Section 1
Polynomial interpolation

1.1 The interpolation problem and an application to rootfinding

Polynomials of degree n simultaneously play the important roles of the simplest nonlinear functions, and the simplest linear space of functions. In this section we will consider the problem of finding the simplest polynomial, that is, the one of smallest degree, whose value at $x = x_i$ agrees with given numerical data y_i for the $n+1$ distinct points x_i, $i = 0, 1, \ldots, n$. We usually think of the y_i as values of an existing function $f(x_i)$ which we are trying to approximate, even if a function f is not explicitly given. Calling right hand sides $f(x_i), y_i$, or $p_n(x_i)$ is a matter of context and connotation. The x_i's are known as nodes of the interpolation, or centers when they appear in a certain representaion of the polynomial discussed below. They are also, by definition, roots of $f(x) - p_n(x)$. The reasons for the existence and uniqueness of such a polynomial of degree $\leq n$ will become apparent shortly. We will see some ways in which this situation arises in rootfinding problems, and later how it is extremely useful in integrating functions and differential equations numerically. We will also observe how it sits at one end of a spectrum of polynomial approximation problems with Taylor approximation at the other, though there is more variety among interpolation polynomials which depend upon many points, as opposed to just one. We will investigate good choices of these points in the later context of an even more general range of polynomial approximation problems. The interpolation problem is a source of linear systems of equations, as waas rootfinding, and understanding the theoretical and practical aspects of their solution will naturally lead us to our study of linear algebra.

The secant method and Muller's method have already suggested the need for finding polynomials to replace functions with particular values at various points. In these linear and quadratic cases it is not necessary to be too systematic, but it is easy to see the usefulness of replacing functions by better polynomial approximation to take advantage of the ease of evaluating, differentiating, and integrating polynomials. This will clearly require a more systematic approach.

$\triangleright$ Exercise 1.1 Show by example that the region of validity of the Taylor polynomial of $f(x)$ at a is too small to be generally useful for the purpose of approximation f over large intervals. Recall the restrictions on the real radius of convergence of the Taylor series imposed by singular behavior or f in the complex plane.

▷ Exercise 1.2 Taylor polynomials put differentiation in Jordan form, exponentials diagonalize it. Consider interpolating and approximating functions by other classes of functions. When would it be useful to have piecewise polynomial or trigonometric approximations?

▷ Exercise 1.3 Show that good polynomial approximations do not need to agree with f at any points, by adding a small positive function (nonpolynomial) to a polynomial.

A simple application of interpolation evolving from our rootfinding concerns Proceeds as follows. If we tried to generalize Muller's method and interpolate four successive iterates by a cubic, and so forth, we would quickly reach the point of diminishing or vanishing returns where we would no longer have a simple (or any for that matter) formula for the roots of that polynomial. We would have to employ a simpler rootfinding method to find it!

An alternative is to interpolate the inverse function by polynomials which need only be *evaluated* at 0 to find the corresponding approximation to the root. Recall that ordered pairs of the inverse of f are found by simply reversing the order of pairs of f.

▷ Exercise 1.4 Implement this idea with linear approximation. You should get the same method as the secant method, since the inverse of a linear function is linear. Then try for quadratic. The difference now from Muller's method points out the subtle point that there is more than one parabolic shape through three points, although there is exactly one quadratic function. In other words, different choice of axes will give different parabolas through the points, but not different lines! The inverse of a quadratic is not quadratic. Explain the meaning of the preservation of linear functions under not just inversion, but all rotations of axes.

Now as each new approximation x_n to the root is found by this procedure, we may evaluate $f(x_n)$ and might hope to use the pair $(f(x_n), x_n)$ belonging to the inverse function along with our other pairs to form an approximation polynomial of one degree higher. This concrete application suggests a feature not apparent in the original formulation of the problem, the need to be able to efficiently adjust the interpolation polynomial for the addition of new data points. This would also allow one to reach a given degree of approximation to be verified empirically as we add points.

1.2 Newton's solution

The simple solution to this problem provides one of many insights as to why we may interpolate $n + 1$ points by a polynomial of degree n.

We may interpolate f at x_0 by a polynomial of degree 0, a constant function $p_0(x) =$

$f(x_0)$. If we wish to add a point $(x_1, f(x_1))$ to our interpolation, we may do so by using the form $p_1(x) = p_0(x) + a_1(x - x_0)$ to ensure we do not affect our previous success. The value of a_1 is uniquely determined by our requirement $p_1(x_1) = p_0(x_1) + a_1(x_1 - x_0) = f(x_1)$, as long as $x_1 \neq x_0$.

$$a_1 = \frac{f(x_1) - p_0(x_1)}{(x_1 - x_0)}$$

▷ Exercise 1.5 Confirm the generalization, that if $p_n(x)$ interpolates f at $x_0, x_1, \ldots, x_n$ then $p_{n+1}(x) = p_n(x) + a_{n+1}(x - x_0)\ldots(x - x_n)$ interpolates f at $x_0, x_1, \ldots, x_n$ and if

$$a_{n+1} \equiv \frac{f(x_{n+1}) - p_n(x_{n+1})}{(x_{n+1} - x_0)\ldots(x_{n+1} - x_n)}$$

then $p_n(x_{n+1}) = f(x_{n+1})$.

This essentially solves the problem, but we will look a little more closely at the interpolating polynomial to understand its error at other points, so when we may choose the points we can do it wisely, and find other forms of a_n which may be computed more effectively. Note that all of the coefficients of the standard form are changed at each stage of this procedure.

▷ Exercise 1.6 Recognize that there are situations in which you may choose the interpolation points, and others in which they are chosen for you by convenience or necessity.

▷ Answer 1.6 Inverse interpolation determines them, as does a table of values, or a function which is easier to compute at some points than others.

▷ Exercise 1.7 Familiarize yourself with the multiple center form

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

of the polynomial which has arisen naturally in this context. In analogy with the Taylor and standard forms, can you find another interpretation of the coefficients?

