Numerical Methods

Numerical Differentiation

- Finite Differences

2022-2023

[DΣIM]

Numerical Derivation

- We have rules to derive analytically derivable functions, these rules, however, can not be used when the data are in tabular form
- Moreover, sometimes the derivation rules give analytical functions that can be difficult to evaluate
- It is more useful in these cases, to use a numerical rule that, given an initial data set, computes an approximate value of de derivative at a particular point

Numerical Derivation

- We are interested in calculating the numerical derivative of a function *f,* not his analytical derivative:

$$f \nRightarrow f'$$

- We need to estimate accurately the value of the derivative in a point when we have a data set from an interval including the point.

Numerical Derivative

- First, we need to approximate the function that we want to derive by a polynomial

$$f(x) \approx P_n(x)$$

- Then, we derivate the polynomial and we approximate the value of the derivative of the function by the value of the derivative of the polynomial

$$f'(x) \approx P'_n(x)$$

Numerical Derivative

- If we want to derive numerically a function at the point $x = a$, we need first a set of interpolation points from an interval around the point $x = a$:

$$\{(x_0, f(x_0)), (x_1, f(x_1)), \ldots, (x_m, f(x_m))\}$$

- With these points, we will build an interpolating polynomial of degree $m, P_m(x)$, derive this polynomial, and make the approximation:

$$f'(a) = P'_m(a)$$

Case $m = 1$

- In this case we have two data points at x_0 and x_1. Let's take $x_0=a$ and $x_1=a+h$. The table for the Newton interpolating polynomial is:

$$\begin{array}{ccc} a & f(a) & \searrow \dfrac{f(a+h)-f(a)}{h} \\ a+h & f(a+h) & \nearrow \end{array}$$

- And the resulting polynomial is:

$$P_1(x) = f(a) + \frac{f(a+h)-f(a)}{h}(x-a)$$

Case $m = 1$

- Deriving this polynomial, we obtain the approximation:

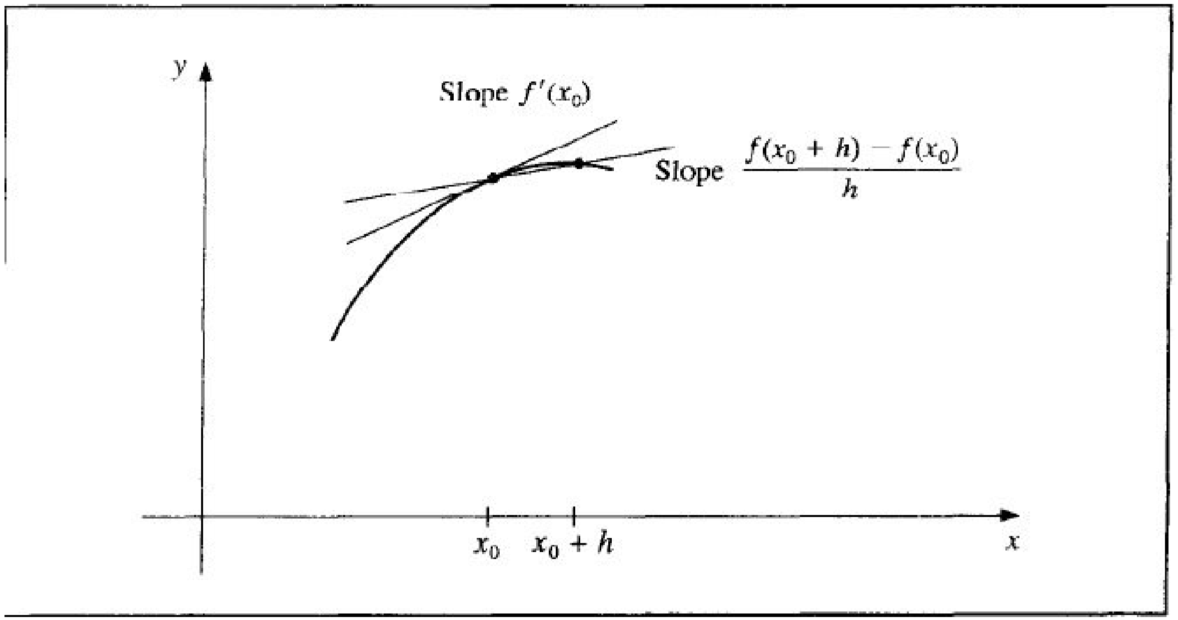
$$\boxed{f'(a) \approx \frac{f(a+h) - f(a)}{h}}$$

- This is the ***first order progressive formula***. Note that this expression reminds the exact definition of the derivative:

$$f'(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$$

[DΣIM]

Numerical Derivative



2022-2023

Case $m = 1$

- This expression can be obtained from the Taylor series of the function to be derived around the point $x = a$. The advantage of this method is that it gives also an estimate of the error of the approximation:

$$f(a+h) = f(a) + f'(a)h + \frac{1}{2} f''(\xi)h^2$$

- And then:

$$\frac{f(a+h) - f(a)}{h} = f'(a) + \frac{1}{2} f''(\xi)h$$

Case $m = 1$

- The error term in this expression is:

$$E_a = \frac{1}{2} f''(\xi) h$$

- It depends on the function to be derived through its second derivative, but it depends also on the distance between two consecutive interpolation points: h. This magnitude will be called the ***step.*** We say that this expression if of $\boldsymbol{O(h)}$.

Case $m = 1$

- For each step h, we obtain a numerical value of the derivative. These values represent the slopes of the secants to the curve. In the limit when $h \to 0$, we should recover the slope of the tangent, which is of course, the value of the derivative.

