Chapter 11

Numerical Differentiation

Differentiation is a basic mathematical operation with a wide range of applications in many areas of science. It is therefore important to have good methods to compute and manipulate derivatives. You probably learnt the basic rules of differentiation in school — symbolic methods suitable for pencil-and-paper calculations. Such methods are of limited value on computers since the most common programming environments do not have support for symbolic computations.

Another complication is the fact that in many practical applications a function is only known at a few isolated points. For example, we may measure the position of a car every minute via a GPS (Global Positioning System) unit, and we want to compute its speed. When the position is known at all times (as a mathematical function), we can find the speed by differentiation. But when the position is only known at isolated times, this is not possible.

The solution is to use approximate methods of differentiation. In our context, these are going to be numerical methods. We are going to present several such methods, but more importantly, we are going to present a general strategy for deriving numerical differentiation methods. In this way you will not only have a number of methods available to you, but you will also be able to develop new methods, tailored to special situations that you may encounter.

The basic strategy for deriving numerical differentiation methods is to evaluate a function at a few points, find the polynomial that interpolates the function at these points, and use the derivative of this polynomial as an approximation to the derivative of the function. This technique also allows us to keep track of the so-called *truncation error*, the mathematical error committed by differentiating the polynomial instead of the function itself. In addition to the truncation error, there are also *round-off* errors, which are unavoidable when we use floating-point numbers to perform calculations with real numbers. It turns out that numerical differentiation is very *sensitive* to round-off errors, but these errors are quite easy to analyse.

The general idea of the chapter is to introduce the simplest method for numerical differentiation in section 11.1, with a complete error analysis. This may appear a bit overwhelming, but it should not be so difficult since virtually all the details are included. You should therefore study this section carefully, and if you understand this, the simplest of the methods and its analysis, you should have no problems understanding the others as well, since both the derivation and the analysis is essentially the same for all the methods. The general strategy for deriving and analysing numerical differentiation methods is then summarised in section 11.2. In the following sections we introduce three more differentiation methods, including one for calculating second derivatives. For these methods we just state the error estimates; the derivation of the estimates is left for the exercises. Note that the methods for numerical integration in Chapter 12 are derived and analysed in much the same way as the differentiation methods in this chapter.

11.1 Newton's difference quotient

We start by introducing the simplest method for numerical differentiation, derive its error, and its sensitivity to round-off errors. The procedure used here for deriving the method and analysing the error is used over again in later sections to derive and analyse the other methods.

Let us first explain what we mean by numerical differentiation.

Problem 11.1 (Numerical differentiation)**.** *Let f be a given function that is known at a number of isolated points. The problem of numerical differentiation is to compute an approximation to the derivative f' of f by suitable combinations of the known function values of f.*

A typical example is that f is given by a computer program (more specifically a function, procedure or method, depending on your choice of programming language), and you can call the program with a floating-point argument x and receive back a floating-point approximation of $f(x)$. The challenge is to compute an approximation to $f'(a)$ for some real number a when the only aid we have at our disposal is the program to compute values of f.

11.1.1 The basic idea

Since we are going to compute derivatives, we must be clear about how they are defined. The standard definition of $f'(a)$ is by a limit process,

$$f'(a) = \lim_{h\to 0} \frac{f(a+h)-f(a)}{h}. \tag{11.1}$$

In the following we will assume that this limit exists, in other words that f is differentiable at $x = a$. From the definition (11.1) we immediately have a natural approximation of $f'(a)$: We simply pick a positive number h and use the approximation

$$f'(a) \approx \frac{f(a+h)-f(a)}{h}. \tag{11.2}$$

Recall that the straight line p_1 that interpolates f at a and $a+h$ (the secant based at these points) is given by

$$p_1(x) = f(a) + \frac{f(a+h)-f(a)}{h}(x-a).$$

The derivative of this secant is exactly the right-hand side in (11.2) and corresponds to the secant's slope. The approximation (11.2) therefore corresponds to approximating f by the secant based at a and $a+h$, and using its slope as an approximation to the slope of f at a, see figure 11.1.

The tangent to f at a has the same slope as f at a, so we may also obtain the approximation (11.2) by considering the secant based at a and $a+h$ as an approximation to the tangent at a, see again figure 11.1.

Observation 11.2 (Newton's difference quotient)**.** *The derivative of f at a can be approximated by*

$$f'(a) \approx \frac{f(a+h)-f(a)}{h}. \tag{11.3}$$

This approximation is referred to as Newton's difference quotient or just Newton's quotient.

An alternative to the approximation (11.3) is the left-sided version

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

Not surprisingly, this approximation behaves similarly, and the analysis is also completely analogous to that of the more common right-sided version.

In later sections, we will derive several formulas like (11.2). Which formula to use in a particular situation, and exactly how to apply it, will have to be decided in each case.

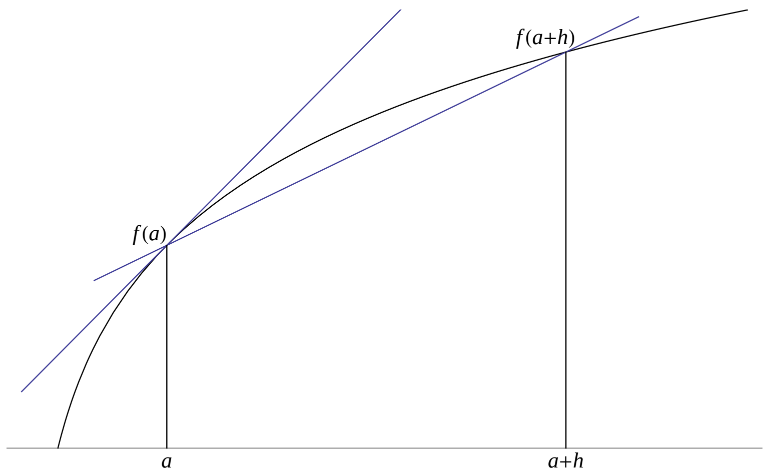

Figure 11.1. The secant of a function based at a and $a+h$, as well as the tangent at a.

Example 11.3. Let us test the approximation (11.3) for the function $f(x) = \sin x$ at $a = 0.5$ (using 64-bit floating-point numbers). In this case we know that the exact derivative is $f'(x) = \cos x$ so $f'(a) \approx 0.8775825619$ with 10 correct digits. This makes it is easy to check the accuracy of the numerical method. We try with a few values of h and find

h	$\bigl(f(a+h) - f(a)\bigr)/h$	$E(f;a,h)$
10^{-1}	0.8521693479	2.5×10^{-2}
10^{-2}	0.8751708279	2.4×10^{-3}
10^{-3}	0.8773427029	2.4×10^{-4}
10^{-4}	0.8775585892	2.4×10^{-5}
10^{-5}	0.8775801647	2.4×10^{-6}
10^{-6}	0.8775823222	2.4×10^{-7}

where $E(f;a,h) = f'(a) - \bigl(f(a+h) - f(a)\bigr)/h$. We observe that the approximation improves with decreasing h, as expected. More precisely, when h is reduced by a factor of 10, the error is reduced by the same factor. ∎

11.1.2 The truncation error

Whenever we use approximations, it is important to try and keep track of the error, if at all possible. To analyse the error in numerical differentiation, Taylor polynomials with remainders are useful. We start by doing a linear Taylor expansion of $f(a+h)$ about $x = a$ which results in the relation

$$f(a+h) = f(a) + hf'(a) + \frac{h^2}{2} f''(\xi_h), \tag{11.4}$$

where ξ_h lies in the interval $(a, a+h)$. This formula may be rearranged to give an expression for the error,

$$f'(a) - \frac{f(a+h) - f(a)}{h} = -\frac{h}{2} f''(\xi_h). \tag{11.5}$$

This is often referred to as the *truncation error* of the approximation.