▷ Answer 1.7 Let $\tilde{p_1}(x) \equiv p(x)$. Then $a_0 = \tilde{p}_1(x_0)$. Defining $\tilde{p}_2(x) = \frac{\tilde{p_1}(x) - a_0}{(x - x_0)}$ we have a polynomial of the same form beginning with $a_1 + a_2(x - x_1) + \ldots$ So $a_1 = \tilde{p}_2(x_1) = \frac{p(x_1) - p(x_0)}{x_1 - x_0}$. Recursively, $\tilde{p}_{j+1}(x) = \frac{\tilde{p}_j(x) - \tilde{p}_j(x_j)}{(x - x_j)}$ and $a_j = \tilde{p}_{j+1}(x_{j+1}) = \frac{\tilde{p}_j(x_{j+1}) - \tilde{p}_j(x_j)}{x_{j+1} - x_j}$.

▷ Exercise 1.8 Count the relative number of operations involved in evaluating the multiple center form directly and using nested evaluation.

▷ Exercise 1.9 Verify that the intermediate coefficients in nested evaluation may be used to change the centers. In particular, if

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

and $a'_j \equiv a_j + (z - c_{j+1})a'_{j+1}$, then $a'_0 = p(z)$ and $p(x) = p(z) + (x - z)q(x)$.

▷ Answer 1.9 Note that $p(x) - p(z)$ has a root at $x = z$ and hence a factor of $(x - z)$, with quotient we call $q(x)$, which may be written in any form we please. The point is that $q(x)$ may be written in a known form with respect to the centers $x_0, \ldots, x_{n-2}$:

$$q(x) = a'_1 + a'_2(x - x_0) + \ldots + a'_{n-1}(x - x_0)\ldots(x - x_{n-2})$$

(Note where x_{n-1} is treated differently from the other centers). To confirm this, we begin with $a'_0 + (x-z)(a'_1 + (x-x_0)(a'_2 + \ldots$ and writing $(x-z)$ as $(x-x_i)-(z-x_i)$ beginning with $i = 0$ we obtain

$$a'_0 - (z - x_0)a'_1 + (x - x_0)(a'_1 + [(x - x_0) - (z - x_0)](a'_2 + \ldots$$

Rearranging to find $a_j = a'_j - (z - c_{j+1})a'_{j+1}$ we recognize this as $a_0 + (x - x_0)(a'_1 + (x - z)(a'_2 + \ldots$ and we proceed as before with $i = 1, 2, \ldots$ until we finally only $\ldots + a'_{n-2} + [(x - x_{n-1}) - (z - x_{n-1})]a'_{n-1} = \ldots + a_{n-2} + (x - x_{n-1})a_{n-1}$ remains.

▷ Exercise 1.10 Use the preceding exercise and the relation $p'(x) = (x - z)q'(x) + q(x)$ and consequently $p'(z) = q(z)$ to describe an efficient implementation of Newton's method for polynomials. Notice in particular that the coefficients of q are provided just in time to allow you to compute $p'(z)$ by nested evaluation simultaneously with $p(z)$. Specialize to polynomials in standard form.

▷ Exercise 1.11 When a root x_0 of $p(x)$ is found within a given tolerance by the above method, we automatically know $\frac{p(x)}{(x-r_0)} = q(x)$. Describe a procedure, called deflation, which finds all of the roots of $p(x)$ finding a root r_1 of $q(x)$, then of $\frac{q(x)}{(x-r_1)}$ and so forth. Show that by storing the intermediate coefficients in nested evaluation of successively deflated polynomials, implementation of deflation means merely to move these coefficients to the right place at the right time. What occurs if $p(x)$ has multiple roots? Examine the possibility of approximate roots of $q(x)$ to a specific tolerance failing to be approximate roots of $p(x)$ to that same tolerance. Justify the advisability of "polishing" the roots by using them as initial guesses for roots of the original $p(x)$.

▷ Exercise 1.12 Choose a polynomial which arises from interpolation and implement the above techniques to find all of its roots. (See the appendix for sample implementations in Fortran.)

▷ Exercise 1.13 The above shows that nested evaluation may be used to find the quotient and remainder when $p(x)$ is divided by $(x - z)$ (synthetic division). Show that the formula for synthetic division is essentially the same as the familiar long division algorithm. Hint: Try the case of a polynomial in standard form (all centers = 0).

▷ Exercise 1.14 Review the theoretical estimates for the number and location of real and complex roots of polynomials (Descartes rule, Laguerre's, etc.).

▷ Exercise 1.15 Design an efficient algorithm to generate equally spaced points hierarchically on a given interval.

▷ Answer 1.15 Begin with 'reflected binary' counting, $.0, .1, .01, .11, .001, \ldots$ which can be implemented in binary data types. This 'VanderKorput' sequence is extremely useful, deterministically providing uniformly distributed sample points at an equal rate to all subintervals.

1.3 Divided differences and the Lagrange polynomials

There is a simple interpretation of the interpolation problem as a linear systems of equations. We will soon study these systems in their own right, motivated by this and the examples arising in nonlinear systems.

In standard form, the conditions for a solution to the interpolation problem may be given as

$$\begin{aligned}
a_0 + a_1 x_0 + a_2 x_0^2 + \ldots + a_n x_0^n &= y_0 \\
a_0 + a_1 x_1 + a_2 x_1^2 + \ldots + a_n x_1^n &= y_1 \\
&\vdots \\
a_0 + a_1 x_n + a_2 x_n^2 + \ldots + a_n x_n^n &= y_n
\end{aligned}$$

This is a system of $n + 1$ linear equations having $n + 1$ unknowns, a_j.

Convince yourself that everything but the a_i's in this system represents known *numbers* in a concrete case, even if it all looks like unknown letters in its current form!

When we study the linear system above in the next chapter, we will determine the property of the coefficients x_i^j which correspond to the unique solvability of the system for all right hand sides $\vec{y}$.

A general fact about linear systems which we will examine in detail, and which is easy to understand in this example, suffices here. The solution to system with a general right hand side y_j is easy to find if we know the solutions for the right hand sides $\mathbf{e_i} = \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix}$ with the 1 in the ith row.

Phrased differently, if we know the polynomials of degree n, $l_i(x)$ such that $l_i(x_j) = 0, j \neq i$ and $l_i(x_i) = 1$ for each i then quite simply

$$p(x) = y_0 l_0(x) + \ldots + y_n l_n(x)$$

is the polynomial of degree n having $p(x_i) = y_i$.