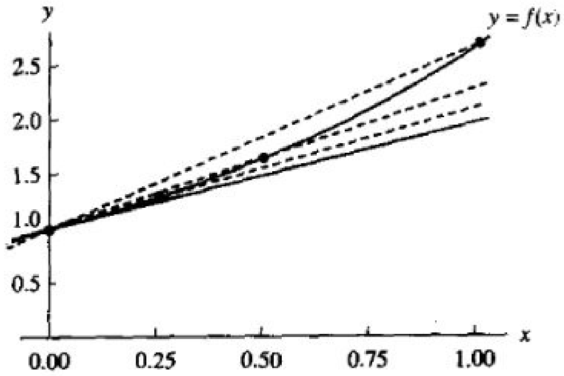


Figure 6.2 Several secant line for $y = e^x$.

Case $m = 1$

Table 6.1 Finding the Difference Quotients $D_k = (e^{1+h_k} - e)/h_k$

h_k	$f_k = f(1+h_k)$	$f_k - e$	$D_k = (f_k - e)/h_k$
$h_1 = 0.1$	3.004166024	0.285884196	2.858841960
$h_2 = 0.01$	2.745601015	0.027319187	2.731918700
$h_3 = 0.001$	2.721001470	0.002719642	2.719642000
$h_4 = 0.0001$	2.718553670	0.000271842	2.718420000
$h_5 = 0.00001$	2.718309011	0.000027183	2.718300000
$h_6 = 10^{-6}$	2.718284547	0.000002719	2.719000000
$h_7 = 10^{-7}$	2.718282100	0.000000272	2.720000000
$h_8 = 10^{-8}$	2.718281856	0.000000028	2.800000000
$h_9 = 10^{-9}$	2.718281831	0.000000003	3.000000000
$h_{10} = 10^{-10}$	2.718281828	0.000000000	0.000000000

Case $m = 1$

- Now, using the table:

$$\begin{array}{ccc} a-h & f(a-h) & \searrow \\ & & \dfrac{f(a)-f(a-h)}{h} \\ a & f(a) & \nearrow \end{array}$$

- We can obtain the polynomial:

$$P_1(x) = f(a) + \frac{f(a)-f(a-h)}{h}(x-a+h)$$

Case $m = 1$

- Deriving this polynomial we obtain a new derivation formula, the ***fist order regressive formula***:

$$\boxed{f'(a) \approx \frac{f(a) - f(a-h)}{h}}$$

- And as in the former case the error will be of the order of:

$$E_a = \frac{1}{2} f''(\xi) h$$

- Meaning that it is also a $\boldsymbol{O(h)}$ expression.

Case $m = 1$

- Another possibility would be to use the table:

$$\begin{array}{lll} a-h & f(a-h) & \searrow \\ & & \dfrac{f(a+h)-f(a-h)}{2h} \\ a+h & f(a+h) & \nearrow \end{array}$$

- Which gives the polynomial:

$$P_1(x) = f(a-h) + \frac{f(a+h)-f(a-h)}{2h}(x-a+h)$$

Case $m = 1$

- The derivation of this polynomial gives another derivation formula, called the ***centered derivation formula***:

$$\boxed{f'(a) \approx \frac{f(a+h) - f(a-h)}{2h}}$$

- Note that in order not calculate the value of $f'(a)$ we do not use the function value at this point, $f(a)$

Case $m = 1$

- It is interesting to know the error associated to this new expression. From the Taylor series around $x = a$, we obtain:

$$f(a+h) = f(a) + f'(a)h + \frac{1}{2} f''(a)h^2 + \frac{1}{3!} f^{(3)}(\xi_1)h^3$$

$$f(a-h) = f(a) - f'(a)h + \frac{1}{2} f''(a)h^2 - \frac{1}{3!} f^{(3)}(\xi_2)h^3$$

Case $m = 1$

- Combining these expressions, we have:

$$\frac{f(a+h)-f(a-h)}{2h} = f'(a) + \frac{h^2}{12}\left[f^{(3)}(\xi_1) + f^{(3)}(\xi_2)\right]$$

- The error term can be simplified if we assume that $f^{(3)}$ is continuous on $[x-h, x+h]$. Let M and m denote the greatest and least values of $f^{(3)}$ in this interval. Then $f^{(3)}(\xi_1), f^{(3)}(\xi_2)$ and $c = \frac{[f^{(3)}(\xi_1)+f^{(3)}(\xi_2)]}{2}$ all lie in the interval $[m, M]$. Since $f^{(3)}$ is continuous is assumes the value c at some point ξ in $[x-h, x+h]$.

Case m=1

- At this point we have:

$$f^{(3)}(\xi) = \frac{1}{2}\left[f^{(3)}(\xi_1) + f^{(3)}(\xi_2) \right]$$

- And the error term can be written:

