Interpolation

A standard idea in interpolation now is to find a polynomial $p_n(x)$ of degree n (or less) that assumes the given values; thus

(1) $$p_n(x_0) = f_0, \qquad p_n(x_1) = f_1, \qquad \cdots, \qquad p_n(x_n) = f_n.$$

We call this $p_n(x)$ an interpolation polynomial and x0, · · · ·, xn the nodes. And if f(x) is a mathematical function, we call $p_n(x)$ a polynomial approximation of f.

We use $p_n(x)$ to get (approximate) values of f for x's between x0 and xn ("interpolation") or sometimes outside this interval ("extrapolation").

continued

Lagrange Interpolation

Linear interpolation is interpolation by the straight line through (x_0, f_0), (x_1, f_1); see Fig. 428. Thus the linear Lagrange polynomial p_1 is a sum $p_1 = L_0 f_0 + L_1 f_1$ with L_0 the linear polynomial that is 1 at x_0 and 0 at x_1; similarly, L_1 is 0 at x_0 and 1 at x_1. Obviously,

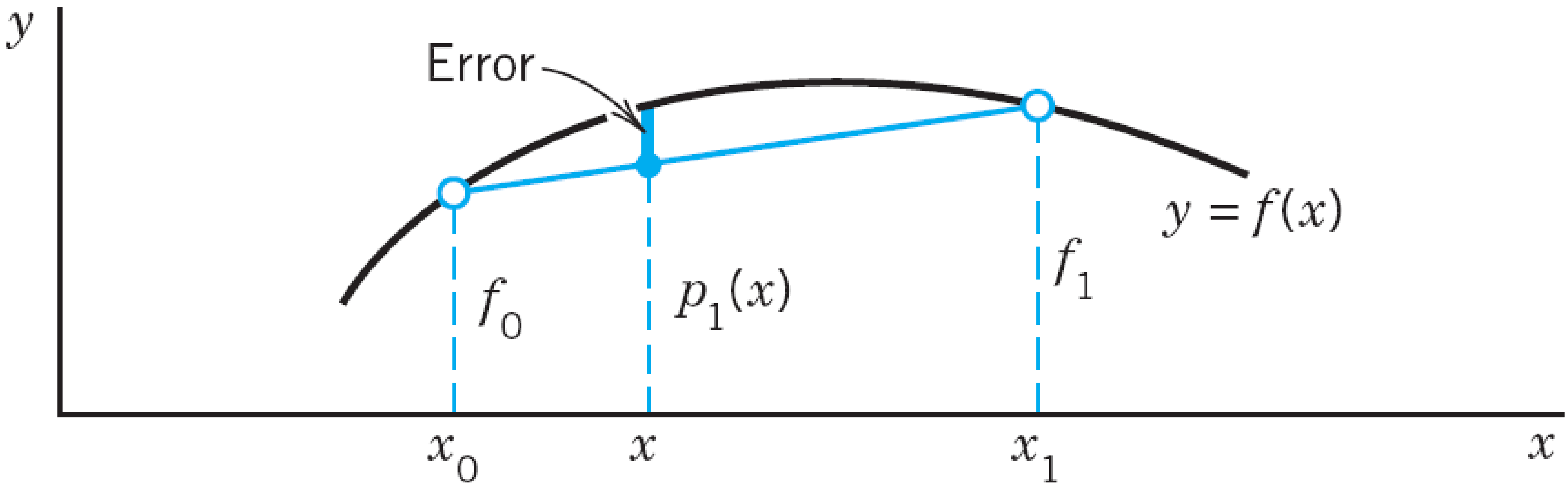
$$L_0(x) = \frac{x - x_1}{x_0 - x_1}, \qquad L_1(x) = \frac{x - x_0}{x_1 - x_0}.$$

This gives the linear Lagrange polynomial

(2) $$p_1(x) = L_0(x) f_0 + L_1(x) f_1 = \frac{x - x_1}{x_0 - x_1} \cdot f_0 + \frac{x - x_0}{x_1 - x_0} \cdot f_1.$$

continued

Fig. 428. Linear interpolation



- Thus $|\varepsilon_n(x)|$ is 0 at the nodes and small near them, because of continuity. The product $(x - x_0)$ .... $(x - x_n)$ is large for x away from the nodes. This makes extrapolation risky. And interpolation at an x will be best if we choose nodes on both sides of that x. Also, we get error bounds by taking the smallest and the largest value of $f^{(n+1)}(t)$ in (5) on the interval $x_0 \leq t \leq x_n$ (or on the interval also containing x if we *extra*polate).
- Most importantly, since p_n is unique, as we have shown, we have