$\triangleright$ Exercise 1.16 Check this fact, and draw the corresponding graph for three points and quadratic functions. Express this result in terms of matrices and their inverses.

Be aware that the simplifying notation $l_i(x)$ disguises the dependence upon $x_0, \ldots, x_n$.

The Lagrange polynomials would be nice enough from this property, but there is an independent fact that we do not have to solve the corresponding systems to find them, we may just write them down. From the fact $l_i(x_j) = 0$, and their degree, we know $l_i(x) = c(x - x_0) \ldots (x - x_{i-1})(x - x_{i+1}) \ldots (x - x_n)$ The constant c is determined by $l_i(x_i) = 1$, so

$$c = \frac{1}{(x_i - x_0) \ldots (x_i - x_{i-1})(x_i - x_{i+1}) \ldots (x_i - x_n)}$$

and

$$l_i(x) = \frac{(x - x_0) \ldots (x - x_{i-1})(x - x_{i+1}) \ldots (x - x_n)}{(x_i - x_0) \ldots (x_i - x_{i-1})(x_i - x_{i+1}) \ldots (x_i - x_n)}$$

$\triangleright$ Exercise 1.17 Observe how the coefficients of the Lagrange polynomials provide the inverse of matrix of coefficients $\{x_i^j\}$. Compare the efficiency of this method (include the multiplications to find the coefficients) and standard inversion procedures.

To find the corresponding a_j's we would only have to multiply this out, but this form is more suggestive. Going from standard form to Lagrange polynomial form is even more convenient.

▷ Exercise 1.18 Given a polynomial in standard form, show that it may be written $p(x) = \sum_0^n p(x_i)l_i(x)$. Does it matter that $p(x)$ is in standard form?

The coefficients in a Lagrange polynomial expansion are simply the values of the function at the points corresponding to the 1 for each polynomial. We will return to this.

▷ Exercise 1.19 For a concrete choice of x_i's, demonstrate the general corollary that $1 = \sum_0^n l_i(x)$ and that $x^k = \sum_0^n x_i^k l_i(x)$ for each $k \le n$.

The Lagrange polynomials provide an explicit formula for the coefficients of the Newton interpolation polynomials. Since a_n is the leading coefficient of $p_n(x)$, it must also be the leading coefficient of $p_n(x)$, when it is written as a sum of Lagrange polynomials. Because the the x^n coefficient of each Lagrange polynomial is just the denominator in the expression above, (the coefficient of x^n in the numerator is 1), we obtain

$$a_n = \sum_{i=0}^{n} \frac{f(x_i)}{(x_i - x_0)\dots(x_i - x_{i-1})(x_i - x_{i+1})\dots(x_i - x_n)}$$

The notation which indicates the dependence of this coefficient upon $x_0, \dots, x_n$ and the values of f at those points is $f[x_0, \dots, x_n] \equiv a_n$. That is, $f[x_0, \dots, x_n]$. is the leading coefficient of the unique polynomial $p_n(x)$ of degree n having $p_n(x_i) = f(x_i)$ for $i = 0, \dots, n$, and is given by either of the above expressions for a_n.

▷ Exercise 1.20 Use the formula to show that $f[x_0, \dots, x_n]$ doesn't depend upon the order of the x_i's, i.e. it is invariant under permutation of its arguments.

▷ Answer 1.20 We know this is true, since it is defined as the leading coefficient of $p_n(x)$, and $p_n(x)$ doesn't depend upon the order of the interpolation points. The explicit formula is just permuted.

▷ Exercise 1.21 Demonstrate that $f[x_0, \dots, x_n] = 0$ if f is a polynomial of degree $< n$, nd that $f[x_0, \dots, x_n]$ is independent of its arguments if f is a polynomial of degree n.

The original expression for a_j identified $f[x_0, \dots, x_n] = \frac{u - f[x_0, \dots, x_{n-1}]}{(x_{n-1} - x_n)}$ and we will discover now that more generally,

$$f[x_0, \dots, x_n] = \frac{f[x_0, \dots, \hat{x}_i, \dots, x_n] - f[x_0, \dots, \hat{x}_j, \dots, x_n]}{(x_j - x_i)}$$

where $x_0, \dots, \hat{x}_i, \dots, x_n$ indicates x_i has been omitted.

▷ Exercise 1.22 Assuming this is true, solve for u. Explain the result from its earlier definition as $\tilde{p}_{n-1}(x_n)$.

▷ Exercise 1.23 Put terms of $f[x_0,\ldots,\hat{x_i},\ldots,x_n]-f[x_0,\ldots,\hat{x_j},\ldots,x_n]$ involving $f(x_k)$ in the Lagrange expression above over a common denominator to prove the divided difference formula.

▷ Answer 1.23 The numerator is then $f(x_k)((x_k-x_i)-(x_k-x_i)) = f(x_k)(x_j-x_i)$. Check the special cases for $f(x_i), f(x_j)$.

An alternative confirmation of this formula returns to the meaning of the expressions involved. If we wish to find a polynomial of degree n which interpolates $f(x)$ at $x_0,\ldots,x_n$, we might generalize our first approach by imagining a correction formed out of $p_{\hat{i}}(x)$ and $p_{\hat{j}}(x)$ of degree $n-1$ which respectively interpolate $f(x)$ at $x_0,\ldots,\hat{x_i},\ldots,x_n$ and $x_0,\ldots,\hat{x_j},\ldots,x_n$ where again the hat (caret) ˆ indicates omission. Since we know nothing about $p_{\hat{i}}(x)$ at x_i we multiply it by $(x-x_i)$ to let $p_{\hat{j}}(x)$ take care of things at x_i, and symmetrically, we should write $(x-x_j)p_{\hat{j}}(x)$. At $x=x_i$ this factor introduces a factor of (x_i-x_j) which we must divide out in order not to alter the correct value of $p_{\hat{j}}(x_i)$. With the corresponding correction for $p_{\hat{i}}$ we are led the polynomial of degree n

$$p_n(x) = \frac{(x-x_i)}{(x_j-x_i)}p_{\hat{i}}(x) + \frac{(x-x_j)}{(x_i-x_j)}p_{\hat{j}}(x)$$

which we confirm interpolates $f(x)$ at $x_0,\ldots,x_n$,

$$p_n(x_k) = \frac{(x_k-x_i)}{(x_j-x_i)}p_{\hat{i}}(x_k) + \frac{(x_k-x_j)}{(x_i-x_j)}p_{\hat{j}}(x_k) = f(x_k)\frac{x_j-x_k+x_k-x_i}{x_j-x_i} = f(x_k)$$

Equating their leading coefficients gives the divided difference formula.