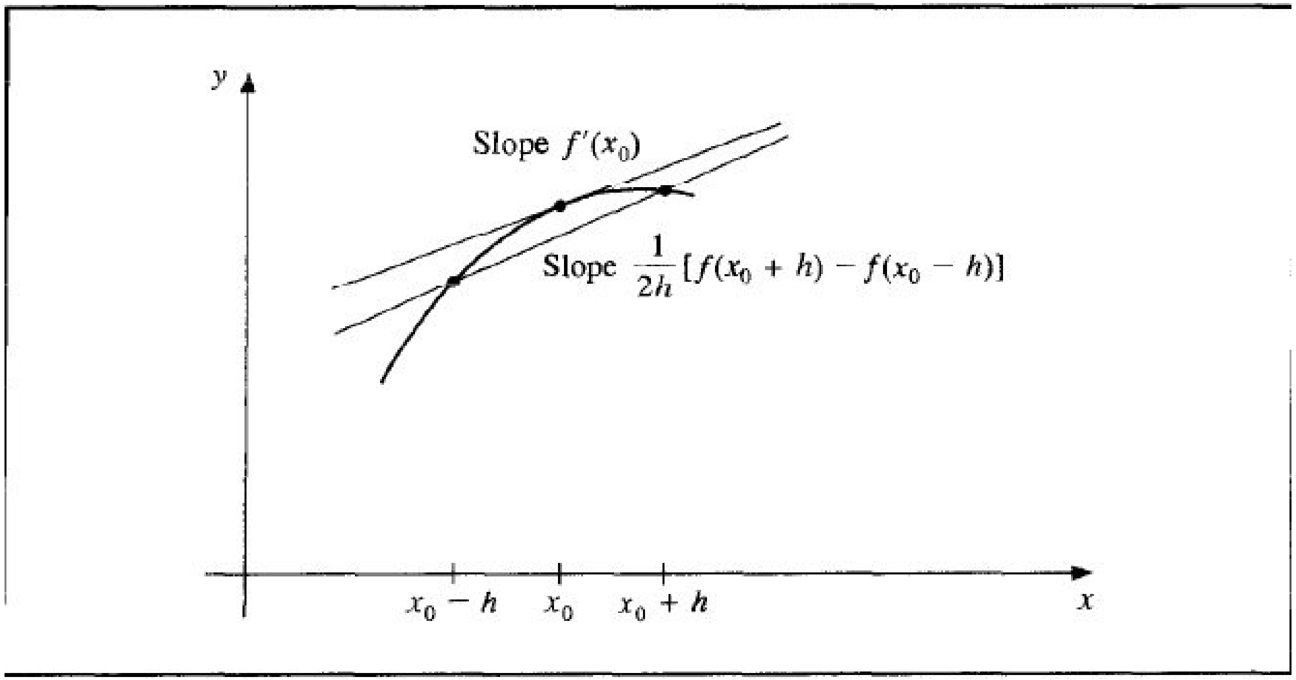
$$E_a = \frac{h^2}{12}\left[f^{(3)}(\xi_1) + f^{(3)}(\xi_2) \right] = \frac{h^2}{6} f^{(3)}(\xi)$$

- Showing that this approximation is $\boldsymbol{O(h^2)}$.

Case $m = 1$

- When using the centered formula, the error is proportional to the square power of the step: $E_a \sim h^2$, while in the progressive or regressive formulae, the error is $E_a \sim h$
- This means that while the ***regressive or the o progressive are first order formulae,*** the ***centered formula is of second order, i.e., $\boldsymbol{O(h^2)}$***.
- Using this formula, we can double the precision in a similar calculation using also two nodes.

Numerical Derivative



Case $m = 2$

- Given a table, we can build different interpolating polynomials. Their derivatives can be taken as different approximations to the derivative of the function.

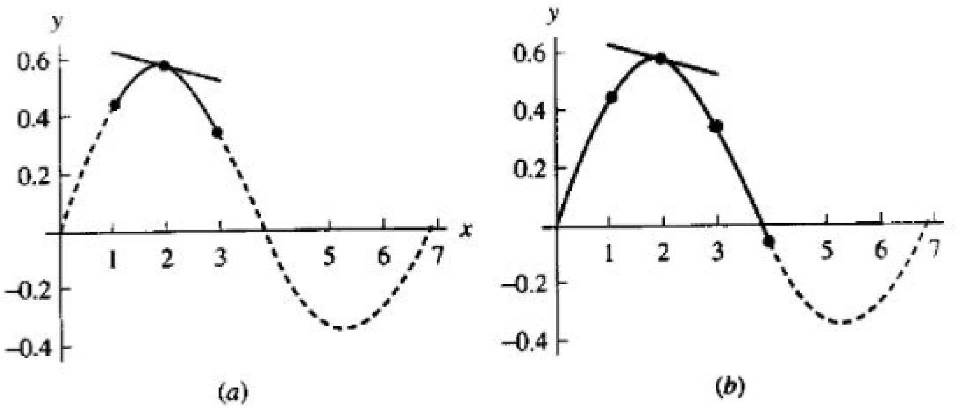


Figure 6.1 (a) The tangent to $p_2(x)$ at $(2, 0.5767)$ with slope $p_2'(2) = -0.0505$. (b) The tangent to $p_4(x)$ at $(2, 0.5767)$ with slope $p_4'(2) = -0.0618$.

Case $m = 2$

- In the case $m = 2$, we will use three interpolation points: $x_0 = a - h, x_1 = a,$ and $x_2 = a + h$. The Newton's divided differences are:

$$\begin{array}{cccc}
a-h & f(a-h) & & \\
a & f(a) & \dfrac{f(a)-f(a-h)}{h} & \\
a+h & f(a+h) & \dfrac{f(a+h)-f(a)}{h} & \dfrac{f(a+h)-2f(a)+f(a-h)}{2h^2}
\end{array}$$

Case m=2

- The resulting interpolating polynomial is:

$$P_2(x) = f(a-h) + \frac{f(a)-f(a-h)}{h}(x-a+h)$$
$$+\frac{f(a+h)-2f(a)+f(a-h)}{2h^2}(x-a+h)(x-a)$$

- And its derivative is a straight line:

$$P_2'(x) = \frac{f(a)-f(a-h)}{h}$$
$$+\frac{f(a+h)-2f(a)+f(a-h)}{2h^2}(2x-2a+h)$$

Case $m = 2$

- To obtain the derivative we evaluate the line at $x = a$ resulting in:

$$P_2'(a) = \frac{f(a) - f(a-h)}{h} + \frac{f(a+h) - 2f(a) + f(a-h)}{2h^2}(2a - 2a + h)$$
$$= \frac{2f(a) - 2f(a-h) + f(a+h) - 2f(a) + f(a-h)}{2h}$$
$$= \frac{f(a+h) - f(a-h)}{2h}$$

- This is the same expression as in the case $m = 1$ using centered derivatives.

Case $m = 2$

- We already know that the error in this expression is:

$$E_a = \frac{h^2}{6} f^{(3)}(\xi)$$

- The centered derivative of first order is equivalent to the progressive second order formulae. For a better precision it will be necessary to use more data points to get interpolating polynomials of higher degree.

