5.7 Numerical Differentiation

Numerical approximations to derivatives are used mainly in two ways. First, we are interested in calculating derivatives of given data that is often obtained empirically. Second, numerical differentiation formulas are used in deriving numerical methods for solving ordinary and partial differential equations. We begin this section by deriving some of the most commonly used formulas for numerical differentiation.

The problem of numerical differentiation is in some ways more difficult than that of numerical integration. When using empirically determined function values, the error in these values will usually lead to instability in the numerical differentiation of the function. In contrast, numerical integration is stable when faced with such errors; see problem 13. The instability of the classical numerical differentiation formulas is analyzed later in the section. Some recent literature for improved, and stabilized, numerical differentiation procedures is then reviewed.

The classical formulas One of the main approaches to deriving a numerical approximation to $f'(x)$ is to use the derivative of a polynomial $p_n(x)$ that interpolates $f(x)$ at a given set of node points. Let $x_0, x_1, \ldots, x_n$ be given, and let $p_n(x)$ interpolate $f(x)$ at these nodes. Usually $\{x_i\}$ are evenly spaced. Then use

$$f'(x) \doteq p_n'(x) \tag{5.7.1}$$

From (3.1.6), (3.2.4), and (3.2.11):

$$p_n(x)=\sum_{j=0}^{n} f(x_j)\ell_j(x)$$

$$\ell_j(x) = \frac{\Psi_n(x)}{(x-x_j)\Psi_n'(x_j)} = \frac{(x-x_0)\dots(x-x_{j-1})(x-x_{j+1})\dots(x-x_n)}{(x_j-x_0)\dots(x_j-x_{j-1})(x_j-x_{j+1})\dots(x_j-x_n)}$$

$$\Psi_n(x) = (x-x_0)\dots(x-x_n) \tag{5.7.2}$$

$$f(x)-p_n(x) = \Psi_n(x)f[x_0,\dots,x_n,x]$$

Thus

$$f'(x)\doteq p_n'(x) = \sum_{j=0}^{n} f(x_j)\ell_j'(x)\equiv D_h f(x) \tag{5.7.3}$$

$$f'(x)-D_h f(x)=\Psi_n'(x)f[x_0,\dots,x_n,x] + \Psi_n(x)f[x_0,\dots,x_n,x,x], \tag{5.7.4}$$

with the last step using (3.2.17). Applying (3.2.12),

$$f'(x)-D_h f(x) = \Psi_n'(x)\frac{f^{(n+1)}(\xi_1)}{(n+1)!} + \Psi_n(x)\frac{f^{(n+2)}(\xi_2)}{(n+2)!} \tag{5.7.5}$$

with $\xi_1,\xi_2\in\mathscr{H}\{x_0,\dots,x_n,x\}$. Higher order differentiation formulas and their error can be obtained by further differentiation of (5.7.3) and (5.7.4).

The most common application of the above is to evenly spaced nodes $\{x_i\}$. Thus let

$$x_i=x_0+ih, \quad i\geq 0$$

with $h>0$. In this case, it is straightforward to show that

$$\Psi_n(x)=O(h^{n+1}), \quad \Psi_n'(x)=O(h^n) \tag{5.7.6}$$

Thus

$$f'(x)-p_n'(x) = \begin{cases} O(h^n), & \Psi_n'(x)\neq 0 \\ O(h^{n+1}), & \Psi_n'(x)=0 \end{cases} \tag{5.7.7}$$

We will derive examples of each case.

Let n=1, so that $p_n(x)$ is just the linear interpolate of $(x_0, f(x_0))$ and $(x_1, f(x_1))$. Then (5.7.3) yields

$$f'(x_0) \doteq D_h f(x_0) \equiv \frac{1}{h}[f(x_0+h)-f(x_0)] \qquad (5.7.8)$$

From (5.7.5),

$$f'(x_0)-D_h f(x_0) = \frac{h}{2}f''(\xi_1), \quad x_0 \le \xi_1 \le x_1, \qquad (5.7.9)$$

since $\Psi(x_0)=0$.