▷ Exercise 1.24 Show how to use the divided difference formula in multiple ways to efficiently compute the interpolation polynomial using a divided difference table of the following form. Do we need to save the entire table to compute the polynomial of one degree higher interpolating $f(x)$ at one more point?

▷ Answer 1.24 From the new value $f(x_n)$ and the known divided differences $f[x_{n-1}], f[x_{n-2},x_{n-1}],\ldots,f[x_0,x_1,\ldots,x_{n-2},x_{n-1}]$, use the formula to inductively compute

$$f[x_n], f[x_{n-1},x_n],\ldots,f[x_{n-2},x_{n-1},x_n],\ldots,f[x_0,x_1,\ldots,x_{n-1},x_n]$$

This is Neville's method. Alternatively, from $f(x_n)$ and the known divided differences $f[x_0], f[x_0, x_1], \ldots, f[x_0, x_1, \ldots, x_{n-2}, x_{n-1}]$, use the formula to inductively compute

$$f[x_n], f[x_0, x_n], \ldots, f[x_0, x_1, x_n], \ldots, f[x_0, x_1, \ldots, x_{n-1}, x_n]$$

This is Aitken's method. The storage for the divided differences of the previous stage may be reused at the appropriate time. The differences $x_i - x_j$ must be computed as well.

▷ Exercise 1.25 Obtain simplified expressions for the coefficients of an interpolation polynomial with equally spaced nodes from the special forms of the $(x_i - x_j)$ and $(x - x_i)$ and their products, yielding the forward, backward, and centered differencing formulas and their errors.

▷ Exercise 1.26 If $p_n(x)$ is a polynomial of degree n, show that $p_n[x_0, \ldots, x_k]$ is a polynomial of degree $n - k$. Hint: Start with $k = 1$.

▷ Exercise 1.27 Use the divided difference formula to show that

$$f[x_0, \ldots, x_n, x+h, x] = \frac{f[x_0, \ldots, x_n, x+h] - f[x_0, \ldots, x_n, x]}{h}$$

Take the limit as $h \to 0$ to find an expression for $\frac{d}{dx} f[x_0, \ldots, x_n, x]$ This limit will be justified later.

Just as the coefficients for a polynomial in Taylor form are given by its derivatives at a point, the coefficients of the interpolation polynomial in multiple center form may be interpreted as "spread out derivatives" as we will see in the next section.

▷ Exercise 1.28 Write out the first and second divided differences explicitly and give their geometric meaning in terms of secant slopes.

1.4 The error of interpolation.

We wish to understand how accurately the interpolating polynomial approximates the values of $f(x)$, in other words, we want to estimate $|f(t) - p_n(t)|$. In analyzing the Taylor polynomial one identifies the form of the error as like the next term in the expansion by "going one term further" (see Calculus appendix). The same approach will work here.

To identify $f(t) - p_n(t)$ just add one more interpolation point, at t. By defining

$$p_{n+1}(x) = p_n(x) + f[x_0, \ldots, x_n, t](x - x_0) \ldots (x - x_n)$$

we succeed because the error $f(t) - p_{n+1}(t)$ is now zero! It might sound like cheating, but look what it tells us.

$$0 = f(t) - p_n(t) - f[x_0, \ldots, x_n, t](t - x_0) \ldots (t - x_n)$$

or

$$E_n(t) \equiv f(t) - p_n(t) = f[x_0, \ldots, x_n, t](t - x_0) \ldots (t - x_n)$$

and the error has the same form as the next term of the expansion, with a coefficient which is no longer constant. All of the non-polynomial behavior is contained in that coefficient, which must become zero if f is itself a polynomial and we carry out the expansion far enough.

We have many hints that, as for the Taylor series, we may estimate that coefficient by the derivatives of f. The identical tool, Rolle's theorem is responsible for *this* global statement.

▷ Exercise 1.29 Generalized Rolle's Theorem: If $f \in C^{k+1}([a, b])$ and f has $k + 2$ zeroes on $[a, b]$, then $f^{(i)}$ has at least $k + 2 - i$ zeroes on $[a, b]$, $i = 1, \ldots, k + 1$.

▷ Answer 1.29 By Rolle's theorem, f' has a zero between every pair of consecutive zeroes of f. There are $k + 1$ such pairs, giving $k + 1$ zeroes of $f' \in C^k$. Apply Rolle's theorem to f' to guarantee k zeroes of $f'' \in C^{k-1}$, and so forth down to show $f^{k+1}(\xi) = 0$ for $\xi \in [a, b]$.

Estimation of divided differences by derivatives.

We consider $E_{n+1}(x) \equiv f(x) - p_{n+1}(x)$ which vanishes at the $n+2$ points $x_0, \ldots, x_n, t$. Let $I = [a, b]$ be an interval containing $x_0, \ldots, x_n, t$, (which is not necessarily $[x_0, x_n]$) Assuming $f(x) \in C^{n+1}([a, b])$, then so is $E_{n+1}(x)$, so there is a $\xi \in [a, b]$ where $E_{n+1}^{(n+1)}(\xi) = 0$. We can write $E_{n+1}^{(n+1)}(x) = f^{(n+1)}(x) - p_{n+1}^{(n+1)}(x)$ and use the fact that the $n + 1$st derivative of a polynomial of degree $n + 1$ is just $(n + 1)!$ times its leading coefficient a_{n+1}. Since $p_{n+1}(x)$ is a polynomial of degree n plus $f[x_0, \ldots, x_n, t](x - x_0) \ldots (x - x_n)$, its leading coefficient must be $f[x_0, \ldots, x_n, t]$. Combining these observations, the punchline is $0 = f^{(n+1)}(\xi) - (n + 1)! f[x_0, \ldots, x_n, t]$ or

$$f[x_0, \ldots, x_n, t] = \frac{f^{(n+1)}(\xi)}{(n + 1)!}$$

where $\xi \in$ an interval containing $x_0, \ldots, x_n, t$.

