[x_i] + f[x_i, x_{i+1}](x - x_i) + f[x_i, x_{i+1}, x_{i+2}](x - x_i)(x - x_{i+1}) + \cdots \\ \cdots + f[x_i, \cdots, x_{i+k}](x - x_i)(x - x_{i+1}) \cdots (x - x_{i+k-1}). \end{aligned}$$

Proof: by induction on k
For $k = 0$ and $k = 1$ we have already seen that the theorem is true for any value of i. We will assume it is true for some value of k [and all values of i] and denote

$$P_{i,k+1}(x) = P_{i,k}(x) + a_{k+1}(x - x_i)(x - x_{i+1}) \cdots (x - x_{i+k}).$$

We note that the above is a polynomial of degree $k+1$ which interpolates the function $f(x)$ at the points $\{x_{i+j}, i = 0, 1, \cdots, k\}$. Thus we need to choose a_{k+1}, so that

$$P_{i,k+1}(x_{i+k+1}) = f(x_{i+k+1})$$

and show that

$$a_{k+1} = f[x_i, x_{i+1}, \cdots, x_{i+k+1}].$$

We note that

1) the coefficient of x^k in $P_{i,k}(x)$ is $f[x_i, x_{i+1}, \cdots, x_{i+k}]$
2) the coefficient of x^{k+1} in $P_{i,k+1}(x)$ is a_{k+1}

Consider the polynomial $q(x)$ of degree $(k+1)$ defined by

$$q(x) = \frac{(x - x_i) P_{i+1,k}(x) - (x - x_{i+k+1}) P_{i,k}(x)}{x_{i+k+1} - x_i}.$$

This polynomial satisfies

$$\begin{aligned}
&1)\ q(x_i) = P_{i,k}(x_i) = f_i \\
&2)\ q(x_{i+k+1}) = P_{i+1,k}(x_{i+k+1}) = f_{i+k+1} \\
&3)\ q(x_j) = \frac{(x_j - x_i) P_{i+1,k}(x_j) - (x_j - x_{i+k+1}) P_{i,k}(x_j)}{x_{i+k+1} - x_i} \\
&\qquad\quad = \frac{(x_j - x_i) f_j - (x_j - x_{i+k+1}) f_i}{x_{i+k+1} - x_i} \qquad \text{when } i < j < i+k+1 \\
&\qquad\quad = f_j \\
&\therefore \quad q(x) = P_{i,k+1}(x).
\end{aligned}$$

Moreover, the leading coefficient of $q(x)$, that is the coefficient of x^{k+1}, is

$$\frac{f[x_{i+1}, x_{i+2}, \cdots, x_{i+k+1}] - f[x_i, x_{i+1}, \cdots, x_{i+k}]}{x_{i+k+1} - x_i}.$$

Thus the leading coefficient of $P_{i,k+1}(x)$ is also given by the above formula which, by the definition of divided differences, is also the divided difference $f[x_i, x_{i+1}, \cdots, x_{i+k+1}]$. $\square$

Corollary to theorem I3

$$\begin{aligned}
L_n(x) = f[x_0] + (x - x_0) f[x_0, x_1] + (x - x_0)(x - x_1) f[x_0, x_1, x_2] + \cdots \\
\cdots + (x - x_0)(x - x_1) \cdots (x - x_{n-1}) f[x_0, x_1, \cdots, x_n].
\end{aligned}$$

Although the interpolation points $\{x_i\}$ will usually be give in ascending order, there is no requirement for this to be the case. They could be in descending order or indeed in any order.

