RESUMÉ [WITH PROOFS] OF POLYNOMIAL INTERPOLATION

1. Interpolation Problems: Solutions, Existence and Uniqueness.

Let \( \{x_0, \ldots, x_n\} \) be \( n+1 \) distinct points in a real interval \( J \). (There would be no essential difference in the results of this § if the real interval \( J \) were replaced by an open set \( U \) in the complex plane. Only when we consider the form of error terms in the next § will we have to distinguish between the real and complex cases.) Let \( f_0, \ldots, f_n \) be \( n+1 \) real numbers, and consider the problem of finding a polynomial

\[
p(x) = \sum_{j=0}^{n} a_j x^j
\]

of degree \( \leq n \) that satisfies

\[
p(x_j) = f_j \quad \text{for} \quad j = 0, \ldots, n.
\]

{Although we don’t need to know this to state the problem, we should imagine that there is a function \( f(x) \) defined on \( J \) whose values we know at \( x_0, \ldots, x_n \) to be \( f(x_j) = f_j \); our hope is that the interpolating polynomial \( p(x) \) will be a good approximation to \( f(x) \) at other points of the interval \( J \). In the next section we shall see how good the approximation might be.) More generally, since it’s no harder algebraically, suppose that for each \( x_j \) we are given \( m_j \) numbers \( (m_j \geq 1) \)

\[
f(x_j), f'(x_j), \ldots, f^{(m_j-1)}(x_j)
\]

and we seek a polynomial \( p(x) \) of degree \( m = (\sum_j m_j) - 1 \) satisfying \((0 \leq j \leq n)\)

\[
p(x_j) = f(x_j), \quad p'(x_j) = f'(x_j), \ldots, \quad p^{(m_j-1)}(x_j) = f^{(m_j-1)}(x_j).
\]

In words, at each \( x_j \) we want to prescribe the values of \( p(x) \) and its derivatives up to order \( m_j - 1 \). Question: Can such a polynomial be found, and do the conditions which the polynomial is required to satisfy determine it uniquely?

The answer to both sections of the question is affirmative for the same linear-algebraic reason. We need to determine a vector of \( m+1 \) unknown coefficients \( [a_0, \ldots, a_m]^T \) for which the \( m+1 \) linear equations below are satisfied:

\[
\begin{align*}
a_0 + a_1 x_0 + \cdots + a_m x_0^m &= f(x_0) \\
a_1 + \cdots + m a_m x_0^{m-1} &= f'(x_0) \\
&\vdots \\
a_0 + a_1 x_1 + \cdots + a_m x_1^m &= f(x_1) \\
a_1 + \cdots + m a_m x_1^{m-1} &= f'(x_1) \\
&\vdots \\
&\vdots \\
+a_0 + a_1 x_n + \cdots + a_m x_n^m &= f(x_n) \\
a_1 + \cdots + m a_m x_n^{m-1} = f^{(m_n-1)}(x_n)
\end{align*}
\]

The coefficients of the unknown vector \( [a_0, \ldots, a_m]^T \) do not depend directly on anything on the r. h. s. It is a well-known theorem of linear algebra that a given \((m+1) \times (m+1)\) matrix determining \( m+1 \) equations in \( m+1 \) unknowns must obey the following alternative: either the equations have a unique solution for each choice of the r. h. s., or else there is a nonzero solution \([a_0, \ldots, a_m]^T\) of the equations when all the right-hand sides are taken to be zero.

Thus to establish the existence and uniqueness we want, we need only show that no such “nonzero solution of the homogeneous equations” can exist. Well, suppose one did, call it \([a_0, \ldots, a_m]^T\) for lack of imagination. That would mean that at each of the points \( x_j, 0 \leq j \leq n \), the polynomial of degree \( \leq m \)

\[
p(x) = \sum_{k=0}^{m} a_k x^k
\]

would also satisfy

\[
\begin{align*}
a_0 + a_1 x_j + \cdots + a_m x_j^m &= f(x_j) \\
a_1 + \cdots + m a_m x_j^{m-1} &= f'(x_j) \\
&\vdots \\
a_0 + a_1 x_j + \cdots + a_m x_j^m = f(x_j) \\
a_1 + \cdots + m a_m x_j^{m-1} = f'(x_j) \\
&\vdots \\
&\vdots \\
+a_0 + a_1 x_j + \cdots + a_m x_j^m = f(x_j) \\
a_1 + \cdots + m a_m x_j^{m-1} = f^{(m-j-1)}(x_j)
\end{align*}
\]

for lack of
had the property that
\[ p(x_j) = 0, \ p'(x_j) = 0, \ldots, \ p^{(m_j-1)}(x_j) = 0. \]
The Taylor form of \( p(x) \) with center \( x_j \) would then have to look like
\[
p(x) = 0 + 0 \cdot (x - x_j) + \cdots + 0 \cdot (x - x_j)^{m_j-1} + \frac{p^{(m_j)}(x_j)}{m_j!} (x - x_j)^{m_j} + \cdots
\]
\[= (x - x_j)^{m_j} \cdot \text{[some polynomial]}\]
(remember, when you write the Taylor series of a polynomial it terminates at a finite index [the degree of the polynomial] and comes out exact). It follows that each of the “prime power polynomials”
\[
(x - x_0)^{m_0}, (x - x_1)^{m_1}, \ldots, (x - x_n)^{m_n}
\]
divides \( p(x) \) “evenly” without remainder. Because these are powers of distinct primes, their product
\[
(x - x_0)^{m_0} \cdot (x - x_1)^{m_1} \cdots (x - x_n)^{m_n}
\]
also divides \( p(x) \) evenly: one could write
\[
p(x) = (x - x_0)^{m_0} \cdot (x - x_1)^{m_1} \cdots (x - x_n)^{m_n} \cdot q(x)
\]
(where \( q(x) \) is the quotient after division). But that is absurd: the polynomial \( p(x) \) on the l. h. s. has degree \( m \) while the polynomial on the r. h. s. has degree at least equal to \( \sum m_j = m + 1 \). So the \((m + 1) \times (m + 1)\) system of linear equations that we have to solve has no nonzero homogeneous solution, and thus a solution to the problem of interpolating values and derivatives that we gave above always exists and is always unique.

It cannot be emphasized too strongly that this theoretical proof of existence and uniqueness should not be implemented as an algorithm! The interested reader who doesn’t believe this should try even to solve the \(4 \times 4\) system corresponding to the “cubic Hermite interpolation problem” of finding a polynomial of degree \( \leq 3 \) satisfying
\[
p(x_0) = f(x_0) \quad p(x_1) = f(x_1)
\]
\[
p'(x_0) = f'(x_0) \quad p'(x_1) = f'(x_1)
\]
for \( x_0 = 1000 \) and \( x_1 = 1000.5 \), say. It is wrongheaded (to begin with!) to express the polynomial in powers of \( x \)—powers of \( x - 1000 \) would be much more appropriate—and look at all the big numbers that are being divided by small numbers! Practical, fairly recursive and fairly numerically stable methods for interpolation will be discussed below. Cf. Atkinson, p. 132, Proof (ii) of Theorem 3.1.