Substituting in the earlier expression, we have

$$E_n(t) \equiv f(t) - p_n(t) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(t - x_0) \ldots (t - x_n)$$

.

▷ Exercise 1.30 Estimate the polynomial term in the error formula $\Psi(x) \equiv (x-x_0) \ldots (x-x_n)$ for equally spaced points on a fixed interval I. Try some small values of n to begin with. Why does this suggest that extrapolation might be a bad idea. Is there a function $f(x)$ which attains the maximum allowable error?

▷ Exercise 1.31 Observe numerically the divergence of interpolation at equally spaced points on $[-5, 5]$ for $f(x) = (1 + x^2)^{-1}$ as $n \to \infty$ for $|x| > 3.63$? . How do its coefficients in the estimate grow? While one is sometimes forced to interpolate at equally spaced points, for instance with tabulated data on a uniform grid, and the formulas simplify, this example, due to Runge, points out some of the weaknesses of interpolation at equally spaced points. We will return to look for 'optimally spaced' interpolation nodes in a later chapter.

▷ Exercise 1.32 Derive the explicit formula for interpolating this f at equally spaced points, using

$$f[x_0, \ldots, x_n, x] = f(x) \cdot \frac{(-1)^{r+1}}{\prod_{j=0}^{r}(1 + x_j^2)} \cdot \left\{ \begin{array}{l} 1, \text{ if } n = 2r + 1 \\ x, \text{ if } n = 2r \end{array} \right\}$$

Use induction or

$$\frac{1}{1 + x^2} = \frac{1}{2i}\left(\frac{1}{x - i} - \frac{1}{x + i}\right)$$

and for $g(x) = 1/(x + c)$,

$$g[x_0, \ldots, x_n] = (-1)^n \frac{1}{\prod_{j=0}^{n}(x_j + c)}$$

Then show

$$f(x) - p_n(x) = (-1)^{r+1} \cdot f(x) g_n(x), \qquad g_n(x) \equiv \prod_{j=0}^{r} \frac{x^2 - x_j^2}{1 + x_j^2}$$

▷ Exercise 1.33 Demonstrate that $|g_n(x)| \to \infty$ as $n \to \infty$ for $|x| > 3.63\ldots$. Hint: Write $|g_n(x)| = [e^{\delta x \ln |g_n(x)|}]^{\frac{1}{\delta x}}$ to change the product into a Riemann sum converging to an integral which may be analyzed. See Issacson and Keller.

▷ Exercise 1.34 Do the restrictions on the real radius of convergence of the Taylor series imposed by singular behavior or f in the complex plane carry over to convergence of sequences of interpolation polynomials at equally spaced points?

▷ Answer 1.34 Yes, circles of convergence are replaced by lemniscates, which in the case of the singularities $a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \ldots + a_n(x - x_0)\cdots(x - x_{n-1})i$ of $f(z) = \frac{1}{1+z^2}$ intersect the real axis at $a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \ldots + a_n(x - x_0)\cdots(x - x_{n-1})3.63\ldots$. See Issacson and Keller.

▷ Exercise 1.35 Use the estimate upon the divided differences which appeared in the error analysis for the secant method.

▷ Answer 1.35

$$\frac{f[x_{n-1}, x_n, r]}{f[x_{n-1}, x_n]} = \frac{f''(\xi)}{2f'(\eta)}$$

▷ Exercise 1.36 Preview: Choose x_0, x_1, x_2, x_3 to minimize the cubic polynomial $\Psi(x) = (x - x_0)(x - x_1)(x - x_2)(x - x_3)$ on the interval $[-1, 1]$ in the following senses:

a) The maximum absolute value attained by $\Psi(x)$

b) The value of $\int_{-1}^{1} |\Psi(x)|dx$

c) The value of $(\int_{-1}^{1} \Psi^2(x)dx)^{\frac{1}{2}}$

▷ Exercise 1.37 Give rigorous proofs for the existence and uniqueness of the interpolating polynomial of degree n for $n + 1$ distinct points, making use of Lagrange polynomials, Newton's form, and the solution theory of linear systems of equations.

1.5 Osculatory interpolation

Osculatory interpolation is the bridge between the interpolation polynomials and the

Taylor polynomials. We have given an efficient way to interpolate by polynomials and incrementally increase the degree to allow the addition of an interpolation point. We have estimated the error with respect to the placement of the interpolation points and the magnitude of the derivatives of the function being interpolated. There have been many side benefits of this study, and it connects our earlier investigation of rootfinding to our upcoming investigation of linear systems. The results we are obtaining will be useful in future studies, for instance in analyzing the accuracy of interpolation based numerical quadrature, differentiation, and differential equation formulas.

The formal transition from Newton form to Taylor form of a polynomial when its centers coincide suggests the final issue we will consider, the coalescence of two or more interpolation nodes. To develop an intuition we might consider the simplest case of just two points coming together. Geometrically, the interpolating secant becomes the tangent line. The divided difference coefficient $f[x_0, x_1] \to f'(x_0)$ (if it exists) as $x_1 \to x_0$, by definition. Write out the Lagrange polynomials which become singular, and the linear system of equations defining the problem which also becomes singular. An equivalent system of equations which does not become singular is

$$\begin{pmatrix} 1 & x_0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f[x_0, x_1] \end{pmatrix}$$

which has no singularity as $x_0 \to x_1$, nor do the 'Lagrange polynomials' as defined by the special solutions of this system. We will construct a framework which includes the interpolation polynomials and eror estimates regardless of the multiplicity of nodes. In the simple example above, observe that two simple zeroes of $f(x) - p_1(x)$ are coalescing into one zero of multiplicity two. This formulation will work in genaral.