Higher order formulae

- Thus, making $x_{-2} = a - 2h$, $x_{-1} = a - h$, $x_0 = a$, $x_1 = a + h$, and $x_2 = a + 2h$, we can build derivation formulae of higher order:

$$f'(x_0) = \frac{f(x_1) - f(x_0)}{h} + O(h)$$

$$f'(x_0) = \frac{f(x_1) - f(x_{-1})}{2h} + O(h^2)$$

$$f'(x_0) = \frac{-f(x_2) + 4f(x_1) - 3f(x_0)}{2h} + O(h^2)$$

$$f'(x_0) = \frac{-f(x_2) + 8f(x_1) - 8f(x_{-1}) + f(x_{-2})}{12h} + O(h^4)$$

Higher order formulae

- We can compare the behavior of the centered derivative formula of second order with the fourth-degree formula for $f(x)=\cos(x)$

Table 6.2 Numerical Differentiation Using Formulas (3) and (10)

Step size	Approximation by formula (3)	Error using formula (3)	Approximation by formula (10)	Error using formula (10)
0.1	−0.716161095	−0.001194996	−0.717353703	−0.000002389
0.01	−0.717344150	−0.000011941	−0.717356108	0.000000017
0.001	−0.717356000	−0.000000091	−0.717356167	0.000000076
0.0001	−0.717360000	−0.000003909	−0.717360833	0.000004742

Higher order derivatives

- We could repeat the process to calculate higher order derivatives, $f^{(d)}(a)$ for a function in tabular form. The derivative of order d, must be lower or equal to the maximum degree of the polynomial that can be build with this data m.
- Knowing $f(x)$ at the data points $x_0, \dots, x_m$ with $d \leq m$, we can compute the interpolating polynomial $P_d(x)$ at $x_0, \dots, x_d$ and deriving d times, we can compute the approximation:

$$f^{(d)}(a) \approx d!\,f\left[x_0, \dots, x_d\right] \quad \text{on } a \in (x_0, x_d)$$

Second derivatives

- Following this procedure, we can obtain the following set of formulae for approximating the second derivatives:

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

$$f''(x_0) = \frac{f(x_1) - 2f(x_0) + f(x_{-1})}{h^2} + O(h^2)$$

$$f''(x_0) = \frac{-f(x_3) + 4f(x_2) - 5f(x_1) + f(x_0)}{h^2} + O(h^2)$$

$$f''(x_0) = \frac{-f(x_2) + 16f(x_1) - 30f(x_0) + 16f(x_{-1}) - f(x_{-2})}{12h^2} + O(h^4)$$

Third and Fourth Derivative

- Similarly for the third derivatives:

$$f^{(3)}(x_0) = \frac{f(x_3) - 3f(x_2) + 3f(x_1) - f(x_0)}{h^3} + O(h)$$

$$f^{(3)}(x_0) = \frac{f(x_2) - 2f(x_1) + 2f(x_{-1}) - f(x_{-2})}{2h^3} + O(h^2)$$

- And for the fourth order:

$$f^{(4)}(x_0) = \frac{f(x_4) - 4f(x_3) + 6f(x_2) - 4f(x_1) + f(x_0)}{h^4} + O(h)$$

$$f^{(4)}(x_0) = \frac{f(x_2) - 4f(x_1) + 6f(x_0) - 4f(x_{-1}) + f(x_{-2})}{h^4} + O(h^2)$$

Second Derivative

- All these expressions are known as ***finite difference approximations***. As in the case of the first order expressions, the finite differences can be obtained from the Taylor series. In the case of second derivatives, adding the following expressions:

$$f(x+h) = f(x) + hf'(x) + \frac{1}{2!}h^2 f''(x) + \frac{1}{3!}h^3 f'''(x) + \frac{1}{4!}h^4 f^{iv}(\xi)$$

$$f(x-h) = f(x) - hf'(x) + \frac{1}{2!}h^2 f''(x) - \frac{1}{3!}h^3 f'''(x) + \frac{1}{4!}h^4 f^{iv}(\varsigma)$$

- We obtain

$$f(x+h) + f(x-h) = 2f(x) + h^2 f''(x) + \frac{2}{4!}h^4 f^{iv}(\zeta)$$

Second Derivative

- And we can write the following approximation:

$$\boxed{f''(x) \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}}$$

- Taylor series give explicitly the error associated with the approximation:

$$-\frac{1}{12}h^2 f^{(4)}(\zeta)$$

- Thus, in this case we have an error of the order $\boldsymbol{O(h^2)}$

Undetermined Coefficients

- We have derived the numerical differentiation formulas by differentiating the interpolating polynomials. In many cases it is simpler to derive the formulas using the method of undetermined coefficients, demanding that the resulting formula should be exact for polynomials up to a certain degree.
- Consider a five-point formula of the following form:

$$f'(x)=\frac{c_{-2}f(x_{-2})+c_{-1}f(x_{-1})+c_0f(x_0)+c_1f(x_1)+c_2f(x_2)}{h}$$

Undetermined Coefficients

- Since the formula is linear in the function values, it is enough to ensure that it is exact when $f(x) = 1, x, x^2, x^3, x^4$. Then the required conditions can be written as follows:

$$
\begin{aligned}
f(x) = 1: &\quad c_{-2} + c_{-1} + c_0 + c_1 + c_2 = 0 \\
f(x) = x: &\quad 2(c_2 - c_{-2}) + (c_1 - c_{-1}) = 1 \\
f(x) = x^2: &\quad 4(c_2 + c_{-2}) + (c_1 + c_{-1}) = 0 \\
f(x) = x^3: &\quad 8(c_2 - c_{-2}) + (c_1 - c_{-1}) = 0 \\
f(x) = x^4: &\quad 16(c_2 + c_{-2}) + (c_1 + c_{-1}) = 0
\end{aligned}
$$

Undetermined Coefficients

- Solving the linear system, we obtain the expression:

$$f'(x) = \frac{f(x_{-2}) - 8f(x_{-1}) + 8f(x_1) + f(x_2)}{12h} + \frac{h^4}{30} f^{(5)}(\xi)$$

- Here, the error term can be estimated by assuming it to be of the form $a_e h^4 f^{(5)}(\xi)$. The unknown coefficient a_e can be determined using $f(x) = x^5$ in the above formula. Though this is not a rigorous derivation, we can expect to obtain a reasonable estimate.