EXAMPLE1 Linear Lagrange Interpolation

Compute a 4D-value of ln 9.2 from ln 9.0 = 2.1972, ln 9.5 = 2.2513 by linear Lagrange interpolation and determine the error, using ln 9.2 = 2.2192 (4D).

Solution. $x_0 = 9.0$, $x_1 = 9.5$, $f_0 = \ln 9.0$, $f_1 = \ln 9.5$. In (2) we need

$$L_0(x) = \frac{x - 9.5}{-0.5} = -2.0(x - 9.5), \qquad L_0(9.2) = -2.0(-0.3) = 0.6$$

$$L_1(x) = \frac{x - 9.0}{0.5} = 2.0(x - 9.0), \qquad L_1(9.2) = 2 \cdot 0.2 = 0.4$$

and obtain the answer

$$\ln 9.2 \approx p_1(9.2) = L_0(9.2)f_0 + L_1(9.2)f_1 = 0.6 \cdot 2.1972 + 0.4 \cdot 2.2513 = 2.2188.$$

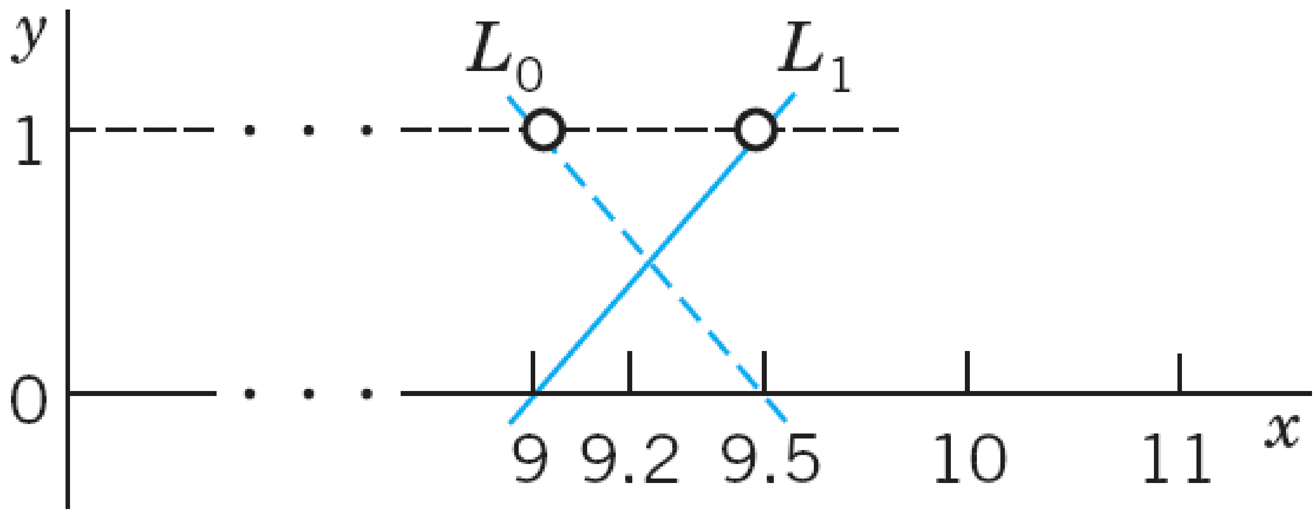
continued

The error is $\varepsilon = a - a = 2.2192 - 2.2188 = 0.0004$.