It is crucial that all the derivatives from the 0-th (= value) to the \( (m_j - 1) \)-st were interpolated at each \( x_j \). Problem set #1, Problem 3 shows that the problem of interpolating \( p(x_0), p'(x_1) \) and \( p(x_2) \) for \( x_0 < x_1 < x_2 \)—with no condition on \( p(x_1) \)—behaves somewhat differently.

2. Error Expressions.

Let us think of ourselves as remaining in the context of the previous § and think about the following subject: if the function \( f(x) \) that \( p(x) \) interpolates is an arbitrary continuous function, then of course the fact that agrees with \( p(x) \) at \( x_n, \ldots, x_n \) tells us little about its behavior at other points of the interval \( J \). However, it can be hoped that if \( f(x) \) is very smooth in \( J \) then the size of its derivatives may control the extent to which \( f(x) \) and \( p(x) \) deviate from each other between points where they are known to agree.

Let us agree to give the name error factor, or error product, to the product
\[
\Psi(x) = (x - x_0)^{m_0} \cdot (x - x_1)^{m_1} \cdots (x - x_n)^{m_n}
\]
(for a given set of nodes—possibly with repetitions), since it will occur quite frequently in our considerations.
For real-valued functions the basic error estimate is given by

**Theorem:** Let \( p(x) \) be the solution of the interpolation problem of the preceding §, and suppose the interpolated function \( f(x) \) has derivatives of order up to and including \( \sum_j m_j \). Then for each point \( x \in J \) there exists (at least one) \( \xi_x \in J \)—depending on \( x \)—for which

\[
\begin{align*}
f(x) &= p(x) + \frac{f^{(\sum m_j)}(\xi_x)}{(\sum m_j)!} \cdot (x - x_0)^{m_0} \cdot (x - x_1)^{m_1} \cdots (x - x_n)^{m_n} \\
&= p(x) + \frac{f^{(\sum m_j)}(\xi_x)}{(\sum m_j)!} \cdot \Psi(x).
\end{align*}
\]

{The simplest case of this theorem is the one in which all the \( m_j \) = 1; it is Atkinson’s Thm. 3.2, pp. 134–135.}

**Proof.** Let \( x \in J \) be given: it will be held fixed throughout the following argument. Since \( p(x) \) takes the same value as \( f(x) \) and “the error is zero” whenever \( x \) equals some \( x_j \)—so our error term is correct for trivial reasons—we can assume with no loss of generality that \( x \) is not one of the \( x_j \)’s. Consequently the value \( \Psi(x) = (x - x_0)^{m_0} \cdot (x - x_1)^{m_1} \cdots (x - x_n)^{m_n} \) does not \( \neq 0 \), and so it is trivial to find a constant \( K \) that is a solution of the equation

\[
f(x) = p(x) + K \cdot (x - x_0)^{m_0} \cdot (x - x_1)^{m_1} \cdots (x - x_n)^{m_n}.
\]

{Remember, \( x \) is a fixed number, not a dummy variable! Yes, one can explicitly compute \( K \), but don’t bother—it’s the fact that it exists that will be important.} Now using that constant \( K \), consider the function of the “new variable” \( t \) given by

\[
G(t) = f(t) - p(t) - K \cdot \Psi(t) = f(t) - p(t) - K \cdot (t - x_0)^{m_0} \cdot (t - x_1)^{m_1} \cdots (t - x_n)^{m_n}.
\]

The interpolation properties of \( p(t) \) relative to \( f(t) \) and its derivatives tell us that

\[
G(x_j) = 0, \ G'(x_j) = 0, \ldots, \ G^{(m_j - 1)}(x_j) = 0
\]

since \( \Psi(t) \) has a zero of order \( m_j \) at each \( x_j \). Moreover, \( K \) has been chosen to make \( G(x) = 0 \) (remember, this means \( G(t) = 0 \) when you plug in \( t = x \)). Let \( I \) be the smallest closed interval containing \( x_0, \ldots, x_n \) and also \( x \) (\( x \) may lie to the right of the largest \( x_j \) or to the left of the smallest, if we wish). Then \( G(t) \) has a zero at each endpoint of \( I \).

For a natural number \( k \), let us say that a (sufficiently differentiable) function \( H(t) \) has at least \( k \) zeros counted according to multiplicity in the interval \( I \) if there are distinct points \( t_1, \ldots, t_r \ (r \geq 1) \) in \( I \) and natural numbers \( 1 \leq n_1, \ldots, n_r \) such that

\[
H(t_i) = 0, \ldots, \ H^{(n_i - 1)}(t_i) = 0
\]

for each \( i = 1, \ldots, r \), with \( \sum_i n_i \geq k \). Then if \( H(t) \) has at least \( k \) zeros in \( I \) counted according to multiplicity, its derivative \( H'(t) \) must have at least \( k - 1 \) of them. The reason is that \( H'(t) \) has at least \( r - 1 \) zeros in \( I \) that are not the same as the \( t_i \)’s—namely, the zeros whose existence is guaranteed by Rolle’s theorem. Moreover, \( H'(t) \) has at least \( n_i - 1 \) zeros at \( t_i \) whenever \( n_i > 1 \)—in the obvious sense that its derivatives are zero at \( t_i \) out to order at least \( n_i - 2 \).(1) Thus \( H'(t) \) has at least

\[
\sum_{i=1}^{r} (n_i - 1) + (r - 1) \geq \sum_{i=1}^{r} n_i - 1 \geq k - 1
\]

zeros in \( I \) counted according to multiplicity.

---

(1) Look at the Taylor series with base point \( t_i \).
Returning to our function $G(t)$, we observe that it has at least $\sum_j m_j$ zeros in $I$ counted according to multiplicity. It follows by applying the argument of the preceding paragraph $\sum_j m_j$ times that $G'(t)$ has at least $\sum_j m_j$ zeros, . . . , and $G'(\sum_j m_j)(t)$ has at least one zero $\xi_x \in I$. It is much easier to compute the $(\sum_j m_j)$-th derivative of $G(t)$ than it looks, because

$$\Psi(x) = (x - x_0)^{m_0} \cdot (x - x_1)^{m_1} \cdots (x - x_n)^{m_n}$$

$$= x^{(\sum_j m_j)} + \text{lower order terms}$$

and the $(\sum_j m_j)$-th derivative of that is obviously just

$$(\sum_j m_j)! + 0 + \cdots + 0 = (\sum_j m_j)!.$$ Moreover, since $\deg p \leq (\sum_j m_j) - 1$, its $(\sum_j m_j)$-th derivative is $\equiv 0$. So

$$0 = G'(\sum_j m_j)(\xi_x) = f'(\sum_j m_j)(\xi_x) - K \cdot (\sum_j m_j)!.$$ Solve this equation for $K = \frac{f'(\sum_j m_j)(\xi_x)}{(\sum_j m_j)!}$, and the original equation

$$f(x) = p(x) + K \cdot (x - x_0)^{m_0} \cdot (x - x_1)^{m_1} \cdots (x - x_n)^{m_n}$$

that defined $K$ can be rewritten—by plugging in this value of $K$—as

$$f(x) = p(x) + \frac{f'(\sum_j m_j)(\xi_x)}{(\sum_j m_j)!} \cdot (x - x_0)^{m_0} \cdot (x - x_1)^{m_1} \cdots (x - x_n)^{m_n}$$

which is what the theorem advertised.