While the details of these constructions and the corresponding error estimates are more involved when more points are involved and are coinciding, the essence is the same, with forms intermediate to distinct interpolation and Taylor, to divided differences and derivatives, and having intermediate desingularized linear systems and 'Lagrange polynomials'. The key term here is desingularized, in that we are trying to find a continuous extension of a related set of formulas which are currently undefined on a thin but important set of data. This is the same procedure by which we fill in the undefined difference quotient with the 'derivative' when $h \to 0$. The topic is known as osculatory interpolation (etymology: kissing) or generalized Hermite interpolation for reasons that will become clear shortly.

▷ Exercise 1.38 Experiment with the cases of quadratic interpolation with three nodes which coalesce simultaneously, or with a pair of nodes which coalesce first.

▷ Answer 1.38 It is instructive to write the system of equations for interpolation in a equivalent forms (obtained by elementary row operations, thus having the same solution) which, for f sufficiently smooth, have well defined limits as points coalesce.

$$\begin{pmatrix} 1 & x_0 & x_0^2 \\ 0 & 1 & x_0 + x_1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f[x_0, x_1] \\ f[x_0, x_1, x_2] \end{pmatrix}$$

Observe the equivalence of the original system and a system which has the divideded difference as its leading coefficient.

When $x_1 \to x_0$ this becomes

$$\begin{pmatrix} 1 & x_0 & x_0^2 \\ 0 & 1 & 2x_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f'(x_0) \\ \frac{f[x_0,x_2]-f'(x_0)}{(x_2-x_0)} \end{pmatrix}$$

When all three nodes coincide, the limiting behavior of the derivative estimate for the divided difference suggests convergence to

$$\begin{pmatrix} 1 & x_0 & x_0^2 \\ 0 & 1 & 2x_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f'(x_0) \\ \frac{f''(x_0)}{2} \end{pmatrix}$$

▷ Exercise 1.39 Describe the row operations performed above in terms of divided differences.

▷ Exercise 1.40 Formulate a correspondence principle which is illustrated by the analogies between interpolation and Taylor polynomials.

▷ Answer 1.40 When two constructions agree in some limit, related formulas and theorems may be formulated and perhaps proven in a unified manner which reduce to the original formulations in the appropriate regime.

With this background, we proceed to reformulate the interpolation problem, and the definition of divided differences in a continuous manner which is in agreement with the earlier definitions when the nodes are distinct, and which is well defined for nondistinct nodes as well.

▷ Exercise 1.41 Use the explicit formula for divided differences to describe their continuity and differentiability as a function of (distinct) arguments, in terms of the degree of smoothness of f.

If we define the interpolation polynomial of $f(x)$ at the (perhaps nondistinct) points $x_0, \ldots, x_0, \ldots, x_n, \ldots, x_n$ with x_i occuring $m_i \geq 1$ times, by the criterion that $f(x) - p(x)$ has a root of multiplicity m_i at x_i, i.e., $(f-p)^{(k)}(x_i) = 0$ for $k = 0, \ldots, m_i - 1$, and p has degree $\leq d$, with $d + 1 = \sum_{i=0}^{n} m_i$. Sorting this out, the osculatory interpolation problem may also be formulated by giving values $p^{(k)}(x_i) = y_{i,k}$ without reference to a particular function f. We have agreement with the definition of interpolation and Taylor polynomials, the extreme cases of d distinct roots and one root of multiplicity d, respectively. There are numerous intermediate cases.

▷ Exercise 1.42 Enumerate the distinct possibilities.

▷ Exercise 1.43 Generalize the proof of uniqueness of interpolating polynomials which uses the lemma that a polynomial of degree n with $n+1$ distinct zeroes is identically zero to nondistinct nodes.

▷ Exercise 1.44 To understand why we do not use the condition of equality of order m_i at x_i: $\left|\frac{f(x)-p(x)}{(x-a)^n}\right| \to 0$ as $x \to a$, consider the pathological case of

$$\begin{aligned} f(x) &= x^n \text{ for } x \text{ irrational} \\ &= 0 \text{ for } x \text{ rational} \end{aligned}$$

which has equality of order n to zero at $x = 0$, yet f' does not exist for $x \neq 0$ and f" does not exist anywhere. For sufficiently smooth f however, equality of order n at a is equivalent to requiring $f^{(k)}(a) = p^{(k)}(a) \quad i = 0, \ldots, n$. Prove this.

We will examine the central topics of interpolation in light of this more general formulation. These are, the formulation of the problem as a uniquely solveable linear system of equations, the definition of the Lagrange polynomials as solutions for special right hand sides, the direct construction of the Lagrange polynomials from the data, the construction of the Newton form by successively adding nodes and conditions, the divided difference formula for the coefficients in Newton form, and the error formulas in divided difference and derivative forms.

Aside from instructional value, the most important intermediate case is when each root has multiplicity two, or equivalently the value of f and f' are interpolated at each point. This case is common enough in applications to have a name, Hermite interpolation. We will emphasize as examples this case in which pairs of points coalesce, and the general cubic case, which contains the seed of the procedures for general n.

▷ Exercise 1.45 In what sense is the interpolation matrix $\begin{pmatrix} 1 & x_o \\ 1 & x_1 \end{pmatrix}$ related to the matrix for the Taylor polynomial, $\begin{pmatrix} 1 & x_o \\ 0 & 1 \end{pmatrix}$?

▷ Exercise 1.46 Describe the linear syystem of equations solved by the coefficients of the cubic polynomial which interpolates f at

a) x_0, x_1, x_2, x_3

b) x_0, x_0, x_1, x_2

c) x_0, x_0, x_1, x_1

d) x_0, x_0, x_0, x_1

e) x_0, x_0, x_0, x_0

Compare the coalescence of the simple root to the triple root, with the coalescence of the two double roots.

Do the same for the degree $2n + 1$ Hermite interpolation at $x_0, \ldots, x_n$.

▷ Exercise 1.47 Let $\vec{f}(t) = \begin{pmatrix} 1 \\ t \\ t^2 \\ t^3 \end{pmatrix}$. Write the rows of the above matrices in terms of f and its derivatives at x_i. Using the chain rule and the linearity of the determinant as a multi-linear, antisymmetric function of each row of a matrix, determine the determinants of the matrices above. You may also use that the determinant is a homogeneous polynomial of degree n in its entries. (See Linear Algebra appendix for more on determinants.)