Undetermined Coefficients

- This procedure can be automated if we introduce directly the Taylor series in the formula as the coefficients of the linear system will be related to the Taylor coefficients.
- For instance, if we want to derive the coefficients of second derivative centred formula:

$$f''(x) = \frac{c_{-2}f(x_{-2}) + c_{-1}f(x_{-1}) + c_0 f(x_0) + c_1 f(x_1) + c_2 f(x_2)}{h^2}$$

Undetermined Coefficients

- If we introduce the Taylor series of each term we obtain:

$$f''(x)=\frac{1}{h^2}\left[\begin{array}{l} c_2\left(f_0+2hf_0'+\frac{(2h)^2}{2!}f_0^{(2)}+\frac{(2h)^3}{3!}f_0^{(3)}+\frac{(2h)^4}{4!}f_0^{(4)}+\dots\right)\\ +c_1\left(f_0+hf_0'+\frac{(h)^2}{2!}f_0^{(2)}+\frac{(h)^3}{3!}f_0^{(3)}+\frac{(h)^4}{4!}f_0^{(4)}+\dots\right)+c_0f_0\\ +c_{-1}\left(f_0-hf_0'+\frac{(h)^2}{2!}f_0^{(2)}-\frac{(h)^3}{3!}f_0^{(3)}+\frac{(h)^4}{4!}f_0^{(4)}-\dots\right)\\ +c_{-2}\left(f_0-2hf_0'+\frac{(2h)^2}{2!}f_0^{(2)}-\frac{(2h)^3}{3!}f_0^{(3)}+\frac{(2h)^4}{4!}f_0^{(4)}+\dots\right)\end{array}\right]$$

Undetermined Coefficients

- And rearranging terms:

$$f''(x)=\frac{1}{h^2}\left[\begin{array}{l}\left(c_2+c_1+c_0+c_{-1}+c_2\right)f_0+h(2c_2+c_1-c_{-1}-2c_{-2})f_0{}'\\ +h^2\left(\dfrac{2^2}{2}c_2+\dfrac{1}{2}c_1+\dfrac{1}{2}c_{-1}+\dfrac{2^2}{2}c_{-2}\right)f_0^{(2)}\\ +h^3\left(\dfrac{2^3}{3!}c_2+\dfrac{1}{3!}c_1-\dfrac{1}{3!}c_{-1}-\dfrac{2^3}{3}c_{-2}\right)f_0^{(3)}\\ +h^4\left(\dfrac{2^4}{4!}c_2+\dfrac{1}{4!}c_1+\dfrac{1}{4!}c_{-1}+\dfrac{2^4}{4!}c_{-2}\right)f_0^{(4)}\end{array}\right]$$

Undetermined Coefficients

- Finally making the expression conform to the second derivative $f_0^{(2)}$ at $x + 0h = x$ we get the system

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 2 & 1 & 0 & -1 & -2 \\ 2^2/2! & 1/2! & 0 & 1/2! & 2^2/2! \\ 2^3/3! & 1/3! & 0 & -1/3! & -2^3/3! \\ 2^4/4! & 1/4! & 0 & 1/4! & 2^4/4! \end{pmatrix} \begin{pmatrix} c_2 \\ c_1 \\ c_0 \\ c_{-1} \\ c_{-2} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

Error in the numerical derivative

- Can we reduce the time step h as long as we wish to minimize the value of the numerical derivative?
- Consider for instance the centred derivative expression:

$$f'(a) = \frac{f(a+h) - f(a-h)}{2h} - \frac{h^2 f^{(3)}(\xi)}{6}$$

- Reducing the time step h, certainly the error associated with this approximation goes to zero.

Optimal Step

- The error in the derivation formulae depend on powers of the step h, $(h, h^2, \ldots)$. We could think that it would be better to take a smaller value of h as possible to minimize the error.
- Note however, that taking a very small value of h will lead to numerical error due to the cancellation of terms as all the terms of the approximation tend to be equal:

$$f(a+h) \approx f(a) \approx f(a-h)$$

Optimal Step

- Consider for instance the following table:

Table 5.1 The Forward Difference Approximation (5.1.4) for the First Derivative of $f(x) = \sin x$ and Its Error from the True Value ($\cos \pi/4 = 0.7071067812$) Depending on the Step Size h

$h_k = 10^{-k}$	$D_{1k\|x=\pi/4}$	$D_{1k} - D_{1(k-1)}$	$D_{1k\|x=\pi/4} - \cos(\pi/4)$
$h_1 = 0.1000000000$	0.6706029729		−0.03650380828
$h_2 = 0.0100000000$	0.7035594917	0.0329565188	−0.00354728950
$h_3 = 0.0010000000$	0.7067531100	0.0031936183	−0.00035367121
$h_4 = 0.0001000000$	0.7070714247	0.0003183147	−0.00003535652
$h_5 = 0.0000100000$	0.7071032456	0.0000318210	−0.00000353554
$h_6 = 0.0000010000$	0.7071064277	0.0000031821	−0.00000035344
$h_7 = 0.0000001000$	0.7071067454	0.0000003176	−0.00000003581
$h_8 = 0.0000000100^*$	0.7071067842	0.0000000389	0.00000000305*
$h_9 = 0.0000000010$	0.7071068175	0.0000000333*	0.00000003636
$h_{10} = 0.0000000001$	0.7071077057	0.0000008882	0.00000092454
$h_o = 0.0000000168$ (the optimal value of h obtained from Eq. (5.2.2))			

Optimal Step

- This phenomenon is also present in we use the second order expressions:

Table 5.2 The Forward Difference Approximation (5.1.8) for the First Derivative of $f(x) = \sin x$ and Its Error from the True Value ($\cos \pi/4 = 0.7071067812$) Depending on the Step Size h