The following material is not part of the “official” content of the course and may be omitted by the nonmathematical, but it will be useful to anyone who wishes to read the paper on the Runge example. For complex-valued analytic functions $f(z)$ defined on an open set $U$ in the complex plane containing the points $z_0, \ldots, z_n$, it would seem natural to get an expression for the error $f(z) - p(z)$ in the form of a Cauchy integral. Here, briefly, is how to do it in the case where only the values—not the derivatives—of $f(z_0), \ldots, f(z_n)$ are being interpolated. {We shall indicate below why it suffices to handle only this case for analytic $f(z).}$ Let $\Gamma$ be a contour winding around the points $z_0, \ldots, z_n$ and $z$ as in the Cauchy integral theorem; then since

$$f(z) = \frac{1}{2\pi i} \int_{\Gamma} f(\zeta) \cdot \frac{1}{\zeta - z} d\zeta$$

for $z = z_0, \ldots, z_n, x$, we can use the Lagrange interpolating polynomials

$$\ell_j(z) = \frac{\Psi(z)}{\Psi'(z_j) \cdot (z - z_j)}$$

(cf. Atkinson, p. 139, formula (3.2.4)) and the Cauchy integral theorem together to write

$$p(z) = \sum_{j=0}^n f(z_j) \cdot \ell_j(z)$$

$$= \sum_{j=0}^n \frac{1}{2\pi i} \int_{\Gamma} f(\zeta) \cdot \frac{1}{\zeta - z} \frac{\Psi(z)}{\Psi'(z_j) \cdot (z - z_j)} d\zeta$$

and therefore finally

$$f(z) - p(z) = \frac{1}{2\pi i} \int_{\Gamma} f(\zeta) \cdot \left\{ \frac{1}{\zeta - z} - \sum_{j=0}^n \frac{\Psi(z)}{\Psi'(z_j) \cdot (z - z_j)(\zeta - z_j)} \right\} d\zeta. \quad (*)$$
The expression in curly braces $\{\}$, considered as a function of $\zeta$, has residues 1 at $z$ and $\frac{\Psi(z)}{\Psi'(z_j)(z-z_j)}$ at each $z_j$, $j = 0, \ldots, n$; those points are simple poles of the expression as a function of $\zeta$. These are the same as the residues of the function of $\zeta$ given by $\frac{1}{\zeta-z} \cdot \frac{\Psi(z)}{\Psi'(\zeta)}$ at $z$ and each $z_j$, as the reader can easily verify. Therefore the integral on the r. h. s. of the equation $(\star)$ has the same value as that on the r. h. s. of

$$f(z) - p(z) = \frac{1}{2\pi i} \int_{\Gamma} \frac{f(\zeta)}{\zeta-z} \cdot \frac{\Psi(z)}{\Psi'(\zeta)} d\zeta$$

and so that integral is the error term:

$$f(z) = p(z) + \frac{1}{2\pi i} \int_{\Gamma} \frac{f(\zeta)}{\zeta-z} \cdot \frac{\Psi(z)}{\Psi'(\zeta)} d\zeta.$$  

3. The Lagrange Form of the Interpolating Polynomial; Hermite Interpolating Polynomials.

I have nothing to add to Atkinson, p. 133 and §3.6.

4. Computing Interpolating Values Recursively: Neville’s Algorithm.

It is amusing to know that for the “multiplicity-1” interpolation problem, where we only require a polynomial $p(\cdot)$ of degree $\leq n$ that interpolates $p(x_j) = y_j$ for $j = 0, \ldots, n$, there is a recursive method for computing the value $p(x)$ at any $x$ (as distinct from “computing $p(\cdot)$ itself,” i.e., computing its coefficients (or something equivalent to its coefficients). The recursion goes like this:

For one node, $p(x) = y_0$ is a polynomial of degree 0 that takes the correct value at $x_0$; done.

For nodes $x_0$ and $x_1$, let $p_0(x) = y_0$ and $p_1(x) = y_1$ be the constant functions that take the correct values at $x = x_0$ and $x = x_1$ respectively. Then

$$p_{01}(x) = \frac{x - x_0}{x_1 - x_0} \cdot p_0(x) + \frac{x_1 - x}{x_1 - x_0} \cdot p_1(x)$$

$$= \frac{x - x_0}{x_1 - x_0} \cdot y_0 + \frac{x_1 - x}{x_1 - x_0} \cdot y_1$$

is a polynomial of degree 1 that takes the correct values at $x_0$ and $x_1$; indeed, the second way we wrote it is just the two-point form of the line joining the points $(x_0, y_0)$ and $(x_1, y_1)$. For any particular value of $x$ this formula yields the correct value of the 1st-degree polynomial $p_{01}(x)$ interpolating the values $y_0$ and $y_1$ at $x_0$ and $x_1$ respectively.

\ldots

Suppose one knows the polynomials $p_0, \ldots, n-1(x)$ and $p_1, \ldots, n(x)$, each of degree $\leq (n-1)$, that interpolate the values $y_j$ at $x_j$ for $j = 0, \ldots, (n-1)$ and for $j = 1, \ldots, n$ respectively (or perhaps one just knows their values at a particular $x$). Then the expression given by

$$p_{0,\ldots,n}(x) = \frac{x - x_0}{x_n - x_0} \cdot p_0,\ldots,n-1(x) + \frac{x_n - x}{x_n - x_0} \cdot p_1,\ldots,n(x)$$

is a polynomial function of $x$ of degree $\leq n$. Moreover, it takes the value $y_j$ at each $x_j$, $j = 0, \ldots, n$:

At $x = x_0$,

$$p_{0,\ldots,n}(x_0) = \frac{x_n - x_0}{x_n - x} \cdot p_0,\ldots,n-1(x_0) + 1 \cdot y_0 = y_0$$

At $x = x_j$ for $1 \leq j \leq n$,

$$p_{0,\ldots,n}(x_j) = \frac{x_n - x_j}{x_n - x} \cdot p_0,\ldots,n-1(x_j) + \frac{x_j - x_0}{x_n - x_0} \cdot p_1,\ldots,n(x_j)$$

$$= \frac{x_n - x_j}{x_n - x} \cdot y_j + \frac{x_j - x_0}{x_n - x_0} \cdot y_j = y_j$$

At $x = x_n$, $p_{0,\ldots,n}(x_n) = y_n$ in a “mirror image” of the case $x = x_0$. 