▷ Exercise 1.48 Show that the Hermite polynomial for the point x_0 is the tangent line for f at x_0.

The nonsingular linear systems which define the solution to the osculatory interpolation problem are well defined limits of systems that are equivalent to the system for interpolation at distinct nodes as they coalesce. The Lagrange polynomials are connected to the particular form of the system, not intrinsically to the system. As two points coalesce, slopes of corresponding Lagrange polynomials become infinite in between. The appropriate Lagrange polynomials for the osculatory problem are again the special solutions of the corresponding system for right hand sides $\mathbf{e_i}$. We can see how they may be obtained in a manner similar to the usual Lagrange polynomials by the following procedure. Note that this gives an inverse for the matrix for the osculatory interpolation system. The polynomials

$$p_{x_k,m_k}(x) = \prod_{k \neq i} (x - x_i)^{m_i} (x - x_k)^{m_k - 1}$$

satisfy $p^{(j)}_{x_k,m}(x_i) = 0$ for $0 \leq j \leq m_i$ and $p^{(j)}_{x_k,m}(x_k) = \delta^m_j$ for $0 \leq j \leq m_k$. for $m = m_k$. Inductively, we construct $p_{x_k,m_k-l}(x) \quad l = 0, \ldots, m_k$ starting from

$$\tilde{p}_{x_k,m_k-l}(x) = \prod_{k \neq i} (x - x_i)^{m_i} (x - x_k)^{m_k - l - 1}$$

and use the $p_{x_k,m_k-l}(x)$ from previous l's to correct the nonzero higher derivatives of $\tilde{p}_{x_k,m_k-l}(x)$. The final form is

$$p_{x_k,m_k-l}(x) = \prod_{k \neq i} (x - x_i)^{m_i} (x - x_k)^{m_k - l - 1} q(x)$$

where $q(x)$ is a polynomial of degree l.

▷ Exercise 1.49 Write the Lagrange polynomials for the distinct points x_i in terms of the function $\Psi(x) = \prod (x - x_i)$ and $\Psi'(x)$. Confirm the form of the Hermite polynomials

$$H_{x_k}(x) = \alpha_k + \beta_k (x - x_k) \prod_{k \neq i} (x - x_i)^2$$

$$\hat{H}_{x_k}(x) = \gamma_k \prod_{k \neq i} (x - x_i)^2 (x - x_k)$$

and find the coefficients in a standard form involving the Lagrange polynomials.

▷ Exercise 1.50 The iterated product rule is useful in the constructions above and below. Show that $fg^{(n)} = \sum_{k=0}^{n} \binom{n}{k} f^{(n-k)} g^{(k)}$.

We again define the divided difference $f[x_0, \ldots, x_0, \ldots, x_n, \ldots, x_n]$ in the case of perhaps nondistinct points to be the leading coefficient of the unique osculatory interpolating polynomial for these points. You should confirm that the uniqueness goes through as easily as the existence.

▷ Exercise 1.51 Derive an explicit formula for the divided difference with nondistinct nodes. The osculatory interpolating polynomial is still independent of the order in which we list the nodes. Is this visible from the formula?

The proof of the divided difference formula also goes through with modifications. When the omitted points x_i and $x_j$are distinct,

$$p_n(x) = \frac{(x - x_i)}{(x_j - x_i)} p_{\hat{i}}(x) + \frac{(x - x_j)}{(x_i - x_j)} p_{\hat{j}}(x)$$

still interpolates $f(x)$ at $x_0, \ldots, x_n$. By the iterated product rule, for x_k distinct from x_i or x_j, and $0 \le l \le m_k - 1$, then

$$\begin{aligned} p_n^{(l)}(x_k) &= \frac{(x_k - x_i)}{(x_j - x_i)} p_{\hat{i}}^{(l)}(x_k) + \frac{(x_k - x_j)}{(x_i - x_j)} p_{\hat{j}}^{(l)}(x_k) \\ &\quad + l \frac{1}{(x_j - x_i)} p_{\hat{i}}^{(l-1)}(x_k) + l \frac{1}{(x_i - x_j)} p_{\hat{j}}^{(l-1)}(x_k) = f^{(l)}(x_k) \end{aligned}$$

▷ Exercise 1.52 Check that $p_n^{(l)}(x_i) = f^{(l)}(x_i)$ for $0 \le l \le m_i - 1$.

For all nodes equal to each other, we know that the Taylor polynomial with leading coefficient

$$f[x_0, \ldots, x_0] = \frac{f^{(k)}(x_0)}{k!}$$

is the osculatory interpolating polynomial. These two cases suffice to construct arbitrary divided differences by the Neville algorithm, according to our ordering common nodes consecutively. By computing all divided differences involving only one repeated node using Taylor coefficients, and the divided difference formula for the intermediate cases, we can extend our efficient algorithm for adding a node to nondistinct nodes.

For instance, to compute the polynomial which interpolates $f(x)$ at x_0, x_0, x_0, x_1, x_1, we compute

x_0	$f(x_0)$	$f[x_0, x_0] = f'(x_0)$	$f[x_0, x_0, x_0] = \frac{f''(x_0)}{2}$	$f[x_0, x_0, x_0, x_1]$	$f[x_0, x_0, x_0, x_1, x_1]$
x_0	$f(x_0)$	$f[x_0, x_0] = f'(x_0)$	$f[x_0, x_0, x_1]$	$f[x_0, x_0, x_1, x_1]$	
x_0	$f(x_0)$	$f[x_0, x_1]$	$f[x_0, x_1, x_1]$		
x_1	$f(x_1)$	$f[x_1, x_1] = f'(x_1)$			
x_1	$f(x_1)$				

where, if no derivative is given, the divided difference may be obtained from the formula from the two neighboring coefficients to the left by omission of distinct points. The osculatory interpolating polynomial will be

$$\begin{aligned} p(x) &= f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2}(x - x_0)^2 \\ &\quad + f[x_0, x_0, x_0, x_1](x - x_0)^3 + f[x_0, x_0, x_0, x_1, x_1](x - x_0)^3(x - x_1) \end{aligned}$$

▷ Exercise 1.53 Confirm the argument that the multiple centers form is still valid when the nodes are not distinct. The leading coefficient is clearly correct, and inductively, the addition of each new (possibly repeated) node does not upset the conditions which are met at the previous stages.