Hence linear interpolation is not sufficient here to get 4D-accuracy; it would suffice for 3D-accuracy.

continued

Fig. 429. L_0 and L_1 in Example 1



Quadratic interpolation

The interpolation of given (x_0, f_0), (x_1, f_1), (x_2, f_2) by a second-degree polynomial $p_2(x)$, which by Lagrange's idea is

(3a) $$p_2(x) = L_0(x)f_0 + L_1(x)f_1 + L_2(x)f_2$$

with $L_0(x_0) = 1$, $L_1(x_1) = 1$, $L_2(x_2) = 1$, and $L_0(x_1) = L_0(x_2) = 0$, etc. We claim that

(3b)
$$L_0(x) = \frac{l_0(x)}{l_0(x_0)} = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)}$$

$$L_1(x) = \frac{l_1(x)}{l_1(x_1)} = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)}$$

$$L_2(x) = \frac{l_2(x)}{l_2(x_2)} = \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}.$$

continued

EXAMPLE 2 Quadratic Lagrange Interpolation

Compute ln 9.2 by (3) from the data in Example 1 and the additional third value ln 11.0 = 2.3979.

Solution. In (3),

$$L_0(x) = \frac{(x-9.5)(x-11.0)}{(9.0-9.5)(9.0-11.0)} = x^2 - 20.5x + 104.5, \qquad L_0(9.2) = 0.5400,$$

$$L_1(x) = \frac{(x-9.0)(x-11.0)}{(9.5-9.0)(9.5-11.0)} = -\frac{1}{0.75}(x^2 - 20x + 99), \qquad L_1(9.2) = 0.4800,$$

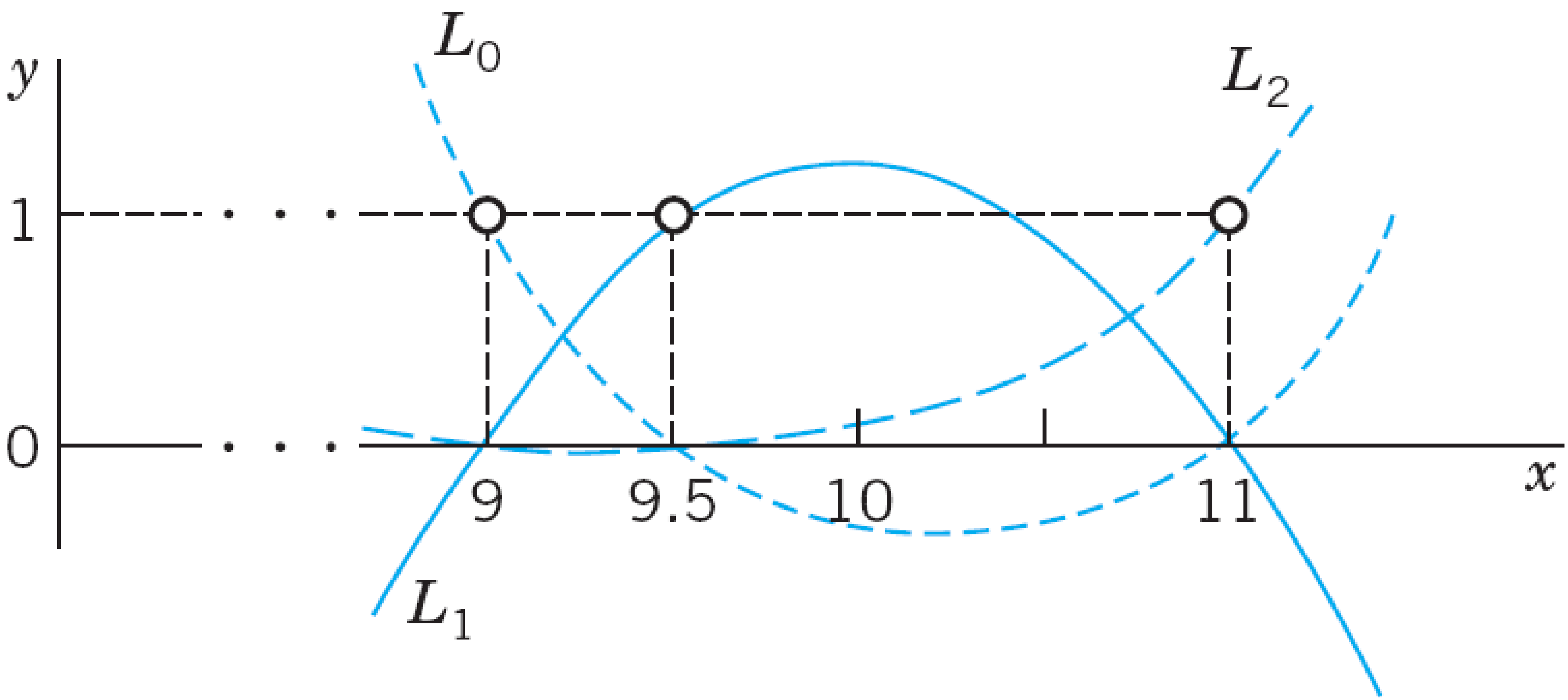
$$L_2(x) = \frac{(x-9.0)(x-9.5)}{(11.0-9.0)(11.0-9.5)} = \frac{1}{3}(x^2 - 18.5x + 85.5), \qquad L_2(9.2) = -0.0200,$$