Example 11.4. Let us check that the error formula (11.5) agrees with the numerical values in example 11.3. We have $f''(x) = -\sin x$, so the right-hand side in (11.5) becomes

$$E(\sin; 0.5, h) = \frac{h}{2} \sin \xi_h,$$

where $\xi_h \in (0.5, 0.5+h)$. We do not know the exact value of ξ_h, but for the values of h in question, we know that $\sin x$ is monotone on this interval. For $h = 0.1$ we therefore have that the error must lie in the interval

$$[0.05 \sin 0.5, \, 0.05 \sin 0.6] = [2.397 \times 10^{-2}, \, 2.823 \times 10^{-2}],$$

and we see that the right end point of the interval is the maximum value of the right-hand side in (11.5).

When h is reduced by a factor of 10, the number $h/2$ is reduced by the same factor, while ξ_h is restricted to an interval whose width is also reduced by a factor of 10. As h becomes even smaller, the number ξ_h will approach 0.5 so $\sin \xi_h$ will approach the lower value $\sin 0.5 \approx 0.479426$. For $h = 10^{-n}$, the error will therefore tend to

$$\frac{10^{-n}}{2} \sin 0.5 \approx \frac{0.2397}{10^n},$$

which is in close agreement with the numbers computed in example 11.3. ∎

The observation at the end of example 11.4 is true in general: If f'' is continuous, then ξ_h will approach a when h goes to zero. But even for small, positive values of h, the error in using the approximation $f''(\xi_h) \approx f''(a)$ is usually acceptable. This is the case since we are almost always only interested in knowing the approximate magnitude of the error, i.e., it is sufficient to know the error with one or two correct digits.

Observation 11.5. *The truncation error when using Newton's quotient to approximate $f'(a)$ is given approximately by*

$$\left| f'(a) - \frac{f(a+h)-f(a)}{h} \right| \approx \frac{h}{2} \left| f''(a) \right|. \tag{11.6}$$

An upper bound on the truncation error

For practical purposes, the approximation (11.6) is usually sufficient. But let us also take the time to present a more precise argument. We will use a technique from chapter 9 and derive an upper bound on the truncation error.

We go back to (11.5) and start by taking absolute values,

$$\left| f'(a) - \frac{f(a+h)-f(a)}{h} \right| = \frac{h}{2} \left| f''(\xi_h) \right|.$$

We know that ξ_h is a number in the interval $(a, a+h)$, so it is natural to replace $|f''(\xi_h)|$ by its maximum in this interval. Here we must be a bit careful since this maximum does not always exist. But recall from the Extreme value theorem that if a function is continuous, then it always attains its maximum on any closed and bounded interval. It is therefore natural to include the end points of the interval $(a, a+h)$ and take the maximum over $[a, a+h]$. This leads to the following lemma.

Lemma 11.6. *Suppose f has continuous derivatives up to order two near a. If the derivative $f'(a)$ is approximated by*

$$\frac{f(a+h)-f(a)}{h},$$

then the truncation error *is bounded by*

$$E(f;a,h) = \left| f'(a) - \frac{f(a+h)-f(a)}{h} \right| \le \frac{h}{2} \max_{x\in[a,a+h]} \left| f''(x) \right|. \tag{11.7}$$

11.1.3 The round-off error

So far, we have just considered the mathematical error committed when $f'(a)$ is approximated by $\big(f(a+h)-f(a)\big)/h$. But what about the round-off error? In fact, when we compute this approximation with small values of h we have to perform the one critical operation $f(a+h)-f(a)$, i.e., subtraction of two almost equal numbers, which we know from chapter 5 may lead to large round-off errors. Let us continue the calculations in example 11.3 and see what happens if we use smaller values of h.

Example 11.7. Recall that we estimated the derivative of $f(x) = \sin x$ at $a = 0.5$ and that the correct value with ten digits is $f'(0.5) \approx 0.8775825619$. If we check values of h for 10^{-7} and smaller we find

h	$\big(f(a+h)-f(a)\big)/h$	$E(f;a,h)$
10^{-7}	0. 8775825372	2.5×10^{-8}
10^{-8}	0.8775825622	-2.9×10^{-10}
10^{-9}	0.8775825622	-2.9×10^{-10}
10^{-11}	0.8775813409	1.2×10^{-6}
10^{-14}	0.8770761895	5.1×10^{-4}
10^{-15}	0.8881784197	-1.1×10^{-2}
10^{-16}	1.110223025	-2.3×10^{-1}
10^{-17}	0.000000000	8.8×10^{-1}

This shows very clearly that something quite dramatic happens. Ultimately, when we come to $h = 10^{-17}$, the derivative is computed as zero. ∎

Round-off errors in the function values

Let us see if we can explain what happened in example 11.7. We will go through the explanation for a general function, but keep the concrete example in mind.

The function value $f(a)$ will usually not be representable exactly in the computer and will therefore be replaced by the nearest floating-point number which we denote $\overline{f(a)}$. We then know from lemma 5.21 that the relative error in this approximation will be bounded by 5×2^{-53} since floating-point numbers are represented in binary ($\beta = 2$) with 53 bits for the significand ($m = 53$). In other words, if we set

$$\epsilon_1 = \frac{\overline{f(a)} - f(a)}{f(a)}, \tag{11.8}$$

we have

$$|\epsilon_1| \le 5\times2^{-53} \approx 6\times10^{-16}. \tag{11.9}$$

This means that $|\epsilon_1|$ is the relative error, while ϵ_1 itself is the signed relative error.

Note that ϵ_1 will depend both on a and f, and in practice, there will usually be better upper bounds on ϵ_1 than the one in (11.9). In the following we will denote the least upper bound by ϵ^*.

Notation 11.8. *The maximum relative error that occurs when real numbers are represented by floating-point numbers, and there is no underflow or overflow, is denoted by ϵ^*.*

We will see later in this chapter that a reasonable estimate for ϵ^* is $\epsilon^* \approx 7 \times 10^{-17}$. We note that equation (11.8) may be rewritten in a form that will be more convenient for us.

Observation 11.9. *Suppose that $f(a)$ is computed with 64-bit floating-point numbers and that no underflow or overflow occurs. Then the computed value $\overline{f(a)}$ satisfies*

$$\overline{f(a)} = f(a)(1+\epsilon_1) \tag{11.10}$$

where $|\epsilon_1| \le \epsilon^*$, and ϵ_1 depends on both a and f.

The computation of $f(a+h)$ is of course also affected by round-off error, so in total we have

$$\overline{f(a)} = f(a)(1+\epsilon_1), \quad \overline{f(a+h)} = f(a+h)(1+\epsilon_2), \tag{11.11}$$

where $|\epsilon_i| \le \epsilon^*$ for $i = 1, 2$. Here we should really write $\epsilon_2 = \epsilon_2(h)$, because the exact round-off error in $\overline{f(a+h)}$ will inevitably depend on h in an apparently random way.