Divided differences are obtained by constructing a divided difference table. For example

$$
\begin{array}{llllll}
x_0 & f_0 = f[x_0] & & & & \\
 & & f[x_0,x_1] & & & \\
x_1 & f_1 = f[x_1] & & f[x_0,x_1,x_2] & & \\
 & & f[x_1,x_2] & & f[x_0,x_1,x_2,x_3] & \\
x_2 & f_2 = f[x_2] & & f[x_1,x_2,x_3] & & f[x_0,x_1,x_2,x_3,x_4] \\
 & & f[x_2,x_3] & & f[x_1,x_2,x_3,x_4] & \\
x_3 & f_3 = f[x_3] & & f[x_2,x_3,x_4] & & \\
 & & f[x_3,x_4] & & & \\
x_4 & f_4 = f[x_4] & & & &
\end{array}
$$

We note how easy it is to add a new function value to the table and construct a few new differences. We also note that $L_4(x)$ for the above set of data values is given by

$$
\begin{aligned}
L_4(x) = {} & f[x_0] + f[x_0,x_1](x-x_0) + f[x_0,x_1,x_2](x-x_0)(x-x_1) + \\
& + f[x_0,x_1,x_2,x_3](x-x_0)(x-x_1)(x-x_2) + \\
& + f[x_0,x_1,x_2,x_3,x_4](x-x_0)(x-x_1)(x-x_2)(x-x_3).
\end{aligned}
$$

Example

A divided difference table for the function values

$$\{f(-1) = 1, f(0) = 1, f(3) = 181, f(-2) = -39, f(4) = 801\}$$

is given by

$$
\begin{array}{llllll}
x_0 = -1 & f_0 = 1 & & & & \\
 & & 0 & & & \\
x_1 = 0 & f_1 = 1 & & 15 & & \\
 & & 60 & & 7 & \\
x_2 = 3 & f_2 = 181 & & 8 & & 3 \\
 & & 44 & & 22 & \\
x_3 = -2 & f_3 = -39 & & 96 & & \\
 & & 140 & & & \\
x_4 = 4 & f_4 = 801 & & & &
\end{array}
$$

Thus

$$
\begin{aligned}
f[x_0,x_1] &= 0 \\
f[x_3,x_4] &= 140 \\
f[x_1,x_2,x_3,x_4] &= 22 \quad \text{etc.}
\end{aligned}
$$

The linear Lagrange polynomial interpolating $\{f(-1) = 1, f(0) = 1\}$ is

$$L_1(x) = [1] + [0](x - [-1]) = 1.$$

The cubic polynomial interpolating $\{f(-1)=1, f(0)=1, f(3)=181, f(-2)=-39\}$ is

$$\begin{aligned} L_3(x) &= [1] + [0](x-[-1]) + [15](x-[-1])(x-[0]) + [7](x-[-1])(x-[0])(x-[3]) \\ &= 1 + 15x(x+1) + 7x(x+1)(x-3). \end{aligned}$$

4.4 Interpolation using finite differences

For interpolating functions when the data is at equally spaced points

$$\text{i.e. } x_i = x_0 + ih$$

then the divided differences simplify. For example

$$\begin{aligned} f[x_i, x_{i+1}] &= \frac{f(x_{i+1}) - f(x_i)}{h} \\ f[x_i, x_{i+1}, x_{i+2}] &= \frac{f(x_{i+2}) - 2f(x_{i+1}) + f(x_i)}{2h^2} \\ f[x_i, x_{i+1}, x_{i+2}, x_{i+3}] &= \frac{f(x_{i+3}) - 3f(x_{i+2}) + 3f(x_{i+1}) - f(x_i)}{6h^3} \\ &\text{etc.} \end{aligned}$$

The idea of **forward differences** can be used to exploit this simplification. The forward difference operator Δ is defined by

$$\Delta(f(x)) = f(x+h) - f(x)$$

and higher order differences are defined recursively via

$$\Delta^k(f(x)) = \Delta^{k-1}(f(x+h)) - \Delta^{k-1}(f(x)).$$

For example

$$\begin{aligned} \Delta^2(f(x)) &= \Delta(f(x+h)) - \Delta(f(x)) \\ &= [f(x+2h) - f(x+h)] - [f(x+h) - f(x)] \\ &= f(x+2h) - 2f(x+h) + f(x). \end{aligned}$$