so that (3a) gives, exact to 4D,

$$\ln 9.2 \approx p_2(9.2) = 0.5400 \cdot 2.1972 + 0.4800 \cdot 2.2513 - 0.0200 \cdot 2.3979 = 2.2192.$$

continued

Fig. 430. L_0, L_1, L_2 in Example 2



General Lagrange Interpolation Polynomial

For general n we obtain

(4a)
$$f(x) \approx p_n(x) = \sum_{k=0}^{n} L_k(x) f_k = \sum_{k=0}^{n} \frac{l_k(x)}{l_k(x_k)} f_k$$

where $L_k(x_k) = 1$ and L_k is 0 at the other nodes, and the L_k are independent of the function f to be interpolated. We get (4a) if we take

(4b)
$$\begin{aligned} l_0(x) &= (x - x_1)(x - x_2) \cdots (x - x_n), \\ l_k(x) &= (x - x_0) \cdots (x - x_{k-1})(x - x_{k+1}) \cdots (x - x_n), \qquad 0 < k < n, \\ l_n(x) &= (x - x_0)(x - x_1) \cdots (x - x_{n-1}). \end{aligned}$$

We can easily see that $p_n(x_k) = f_k$.

continued

Error Estimate

If f is itself a polynomial of degree n (or less), it must coincide with p_n because the $n + 1$ data $(x_0, f_0), \cdots, (x_n, f_n)$ determine a polynomial uniquely, so the error is zero.

Now the special f has its $(n + 1)$st derivative identically zero. This makes it plausible that for a *general* f its $(n + 1)$st derivative $f^{(n+1)}$ should measure the error $\epsilon_n(x) = f(x) - p_n(x)$.

If $f^{(n+1)}$ exists and is continuous, then with a suitable t between x_0 and x_n

(5)

$$\epsilon_n(x) = f(x) - p_n(x) = (x - x_0)(x - x_1) \cdots (x - x_n) \frac{f^{(n+1)}(t)}{(n + 1)!} .$$

continued

THEOREM 1

Error of Interpolation

Formula (5) gives the error for any polynomial interpolation method if $f(x)$ has a continuous $(n + 1)$st derivative.

EXAMPLE 3 Error Estimate (5) of Linear Interpolation. Damage by Roundoff. Error Principle

- Estimate the error in Example 1 first by (5) directly and then by the Error Principle (Sec. 19.1).
- **Solution.** **(A)** ***Estimation by*** **(5).** We have $n = 1$, $f(t) = \ln t$, $f'(t) = 1/t$, $f''(t) = -1/t^2$. Hence

$$\epsilon_1(x) = (x - 9.0)(x - 9.5)\,\frac{(-1)}{2t^2}, \qquad \text{thus} \qquad \epsilon_1(9.2) = \frac{0.03}{t^2}.$$

$t = 9.0$ gives the maximum $0.03/9^2 = 0.00037$ and $t = 9.5$ gives the minimum $0.03/9.5^2 = 0.00033$, so that we get $0.00033 \le \varepsilon_1(9.2) \le 0.00037$, or better, 0.00038 because $0.3/81 = 0.003\,703\ldots$.

- But the error 0.0004 in Example 1 disagrees, and we can learn something! Repetition of the computation there with 5D instead of 4D gives

$$\ln 9.2 \approx p_1(9.2) = 0.6 \cdot 2.19722 + 0.4 \cdot 2.25129 = 2.21885$$

with an actual error $\varepsilon = 2.21920 - 2.21885 = 0.00035$, which lies nicely near the middle between our two error bounds.