Round-off errors in the computed derivative

The next step is to see how these errors affect the computed approximation of $f'(a)$. Recall from example 5.12 that the main source of round-off in subtraction is the replacement of the numbers to be subtracted by the nearest floating-point numbers. We therefore consider the computed approximation to be

$$\frac{\overline{f(a+h)} - \overline{f(a)}}{h},$$

and ignore the error in the division by h. If we insert the expressions (11.11), and also make use of equation (11.5), we obtain

$$\begin{aligned} f'(a) - \frac{\overline{f(a+h)} - \overline{f(a)}}{h} &= f'(a) - \frac{f(a+h) - f(a)}{h} - \frac{f(a+h)\epsilon_2 - f(a)\epsilon_1}{h} \\ &= -\frac{h}{2} f''(\xi_h) - \frac{f(a+h)\epsilon_2 - f(a)\epsilon_1}{h}, \end{aligned} \tag{11.12}$$

where $\xi_h \in (a, a+h)$. This shows that the total error in the computed approximation to the derivative consists of two parts: The truncation error that we derived in the previous section, plus the last term on the right in (11.12), which is due to the round-off when real numbers are replaced by floating-point numbers. The truncation error is proportional to h and therefore tends to 0 when h tends to 0. The error due to round-off however, is proportional to $1/h$ and therefore becomes large when h tends to 0.

In observation 11.5 we obtained an approximate expression for the truncation error, for small values of h, by replacing ξ_h by a. When h is small we may also assume that $f(a+h) \approx f(a)$ so (11.12) leads to the approximate error estimate

$$f'(a) - \frac{\overline{f(a+h)} - \overline{f(a)}}{h} \approx -\frac{h}{2} f''(a) - \frac{\epsilon_2 - \epsilon_1}{h} f(a). \tag{11.13}$$

The most uncertain term in (11.13) is the difference $\epsilon_2 - \epsilon_1$. Since we do not even know the signs of the two numbers ϵ_1 and ϵ_2, we cannot estimate this difference accurately. But we do know that both numbers represent relative errors in floating-point numbers, so the magnitude of each is about 10^{-17}. If they are of opposite signs, this magnitude may be doubled, so we replace the difference $\epsilon_2 - \epsilon_1$ by $2\tilde{\epsilon}(h)$ to emphasise the dependence on h. The error (11.13) then becomes

$$f'(a) - \frac{\overline{f(a+h)} - \overline{f(a)}}{h} \approx -\frac{h}{2} f''(a) - \frac{2\tilde{\epsilon}(h)}{h} f(a). \tag{11.14}$$

Let us check if this agrees with the computations in examples 11.3 and 11.7.

Example 11.10. For large values of h the first term on the right in (11.14) will dominate the error, and we have already seen that this agrees very well with the computed values in example 11.3. The question is how well the numbers in example 11.7 can be modelled when h becomes smaller.

To investigate this, we denote the left-hand side of (11.14) by $E(f;a,h)$ and solve for $\tilde{\epsilon}(h)$,

$$\tilde{\epsilon}(h) \approx -\frac{h}{2f(a)}\Big(E(f;a,h) + \frac{h}{2} f''(a)\Big).$$

From example 11.7 we have corresponding values of h and $E(f;a,h)$ which allow us to estimate $\tilde{\epsilon}(h)$ (recall that $f(x) = \sin x$ and $a = 0.5$ in this example). If we do

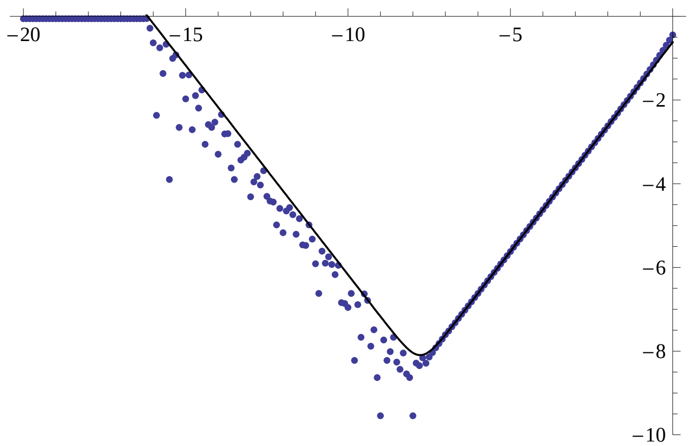

Figure 11.2. Numerical approximation of the derivative of $f(x) = \sin x$ at $x = 0.5$ using Newton's quotient, see lemma 11.6. The plot is a $\log_{10}$-$\log_{10}$ plot which shows the logarithm to base 10 of the absolute value of the total error as a function of the logarithm to base 10 of h, based on 200 values of h. The point −10 on the horizontal axis therefore corresponds $h = 10^{-10}$, and the point −6 on the vertical axis corresponds to an error of 10^{-6}. The solid line is a plot of the error estimate $g(h)$ given by (11.15).

this we can augment the table on page 257 with an additional column

h	$\big(f(a+h)-f(a)\big)/h$	$E(f;a,h)$	$\tilde{\epsilon}(h)$
10^{-7}	0. 8775825372	2.5×10^{-8}	-7.6×10^{-17}
10^{-8}	0.8775825622	-2.9×10^{-10}	2.8×10^{-17}
10^{-9}	0.8775825622	-2.9×10^{-10}	5.5×10^{-19}
10^{-11}	0.8775813409	1.2×10^{-6}	-1.3×10^{-17}
10^{-14}	0.8770761895	5.1×10^{-4}	-5.3×10^{-18}
10^{-15}	0.8881784197	-1.1×10^{-2}	1.1×10^{-17}
10^{-16}	1.110223025	-2.3×10^{-1}	2.4×10^{-17}
10^{-17}	0.000000000	8.8×10^{-1}	-9.2×10^{-18}

We observe that all these values are considerably smaller than the upper limit 6×10^{-16} in (11.9). (Note that in order to compute $\tilde{\epsilon}(h)$ correctly for $h = 10^{-7}$, you need to use the more accurate value 2.4695×10^{-8} for the error in this case.)

Figure 11.2 shows plots of the error. The numerical approximation has been computed for the values $h = 10^{-z}$, for $z = 0, \ldots, 20$ in steps of $1/10$, and the absolute value of the total error plotted in a log-log plot. The errors are shown as isolated dots, and the function

$$g(h) = \frac{h}{2}\sin 0.5 + \epsilon\frac{2}{h}\sin 0.5 \tag{11.15}$$

with $\epsilon = 7\times 10^{-17}$ is shown as a solid graph. This corresponds to adding the absolute value of the truncation error and the round-off error, even in the case where they have opposite signs. It appears that the choice of ϵ makes $g(h)$ a reasonable upper bound on the error so we may consider this to be a decent estimate of ϵ^*. ∎

The estimates (11.13) and (11.14) give the approximate error with sign. In general, it is more convenient to consider the absolute value of the error. Starting with (11.13), we then have

$$
\begin{aligned}
\left| f'(a) - \frac{\overline{f(a+h)} - \overline{f(a)}}{h} \right| &\approx \left| -\frac{h}{2} f''(a) - \frac{\epsilon_2 - \epsilon_1}{h} f(a) \right| \\
&\leq \frac{h}{2} |f''(a)| + \frac{|\epsilon_2 - \epsilon_1|}{h} |f(a)| \\
&\leq \frac{h}{2} |f''(a)| + \frac{|\epsilon_2| + |\epsilon_1|}{h} |f(a)| \\
&\leq \frac{h}{2} |f''(a)| + \frac{2\epsilon(h)}{h} |f(a)|
\end{aligned}
$$

where we used the triangle inequality in the first and second inequality, and $\epsilon(h)$ is the largest of the two numbers $|\epsilon_1|$ and $|\epsilon_2|$.