5

642:573:01 Fall 2000
It follows that \( p_0, \ldots, n(x) \) is in fact the (uniquely determined) interpolating polynomial, and that its value at any given \( x \) may be computed recursively in the manner indicated above. Implementation of this recursion is called Neville’s Algorithm. It has little to recommend it—for the same amount of computation required to produce a single value of the interpolating polynomial, Newton divided-difference methods will produce the coefficients of the polynomial and an efficient way to compute its values. (To find Neville’s algorithm coded, see, e.g., Numerical Recipes, §3.1.) We shall use the representation of the interpolating polynomial that Neville’s algorithm gives in only one situation: it shows us how to compute divided differences recursively when—for technical reasons—we have chosen to define them in a non-computable way.

5. The Newton Form of the Interpolating Polynomial; Divided Differences.

Let \( P_n \) denote the set of polynomials of degree \( \leq n \); this set forms a vector space in an obvious way, since a constant \( (3) \) multiple of \( p(x) \in P_n \) again belongs to \( P_n \) and \( p(x) \pm q(x) \) belongs to \( P_n \) whenever \( p(x) \) and \( q(x) \) do. Since every \( p(x) \in P_n \) can be written in the “usual power form”

\[
p(x) = \sum_{j=0}^{n} a_j x^j
\]

for unique \( a_j \)’s which can only all be zero if \( p(x) = 0 \) identically, it is evident that the set \( \{1 = x^0, x^1, \ldots, x^n\} \) forms a basis of \( P_n \) and thus that \( P_n \) has dimension \( n + 1 \).

Here is a way to produce a large quantity of bases for \( P_n \): for each \( j = 0, \ldots, n \), suppose

\[
q_j(x) = x^j + \text{lower degree powers of } x
\]

is a (without loss of generality, monic) polynomial of degree \( j \). Then \( \{q_0(x), \ldots, q_n(x)\} \) is a basis for \( P_n \). Two ways to see this are:

1. The \( \{q_j : 0 \leq j \leq n\} \) must be linearly independent, for if \( a_0 q_0(x) + \cdots + a_n q_n(x) = 0 \) (identically) were a nontrivial linear dependence, we could let \( a_k \) be the nonzero coefficient of largest index \( k \). Then we would have

\[
0 = a_0 q_0(x) + \cdots + a_n q_n(x)
\]

\[
= a_k x^k + \text{lower degree powers of } x \quad \text{(identically)}
\]

and the \( k \)-th degree polynomial on the r. h. s. is identically zero, so \( a_k = 0 \) contrary to the way we chose it! Thus the \( \{q_j(x) : 0 \leq j \leq n\} \) are linearly independent, and since there are \( n + 1 \) of them they must form a basis.

2. Evidently \( q_0(x) = 1 \) (identically), so every 0-th degree polynomial can be written uniquely as a multiple of \( q_0(x) \). Suppose that we know that every polynomial of degree \( < k \) can be written uniquely as a linear combination of the polynomials \( \{q_j(x) : 0 \leq j < k\} \). Let

\[
p(x) = \sum_{j=0}^{k} a_j x^j
\]

be a polynomial of degree \( k \). Then

\[
p(x) - a_k q_k(x) = \sum_{j=0}^{k-1} a_j x^j
\]

(2) The books Numerical Recipes: The Art of Scientific Computing, Cambridge Univ. Press, have come in various flavors at various times. In the 1986 FORTRAN edition, this was p. 80 ff.; in the 1987 C edition, it was p. 88 ff. Presumably the Pascal edition has been overtaken by events. These are very good cookbooks: look up the latest one in the language of your choice.

(3) Multiplication by non-constant polynomials cannot be permitted, since that would raise the degree, possibly to a value higher than \( n \).
has degree \(< k\)—the monomials in \(x^k\) on the l. h. s. are equal but opposite in sign—and so there exist unique coefficients \(\{b_j : 0 \leq j < k\}\) for which

\[
p(x) - a_k q_k(x) = \sum_{j=0}^{k-1} b_j q_j(x)
\]

and thus

\[
p(x) = \sum_{j=0}^{k-1} b_j q_j(x) + a_k q_k(x)
\]

so \(p(x)\) can be written as a linear combination of \(\{q_j : 0 \leq j \leq k\}\). The linear-combination coefficients are uniquely determined because if \(p(x) = \sum_{j=0}^{k} c_j q_j(x)\) held for some coefficients \(\{c_j : 0 \leq j \leq k\}\), then comparing coefficients of the highest-degree monomial \(x^k\) on both sides would give \(a_k = c_k\), and thus

\[
k \sum_{j=0}^{k-1} c_j q_j(x) = p(x) - a_k q_k(x) = \sum_{j=0}^{k-1} b_j q_j(x),
\]

so by uniqueness for degrees \(< k\), each \(c_j = b_j\) for \(0 \leq j < k\). By mathematical induction, this is true for all \(k\), \(0 \leq k \leq n\), finishing the proof.

For the time being one should think of \(x_0, \ldots, x_k\) in the next definition as being distinct, though we shall later relax that requirement.

**Definition:** The Newton basis for the vector space of polynomials of degree \(\leq n\) determined by the (ordered list of) points \(x_0, \ldots, x_n\) is the sequence of polynomials

\[
\{1, (x - x_0), \ldots, (x - x_0)(x - x_1) \cdots (x - x_{n-1})\}.
\]

**Definition:** Let \(f(x)\) be a function defined in an interval \(J\) on the real line containing the points \(\{x_0, \ldots, x_k\}\). The divided difference of \(f(\cdot)\) at those points, written \(f[x_0, \ldots, x_k]\), is the leading coefficient \(A_k\) in the polynomial

\[
p(x) = A_0 + \cdots + A_k x^k
\]

of degree \(\leq k\) that interpolates \(f(x_0), \ldots, f(x_k)\) at \(x_0, \ldots, x_k\).