The final issue we must consider is the error of the osculatory interpolating polynomial, and the corresponding estimates for the divided differences in terms of derivatives. The error $f(x) - p(x)$ is still given by the term needed to interpolate f at x, in addition to $x_0, \dots, x_n$, thus $f(x) - p(x) = f[x_0, \dots, x_n, x](x - x_0) \cdots (x - x_n)$ holds regardless of the repetition of the arguments.

We approach the estimates of osculatory divided differences by continuity and induction, based upon the following observations. If $f(x)$ is continuous, then $f[x_0] \to f[y_0]$ as $x_0 \to y_0$, so inductively $f[x_0, x_1] = \frac{f[x_1]-f[x_0]}{x_1-x_0} \to \frac{f[y_1]-f[y_0]}{y_1-y_0} \to f[y_0, y_1]$ as $x_i \to y_i$, for distinct y_0, y_1 (which implies x_0, x_1 are eventually distinct so the formula is valid). For nondistinct y_0, y_1 we notice that the estimate

$$f[x_0, \dots, x_n] = \frac{f^{(n)}(\xi)}{n!}$$

for $\xi \in [\min x_i, \max x_i]$ is valid in general, even when the x_i's are nondistinct (see below), and use

$$f[y_0, y_0] = f'(y_0) = \lim_{\substack{x_i \to y_i \\ \xi \in [\min x_i, \max x_i]}} f'(\xi) = \lim_{x_i \to y_i} f[x_0, x_1]$$

as $x_0, x_1 \to y_0$.

Assuming f is sufficiently continuously differentiable, we may procede by induction as follows to obtain continuity for divided differences of arbitrary order, and with nondistinct points. When at least two of the $n + 1$ nodes y_i are distinct, we apply the divided difference formula and the induction hypothesis, (which must necessarily include the case of n identical nodes). That case is handled as above by

$$f[y_0, \dots, y_0] = \frac{f^{(n)}(y_0)}{n!} = \lim_{\substack{x_i \to y_i \\ \xi \in [\min x_i, \max x_i]}} \frac{f^{(n)}(\xi)}{n!} = \lim_{x_i \to y_i} f[x_0, \dots, x_n]$$

for $x_0, \dots, x_n \to y_0$.

To see the last of these equalities, the divided difference estimate for nondistinct nodes x_i, apply the analogous estimate to a set of *distinct* nodes which approximate the x_i's. When at least two of the $x_i' s$ are distinct, we may apply the continuity just proven for $n + 1$ nodes of this type. The case of the x_i's all identical has already been verified.

▷ Exercise 1.54 Check the details of this argument and the formalities of the induction. In particular, check the continuity of the $f(\xi(x_i))$ as the x_i's vary.

▷ Exercise 1.55 Derive the estimate for osculatory divided differences using Rolle's theorem analogously to the proof for distinct points. Hint: In the Hermite case, let $g(x) =$

$f(x)-H(x)-f[x_0,x_0,\ldots,x_n,x_n,t](x-x_0)^2\ldots(x-x_n)^2$ and observe that $g'(x)$ has $2n+2$ simple zeroes.

▷ Exercise 1.56 Derive an integral formula for distinct node divided differences by induction, and derive continuity properties for coalescing nodes from this representation. The formula is

$$f[x_0,\ldots,x_n]=\int_{\substack{t_i\geq 0\\ \sum_0^n t_i=1}} f^{(n)}(t_0x_0+\ldots+t_nx_n)d\vec{t}$$

In the final exercises we will apply the continuity properties for divided differences with repeated arguments to justify and generalize the error estimates for and derivatives of divided differences. Recall that $f[x_0,\ldots,x_n,x+h,x]=\frac{f[x_0,\ldots,x_n,x+h]-f[x_0,\ldots,x_n,x]}{h}$ and so $\frac{d}{dx}f[x_0,\ldots,x_n,x]=f[x_0,\ldots,x_n,x,x]$ is now justified. More generally if we let

$$g(x)=f[x_0,\ldots,x_n,x]$$

we can show by induction and continuity that

$$g[y_0,\ldots,y_m]=f[x_0,\ldots,x_n,y_0,\ldots,y_m]$$

Then

$$g^{(n)}(x)=n!f[x_0,\ldots,x_n,x,\ldots,x]$$

with the x repeated $n+1$ times.

▷ Exercise 1.57 Carry out the details of these statements.

▷ Exercise 1.58 Prove the following two consequences of the previous exercise: If f is m times continuously differentiable, then

$$f[x_0,\ldots,x_p,y_0,\ldots,y_q,z_0,\ldots,z_r]=\frac{1}{p!}\frac{1}{q!}\frac{1}{r!}\frac{\partial^p}{\partial x^p}\frac{\partial^q}{\partial y^q}\frac{\partial^r}{\partial z^r}f[x,y,z]\bigg|_{\xi,\eta,\zeta}$$

with the x's,y's,z's mutually (but not necessarily independently) distinct, and

$$\min_i x_i\leq\xi\leq\max_i x_i$$
$$\min_i y_i\leq\eta\leq\max_i y_i$$
$$\min_i z_i\leq\zeta\leq\max_i z_i$$

and its corollary

$$f[x,\ldots,x,y,\ldots,y,z,\ldots,z]=\frac{1}{p!}\frac{1}{q!}\frac{1}{r!}\frac{\partial^p}{\partial x^p}\frac{\partial^q}{\partial y^q}\frac{\partial^r}{\partial z^r}f[x,y,z]$$

with x, y, z distinct.

▷ Exercise 1.59 Prove the product rule for divided differences of $f(x) = g(x)h(x)$, which generalizes the Leibniz rule from calculus.

$$f[x_0, \ldots, x_n] = \sum_{i=0}^{n} g[x_0, \ldots, x_i]h[x_i, \ldots, x_n]$$

▷ Exercise 1.60 Relate the notions of osculating quadratic of a function to the osculating circle to a curve. Does either depend upon the axes used to describe the curve.