Observation 11.11. *Suppose that f and its first two derivatives are continuous near a. When the derivative of f at a is approximated by Newton's difference quotient* (11.3)*, the error in the computed approximation is roughly bounded by*

$$
\left| f'(a) - \frac{\overline{f(a+h)} - \overline{f(a)}}{h} \right| \lesssim \frac{h}{2} \left| f''(a) \right| + \frac{2\epsilon(h)}{h} \left| f(a) \right|, \tag{11.16}
$$

where $\epsilon(h)$ is the largest of the relative errors in $\overline{f(a)}$ and $\overline{f(a+h)}$, and the notation $\alpha \lesssim \beta$ indicates that α is approximately smaller than β.

An upper bound on the total error

The $\lesssim$ notation is vague mathematically, so we include a more precise error estimate.

Theorem 11.12. *Suppose that f and its first two derivatives are continuous near a. When the derivative of f at a is approximated by*

$$
\frac{f(a+h) - f(a)}{h},
$$

the error in the computed approximation is bounded by

$$\left| f'(a) - \frac{\overline{f(a+h)} - \overline{f(a)}}{h} \right| \le \frac{h}{2} M_1 + \frac{2\epsilon^*}{h} M_2, \tag{11.17}$$

where

$$M_1 = \max_{x\in[a,a+h]} \left| f''(x) \right|, \quad M_2 = \max_{x\in[a,a+h]} \left| f(x) \right|.$$

Proof. To get to (11.17) we start with (11.12), take absolute values, and use the triangle inequality a number of times. We also replace $\left|f''(\xi_h)\right|$ by its maximum on the interval $[a, a+h]$, and we replace $f(a)$ and $f(a+h)$ by their common maximum on $[a, a+h]$. The details are:

$$\begin{aligned}
\left| f'(a) - \frac{\overline{f(a+h)} - \overline{f(a)}}{h} \right| &= \left| \frac{h}{2} f''(\xi_h) - \frac{f(a+h)\epsilon_2 - f(a)\epsilon_1}{h} \right| \\
&\le \frac{h}{2} |f''(\xi_h)| + \frac{|f(a+h)\epsilon_2 - f(a)\epsilon_1|}{h} \\
&\le \frac{h}{2} |f''(\xi_h)| + \frac{|f(a+h)||\epsilon_2| + |f(a)||\epsilon_1|}{h} \\
&\le \frac{h}{2} M_1 + \frac{M_2|\epsilon_2| + M_2|\epsilon_1|}{h} \\
&= \frac{h}{2} M_1 + \frac{|\epsilon_2| + |\epsilon_1|}{h} M_2 \\
&\le \frac{h}{2} M_1 + \frac{2\epsilon^*}{h} M_2. \quad \blacksquare
\end{aligned} \tag{11.18}$$

11.1.4 Optimal choice of h

Figure 11.2 indicates that there is an optimal value of h which minimises the total error. We can find a decent estimate for this h by minimising the upper bound in one of the error estimates (11.16) or (11.17). In practice it is easiest to use (11.16) since the two numbers M_1 and M_2 in (11.17) depend on h (although we could insert some upper bound which is independent of h).

The right-hand side of (11.16) contains the term $\epsilon(h)$ whose exact dependence on h is very uncertain. We therefore replace $\epsilon(h)$ by the upper bound ϵ^*. This gives us the error estimate

$$e(h) = \frac{h}{2} \left| f''(a) \right| + \frac{2\epsilon^*}{h} \left| f(a) \right|. \tag{11.19}$$

To find the value of h which minimises this expression, we differentiate with respect to h and set the derivative to zero. We find

$$e'(h) = \frac{|f''(a)|}{2} - \frac{2\epsilon^*}{h^2}|f(a)|.$$

If we solve the equation $e'(h) = 0$, we obtain the approximate optimal value.

Lemma 11.13. *Let f be a function with continuous derivatives up to order 2. If the derivative of f at a is approximated as in lemma 11.6, then the value of h which minimises the total error (truncation error + round-off error) is approximately*

$$h^* \approx 2\frac{\sqrt{\epsilon^* |f(a)|}}{\sqrt{|f''(a)|}}.$$

It is easy to see that the optimal value of h is the value that balances the two terms in (11.19), i.e., the truncation error and the round-off error are equal.

Example 11.14. Based on example 11.7, we saw above that a good value of ϵ^* is 7×10^{-17}. Let us check what the optimal value of h is in this case. We have $f(x) = \sin x$ and $a = 0.5$ so

$$h^* = 2\sqrt{\epsilon} = 2\sqrt{7\times10^{-17}} \approx 1.7\times10^{-8}.$$

For this value of h we find

$$\frac{\sin(0.5+h^*) - \sin 0.5}{h^*} = 0.877582555644682,$$

and the error in this case is about 6.2×10^{-9}. It turns out that roughly all h in the interval $[3.2\times10^{-9}, 2\times10^{-8}]$ give an error of about the same magnitude which shows that the determination of h^* is quite robust. ∎

Exercises

1 In this exercise we are going to numerically compute the derivative of $f(x) = e^x$ at $a = 1$ using Newton's quotient as described in observation 11.2. The exact derivative to 20 digits is

$$f'(1) \approx 2.7182818284590452354.$$

a) Compute the approximation $\big(f(1+h) - f(1)\big)/h$ to $f'(1)$. Start with $h = 10^{-4}$, and then gradually reduce h. Also compute the error, and determine an h that gives close to minimal error.

b) Determine the optimal h as described in Lemma 11.13 and compare with the value you found in (a).

2 When deriving the truncation error given by (11.7) it is not obvious what the degree of the Taylor polynomial in (11.4) should be. In this exercise you are going to try and increase and reduce the degree of the Taylor polynomial and see what happens.

a) Redo the Taylor expansion in (11.4), but use the Taylor polynomial of degree 2. From this try and derive an error formula similar to (11.5).

b) Repeat (a), but use a Taylor polynomial of degree 0, i.e., just a constant.

c) Why can you conclude that the linear Taylor polynomial and the error term in (11.5) is correct?

11.2 Summary of the general strategy

Before we continue, let us sum up the derivation and analysis of the Newton's difference quotient in section 11.1, since this is standard for all differentiation methods.

The first step is to derive the numerical method. In section 11.1 this was very simple since the method came straight out of the definition of the derivative. Just before observation 11.2 we indicated that the method can also be derived by approximating f by a polynomial p and using $p'(a)$ as an approximation to $f'(a)$. This is the general approach that we will use below.

Once the numerical method is known, we estimate the mathematical error in the approximation, *the truncation error*. This we do by performing Taylor expansions with remainders. For numerical differentiation methods which provide estimates of a derivative at a point a, we replace all function values at points other than a by Taylor polynomials with remainders. There may be a challenge in choosing the correct degree of the Taylor polynomial, see exercise 11.1.2.

The next task is to estimate the total error, including the round-off error. We consider the difference between the derivative to be computed and the computed approximation, and replace the computed function evaluations by expressions like the ones in observation 11.9. This will result in an expression involving the mathematical approximation to the derivative. This can be simplified in the same way as when the truncation error was estimated, with the addition of an expression involving the relative round-off errors in the function evaluations. These estimates can then be simplified to something like (11.16) or (11.17). As a final step, the optimal value of h can be found by minimising the total error.