- This shows that the discrepancy (0.0004 vs. 0.00035) was caused by rounding, which is not taken into account in (5).

continued

- **(B) *Estimation by the Error Principle.*** We calculate $p_1(9.2) = 2.21885$ as before and then $p_2(9.2)$ as in Example 2 but with 5D, obtaining

$$p_2(9.2) = 0.54 \cdot 2.19722 + 0.48 \cdot 2.25129 - 0.02 \cdot 2.39790 = 2.21916.$$

- The difference $p_2(9.2) - p_1(9.2) = 0.00031$ is the approximate error of $p_1(9.2)$ that we wanted to obtain; this is an approximation of the actual error 0.00035 given above.

Newton's Divided Difference Interpolation

The **kth divided difference,** recursively denoted and defined as follows:

$$a_1 = f[x_0, x_1] = \frac{f_1 - f_0}{x_1 - x_0}$$

$$a_2 = f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0}$$

and in general

(8)
$$a_k = f[x_0, \cdots, x_k] = \frac{f[x_1, \cdots, x_k] - f[x_0, \cdots, x_{k-1}]}{x_k - x_0}.$$

$$f(x) = f(x_0) + (x - x_0)\, f[x_0, x]$$

$$f[x_0, x] = f[x_1, x_0] + (x - x_1)\, f[x_1, x_0, x]$$

$$f[x_1, x_0, x] = f[x_2, x_1, x_0] + (x - x_2)\, f[x_2, x_1, x_0, x]$$

$$\cdots$$

$$\begin{aligned} f[x_{n-1}, x_{n-2}, \cdots, x_0, x] &= f[x_n, x_{n-1}, \cdots, x_1, x_0] \\ &\quad + (x - x_n)\, f[x_n, x_{n-1}, \cdots, x_1, x_0, x] \end{aligned}$$

With $p_0(x) = f_0$ by repeated application with $k = 1, \cdots, n$ this finally gives **Newton's divided difference interpolation formula**

(10)

$$f(x) \approx f_0 + (x - x_0)f[x_0, x_1] + (x - x_0)(x - x_1)f[x_0, x_1, x_2]$$
$$+ \cdots + (x - x_0)(x - x_1) \cdots (x - x_{n-1})f[x_0, \cdots, x_n].$$

Table 19.2 Newton's Divided Difference Interpolation

ALGORITHM INTERPOL $(x_0, \cdots, x_n; f_0, \cdots, f_n; \hat{x})$

This algorithm computes an approximation $p_n(\hat{x})$ of $f(\hat{x})$ at $\hat{x}$.

INPUT: Data $(x_0, f_0), (x_1, f_1), \cdots, (x_n, f_n); \hat{x}$

OUTPUT: Approximation $p_n(\hat{x})$ of $f(\hat{x})$

Set $f[x_j] = f_j \quad (j = 0, \cdots, n)$.

continued

For $m = 1, \cdots, n-1$ do:

For $j = 0, \cdots, n-m$ do:

$$f[x_j, \cdots, x_{j+m}] = \frac{f[x_{j+1}, \cdots, x_{j+m}] - f[x_j, \cdots, x_{j+m-1}]}{x_{j+m} - x_j}$$

End

End

Set $p_0(x) = f_0$.

For $k = 1, \cdots, n$ do:

$$p_k(\hat{x}) = p_{k-1}(\hat{x}) + (\hat{x} - x_0) \cdots (\hat{x} - x_{k-1}) f[x_0, \cdots, x_k]$$

End

OUTPUT $p_n(\hat{x})$

End INTERPOL

EXAMPLE 4 Newton's Divided Difference Interpolation Formula

Compute $f(9.2)$ from the values shown in the first two columns of the following table.