$h_k = 10^{-k}$	$D_{2k\|x=\pi/4}$	$D_{2k} - D_{2(k-1)}$	$D_{2k\|x=\pi/4} - \cos(\pi/4)$
$h_1 = 0.1000000000$	0.7059288590		−0.00117792219
$h_2 = 0.0100000000$	0.7070949961	0.0011661371	−0.00001178505
$h_3 = 0.0010000000$	0.7071066633	0.0000116672	−0.00000011785
$h_4 = 0.0001000000$	0.7071067800	0.0000001167	−0.00000000118
$h_5 = 0.0000100000$*	0.7071067812	0.0000000012	−0.00000000001*
$h_6 = 0.0000010000$	0.7071067812	0.0000000001*	0.00000000005
$h_7 = 0.0000001000$	0.7071067804	−0.0000000009	−0.00000000084
$h_8 = 0.0000000100$	0.7071067842	0.0000000039	0.00000000305
$h_9 = 0.0000000010$	0.7071067620	−0.0000000222	−0.00000001915
$h_{10} = 0.0000000001$	0.7071071506	0.0000003886	0.00000036942

$h_o = 0.0000059640$ (the optimal value of h obtained from Eq. (5.2.3))

Optimal Step

- Suppose that we want to compute $f'(a)$ using the formula for the centered derivative

$$f'(a) = \frac{f(a+h) - f(a-h)}{2h} - \frac{h^2 f^{(3)}(\xi)}{6}$$

- The error of this expression depends on the second power of the step. But this is not the sole source of error in our computations.

Optimal Step

- In reality, the values $f(a+h)$ and $f(a-h)$ are represented within the machine with a certain error ε

$$f^*(a+h) = f(a+h) \pm \varepsilon_+$$
$$f^*(a-h) = f(a-h) \pm \varepsilon_-$$

- And we compute the value:

$$f'^*(a) = \frac{f(a+h) \pm \varepsilon_+ - f(a-h) \pm \varepsilon_-}{2h}$$

Optimal Step

- This expression can be decomposed into two parts:

$$
\begin{aligned}
f'^*(a) &= \frac{f(a+h) \pm \varepsilon_+ - f(a-h) \pm \varepsilon_-}{2h} \\
&= \frac{f(a+h) - f(a-h)}{2h} \pm \frac{\varepsilon_+ + \varepsilon_-}{2h} \\
&= f'(a) + \frac{h^2 f^{(3)}(\xi)}{6} \pm \frac{\varepsilon_+ + \varepsilon_-}{2h}
\end{aligned}
$$

Optimal Step

- The expressions f^* are used by the computer after rounding to the machine precision. These are the real expressions of f used, modified by a certain error ε
- When computing operations with numbers affected by certain error, the machine will never improve the precision of the results. The errors will increase due to the propagation of errors

Optimal Step

- There are two sources of error which behave differently when $h \to 0$
- The optimal value of h will be the one that minimizes the total error:

$$\frac{hf^{(3)}(\xi)}{3} - \frac{\varepsilon_{+} \pm \varepsilon_{-}}{h^{2}} = 0$$

- Each value will depend on the function and the interval of derivation

Optimal Step

- The computed value has two sources of error, a theoretical (Taylor series) and a numerical source. Each one behaves differently
- The theoretical term goes to zero as $h \to 0$ while the numerical error goes to ∞ when $h \to 0$. Thus, in each case we will have an optimum value of the step h, which minimizes the error.

Optimal Step

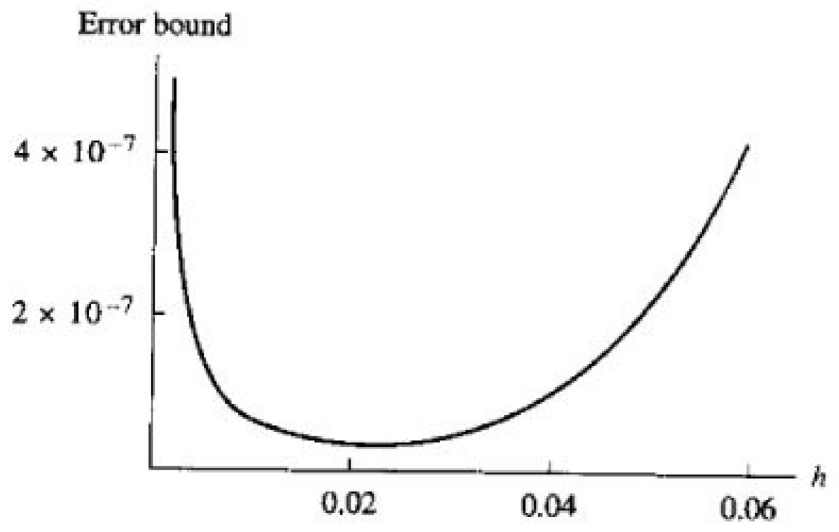


Figure 6.4 Finding the optimum step size $h = 0.022388475$ when formula (26) is applied to $f(x) = \cos(x)$ in Example 6.2.

Optimal Step

- Consider for instance the function $f(x) = e^x$ and calculate numerically $f'(0)$. We have then

$$f(x) = f'(x) = f''(x) = \cdots = f^{(n)}(x) = e^x$$

- Near the point $x = 0$, $f^{(3)}(\xi) \sim 1$. Now if we suppose that we work with a machine with a numerical precision of $\varepsilon = 10^{-8}$

Optimal Step

- The combined error is:

$$g(h) = \frac{2 \cdot 10^{-8}}{2h} + \frac{h^2}{6}$$

- And it will be optimum when

$$g'(h) = -\frac{10^{-8}}{h^2} + \frac{h}{3} = 0$$

- Approximately for $h \sim 0.003$

Optimal Step

- As h is reduced, the truncation component of the error bound shrinks in magnitude like h^2, while the roundoff error grows in magnitude like h^{-1}.