What an ungrateful definition! Since the interpolation problem has a uniquely determined solution, the coefficient \(A_k\) is a well-defined number—it even depends linearly on the function \(f(\cdot)\)—but there seems to be no way to compute it without actually solving the interpolation problem. However, several properties of the number \(f[x_0, \ldots, x_k]\) are almost immediately apparent:

**Proposition:**

1. \(f[x_0, \ldots, x_k]\) is a symmetric function of \(x_0, \ldots, x_k\).
2. If \(\Psi_k(x) = (x - x_0)(x - x_1) \cdots (x - x_k)\), then

\[
f[x_0, \ldots, x_k] = \sum_{j=0}^{k} \frac{f(x_j)}{\Psi_k'(x_j)}.
\]
3. If \(p(x)\) is the polynomial of degree \(\leq k\) that interpolates \(f(x_0), \ldots, f(x_n)\) at \(x_0, \ldots, x_n\), then with respect to the Newton basis \(p(x)\) must have the form

\[
p(x) = f(x_0) + \cdots + f[x_0, \ldots, x_k](x - x_0)(x - x_0) \cdots (x - x_{k-1}) + f[x_0, \ldots, x_n](x - x_0) \cdots (x - x_{n-1}).
\]

**Proof.** Of (1): The polynomial of degree \(\leq k\) that interpolates \(f(x_j)\) at \(x_j\) for all \(0 \leq j \leq k\) is well-determined independently of the order in which \(\{x_0, \ldots, x_k\}\) are listed, and therefore so is its leading
are the same—but it has at least
is of degree
\( \leq \)
as advertised.

cases. The key to developing this formula—Atkinson’s (3.2.7)—is not the “judicious manipulation” that
We still have no general formula with which to compute the concrete numbers
\( f \) respectively. The Neville’s-algorithm construction says that the polynomial
\( p \)
and since the leading coefficient in
\( f \) interpolates these values and also interpolates
\( p(\text{x}) = f(\text{x}) \); then the leading coefficient of
\( p \)
is the divided difference
\( f[\text{x}_0, \ldots, \text{x}_n] \) by definition:
\[
p(\text{x}) = f[\text{x}_0, \ldots, \text{x}_n](\text{x} - \text{x}_0) \cdots (\text{x} - \text{x}_{n-1}) + \text{lower degree terms}.
\]
It follows that the polynomial
\( q(\text{x}) \) defined by
\[
q(\text{x}) = p(\text{x}) - \{ f[\text{x}_0] + \cdots + f[\text{x}_{n-1}](\text{x} - \text{x}_0) \cdots (\text{x} - \text{x}_{n-2}) + f[\text{x}_0, \ldots, \text{x}_n](\text{x} - \text{x}_0) \cdots (\text{x} - \text{x}_{n-1}) \}
\]
is of degree \( \leq (n - 1) \)—the \( n \)-th degree terms of the two polynomials on the r. h. s. of the defining equation
are the same—but it has at least \( n \) zeros: the induction hypothesis guarantees that the two polynomials on
the r. h. s. of the defining equation take the same values at \( \text{x}_0, \ldots, \text{x}_{n-1} \). It must therefore be the case that
\( q(\text{x}) \equiv 0 \) identically, or
\[
p(\text{x}) \equiv f[\text{x}_0] + \cdots + f[\text{x}_{n-1}](\text{x} - \text{x}_0) \cdots (\text{x} - \text{x}_{n-2}) + f[\text{x}_0, \ldots, \text{x}_n](\text{x} - \text{x}_0) \cdots (\text{x} - \text{x}_{n-1})
\]
as advertised.

We still have no general formula with which to compute the concrete numbers \( f[\text{x}_0, \ldots, \text{x}_k] \) in concrete
cases. The key to developing this formula—Atkinson’s (3.2.7)—is not the “judicious manipulation” that
Let distinct \( \text{x}_0, \ldots, \text{x}_n \) be given. Then we can find the polynomials of degree \( \leq (n - 1) \) that interpolate
\( f(\text{x}_0), \ldots, f(\text{x}_n) \) at \( \text{x}_0, \ldots, \text{x}_{n-1} \) and \( \text{x}_1, \ldots, \text{x}_n \) respectively, and our definition of divided difference tells
us their leading coefficients: they are
\[
p_{0,\ldots,n-1}(\text{x}) = f[\text{x}_0, \ldots, \text{x}_{n-1}] \cdot \text{x}^{n-1} + \text{lower degree terms}
\]
\[
p_{1,\ldots,n}(\text{x}) = f[\text{x}_1, \ldots, \text{x}_n] \cdot \text{x}^{n-1} + \text{lower degree terms}
\]
respectively. The Neville’s-algorithm construction says that the polynomial
\( p(\text{x}) \) of degree \( \leq n \) that interpolates
\( f(\text{x}_0), \ldots, f(\text{x}_n) \) must then be
\[
\frac{\text{x}_n - \text{x}}{\text{x}_n - \text{x}_0} \cdot p_{0,\ldots,n-1}(\text{x}) + \frac{\text{x} - \text{x}_0}{\text{x}_n - \text{x}_0} \cdot p_{1,\ldots,n}(\text{x})
\]
\[
= \frac{\text{x}_n - \text{x}}{\text{x}_n - \text{x}_0} \cdot f[\text{x}_0, \ldots, \text{x}_{n-1}] \cdot \text{x}^{n-1} + \frac{\text{x} - \text{x}_0}{\text{x}_n - \text{x}_0} \cdot f[\text{x}_1, \ldots, \text{x}_n] \cdot \text{x}^{n-1} + \text{lower degree terms}
\]
\[
= \frac{f[\text{x}_1, \ldots, \text{x}_n] - f[\text{x}_0, \ldots, \text{x}_{n-1}]}{\text{x}_n - \text{x}_0} \cdot \text{x}^{n} + \text{lower degree terms}
\]
and since the leading coefficient in
\( p(\text{x}) \) is \( f[\text{x}_0, \ldots, \text{x}_n] \) by definition, we must have

**Theorem:** The divided differences of \( f(\text{x}) \) at distinct points \( \text{x}_0, \ldots, \text{x}_n \) are given by the recursion
\[
f[\text{x}_0, \ldots, \text{x}_n] = \frac{f[\text{x}_1, \ldots, \text{x}_n] - f[\text{x}_0, \ldots, \text{x}_{n-1}]}{\text{x}_n - \text{x}_0}.
\]
This is Atkinson’s (3.2.7), p. 139.

The basic fact in this area—and I intend to use it essentially without proof—is the Hermite-Gennochi formula of Atkinson’s Theorem 3.3, p. 144 ff.: if \( f(x) \) is \( n \)-times continuously differentiable on an interval containing \( x_0, \ldots, x_n \), then

\[
f[x_0, \ldots, x_n] = \int \cdots \int_{\tau_n} f^{(n)}(t_0x_0 + \cdots + t_nx_n) \, dt_1 \cdots dt_n
\]

where \( t_0 = 1 - \sum_{j=1}^{n} t_j \) and the integral is extended over an \( n \)-dimensional simplex. Atkinson discusses the cases \( n = 1, 2 \) in some detail; I do not intend to check or require the details in the \( n \)-dimensional general case.