To improve on this with linear interpolation, choose $x=m\equiv(x_0+x_1)/2$. Then

$$f'(m) \doteq \frac{1}{h}[f(x_1)-f(x_0)].$$

We usually rewrite this by letting $\delta=h/2$, to obtain

$$f'(m) \doteq D_\delta f(m)=\frac{1}{2\delta}[f(m+\delta)-f(m-\delta)] \qquad (5.7.10)$$

For the error, using (5.7.5) and $\Psi_1'(m)=0$,

$$f'(m)-D_\delta f(m) = \frac{-\delta^2}{6}f''(\xi_2), \quad m-\delta \le \xi_2 \le m+\delta \qquad (5.7.11)$$

In general to obtain the higher order case in (5.7.7), we want to choose the nodes $\{x_i\}$ to have $\Psi_n'(x)=0$. This will be true if n is odd and the nodes are placed symmetrically about x, as in (5.7.10).

To obtain higher order formulas in which the nodes all lie on one side of x, use higher values of n in (5.7.3). For example, with $x=x_0$ and n=2,

$$f'(x_0) \doteq D_h f(x_0) \equiv \frac{1}{2h}[-3f(x_0)+4f(x_1)-f(x_2)] \qquad (5.7.12)$$

$$f'(x_0)-D_h f(x_0) = \frac{h^2}{3} f^{(3)}(\xi_1), \qquad x_0 \le \xi_1 \le x_2. \qquad (5.7.13)$$

<u>The method of undetermined coefficients</u> Another method to derive formulas for numerical integration, differentiation, and interpolation is called the method of undetermined coefficients. It is often equivalent to the formulas obtained from a polynomial

interpolation formula, but sometimes it results in a simpler derivation. We will illustrate the method by deriving a formula for $f''(x)$.

Assume

$$f''(x) \doteq D_h^{(2)}f(x)=Af(x+h)+Bf(x)+Cf(x-h) \qquad (5.7.14)$$

with A, B, and C unspecified. Replace $f(x+h)$ and $f(x-h)$ by the Taylor expansions

$$f(x\pm h) = f(x)\pm hf'(x)+\frac{h^2}{2}f''(x)\pm\frac{h^3}{6}f^{(3)}(x)+\frac{h^4}{24}f^{(4)}(\xi_\pm)$$

with $x-h\leq\xi\leq x\leq\xi\leq x+h$. Substitute into (5.7.14) and rearrange into a polynomial in powers of h:

$$Af(x+h)+Bf(x)+Cf(x-h) = (A+B+C)f(x)+h(A-C)f(x)+\frac{h^2}{2}(A+C)f''(x)$$
$$+\frac{h^3}{6}(A-C)f^{(3)}(x)+\frac{h^4}{24}[Af^{(4)}(\xi_+)+Bf^{(4)}(\xi_-)]$$

In order for this to equal $f''(x)$, we set

$$A+B+C=0, \quad A-C=0, \quad A+C=\frac{2}{h^2}$$

The solution of this system is

$$A=C = \frac{1}{h^2}, \qquad B= \frac{-2}{h^2} \qquad (5.7.16)$$

This yields the formula

$$D_h^{(2)}f(x) = \frac{f(x+h)-2f(x)+f(x-h)}{h^2} \qquad (5.7.17)$$

For the error, substitute (5.7.16) into (5.7.15) and use (5.7.17). This yields

$$f''(x)-D_h^{(2)}f(x) = \frac{h^2}{24}\left[f^{(4)}(\xi_+)+f^{(4)}(\xi_-)\right]$$

Using problem 1 of Chapter 1, and assuming $f(x)$ is four times continuously differentiable,

$$f''(x)-D_h^{(2)}f(x) = -\frac{h^2}{12}f^{(4)}(\xi) \qquad (5.7.18)$$

for some $x-h \le \xi \le x+h$. The formulas (5.7.17) and (5.7.18) could have been derived by calculating $p_2''(x)$ for the quadratic polynomial interpolating f(x) at x-h, x, x+h; but the above is probably simpler.

The general idea of the method of undetermined coefficients is to choose the Taylor coefficients in an expansion in h so as to obtain the desired derivative (or integral) as closely as possible. Further illustrations are given in problem 44.