Procedure 11.15. *The following is a general procedure for deriving numerical methods for differentiation:*

1. *Interpolate the function f by a polynomial p at suitable points.*
2. *Approximate the derivative of f by the derivative of p. This makes it possible to express the approximation in terms of function values of f.*
3. *Derive an estimate for the error by expanding the function values (other than the one at a) in Taylor series with remainders.*
4. *Derive an estimate of the round-off error by assuming that the relative errors in the function values are bounded by ϵ^*. By minimising the total error, an optimal step length h can be determined.*

Exercises

1 Determine an approximation to the derivative $f'(a)$ using the function values $f(a)$, $f(a+h)$ and $f(a+2h)$ by interpolating f by a quadratic polynomial p_2 at the three points a, $a+h$, and $a+2h$, and then using $f'(a) \approx p_2'(a)$.

11.3 A symmetric version of Newton's quotient

The numerical differentiation method in section 11.1 is not symmetric about a, so let us try and derive a symmetric method.

11.3.1 Derivation of the method

We want to find an approximation to $f'(a)$ using values of f near a. To obtain a symmetric method, we assume that $f(a-h)$, $f(a)$, and $f(a+h)$ are known values, and we want to find an approximation to $f'(a)$ using these values. The strategy is to determine the quadratic polynomial p_2 that interpolates f at $a-h$, a and $a+h$, and then we use $p_2'(a)$ as an approximation to $f'(a)$.

We start by writing p_2 in Newton form,

$$\begin{aligned} p_2(x) = f[a-h] &+ f[a-h,a](x-(a-h)) \\ &+ f[a-h,a,a+h](x-(a-h))(x-a). \end{aligned} \tag{11.20}$$

We differentiate and find

$$p_2'(x) = f[a-h,a] + f[a-h,a,a+h](2x-2a+h).$$

Setting $x = a$ yields

$$p_2'(a) = f[a-h, a] + f[a-h, a, a+h]h.$$

To get a practically useful formula we must express the divided differences in terms of function values. If we expand the second divided difference we obtain

$$p_2'(a) = f[a-h, a] + \frac{f[a, a+h] - f[a-h, a]}{2h} h = \frac{f[a, a+h] + f[a-h, a]}{2}. \quad (11.21)$$

The two first order differences are

$$f[a-h, a] = \frac{f(a) - f(a-h)}{h}, \quad f[a, a+h] = \frac{f(a+h) - f(a)}{h},$$

and if we insert this in (11.21) we end up with

$$p_2'(a) = \frac{f(a+h) - f(a-h)}{2h}.$$

We note that the approximation to the derivative given by $p_2'(a)$ agrees with the slope of the secant based at $a-h$ and $a+h$.

Lemma 11.16 (Symmetric Newton's quotient)**.** *Let f be a given function, and let a and h be given numbers. If $f(a-h)$, $f(a)$, $f(a+h)$ are known values, then $f'(a)$ can be approximated by $p_2'(a)$ where p_2 is the quadratic polynomial that interpolates f at $a-h$, a, and $a+h$. The approximation is given by*

$$f'(a) \approx p_2'(a) = \frac{f(a+h) - f(a-h)}{2h}, \quad (11.22)$$

and agrees with the slope of the secant based at $a-h$ and $a+h$.

The symmetric Newton's quotient is illustrated in figure **??**. The derivative of f at a is given by the slope of the tangent, while the approximation defined by $p_2'(a)$ is given by the slope of tangent of the parabola at a (which is the same as the slope of the secant of f based at $a-h$ and $a+h$).

Let us test this approximation on the function $f(x) = \sin x$ at $a = 0.5$ so we can compare with the original Newton's quotient that we discussed in section 11.1.

Example 11.17. We test the approximation (11.22) with the same values of h as in examples 11.3 and 11.7. Recall that $f'(0.5) \approx 0.8775825619$ with 10 correct digits. The results are

h	$\big(f(a+h)-f(a-h)\big)/(2h)$	$E(f;a,h)$
10^{-1}	0.8761206554	1.5×10^{-3}
10^{-2}	0.8775679356	1.5×10^{-5}
10^{-3}	0.8775824156	1.5×10^{-7}
10^{-4}	0.8775825604	1.5×10^{-9}
10^{-5}	0.8775825619	1.8×10^{-11}
10^{-6}	0.8775825619	-7.5×10^{-12}
10^{-7}	0.8775825616	2.7×10^{-10}
10^{-8}	0.8775825622	-2.9×10^{-10}
10^{-11}	0.8775813409	1.2×10^{-6}
10^{-13}	0.8776313010	-4.9×10^{-5}
10^{-15}	0.8881784197	-1.1×10^{-2}
10^{-17}	0.0000000000	8.8×10^{-1}

If we compare with examples 11.3 and 11.7, the errors are generally smaller for the same value of h. In particular we note that when h is reduced by a factor of 10, the error is reduced by a factor of 100, at least as long as h is not too small. However, when h becomes smaller than about 10^{-6}, the error starts to increase. It therefore seems like the truncation error is smaller than for the original method based on Newton's quotient, but as before, the round-off error makes it impossible to get accurate results for small values of h. The optimal value of h seems to be $h^* \approx 10^{-6}$, which is larger than for the first method, but the error is then about 10^{-12}, which is smaller than the best we could do with the asymmetric Newton's quotient. ∎

11.3.2 The error

We analyse the error in the symmetric Newton's quotient in exactly the same way as we analysed the original Newton's quotient in section 11.1. The idea is to replace $f(a-h)$ and $f(a+h)$ with Taylor expansions about a. Some trial and error will reveal that the correct degree of the Taylor polynomials is quadratic, and the Taylor polynomials with remainders may be combined into the expression

$$f'(a) - \frac{f(a+h)-f(a-h)}{2h} = -\frac{h^2}{12}\big(f'''(\xi_1)+f'''(\xi_2)\big). \tag{11.23}$$

The error formula (11.23) confirms the numerical behaviour we saw in example 11.17 for small values of h since the error is proportional to h^2: When h is reduced by a factor of 10, the error is reduced by a factor 10^2.

The analysis of the round-off error is completely analogous to what we did in section 11.1.3: Start with (11.23), take into account round-off errors, obtain a relation similar to (11.12), and then derive the error estimate through a string of equalities and inequalities as in (11.18). The result is a theorem similar to theorem 11.12.

Theorem 11.18. *Let f be a given function with continuous derivatives up to order three, and let a and h be given numbers. Then the error in the symmetric Newton's quotient approximation to $f(a)$,*

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

including round-off error and truncation error, is bounded by

$$\left| f'(a) - \frac{\overline{f(a+h)} - \overline{f(a-h)}}{2h} \right| \leq \frac{h^2}{6} M_1 + \frac{\epsilon^*}{h} M_2, \tag{11.24}$$

where

$$M_1 = \max_{x \in [a-h, a+h]} \left| f'''(x) \right|, \quad M_2 = \max_{x \in [a-h, a+h]} \left| f(x) \right|. \tag{11.25}$$

The most important feature of this theorem is that it shows how the error depends on h. The first term on the right in (11.24) stems from the truncation error (11.23) which clearly is proportional to h^2, while the second term corresponds to the round-off error and depends on h^{-1} because we divide by h when calculating the approximation.

It may be a bit surprising that the truncation error is smaller for the symmetric Newton's quotient than for the asymmetric one, since both may be viewed as coming from a secant approximation to f. The reason is that in the symmetric case, the secant is just a special case of a parabola which is generally a better approximation than a straight line.