Given the Hermite-Gennochi formula, standard theorems about integrals of functions with parameters in them tell us that if \( f(x) \) is sufficiently differentiable, then \( f[x_0, \ldots, x_n] \), considered as a function of the \( n + 1 \) nodes, has a unique continuous extension defined on \( J \times \cdots \times J \)—that is, the divided differences have well-defined values, obtained by taking limits, even when some or all of the nodes are equal to each other. Moreover, if \( f(x) \) has sufficiently many continuous derivatives then \( f[x_0, \ldots, x_n] \) is even a differentiable function of the nodes: e.g.,

\[
\frac{\partial f[x_0, \ldots, x_n]}{\partial x_j} = \int \cdots \int_{\tau_n} t_j f^{(n+1)}(t_0x_0 + \cdots + t_nx_n) \, dt_1 \cdots dt_n.
\]

It cannot be emphasized too strongly that, in general, one does not compute these partials using this formula! The formula assures us that they must exist, and we then use cunning to exhibit formulas by which they may be computed. For example, we have

\[
\frac{d}{dx} f[x_0, \ldots, x_{n-1}, x] = f[x_0, \ldots, x_{n-1}, x, x]
\]

as we showed in class, or per Atkinson, pp. 146–147 (3.2.17). The higher-order versions of this formula, incidentally, are not quite what one might expect at first guess. For example, to get a formula involving second partials or three equal nodes, one should consider the function of \( x \) defined by

\[
g(x) = f[x_0, \ldots, x_{n-1}, x, x, x, x+1] \text{ with } x_n = x \text{ and } x_{n+1} = x.
\]

Computing \( g'(x) \) by the chain rule gives

\[
g'(x) = f[x_0, \ldots, x_n, x, x+1] \cdot 1 + f[x_0, \ldots, x_n, x_{n+1}, x] \cdot 1
\]

which with \( x_n = x \) and \( x_{n+1} = x \) gives

\[
g'(x) = f[x_0, \ldots, x_{n-1}, x, x, x] + f[x_0, \ldots, x, x, x]
\]

so \( \frac{d^2}{dx^2} f[x_0, \ldots, x_{n-1}, x] = 2 \cdot f[x_0, \ldots, x_{n-1}, x, x, x] \) and in general

\[
\frac{d^k}{dx^k} f[x_0, \ldots, x_{n-1}, x] = k! \cdot f[x_0, \ldots, x_{n-1}, x, \ldots, x] \quad (x \text{ repeated } k + 1 \text{ times}).
\]

In particular, \( \frac{d^k}{dx^k} f = k! \cdot f \quad (x \text{ repeated } k + 1 \text{ times}), \)

and the reader may find it reassuring to check the cases \( k = 1, 2 \) of this relation by letting \( x_0, x_1 \) and \( x_2 \) approach \( x \) in the explicit quotient representations of \( f[x_0, x_1] \) and \( f[x_0, x_1, x_2] \).
Incidentally, since \( f[x_0, \ldots, x_n] \) is a symmetric function of its arguments for distinct values of its arguments, the symmetry also holds (by taking limits in the Hermite-Gennochi formula) when some of the arguments are equal. Thus, even in the equal-argument case, the order in which the arguments are listed has no effect on the value. However, it is easiest to state and prove the next theorem—which is the principal result of this \( \S \)—if equal arguments are kept next to each other.

**Theorem:** Let \( x_0, \ldots, x_m \) be \( m + 1 \) points in an interval \( J \), of which the first \( m_1 \) are equal, the next \( m_2 \) are equal, \ldots, with \( m = \sum_j m_j - 1 \). Then the Newton polynomial of degree \( m \)

\[
p(x) = f[x_0] + \cdots + f[x_0, \ldots, x_k](x-x_0)\cdots(x-x_{k-1}) + \cdots,
\]

in which divided differences with some arguments equal are evaluated by differentiation, satisfies

\[
p(x_0), \ldots, p^{(m_0-1)}(x_0) = f^{(m_0-1)}(x_0)
\]

and the similar relations for the other points, and thus \( \{ \) with a slight difference of notation \( \} \) is the solution of the general interpolation problem of this section.

The following lemma will help us to recognize how the polynomial constructed by the Newton-polynomial-and-divided-difference method satisfies the conditions that we are trying to impose on derivatives.

**Lemma:** Let \( F(x) \) and \( G(x) \) be functions which are \( m \)-times-continuously-differentiable in some open interval containing the point \( a \). Then the three following conditions are equivalent:

1. \( F^{(k)}(a) = G^{(k)}(a) \) for \( k = 0, \ldots, m-1 \);
2. The \((m-1)\)-st degree Taylor polynomials of \( F(x) \) and \( G(x) \) with center \( a \),

\[
\sum_{k=0}^{m-1} \frac{F^{(k)}(a)}{k!} (x-a)^k \quad \text{and} \quad \sum_{k=0}^{m-1} \frac{G^{(k)}(a)}{k!} (x-a)^k,
\]

are equal;
3. There is a function \( R(x) \) defined for \( x \) in some open interval containing the point \( a \), such that \( R(x) \) has the property that \( \lim_{x \to a} R(x) = 0 \) and such that \( F(x) - G(x) = R(x) \cdot (x-a)^{m-1} \) for all \( x \) sufficiently close to \( a \).

**Proof of the Lemma.** Conditions (1) and (2) are obviously equivalent—they just say the same thing about the Taylor polynomials in two slightly different ways.

To see that (1) \( \Rightarrow \) (3), write the Taylor polynomial of degree \((m-1)\) with center \( a \) of the function \( F(x) - G(x) \). Since the derivatives of \( F(x) \) and \( G(x) \) agree at \( a \) for orders \( k = 0, \ldots, m-1 \), the Taylor polynomial in question is identically zero and we get only the remainder term instead of the polynomial: for each \( x \),

\[
F(x) - G(x) = \frac{F^{(m)}(\xi_x) - G^{(m)}(\xi_x)}{m!} \cdot (x-a)^m = \left\{ (x-a) \cdot \frac{F^{(m)}(\xi_x) - G^{(m)}(\xi_x)}{m!} \right\} \cdot (x-a)^{m-1}
\]

for some \( \xi_x \) between \( x \) and \( a \). If we choose such a \( \xi_x \) for each such \( x \), then the expression in \( \{ \} \)’s defines a function \( R(x) \). Since \( x \to a \) implies \( \xi_x \to a \), we see that \( R(x) \to 0 \) as \( x \to a \). Thus

\[
F(x) - G(x) = R(x) \cdot (x-a)^{m-1}
\]

with \( R(x) \to 0 \) as \( x \to a \), as desired.