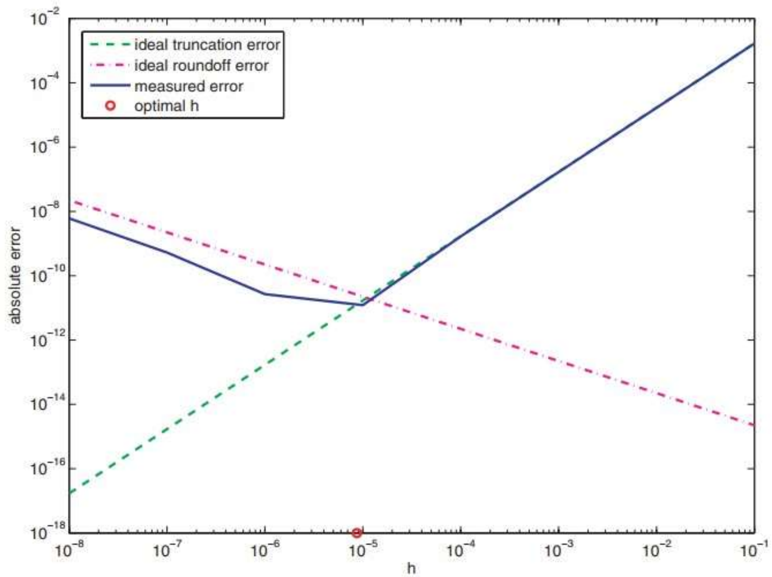


Figure 14.2. *The measured error roughly equals truncation error plus roundoff error. The former decreases but the latter grows as h decreases. The "ideal roundoff error" is just η/h. Note the log-log scale of the plot. A red circle marks the "optimal h" value for Example* 14.1.

Optimal Step

- It is impossible to avoid severe cancellation errors with a very small h and there is always a compromise
- Formulas from numerical differentiation are frequently used in the numerical solution of differential equations. In this case h must be such that truncation error dominates roundoff. Therefore, double precision is the default option in scientific computing. Also, h is usually not small for reasons of efficiency.

Optimal Step

- If an approximate value for a function derivative is desired, then h is typically much smaller than what is employed in the numerical solution of differential equations. Here the roundoff problem is more important.
- This is of particular importance in the case of noisy measurements. Noise is normally a high frequency effect and in this case the error can be strongly amplified.

Optimal Step

- Consider the error of a polynomial approximation $p(x)$ to $f(x)$ with an error $e(x) = f(x) - p(x)$. If we use $p'(x)$ to approximate $f'(x)$, then the error is $e'(x)$.
- Consider the Fourier series of the error

$$e(x) = \sum_{j=-\infty}^{\infty} c_j e^{i\pi jx}$$

- The error for the derivative approximation is then:

$$e'(x) = i\pi \sum_{j=-\infty}^{\infty} j c_j e^{i\pi jx}$$

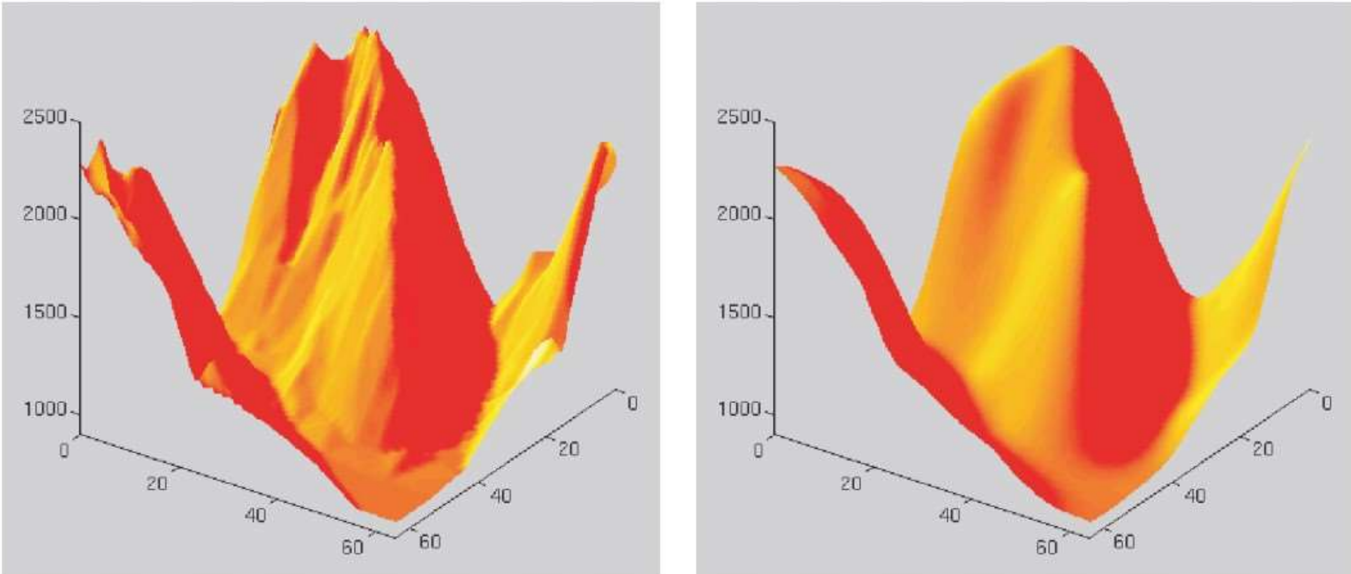
Optimal Step

- So, the amplitude $|c_j|$ for the jth frequency is replaced by $|jc_j|$ in the derivative expansion. The higher the frequency, the larger the amplification $|j|$.
- Truncation error is smooth. The high-frequency components in the Fourier expansion have vanishing amplitudes. Roundoff error is not smooth, and it is rich in high-frequency components. As $j \to \pm\infty$, corresponding to $h \to 0$, the magnification of these error components is unbounded.