In practice, the interesting values of h will usually be so small that there is very little error in using the approximations

$$M_1 = \max_{x \in [a-h, a+h]} \left| f'''(x) \right| \approx \left| f'''(a) \right|, \quad M_2 = \max_{x \in [a-h, a+h]} \left| f(x) \right| \approx \left| f(a) \right|,$$

in (11.24), particularly since we are only interested in the magnitude of the error with only 1 or 2 digits of accuracy. If we make these simplifications we obtain a slightly simpler error estimate.

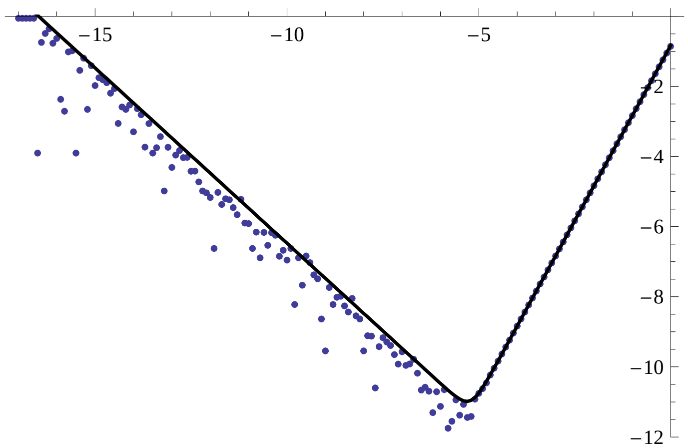

Figure 11.3. Log-log plot of the error in the approximation to the derivative of $f(x) = \sin x$ at $x = 1/2$ for values of h in the interval $[0, 10^{-17}]$, using the symmetric Newton's quotient in theorem 11.18. The solid graph represents the right-hand side of (11.26) with $\epsilon^* = 7 \times 10^{-17}$, as a function of h.

Observation 11.19. *The error in the symmetric Newton's quotient is approximately bounded by*

$$\left| f'(a) - \frac{\overline{f(a+h)} - \overline{f(a-h)}}{2h} \right| \lesssim \frac{h^2}{6} \left| f'''(a) \right| + \frac{\epsilon^* \left| f(a) \right|}{h}. \tag{11.26}$$

A plot of how the error behaves in the symmetric Newton's quotient, together with the estimate of the error on the right in (11.26), is shown in figure 11.3.

11.3.3 Optimal choice of h

As for the asymmetric Newton's quotient, we can find an optimal value of h which minimises the error. We can find this value of h if we differentiate the right-hand side of (11.24) with respect to h and set the derivative to 0. This leads to the equation

$$\frac{h}{3} M_1 - \frac{\epsilon^*}{h^2} M_2 = 0$$

which has the solution

$$h^* = \frac{\sqrt[3]{3\epsilon^* M_2}}{\sqrt[3]{M_1}} \approx \frac{\sqrt[3]{3\epsilon^* \left| f(a) \right|}}{\sqrt[3]{\left| f'''(a) \right|}}. \tag{11.27}$$

At the end of section 11.1.4 we saw that a reasonable value for ϵ^* was $\epsilon^* = 7 \times 10^{-17}$. The optimal value of h in example 11.17, where $f(x) = \sin x$ and $a = 0.5$, then becomes $h = 4.6 \times 10^{-6}$. For this value of h the approximation is $f'(0.5) \approx 0.877582561887$ with error 3.1×10^{-12}.

Exercises

1 In this exercise we are going to check the symmetric Newton's quotient and numerically compute the derivative of $f(x) = e^x$ at $a = 1$, see exercise 11.1.1. Recall that the exact derivative with 20 correct digits is

$$f'(1) \approx 2.71828182845904523\,54.$$

a) Compute the approximation $\big(f(1+h) - f(1-h)\big)/(2h)$ to $f'(1)$. Start with $h = 10^{-3}$, and then gradually reduce h. Also compute the error, and determine an h that gives close to minimal error.

b) Determine the optimal h given by (11.27) and compare with the value you found in (a).

2 Determine $f'(a)$ numerically using the two asymmetric Newton's quotients

$$f_r(x) = \frac{f(a+h) - f(a)}{h}, \qquad f_l(x) = \frac{f(a) - f(a-h)}{h}$$

as well as the symmetric Newton's quotient. Also compute and compare the relative errors in each case.

a) $f(x) = x^2$; $a = 2$; $h = 0.01$.

b) $f(x) = \sin x$; $a = \pi/3$; $h = 0.1$.

c) $f(x) = \sin x$; $a = \pi/3$; $h = 0.001$.

d) $f(x) = \sin x$; $a = \pi/3$; $h = 0.00001$.

e) $f(x) = 2^x$; $a = 1$; $h = 0.0001$.

f) $f(x) = x \cos x$; $a = \pi/3$; $h = 0.0001$.

3 In this exercise we are going to derive the error estimate (11.24). For this it is a good idea to use the derivation in sections 11.1.2 and 11.1.3 as a model, and try and follow the same strategy.

a) Derive the relation (11.23) by replacing $f(a-h)$ and $f(a+h)$ with appropriate Taylor polynomials with remainders around $x = a$.

b) Estimate the total error by replacing the values $f(a-h)$ and $f(a+h)$ by the nearest floating-point numbers $\overline{f(a-h)}$ and $\overline{f(a+h)}$. The result should be a relation similar to equation (11.12).

c) Find an upper bound on the total error by using the same steps as in (11.18).

4 **a)** Show that the approximation to $f'(a)$ given by the symmetric Newton's quotient is the average of the two asymmetric quotients

$$f_r(x) = \frac{f(a+h) - f(a)}{h}, \qquad f_l(x) = \frac{f(a) - f(a-h)}{h}.$$

b) Sketch the graph of the function

$$f(x) = \frac{-x^2 + 10x - 5}{4}$$

on the interval $[0, 6]$ together with the three secants associated with the three approximations to the derivative in (a) (use $a = 3$ and $h = 2$). Can you from this judge which approximation is best?

c) Determine the three difference quotients in (a) numerically for the function $f(x)$ using $a = 3$ and $h_1 = 0.1$ and $h_2 = 0.001$. What are the relative errors?

d) Show that the symmetric Newton's quotient at $x = a$ for a quadratic function $f(x) = ax^2 + bx + c$ is equal to the derivative $f'(a)$.

5 Use the symmetric Newton's quotient and determine an approximation to the derivative $f'(a)$ in each case below. Use the values of h given by $h = 10^{-k}$ $k = 4, 5, \ldots, 12$ and compare the relative errors. Which of these values of h gives the smallest error? Compare with the optimal h predicted by (11.27).

a) The function $f(x) = 1/(1 + \cos(x^2))$ at the point $a = \pi/4$.

b) The function $f(x) = x^3 + x + 1$ at the point $a = 0$.

11.4 A four-point differentiation method

The asymmetric and symmetric Newton's quotients are the two most commonly used methods for approximating derivatives. Whenever possible, one would prefer the symmetric version whose truncation error is proportional to h^2. This means that the error goes to 0 more quickly than for the asymmetric version, as was clearly evident in examples 11.3 and 11.17. In this section we derive another method for which the truncation error is proportional to h^4. This also illustrates the procedure 11.15 in a more complicated situation.

The computations below may seem overwhelming, and have in fact been done with the help of a computer to save time and reduce the risk of miscalculations. The method is included here just to illustrate that the principle for deriving both the method and the error terms is just the same as for the simpler Newton's quotient.