Effect of error in function values The above formulas are useful when deriving methods for solving ordinary and partial differential equations, but they can lead to serious errors when applied to function values that are obtained empirically. To illustrate a method for analyzing the effect of such errors, we will consider the second derivative approximation (5.7.17).

Begin by rewriting (5.7.17) as

$$f''(x_1) \doteq D_h^{(2)} f(x_1) = \frac{f(x_2)-2f(x_1)+f(x_0)}{h^2},$$

with $x_j = x_0 + jh$. Let the actual values used be $\tilde{f}_i$ with

$$f(x_i) = \tilde{f}_i + \epsilon_i, \quad i=0,1,2. \qquad (5.7.19)$$

The actual numerical derivative computed is

$$\tilde{D}_h^{(2)} f(x_1) = \frac{\tilde{f}_2 - 2\tilde{f}_1 + \tilde{f}_0}{h^2} \qquad (5.7.20)$$

For its error, substitute (5.7.19) into (5.7.20), obtaining

$$f''(x_1)-\tilde{D}_h^{(2)}f(x_1) = f''(x_1) - \frac{f(x_2)-2f(x_1)+f(x_0)}{h^2} + \frac{\epsilon_2-2\epsilon_1+\epsilon_0}{h^2}$$

$$= \frac{-h^2}{12}f^{(4)}(\xi) + \frac{\epsilon_2-2\epsilon_1+\epsilon_0}{h^2} \tag{5.7.21}$$

For the term involving $\{\epsilon_i\}$, assume these errors are random within some interval $-E \le \epsilon \le E$. Then

$$|f''(x_1)-\tilde{D}_h^{(2)}f(x_1)| \le \frac{h^2}{12}|f^{(4)}(\xi)| + \frac{4E}{h^2}, \tag{5.7.22}$$

and the last bound would be attainable in many situations. An example of such errors would be rounding errors, with E a bound on their magnitude.

The error bound in (5.7.22) will initially get smaller as h decreases; but for h sufficiently close to zero, the error will begin to increase again. There is an optimal value of h, call it h^*, to minimize the right side of (5.7.22); and presumably there is a similar value for the actual error $f''(x_1)-\tilde{D}_h^{(2)}f(x_1)$.

<u>Example</u> Let $f(x)=-\cos(x)$, and compute $f''(0)$ using the numerical approximation (5.7.17). In Table 5.25, we give the errors in (1) $D_h^{(2)}f(0)$, computed exactly, and (2) $\tilde{D}_h^{(2)}f(0)$, computed using 8 digit rounded decimal arithmetic. In this last case,

$$|f''(0)-\tilde{D}_h^{(2)}f(0)| \le \frac{h^2}{12} + \frac{2\times10^{-8}}{h^2} \tag{5.7.23}$$

This bound is minimized at $h^*=.0022$, and this is consistent with the errors $f''(0)-\tilde{D}_h^{(2)}f(0)$ given in the table. For the exactly computed $D_h^{(2)}f(0)$, note that the errors decrease by 4 whenever h is halved, consistent with the error formula (5.7.18).

Table 5.25 Example of $D_h^{(2)}f(0)$ and $\tilde{D}_h^{(2)}f(0)$

h	$f''(0)-D_h^{(2)}f(0)$	Ratio	$f''(0)-\tilde{D}_h^{(2)}f(0)$
.5	2.07E-2		2.07E-2
.25	5.20E-3	3.98	5.20E-3
.125	1.30E-3	3.99	1.30E-3
.0625	3.25E-4	4.00	3.25E-4
.03125	8.14E-5	4.00	8.45E-5
.015625	2.03E-5	4.00	2.56E-6
.0078125	5.09E-6	4.00	-7.94E-5
.00390625	1.27E-6	4.00	-7.94E-5
.001953125	3.18E-7	4.00	-1.39E-3