x_j	$f_j = f(x_j)$	$f[x_j, x_{j+1}]$	$f[x_j, x_{j+1}, x_{j+2}]$	$f[x_j, \cdots, x_{j+3}]$
8.0	2.079 442			
		0.117 783		
9.0	2.197 225		−0.006 433	
		0.108 134		0.000 411
9.5	2.251 292		−0.005 200	
		0.097 735		
11.0	2.397 895			

continued

Solution

We compute the divided differences as shown. Sample computation:

$$(0.097\,735 - 0.108\,134)/(11 - 9) = -0.005\,200.$$

The values we need in (10) are circled. We have

$$f(x) \approx p_3(x) = 2.079\,442 + 0.117\,783(x - 8.0) - 0.006\,433(x - 8.0)(x - 9.0)$$
$$+ 0.000\,411(x - 8.0)(x - 9.0)(x - 9.5).$$

$$f(9.2) \approx 2.079\,442 + 0.141\,340 - 0.001\,544 - 0.000\,030 = 2.219\,208.$$

The value exact to 6D is $f(9.2) = \ln 9.2 = 2.219\ 203$. Note that we can nicely see how the accuracy increases from term to term:

$$p_1(9.2) = 2.220\,782, \qquad p_2(9.2) = 2.219\,238, \qquad p_3(9.2) = 2.219\,208.$$

continued

Equal Spacing: Newton's Forward Difference Formula

Newton's formula (10) is valid for *arbitrarily spaced* nodes as they may occur in practice in experiments or observations.

However, in many applications the x_j's are *regularly spaced*—for instance, in measurements taken at regular intervals of time. Then, denoting the distance by h, we can write

(11)

$$x_0, \quad x_1 = x_0 + h, \quad x_2 = x_0 + 2h, \quad \cdots, \quad x_n = x_0 + nh.$$

We can show how (8) and (10) now simplify considerably!

continued

Let us define the *first forward difference* of f at x_j by

$$\Delta f_j = f_{j+1} - f_j,$$

and the *second forward difference* of f at x_j by

$$\Delta^2 f_j = \Delta f_{j+1} - \Delta f_j,$$

and, continuing in this way, the ***k*th forward difference** of f at x_j by

(12) $$\boxed{\Delta^k f_j = \Delta^{k-1} f_{j+1} - \Delta^{k-1} f_j} \qquad (k = 1, 2, \cdots).$$

Formula (10) becomes **Newton's** (or *Gregory–Newton's*) **forward difference interpolation formula**

(14)

$$f(x) \approx p_n(x) = \sum_{s=0}^{n} \binom{r}{s} \Delta^s f_0 \qquad (x = x_0 + rh, \quad r = (x - x_0)/h)$$

$$= f_0 + r\Delta f_0 + \frac{r(r-1)}{2!} \Delta^2 f_0 + \cdots + \frac{r(r-1)\cdots(r-n+1)}{n!} \Delta^n f_0$$

where the **binomial coefficients** in the first line are defined by

(15)

$$\binom{r}{0} = 1, \quad \binom{r}{s} = \frac{r(r-1)(r-2)\cdots(r-s+1)}{s!} \qquad (s > 0, \text{ integer})$$

and $s! = 1 \cdot 2 \cdot \cdots \cdot s$.

Error

From (5) we get, with $x - x_0 = rh$, $x - x_1 = (r - 1)h$, etc.,

(16)
$$\epsilon_n(x) = f(x) - p_n(x) = \frac{h^{n+1}}{(n+1)!} r(r-1)\cdots(r-n)f^{(n+1)}(t)$$

with t as characterized in (5).

EXAMPLE5 Newton's Forward Difference Formula. Error Estimation

Compute cosh 0.56 from (14) and the four values in the following table and estimate the error.

j	x_j	$f_j = \cosh x_j$	Δf_j	$\Delta^2 f_j$	$\Delta^3 f_j$
0	0.5	1.127 626			
			0.057 839		
1	0.6	1.185 465		0.011 865	
			0.069 704		0.000 697
2	0.7	1.255 169		0.012 562	
			0.082 266		
3	0.8	1.337 435			

continued

Solution

We compute the forward differences as shown in the table. The values we need are circled. In (14) we have r = (0.56 – 0.50)/0.1 = 0.6, so that (14) gives

$$\cosh 0.56 \approx 1.127\,626 + 0.6 \cdot 0.057\,839 + \frac{0.6(-0.4)}{2} \cdot 0.011\,865 + \frac{0.6(-0.4)(-1.4)}{6} \cdot 0.000\,697$$

$$= 1.127\,626 + 0.034\,703 - 0.001\,424 + 0.000\,039$$

$$= 1.160\,944.$$

continued

Error estimate. From (16), since the fourth derivative is $\cosh^{(4)} t = \cosh t$,