To see that (3) \( \Rightarrow \) (1), write the Taylor polynomial with remainder for the function \( F(x) - G(x) = R(x) \cdot (x-a)^{m-1} \):

\[
R(x) \cdot (x-a)^{m-1} = F(x) - G(x) = \sum_{j=0}^{m-1} \frac{F^{(j)}(a) - G^{(j)}(a)}{j!} (x-a)^j + \frac{F^{(m)}(\xi_x) - G^{(m)}(\xi_x)}{m!} (x-a)^m.
\]
If there is any \( j \leq m - 1 \) for which \( F^{(j)}(a) \neq G^{(j)}(a) \), let \( k \) be the smallest \( j \) for which that is true. Then we can start the Taylor sum at \( j = k \) instead of \( j = 0 \):

\[
R(x) \cdot (x - a)^{m-1} = \sum_{j=k}^{m-1} \frac{F^{(j)}(a) - G^{(j)}(a)}{j!} (x - a)^j + \frac{F^{(m)}(\xi_x) - G^{(m)}(\xi_x)}{m!} (x - a)^m.
\]

Divide both sides by \( (x - a)^k \):

\[
R(x) \cdot (x - a)^{m-k-1} = \sum_{j=k}^{m-1} \frac{F^{(j)}(a) - G^{(j)}(a)}{j!} (x - a)^{j-k} + \frac{F^{(m)}(\xi_x) - G^{(m)}(\xi_x)}{m!} (x - a)^{m-k}.
\]

The l. h. s. must approach 0 as \( x \to a \), since \( m - k - 1 \geq 0 \) and \( R(x) \to 0 \). The r. h. s. is a polynomial in \( (x - a) \) plus a remainder that goes to zero as \( x \to a \), so as \( x \to a \) it approaches the constant term of the polynomial, which is \( \frac{F^{(k)}(a) - G^{(k)}(a)}{k!} \). Thus this term equals zero, which contradicts the choice of \( k \). We conclude that all the \( \frac{F^{(j)}(a) - G^{(j)}(a)}{j!} = 0 \), which means the same thing as condition (1).

**Proof of the theorem.** Again the proof proceeds by mathematical induction on the number of nodes, in this case counted according to multiplicity. If the only node is \( x_0 \) (with multiplicity 1), then \( p(x) = f(x_0) = f[x_0] \) solves the interpolation problem trivially, as before. Suppose

\[
p_{k-1}(x) = f[x_0] + \cdots + f[x_0, \ldots, x_{k-1}](x-x_0)\cdots(x-x_{k-2})
\]

satisfies all the interpolation conditions at all the points up to the one preceding \( x_{k-1} \), and also satisfies the derivative-interpolation conditions at \( x_{k-1} \) up to and including order one less than the number of repetitions of \( x_{k-1} \) in the list \( x_0, \ldots, x_{k-1} \).

We distinguish two cases in the induction step. **First:** suppose that \( x_k \) is different from \( x_{k-1} \). Then in the expression

\[
p_k(x) = f[x_0] + \cdots + f[x_0, \ldots, x_{k-1}](x-x_0)\cdots(x-x_{k-2}) + f[x_0, \ldots, x_k](x-x_0)\cdots(x-x_{k-1})
\]

the newly added term has a zero of order \( m_0 \) at \( x_0 = x_1 = \cdots, m_0 \) times \( x_0 \) and similarly at the points up to \( x_{k-1} \). Thus it changes neither the value nor the derivatives up to the \( m_j \)-th at any of these points, and so the induction step will be finished if this expression takes the correct value (no derivatives involved) at \( x_k \). But this expression is a continuous function of the variables \( x_0, \ldots, x_k \), and we know it takes the correct value if \( x_0, \ldots, x_k \) are distinct; therefore, by taking limits, we see that it must also take the correct value when some of them are equal to each other, and that takes care of the induction in this case.

**Second:** suppose \( x_k = x_{k-1} \). Let \( h \neq 0 \) be so small that \( x_k + h \) differs from all the \( x_j \)'s. We know that for distinct \( x_0, \ldots, x_{k-2} \) and \( z \) we shall have

\[
\frac{\partial f[x_0, \ldots, x_{k-2}, z]}{\partial z} = f[x_0, \ldots, x_{k-2}, z, z];
\]

by the continuity of everything in sight this also holds when some of the \( x_j \)'s are equal among themselves and/or after differentiation, \( z \) is set equal to some of them. The approximation property of the derivative says, for any differentiable \( g(z) \), that

\[
g(z+h) = g(z) + h \cdot g'(z) + h \cdot R(h) \quad \text{where} \quad R(h) \to 0 \quad \text{as} \quad h \to 0.
\]

Applied to \( f[x_0, \ldots, x_{k-2}, z] \), this then says for \( z = x_{k-1} = x_k \) that

\[
f[x_0, \ldots, x_{k-2}, x_{k-1} + h] = f[x_0, \ldots, x_{k-2}, x_{k-1}] + f[x_0, \ldots, x_{k-2}, x_{k-1}, x_k] \cdot h + h \cdot R(h) \quad (\ast)
\]

11
Now by the first part of the proof—in which the point we are considering was different from the points that preceded it—we know that
\[ f(x) = f[x_0] + \cdots + f[x_0, \ldots, x_{k-1} + h](x - x_0) \cdots (x - x_{k-2}) \]
when \( x = x_k + h \). Substituting \((\ast)\) for the last divided difference in that formula tells us that when \( x = x_k + h \), we have
\[
\begin{align*}
  f(x) &= f[x_0] + \cdots + f[x_0, \ldots, x_{k-1}](x - x_0) \cdots (x - x_{k-2}) \\
  &\quad + f[x_0, \ldots x_{k-1}, x_k] \cdot h \cdot (x - x_0) \cdots (x - x_{k-2}) + h \cdot R(h) \cdot (x - x_0) \cdots (x - x_{k-2}) .
\end{align*}
\]
Replace \( h \) by \( x - x_{k-1} \) (remember, \( x_k = x_{k-1} \)) and this becomes
\[
\begin{align*}
  f(x) &= f[x_0] + \cdots + f[x_0, \ldots, x_{k-1}](x - x_0) \cdots (x - x_{k-2}) \\
  &\quad + f[x_0, \ldots x_{k-1}, x_k] \cdot (x - x_0) \cdots (x - x_{k-2})(x - x_{k-1}) \\
  &\quad + (x - x_{k-1}) \cdot R(x - x_{k-1}) \cdot (x - x_0) \cdots (x - x_{k-2}) .
\end{align*}
\]
We need to know that the sum of all the terms of that expression but the last has derivatives agreeing with those of \( f(\cdot) \) at \( x_k \) up to order one higher than the sum of the terms on the first line of the expression. To see that this is true, rewrite the “remainder term” on the last line as
\[ (x - x_0) \cdots (x - x_{k-2})(x - x_{k-1}) \cdot R(x - x_{k-1}) . \]
Because the last factor goes to zero as \( x \to x_{k-1} = x_k \), this remainder term has the form \( R(x - x_k) \cdot (x - x_k)^{m-1} \), where the factor \( (x - x_k)^{m-1} \) occurs the same number of times as in the preceding term, and \( R(x - x_k) \to 0 \) as \( x \to x_k \). This is the setting of the Lemma we proved before trying to prove this theorem. Therefore, by the lemma, the Taylor polynomial of the function of \( x \) given by
\[
\begin{align*}
  f[x_0] + \cdots + f[x_0, \ldots, x_{k-1}](x - x_0) \cdots (x - x_{k-2}) + f[x_0, \ldots x_{k-1}, x_k] \cdot (x - x_0) \cdots (x - x_{k-2})(x - x_{k-1})
\end{align*}
\]
at \( x = x_k \) agrees with the Taylor polynomial of \( f(x) \) at \( x_k \) to degree equal to the number of such factors in the preceding term, which is one degree higher than
\[
\begin{align*}
  f[x_0] + \cdots + f[x_0, \ldots, x_{k-1}](x - x_0) \cdots (x - x_{k-2})
\end{align*}
\]
does. But the Newton polynomial on the set-off line just-preceding has derivatives at \( x_k \) that agree with those of \( f(\cdot) \) up to one less than the number of repetitions of \( x_k \) in the list \( x_0, \ldots, x_{k-1}, \) by the induction hypothesis. So the new polynomial’s derivatives at \( x_k \) agree with those of \( f(\cdot) \) up to one less than the number of repetitions of \( x_k \) in the list \( x_0, \ldots, x_{k-1}, x_k \). This is the next case of the theorem, and the induction has been completed.