11.4.1 Derivation of the method

We want better accuracy than the symmetric Newton's quotient which was based on interpolation with a quadratic polynomial. It is therefore natural to base the approximation on a cubic polynomial, which can interpolate four points. We have seen the advantage of symmetry, so we choose the interpolation points

$x_0 = a-2h$, $x_1 = a-h$, $x_2 = a+h$, and $x_3 = a+2h$. The cubic polynomial that interpolates f at these points is

$$p_3(x) = f(x_0) + f[x_0,x_1](x-x_0) + f[x_0,x_1,x_2](x-x_0)(x-x_1) \\ + f[x_0,x_1,x_2,x_3](x-x_0)(x-x_1)(x-x_2),$$

and its derivative is

$$p_3'(x) = f[x_0,x_1] + f[x_0,x_1,x_2](2x-x_0-x_1) \\ + f[x_0,x_1,x_2,x_3]\big((x-x_1)(x-x_2)+(x-x_0)(x-x_2)+(x-x_0)(x-x_1)\big).$$

If we evaluate this expression at $x=a$ and simplify (this is quite a bit of work), we find that the resulting approximation of $f'(a)$ is

$$f'(a) \approx p_3'(a) = \frac{f(a-2h) - 8f(a-h) + 8f(a+h) - f(a+2h)}{12h}. \tag{11.28}$$

11.4.2 The error

To estimate the error, we expand the four terms in the numerator in (11.28) in Taylor polynomials of degree 4 with remainders. We then insert these into the formula for $p_3'(a)$ and obtain an analog to equation 11.12,

$$f'(a) - \frac{f(a-2h) - 8f(a-h) + 8f(a+h) - f(a+2h)}{12h} = \\ \frac{h^4}{45}f^{(v)}(\xi_1) - \frac{h^4}{180}f^{(v)}(\xi_2) - \frac{h^4}{180}f^{(v)}(\xi_3) + \frac{h^4}{45}f^{(v)}(\xi_4),$$

where $\xi_1 \in (a-2h,a)$, $\xi_2 \in (a-h,a)$, $\xi_3 \in (a,a+h)$, and $\xi_4 \in (a,a+2h)$. We can simplify the right-hand side and obtain an upper bound on the truncation error if we replace the function values by upper bounds. The result is

$$\left| f'(a) - \frac{f(a-2h) - 8f(a-h) + 8f(a+h) - f(a+2h)}{12h} \right| \le \frac{h^4}{18}M \tag{11.29}$$

where

$$M = \max_{x\in[a-2h,a+2h]} \left| f^{(v)}(x) \right|.$$

The round-off error is derived in the same way as before. The quantities we actually compute are

$$\overline{f(a-2h)} = f(a-2h)(1+\epsilon_1), \quad \overline{f(a+2h)} = f(a+2h)(1+\epsilon_3),$$
$$\overline{f(a-h)} = f(a-h)(1+\epsilon_2), \quad \overline{f(a+h)} = f(a+h)(1+\epsilon_4).$$

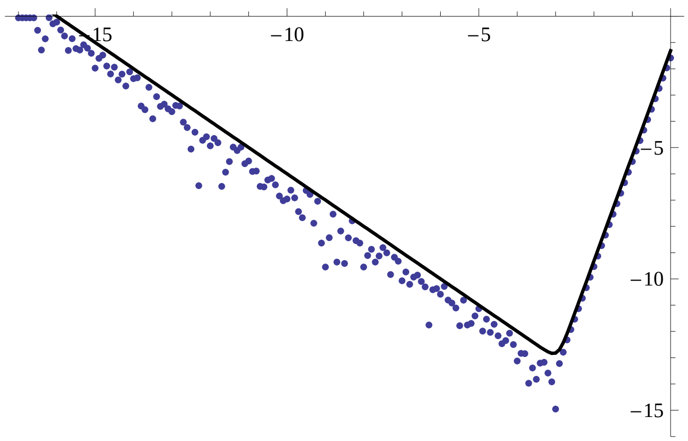

Figure 11.4. Log-log plot of the error in the approximation to the derivative of $f(x) = \sin x$ at $x = 1/2$, using the method in observation 11.20, with h in the interval $[0, 10^{-17}]$. The function plotted is the right-hand side of (11.30) with $\epsilon^* = 7 \times 10^{-17}$.

We estimate the difference between $f'(a)$ and the computed approximation, make use of the estimate (11.29), combine the function values that are multiplied by ϵs, and approximate the maximum values by function values at a, completely analogously to what we did for Newton's quotient.

Observation 11.20. *Suppose that f and its first five derivatives are continuous. If $f'(a)$ is approximated by*

$$f'(a) \approx \frac{f(a-2h) - 8f(a-h) + 8f(a+h) - f(a+2h)}{12h},$$

the total error is approximately bounded by

$$\left| f'(a) - \frac{\overline{f(a-2h)} - 8\overline{f(a-h)} + 8\overline{f(a+h)} - \overline{f(a+2h)}}{12h} \right| \lesssim \frac{h^4}{18} \left| f^{(v)}(a) \right| + \frac{3\epsilon^*}{h} \left| f(a) \right|. \quad (11.30)$$

We could of course also derive a more formal upper bound on the error, similar to (11.17) and (11.24).

A plot of the error in the approximation for the $\sin x$ example that we used for the previous approximations is shown in figure 11.4.

From observation 11.20 we can compute the optimal value of h by differentiating the right-hand side with respect to h and setting it to zero. This leads to the equation

$$\frac{2h^3}{9}\left|f^{(v)}(a)\right| - \frac{3\epsilon^*}{h^2}\left|f(a)\right| = 0$$

which has the solution

$$h^* = \frac{\sqrt[5]{27\epsilon^*\left|f(a)\right|}}{\sqrt[5]{2\left|f^{(v)}(a)\right|}}. \tag{11.31}$$

For the example $f(x) = \sin x$ and $a = 0.5$ the optimal value of h is $h^* \approx 8.8 \times 10^{-4}$. The actual error is then roughly 10^{-14}.

Exercises

1 In this exercise we are going to check the four-point method and numerically compute the derivative of $f(x) = e^x$ at $a = 1$. For comparison, the exact derivative to 20 digits is

$$f'(1) \approx 2.7182818284590452354.$$

a) Compute the approximation $\big(f(a-2h) - 8f(a-h) + 8f(a+h) - f(a+2h)\big)/(12h)$ to $f'(1)$. Start with $h = 10^{-3}$, and then gradually reduce h. Also compute the error, and determine an h that gives close to minimal error.

b) Determine the optimal h given by (11.31) and compare with the experimental value you found in (a).

2 Derive the estimate (11.29), starting with the relation just preceding (11.29).

11.5 Numerical approximation of the second derivative

We consider one more method for numerical approximation of derivatives, this time of the second derivative. The approach is the same: We approximate f by a polynomial and approximate the second derivative of f by the second derivative of the polynomial. As in the other cases, the error analysis is based on expansion in Taylor series.

11.5.1 Derivation of the method

Since we are going to find an approximation to the second derivative, we have to approximate f by a polynomial of degree at least two, otherwise the second derivative is identically 0. The simplest is therefore to use a quadratic polynomial, and for symmetry we want it to interpolate f at $a-h$, a, and $a+h$. The resulting polynomial p_2 is the one we used in section 11.3 and it is given in equation (11.20). The second derivative of p_2 is constant, and the approximation of $f'(a)$ is

$$f''(a) \approx p_2''(a) = f[a-h, a, a+h].$$

The divided difference is easy to expand.