It is not difficult to see that the divided differences satisfy,

$$f[x_i, x_{i+1}, \cdots, x_{i+k}] = \frac{\Delta^k(f(x_i))}{k!h^k} = \frac{\Delta^k f_i}{k!h^k}.$$

Hence, taking the Newton divided difference form of the Lagrange polynomial on equi-spaced data points gives,

$$\begin{aligned} L_n(x) =& f[x_0] + f[x_0, x_1](x-x_0) + f[x_0, x_1, x_2](x-x_0)(x-x_1) + \cdots \\ &+ f[x_0, \cdots, x_n](x-x_0)\cdots(x-x_{n-1}) \\ =& f_0 + \frac{\Delta f_0}{h}(x-x_0) + \frac{\Delta^2 f_0}{2h^2}(x-x_0)(x-x_1) + \cdots \\ &+ \frac{\Delta^n f_0}{n!h^n}(x-x_0)\cdots(x-x_{n-1}). \end{aligned}$$

Now, denoting $x = x_0 + uh$ gives

$$\begin{aligned} L_n(x_0 + uh) =& f_0 + u\Delta f_0 + \frac{u(u-1)}{2!}\Delta^2 f_0 + \frac{u(u-1)(u-2)}{3!}\Delta^3 f_0 + \cdots \\ &+ \frac{u(u-1)\cdots(u-n+1)}{n!}\Delta^n f_0 \\ =& f_0 + \begin{pmatrix} u \\ 1 \end{pmatrix} \Delta f_0 + \begin{pmatrix} u \\ 2 \end{pmatrix} \Delta^2 f_0 + \cdots + \begin{pmatrix} u \\ n \end{pmatrix} \Delta^n f_0 \end{aligned}$$

which is the well-known **Newton's forward formula.** Since this is still Lagrange interpolation we can use the Lagrange expression for the error from Theorem I2:

$$\varepsilon_n(x) = f(x) - L_n(x) = \frac{(x-x_0)(x-x_1)\cdots(x-x_n)}{(n+1)!} f^{(n+1)}(\eta)$$

where

$$\min\{x_0, x_1, \cdots, x_n, x\} < \eta < \max\{x_0, x_1, \cdots, x_n, x\}.$$

Or we can express the error as

$$\varepsilon_n(x_0 + uh) = h^{n+1}\frac{u(u-1)\cdots(u-n)}{(n+1)!} f^{(n+1)}(x_0 + \theta h)$$

where

$$\min\{0, 1, \cdots, n, u\} < \theta < \max\{0, 1, \cdots, n, u\}$$

assuming that f is sufficiently differentiable.

A **forward step operator** E is defined by

$$E(f(x)) = f(x+h)$$

and we note that

1) $E^2(f(x)) = E(E(f(x))) = E(f(x+h)) = f(x+2h)$
2) $E^2 f_i = f_{i+2}$
3) $\Delta = E - I$ where I is the identity operator
4) $\Delta^2 = E^2 - 2E + I$
5) $\Delta^k = \sum_{r=0}^{k} (-1)^r \begin{pmatrix} k \\ r \end{pmatrix} E^{k-r}.$

Further, since $E^u f_0 = f(x_0 + uh)$ we have,

$$\begin{aligned} f(x_0 + uh) &= (I + \Delta)^u f_0 \\ &= \left[I + u\Delta + \begin{pmatrix} u \\ 2 \end{pmatrix} \Delta^2 + \cdots + \begin{pmatrix} u \\ n \end{pmatrix} \Delta^n + \cdots \right] f_0 \end{aligned}$$

and we see that the Newton's forward formula, above, is obtained by truncating this infinite series.