(Note: To make the argument above absolutely rigorous, we would have to be sure that some of the limits were taken, and remainder terms estimated, uniformly in the various variables. This is easy to do by using the Hermite-Gennochi formula, assuming continuous differentiability of \( f(\cdot) \) to a sufficiently high order; any interested reader can fill in the details.)

The importance of the result of this § is that it extends the difference-table algorithm to cover the case where one interpolates derivatives: all that happens, essentially, is that formal divided differences with zero denominators are replaced by appropriate derivatives (divided by factorials where necessary). It also extends the Cauchy-integral form of the error term (see pp. 4–5 above) to the case where some of the nodes \( z_0, \ldots, z_n \) are equal: the integral is obviously a continuous (and in fact analytic) function of the (complex) nodes \( z_0, \ldots, z_n \), so by taking limits from the case where \( z_0, \ldots, z_n \) are distinct, we see that the Cauchy-integral form of the error term with some \( z_j \) equal to others is still correct for the Newton polynomial
\[ p_k(z) = f[z_0] + \cdots + f[z_0, \ldots, z_k](z - z_0) \cdots (z - z_{k-1}) \]
in which some of the nodes may be equal. But by what we just proved in the real case, this is the correct interpolating polynomial on the real line. Since values and derivatives determined on the real line are the same as those determined in the complex domain, the Newton polynomial is also the correct complex interpolating polynomial, and so the integral error term is the correct error term for the interpolating polynomial, at least when \( z \) is real. But by the principle of analytic continuation (or continuation of functional relations), it then continues to be correct even for complex values of \( z \). {Of course all this stuff is correct even for complex nodes \( z_0, \ldots, z_n \)—but this course isn’t even supposed to be talking about complex variables at all. If it were, I would discuss the error term in the complex case by pure complex-variable methods rather than the hybrid I just gave.}


I really have only two things to add to Atkinson’s treatment of these subjects on pp. 139–144.

(1) Newton coefficients can be computed by divided-difference tables even when one wants to interpolate derivatives—and therefore some of the nodes are equal. One simply replaces “indicated divided differences” involving equal nodes with appropriate values of \( \frac{f^{(k)}(x)}{k!} \). For example, to find the 3rd-degree polynomial with \( f(0) = 0, f(1) = 1, f'(1) = 3, f''(1) = 6 \) \{and which we know will have to be \( x^3 \}\), we put a single node \( x_0 = 1 \) and 3 nodes \( x_1 = x_2 = x_3 \) at 1; the divided-difference table takes the form

\[
\begin{array}{cccc}
 0 & 0 & 1 & 2 & 1 \\
 1 & 1 & 3/1 & 6/2 \\
 1 & 1 & 3/1 \\
 1 & 1 \\
\end{array}
\]

and consequently

\[
f(x) = 0 + (x - 0) + 2 \cdot (x - 0)(x - 1) + 1 \cdot (x - 0)(x - 1)^2 = x + 2x^2 - 2x + x^3 - 2x^2 + x = x^3
\]

as expected.

(2) While the algorithm \( \text{Interp}(d, x, n, t, p) \) on pp. 141–142 is the efficient nested-multiplication way to compute the value of a polynomial whose Newton-divided-difference coefficients are known, one computes valuable information in the course of the evaluation and sometimes that information is worth saving. Suppose that

\[ p(x) = d_0 + d_1(x - x_0) + \cdots + d_n(x - x_0) \cdots (x - x_{n-1}) \]

and suppose that we set

\[ b_n = d_n \]

and then for \( j = n - 1, n - 2, \ldots, 0 \) do

\[ b_j = d_j + b_{j+1} \cdot (t - x_j). \]

This is just \( \text{Interp}(d, x, n, t, p) \), except that we are saving the result \( p \) at the \( j \)-th step, \( j = n, n - 1, \ldots, 0 \) under the name \( b_j \). If we rewrite the defining equations of the recursion—with a shift of indices—in the form

\[ d_{j-1} = b_j \cdot (x_{j-1} - t) + b_{j-1} \]
and substitute these into the original form of \( p(x) \)—but written with its terms in decreasing order of degrees (and indices)—we get

\[
p(x) = b_n(x - x_{n-1})(x - x_{n-2}) \cdots (x - x_1)(x - x_0)
+ [b_n(x_{n-1} - t) + b_{n-1}](x - x_{n-2}) \cdots (x - x_1)(x - x_0)
+ \cdots
+ [b_2(x_1 - t) + b_1](x - x_0)
+ [b_1(x_0 - t) + b_0]
\]

and if we combine the term containing \( b_j \) on the \( j \)-th line \((j = n, n-1, \ldots \) reading downward) with the term containing \( b_j \) on the line immediately below it, this becomes

\[
p(x) = b_n(x - x_{n-2}) \cdots (x - x_1)(x - x_0)(x - t)
+ b_{n-1}(x - x_{n-3}) \cdots (x - x_1)(x - x_0)(x - t)
+ \cdots
+ b_2(x - x_0)(x - t) + b_1(x - t) + b_0
\]

which is the Newton form for \( p(x) \) with centers \( t, x_0, \ldots, x_{n-1} \): the number \( t \) is “shifted in” as the 0-th node, the previous \( n \)-th node is “shifted out,” and all the other nodes have their indices shifted up one click. If the polynomial is thought of with these new centers, then it is well adapted for calculations at points \( x \) near \( t \), since the factor \( x - t \) tends to keep the later terms small. If the process is repeated so that the nodes become \( t, t, x_0, \ldots, x_{n-2} \), then in the expansion

\[
p(x) = c_0 + c_1(x - t) + c_2(x - t)^2 + \cdots
\]

it is evident that \( c_1 = p[t, t] = p'(t) \), so one has computed the derivative of \( p(x) \) at \( x = t \); iteration of this process will eventually put \( p(x) \) into the Taylor form

\[
p(x) = p(t) + \frac{p'(t)}{1!} (x - t) + \frac{p''(t)}{2} (x - t)^2 + \cdots
\]

A clever person can use this algorithm in all sorts of interesting ways. At the very least, it gives a non-obvious way of differentiating Newton polynomials—or polynomials of any kind, since (if repeated nodes are allowed) the usual Taylor-power-basis monomials \( (x - a)^k \) are Newton polynomials.