Lemma 11.21 (Three-point approximation of second derivative)**.** *The second derivative of a function f at a can be approximated by*

$$f''(a) \approx \frac{f(a+h) - 2f(a) + f(a-h)}{h^2}. \tag{11.32}$$

11.5.2 The error

Estimation of the error follows the same pattern as before. We replace $f(a-h)$ and $f(a+h)$ by cubic Taylor polynomials with remainders and obtain an expression for the truncation error,

$$f''(a) - \frac{f(a+h) - 2f(a) + f(a-h)}{h^2} = -\frac{h^2}{24}\big(f^{(iv)}(\xi_1) + f^{(iv)}(\xi_2)\big), \tag{11.33}$$

where $\xi_1 \in (a-h, a)$ and $\xi_2 \in (a, a+h)$.

The round-off error can also be estimated as before. Instead of computing the exact values, we actually compute $\overline{f(a-h)}$, $\overline{f(a)}$, and $\overline{f(a+h)}$, which are linked to the exact values by

$$\overline{f(a-h)} = f(a-h)(1+\epsilon_1), \quad \overline{f(a)} = f(a)(1+\epsilon_2), \quad \overline{f(a+h)} = f(a+h)(1+\epsilon_3),$$

where $|\epsilon_i| \le \epsilon^*$ for $i = 1, 2, 3$. We can then derive a relation similar to (11.12), and by reasoning as in (11.18) we end up with an estimate of the total error.

Theorem 11.22. *Suppose f and its first three derivatives are continuous near a, and that $f''(a)$ is approximated by*

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

Then the total error (truncation error + round-off error) in the computed approximation is bounded by

$$\left| f''(a) - \frac{\overline{f(a+h)} - 2\overline{f(a)} + \overline{f(a-h)}}{h^2} \right| \le \frac{h^2}{12} M_1 + \frac{3\epsilon^*}{h^2} M_2, \tag{11.34}$$

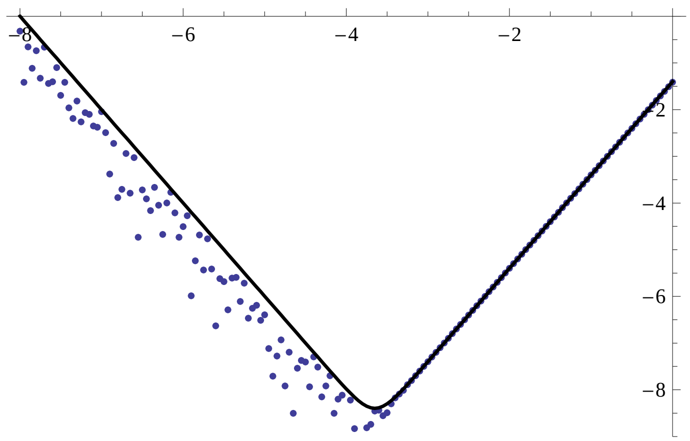

Figure 11.5. Log-log plot of the error in the approximation to the derivative of $f(x) = \sin x$ at $x = 1/2$ for h in the interval $[0, 10^{-8}]$, using the method in theorem 11.22. The function plotted is the right-hand side of (11.30) with $\epsilon^* = 7\times 10^{-17}$.

where

$$M_1 = \max_{x\in[a-h,a+h]} \left|f^{(iv)}(x)\right|, \quad M_2 = \max_{x\in[a-h,a+h]} \left|f(x)\right|.$$

As for the previous methods, we can simplify the right-hand side to

$$\left|f''(a) - \frac{\overline{f(a+h)} - 2\overline{f(a)} + \overline{f(a-h)}}{h^2}\right| \lesssim \frac{h^2}{12}\left|f^{(iv)}(a)\right| + \frac{3\epsilon^*}{h^2}\left|f(a)\right| \tag{11.35}$$

if we can tolerate an approximate upper bound.

Figure 11.5 shows the errors in the approximation to the second derivative given in theorem 11.22 when $f(x) = \sin x$ and $a = 0.5$, and for h in the range $[0, 10^{-8}]$. The solid graph gives the function in (11.35) which describes an approximate upper bound on the error as a function of h, with $\epsilon^* = 7\times 10^{-17}$. For h smaller than 10^{-8}, the approximation becomes 0, and the error constant. Recall that for the approximations to the first derivative, this did not happen until h was about 10^{-17}. This illustrates the fact that the higher the derivative, the more problematic is the round-off error, and the more difficult it is to approximate the derivative with numerical methods like the ones we study here.

11.5.3 Optimal value of h

As before, we find the optimal value of h by minimising the right-hand side of (11.35). To do this we find the derivative with respect to h and set it to 0,

$$\frac{h}{6}\left|f'''(a)\right| - \frac{6\epsilon^*}{h^3}\left|f(a)\right| = 0.$$

Observation 11.23. *The upper bound on the total error* (11.34) *is minimised when h has the value*

$$h^* = \frac{\sqrt[4]{36\epsilon^*\left|f(a)\right|}}{\sqrt[4]{\left|f^{(iv)}(a)\right|}}. \tag{11.36}$$

When $f(x) = \sin x$ and $a = 0.5$ this gives $h^* = 2.2 \times 10^{-4}$ if we use the value $\epsilon^* = 7 \times 10^{-17}$. Then the approximation to $f''(a) = -\sin a$ is -0.4794255352 with an actual error of 3.4×10^{-9}.

Exercises

1 We use our standard example $f(x) = e^3$ and $a = 1$ to check the 3-point approximation to the second derivative given in (11.32). For comparison recall that the exact second derivative to 20 digits is

$$f''(1) \approx 2.7182818284590452354.$$

a) Compute the approximation $\big(f(a-h)-2f(a)+f(a+h)\big)/(h^2)$ to $f''(1)$. Start with $h = 10^{-3}$, and then gradually reduce h. Also compute the actual error, and determine an h that gives close to minimal error.

b) Determine the optimal h given by (11.36) and compare with the value you determined in (a).

2 In this exercise you are going to do the error analysis of the three-point method in more detail. As usual the derivation in sections 11.1.2 and 11.1.3 may be useful as a guide.

a) Derive the relation (11.33) by performing the appropriate Taylor expansions of $f(a-h)$ and $f(a+h)$.

b) Starting from (11.33), derive the analog of relation (11.12).

c) Derive the estimate (11.34) by following the same recipe as in (11.18).

3 This exercise illustrates a different approach to designing numerical differentiation methods.

a) Suppose that we want to derive a method for approximating the derivative of f at a which has the form

$$f'(a) \approx c_1 f(a-h) + c_2 f(a+h), \quad c_1, c_2 \in \mathbb{R}.$$

We want the method to be exact when $f(x) = 1$ and $f(x) = x$. Use these conditions to determine c_1 and c_2.

b) Show that the method in (a) is exact for all polynomials of degree 1, and compare it to the methods we have discussed in this chapter.

c) Use the procedure in (a) and (b) to derive a method for approximating the second derivative of f,

$$f''(a) \approx c_1 f(a-h) + c_2 f(a) + c_3 f(a+h), \quad c_1, c_2, c_3 \in \mathbb{R},$$

by requiring that the method should be exact when $f(x) = 1$, x and x^2. Do you recognise the method?

d) Show that the method in (c) is exact for all cubic polynomials.