Example

The difference table for $f(x) = e^x$ with $h = 0.2$ is,

x_j	f_j	Δf_j	$\Delta^2 f_j$	$\Delta^3 f_j$	$\Delta^4 f_j$
0.0	1.0000				
		0.2214			
0.2	1.2214		0.0490		
		0.2704		0.0109	
0.4	1.4918		0.0599		0.0023
		0.3303		0.0132	
0.6	1.8221		0.0731		0.0031
		0.4034		0.0163	
0.8	2.2255		0.0894		0.0033
		0.4928		0.0196	
1.0	2.7183		0.1090		0.0047
		0.6018		0.0243	
1.2	3.3201		0.1333		
		0.7351			
1.4	4.0552				

To approximate $f(0.63) = e^{0.63}$ using the above table we choose $x_0 = 0.6$ (so that u is between 0 and 1) and thus

$$\begin{aligned} x &= x_0 + uh \\ \therefore\ 0.63 &= 0.6 + u0.2 \\ \therefore\ u &= 0.15 \end{aligned}$$

and

$$f_0 = 1.8221,\ \Delta f_0 = 0.4034,\ \Delta^2 f_0 = 0.0894,\ \Delta^3 f_0 = 0.0196,\ \Delta^4 f_0 = 0.0047.$$

Thus, truncating the expansion for $f(x_0 + uh)$ based on forward differences for various values of n yields:

$$\begin{array}{lcll} f(0.63) & \approx & 1.8221 & n=0 \\ f(0.63) & \approx & 1.8221 + 0.15 \times [0.4304] = 1.88261 & n=1 \\ f(0.63) & \approx & 1.88261 + 0.15 \times (-0.85) \times [0.0894]/2 = 1.87691 & n=2 \\ f(0.63) & \approx & 1.87691 + 0.15 \times (-0.85) \times (-1.85) \times [0.0196]/6 = 1.87768 & n=3 \end{array}$$

Now, a **backward difference operator** ∇ is defined by

$$\begin{aligned} \nabla(f(x)) &= f(x) - f(x-h) \\ \nabla f_i &= f_i - f_{i-1}. \end{aligned}$$

Thus

$$
\begin{aligned}
\nabla^k f_j &= \nabla^{k-1} f_j - \nabla^{k-1} f_{j-1} \\
\nabla^k f_j &= \Delta^k f_{j-k} \\
\nabla &= I - E^{-1}.
\end{aligned}
$$

As above, we have

$$
\begin{aligned}
E^{-u} f_0 &= f\left(x_0 - uh\right) \\
\therefore f\left(x_0 - uh\right) &= \left(I - \nabla\right)^u f_0 \\
&= \left[I - u\nabla + \begin{pmatrix} u \\ 2 \end{pmatrix} \nabla^2 + \cdots + (-1)^n \begin{pmatrix} u \\ n \end{pmatrix} \nabla^n + \cdots \right] f_0
\end{aligned}
$$

and we obtain a **backward difference formula** for interpolating a function at the points $\{x_0, x_{-1}, \cdots, x_{-n}\}$ by truncating the above series. That is

$$
f\left(x_0 - uh\right) \approx \left[I - u\nabla + \begin{pmatrix} u \\ 2 \end{pmatrix} \nabla^2 + \cdots + (-1)^n \begin{pmatrix} u \\ n \end{pmatrix} \nabla^n \right] f_0.
$$

This is **Newton's backward formula** for interpolation and is equivalent to Lagrange interpolation on the data points $\{x_0, x_{-1}, \cdots, x_{-n}\}$.

For example to approximate $f(1.08) = e^{1.08}$ using interpolation using the above table, we choose $x_0 = 1.2$ and thus

$$
\begin{aligned}
x &= x_0 - uh \\
&\therefore 1.08 = 1.2 - u0.2 \\
&\therefore u = 0.6
\end{aligned}
$$

$$
\begin{array}{lcll}
f\left(1.08\right) & \approx & 3.3201 & n = 0 \\
f\left(1.08\right) & \approx & 3.3201 - 0.6 \times [0.6018] = 2.95902 & n = 1 \\
f\left(1.08\right) & \approx & 2.95902 + 0.6 \times (-0.4) \times [0.1090]/2 = 2.94594 & n = 2
\end{array}
$$

etc.