Optimal Step

- Consider the following image from satellite-measured depth data of a mountainous terrain near St. Mary Lake, British Columbia, Canada
- If we need to estimate terrain slopes using directly numerical differentiation will frustrate our efforts. It is better to smooth the image first and then apply the numerical differentiation.
- "A First Course in Numerical Methods. Ascher, U.M., Greif, Ch. SIAM (2011)"

Optimal Step



(a) Unsmoothed image. (b) Smoothed image.

Figure 14.4. *An image with noise and its smoothed version. Numerical differentiation must not be applied to the original image.*

Richardson Extrapolation

- This is an algorithm that avoids the error due to cancelation of terms that can appear in the finite difference formulae when the step h is small and can improve the accuracy of any finite difference formulae.
- We assume that $f(x)$ can be represented as a Taylor series:

$$f(x+h)=\sum_{k=0}^{\infty}\frac{1}{k!}h^k f^{(k)}(x)$$

$$f(x-h)=\sum_{k=0}^{\infty}\frac{1}{k!}(-1)^k h^k f^{(k)}(x)$$

Richardson Extrapolation

- Then we can subtract both equations and build the central difference formula as:

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \left[\frac{1}{3!} h^2 f^{(3)}(x) + \frac{1}{5!} h^4 f^{(5)}(x) + \ldots \right]$$

- This equation has the form:

$$L = \phi(h) + a_2 h^2 + a_4 h^4 + a_6 h^6 + \cdots$$

- Note that the error term is expressed as a power series of the step h

Richardson Extrapolation

- Using the same expression with a halved step $\frac{h}{2}$:

$$L = \phi(h/2) + a_2 h^2/4 + a_4 h^4/16 + a_6 h^6/64 + \cdots$$

- We can use both equations to eliminate the first and greater term of the error series as follows:

$$L = \phi(h) + a_2 h^2 + a_4 h^4 + a_6 h^6 + \cdots$$

$$4L = 4\phi(h/2) + a_2 h^2 + a_4 h^4/4 + a_6 h^6/16 + \cdots$$

$$3L = 4\phi(h/2) - \phi(h) - 3a_4 h^4/4 - 15 a_6 h^6/16 + \cdots$$

Richardson Extrapolation

- Then we can write:

$$L = \frac{4}{3}\phi(h/2) - \frac{1}{3}\phi(h) - a_4 h^4/4 - 5a_6 h^6/16 + \cdots$$

- This is the first step in the Richardson extrapolation. Note that we have combined two expressions of order $O(h^2)$ to obtain a new value with doubling the precision $O(h^4)$
- This procedure can continue recursively to obtain better and better approximations of the value L, which represents the derivative $f'(x)$ of the function at a certain point.

Richardson Extrapolation

- This procedure can be generalized to any finite difference formulae, which have error dependent on a power series. Consider that we have an approximation to $f'(x)$ given by $\phi(h)$. The procedure is as follows:
- 1) Compute the values

$$D(n,0) = \phi\left(\frac{h}{2^n}\right), \quad n = 0,1,\ldots,M$$

Richardson Extrapolation

- Next, we compute recursively the additional quantities:

$$D(n,k) = \frac{4^k}{4^k - 1} D(n,k-1) - \frac{1}{4^k - 1} D(n-1,k-1)$$

$$k = 1,2,..,M \qquad n = k,k+1,...,M$$

- And we will get the values:

$$D(n,0) = L + O(h^2)$$

$$D(n,1) = L + O(h^4)$$

$$D(n,2) = L + O(h^6)$$

$$\vdots$$

Richardson Extrapolation

- These formulae given for $D(n,0)$ and $D(n,k)$ can be used to build a triangular array of the form:

$$
\begin{array}{ccccc}
D(0,0) & & & & \\
D(1,0) & D(1,1) & & & \\
D(2,0) & D(2,1) & D(2,2) & & \\
\vdots & \vdots & \vdots & \ddots & \\
D(M,0) & D(M,1) & D(M,2) & \cdots & D(M,M)
\end{array}
$$

- This array can be enlarged until we reach a value $D(M,M)$ with the desired precision within the *eps* of the machine.

Richardson Extrapolation

- Theorem. The quantities $D(n,k)$ defined in the Richardson extrapolation algorithm obey an equation of the form:

$$D(n,k-1) = L + \sum_{j=k}^{\infty} A_{jk} \left(h/2^n\right)^{2j}$$

- And thus, we have:

$$D(n,k-1) = L + O(h^{2k}) \text{ as } h \to 0$$

Richardson Extrapolation

- When $k = 1$, we have by hypothesis that the errors can be written as a power series:

$$D(n,0) = \phi(h/2^n) = L - \sum_{j=1}^{\infty} a_{2j}(h/2^n)^{2j}$$

- And for these values we can write $A_{j1} = -a_{2j}$
- Now, we proceed by induction on the values $D(n,k)$, assuming that this relation is valid for $D(n,k-1)$

Richardson Extrapolation

- We can write the resulting equation for $D(n,k)$ as follows:

$$D(n,k) = \frac{4^k}{4^k - 1}\left[L + \sum_{j=k}^{\infty} A_{jk}\left(\frac{h}{2^n}\right)^{2j}\right] - \frac{1}{4^k - 1}\left[L + \sum_{j=k}^{\infty} A_{jk}\left(\frac{h}{2^{n-1}}\right)^{2j}\right]$$

- Or:

$$D(n,k) = L + \sum_{j=k}^{\infty} A_{jk}\left[\frac{4^k - 4^j}{4^k - 1}\right]\left(\frac{h}{2^n}\right)^{2j}$$

Richardson Extrapolation

- Thus, $A_{j,k+1}$ should be defined by:

$$A_{j,k+1} = A_{j,k}\left[\frac{4^k - 4^j}{4^k - 1}\right]$$

- Note that $A_{k,k+1} = 0$ and thus, we can write finally:

$$D(n,k) = L + \sum_{j=k+1}^{\infty} A_{jk+1}\left(h/2^n\right)^{2j}$$