Regularization of differentiation We begin by showing that computing the derivative of a function is an ill-posed problem, in the sense of §1.6. Following the notation of §1.6, let y(x) be a continuously differentiable function. Consider computing

$$x(t)=\frac{dy(t)}{dt} \tag{5.7.24}$$

say for $0\leq t\leq 1$. To measure the size of x and y, use the norm in (4.4.2):

$$\|y\|_2 = \sqrt{\int_0^1 |y(t)|^2 dt}$$

Consider the following perturbation of (5.7.24):

$$x_n(t)=\frac{dy_n(t)}{dt}, \quad y_n(t)=y(t)+\frac{1}{n}\sin(n\pi t) \tag{5.7.25}$$

Then

$$x_n(t)=x(t)+\pi\cdot\cos(n\pi t) \tag{5.7.26}$$

For the perturbations,

$$\|y-y_n\|_2 = \frac{1}{\sqrt{2}n}, \quad \|x-x_n\|_2 = \frac{\pi}{\sqrt{2}} \tag{5.7.27}$$

As $n\to\infty$, the perturbation in the data y(t) will tend to zero, but the perturbation in the solution x does not tend to zero. This is the definition of ill-posed as given in the remarks following (1.6.1).

Because the operation of exact differentiation is unstable with respect to small perturbations in the data, we should expect a similar instability in the numerical differentiation of discrete data. This was shown in (5.7.21) for the formula (5.7.17) for $D_h^{(2)}f(x)$. Similar results hold for the other numerical differentiation formulas derived earlier in this section, but we leave their derivation to problem 46.

In the early 1960's, A. N. Tikhonov derived a method for solving ill-posed problems, and he called it the *method of regularization*. An ill-posed problem is replaced by a sequence of well-posed problems, with the solutions of the well-posed problem converging to the solution of the ill-posed problem. For a recent account of this theory, see Tikhonov-Arsenin (1977) or Groetsch (1984).

In Cullum (1971), this technique was applied to the differentiation problem (5.7.24). We will very briefly sketch some of the ideas of that paper. Begin by reformulating (5.7.24) as the integral equation.

$$A(x)(t) \equiv \int_0^t x(s)ds - y(t) + y(0) = 0, \quad 0 \le t \le 1. \qquad (5.7.28)$$

It will be assumed that $y(0)=y(1)=0$. [If this is not satisfied, simply replace $y(t)$ by

$$y(t)-[(1-t)y(0)+ty(t)]$$

This will change the derivative by the constant $y(1)-y(0)$.]

The problem of solving (5.7.28) is still an ill-posed one. Introduce the functional

$$C(x,\alpha)=\int_0^1 |A(x)(t)|^2 dt + \left[\int_0^1 x(t)dt\right]^2 + \alpha\int_0^1 \left[|x(t)|^2+|x'(t)|^2\right]dt \qquad (5.7.29)$$

for $0<\alpha$. Under suitable assumptions on $y(t)$, it can be shown that for each α, there is a unique minimizing function $x_\alpha(t)$ for $C(x,\alpha)$. Moreover, as $\alpha\to 0$, the solution x_α will converge to $x=y'$. The problem of minimizing $C(x,\alpha)$ is well-posed, and standard techniques can be used to calculate x_α.

The numerical procedure is based on choosing a fixed small value of α and then minimizing $C(x,\alpha)$. The procedure for finding $x_\alpha(t)$ is described in Cullum (1971). For very small values of α, minimizing $C(x,\alpha)$ will be ill-conditioned (cf. §1.6) and x_α will be near to $x=y'$. For larger values of α, minimizing $C(x,\alpha)$ is well-conditioned, but the solution is not as close to x. We attempt to find an intermediate value α that will avoid ill-conditioning while giving an x_α close to x.

For the numerical differentiation of discrete data containing experimental error, statistical considerations should be used. The above procedure was approached from such a perspective in Anderssen-Bloomfield (1974a), (1974b), and the choice of the optimal value of the regularization parameter α was related to the error in the data being differentiated. Related but more general procedures are given in Wahba (1980), and these are suitable for a wide variety of ill-posed problems. A Fortran program implementing these ideas and including numerical differentiation is given in Woltring (1986). It also contains a review of a number of software packages for data smoothing, and this will remove the main source of instability in numerical differentiation.