$$\epsilon_3(0.56) = \frac{0.1^4}{4!} \cdot 0.6(-0.4)(-1.4)(-2.4) \cosh t$$
$$= A \cosh t,$$

where $A = -0.000\,003\,36$ and $0.5 \le t \le 0.8$. We do not know t, but we get an inequality by taking the largest and smallest $\cosh t$ in that interval:

$$A \cosh 0.8 \leqq \epsilon_3(0.62) \leqq A \cosh 0.5.$$

continued

Since

$$f(x) = p_3(x) + \epsilon_3(x),$$

this gives

$$p_3(0.56) + A \cosh 0.8 \leqq \cosh 0.56 \leqq p_3(0.56) + A \cosh 0.5.$$

Numeric values are

$$1.160\,939 \leqq \cosh 0.56 \leqq 1.160\,941.$$

The exact 6D-value is cosh 0.56 = 1.160 941. It lies within these bounds. Such bounds are not always so tight. Also, we did not consider roundoff errors, which will depend on the number of operations.

Equal Spacing: Newton's Backward Difference Formula

A formula similar to (14) but involving backward differences is **Newton's** (or *Gregory–Newton's*) **backward difference interpolation formula**

(18)
$$
\begin{aligned}
f(x) \approx p_n(x) &= \sum_{s=0}^{n} \binom{r+s-1}{s} \nabla^s f_0 \qquad (x = x_0 + rh,\ r = (x - x_0)/h) \\
&= f_0 + r\nabla f_0 + \frac{r(r+1)}{2!} \nabla^2 f_0 + \cdots + \frac{r(r+1)\cdots(r+n-1)}{n!} \nabla^n f_0.
\end{aligned}
$$

EXAMPLE 6 Newton's Forward and Backward Interpolations

Compute a 7D-value of the Bessel function $J_0(x)$ for $x = 1.72$ from the four values in the following table, using (a) Newton's forward formula (14), (b) Newton's backward formula (18).

j_{for}	j_{back}	x_j	$J_0(x_j)$	1st Diff.	2nd Diff.	3rd Diff.
0	−3	1.7	0.397 9849			
				−0.057 9985		
1	−2	1.8	0.339 9864		−0.000 1693	
				−0.058 1678		0.000 4093
2	−1	1.9	0.281 8186		0.000 2400	
				−0.057 9278		
3	0	2.0	0.223 8908			

continued

Solution. The computation of the differences is the same in both cases. Only their notation differs.

(a) Forward. In (14) we have r = (1.72 – 1.70)/0.1 = 0.2, and j goes from 0 to 3 (see first column). In each column we need the first given number, and (14) thus gives

$$J_0(1.72) \approx 0.397\,9849 + 0.2(-0.057\,9985) + \frac{0.2(-0.8)}{2}(-0.000\,1693) + \frac{0.2(-0.8)(-1.8)}{6} \cdot 0.000\,4093$$

$$= 0.397\,9849 - 0.011\,5997 + 0.000\,0135 + 0.000\,0196 = 0.386\,4183,$$

which is exact to 6D, the exact 7D-value being 0.386 4185.

(b) Backward. For (18) we use j shown in the second column, and in each column the last number. Since r = (1.72 – 2.00)/0.1 = –2.8, we thus get from (18)

$$J_0(1.72) \approx 0.223\,8908 - 2.8(-0.057\,9278) + \frac{-2.8(-1.8)}{2} \cdot 0.000\,2400 + \frac{-2.8(-1.8)(-0.8)}{6} \cdot 0.000\,4093$$

$$= 0.223\,8908 + 0.162\,1978 + 0.000\,6048 - 0.000\,2750$$

$$= 0.386\,4184.$$

Central Difference Notation

This is a third notation for differences. The first central difference of $f(x)$ at x_j is defined by

$$\delta f_j = f_{j+1/2} - f_{j-1/2}$$

and the ***k*th central difference** of $f(x)$ at x_j by

(19) $$\delta^k f_j = \delta^{k-1} f_{j+1/2} - \delta^{k-1} f_{j-1/2} \qquad (j = 2, 3, \cdots).$$