1 Numerical Differentiation and Integration

- We continue to exploit the useful properties of polynomials to develop methods for a computer to do integrations and to find derivatives.
- When the function is explicitly known, we can emulate the methods of calculus. But doing so in getting derivatives requires the subtraction of quantities that are nearly equal and that runs into round-off error.
- However, integration involves only addition, so round-off is not problem; of course, we cannot often find the true answer numerically because the analytical value is the limit of the sum of an infinite number of terms.
- We must be satisfied with approximations for both derivatives and integrals but, for most applications, the numerical answer is adequate.
- If we are working with experimental data that are displayed in a table of $[x, f(x)]$ pairs emulation of calculus is impossible; we must approximate the function behind the data in some way.
  - **Differentiation with a Computer:** Employs the interpolating polynomials to derive formulas for getting derivatives. These can be applied to functions known explicitly as well as those whose values are found in a table.
  - **Numerical Integration-The Trapezoidal Rule:** Approximates, the integrand function with a linear interpolating polynomial to derive a very simple but important formula for numerically integrating functions between given limits.

1.1 Differentiation with a Computer

- The derivative of a function, $f(x)$ at, $x = a$, is defined as

$$\frac{df}{dx}|_{x=a} = lim_{\Delta x \to 0} \frac{f(a + \Delta x) - f(a)}{\Delta x}$$

  This is called a *forward-difference* approximation. The limit could be approached from the opposite direction, giving a *backward-difference* approximation.

- It should be clear that a computer can calculate an approximation to the derivative from

$$\frac{df}{dx}|_{x=a} = \frac{f(a + \Delta x) - f(a)}{\Delta x}$$

Δx	Approximation	Error	Ratio of errors
0.05	4.05010	−0.11528	
0.05/2	4.10955	−0.05583	2.06
0.05/4	4.13795	−0.02743	2.04
0.05/8	4.15176	−0.01362	2.01
0.05/16	4.15863	−0.00675	2.02
0.05/32	4.16199	−0.00389	1.99
0.05/64	4.16382	−0.00156	2.18
0.05/128	**4.16504**	**−0.00034**	**4.67***
0.05/256	4.16504	−0.00034	
0.05/512	4.16504	−0.00034	
0.05/1024	4.16992	0.00454	
0.05/2048	4.17969	0.01430	

Table 1: Forward-difference approximations for $f(x) = e^x sin(x)$.

if a very small value is used for Δx.

- What if we do this, recalculating with smaller and smaller values of x starting from an initial value that is not small? We should expect to find an optimal value for x because round-off errors in the numerator will become great as x approaches zero, and these are magnified by the small value in the denominator.

- When we try this for $f(x) = e^x sin(x)$ at $x = 1.9$, starting with $\Delta x = 0.05$ and halving Δx each time, we find that the errors of the approximation decrease as Δx is reduced until about $\Delta x = 0.05/128$.

- The analytical answer is 4.1653826. Table 1 gives the results. Notice that each successive error is about 1/2 of the previous error as Δx is halved until Δx gets quite small, at which time round off affects the ratio.

- At values for Δx smaller than 0.05/128, the error of the approximation increases due to round off. In effect, the best value for Δx is when the effects of round-off and truncation errors are balanced.

- If a backward-difference approximation is used; similar results are obtained.

- It should be clear that a computer can calculate an approximation to the derivative from

$$\frac{df}{dx}|_{x=a} = \frac{f(a) - f(a - \Delta x)}{\Delta x}$$

- With MATLAB,
  - it can get the analytical answer to the function of Table 1.

```
>> f='exp(x)*sin(x)'
>> df=diff(f,'x')
>> numeric(subs(df,1.9,'x'))
```

  - it can compute numerically;

```
>> x=[1.9 1.9 1.9 1.9 1.9 1.9 1.9 1.9 1.9];
>> del=[.05 .05/2 .05/4 .05/8 .05/16 .05/32
        .05/64 .05/128 .05/256];
>> xplus=x+del;
>> f=exp(x).*sin(x);
>> fplus=exp(xplus).*sin(xplus);
>> num=fplus-f;
>> deriv=num./del;
```

- It is not by chance that the errors are about halved each time. Look at this Taylor series where we have used h for Δx:

$$f(x + h) = f(x) + f'(x) * h + f''(\xi) * h^2/2$$

where the last term is the error. The value of ξ is at some point between x and $x + h$. If we solve this equation for $f'(x)$, we get

$$f'(x) = (f(x + h) - f(x))/h - f''(\xi) * h^2/2 \tag{1}$$

which shows that the errors should be about proportional to h, precisely what Table 1 shows. If we repeat this but begin with the Taylor series for $f(x - h)$, it turns out that

$$f'(x) = (f(x) - f(x - h))/h + f''(\zeta) * h^2/2 \tag{2}$$

where ζ is between x and $x - h$, so the two error terms are not identical though both are $O(h)$.

Δx	Approximation	Error	Ratio of errors
0.05	4.15831	−0.00708	
0.05/2	4.16361	−0.00177	4.00
0.05/4	4.16496	−0.00042	4.21
0.05/8	4.16527	−0.00011	3.80
0.05/16	**4.16534**	**−0.00004**	**2.75**
0.05/32	4.16534	−0.00004	
0.05/64	4.16565	−0.00027	

Table 2: Central-difference approximations for $f(x) = e^x sin(x)$.

- If we add Eqs. 1 and 2, then divide by 2, we get the *central-difference* approximation to the derivative:

$$f'(x) = (f(x+h) - f(x-h))/(2h) - f'''(\xi)h^2/2 \tag{3}$$

  We had to extend the two Taylor series by an additional term to get the error because the $f''(x)$ terms cancel.

- This shows that using a central-difference approximation is a much preferred way to estimate the derivative; even though we use the same number of computations of the function at each step, we approach the answer much more rapidly.

- Table 2 illustrates this, showing that errors decrease about four fold when Δx is halved (as Eq. 3 predicts) and that a more accurate value is obtained.

1.1.1 Extrapolation Techniques

- The errors of a central-difference approximation to $f'(x)$ were of $O(h^2)$. In effect, suggests that the errors are proportional to h^2 although that is true only in the limit as $h \to 0$. Unless h is quite large, we can assume the proportionality.

- So, from two computations with h being half as large in the second, we can estimate the proportionality factor, C. For example, in Table 2; If errors were truly $C(h^2)$, we can write two equations:

$$True\ value = 4.15831 + C(0.05^2)$$
$$True\ value = 4.16361 + C(0.025^2)$$

h	Approximation
0.05	4.15831
0.025	4.16361

from which we can solve for the true value, eliminating the unknown constant C, getting;

$$\begin{aligned} True\ value &= 4.16361 + (1/3) * (4.16361 - 4.15831) \\ &= 4.16538 \end{aligned}$$

which is very close to the exact value for $f'(1.9)$, 4.165382.

- You can easily derive the general formula for improving the estimate, when errors decrease by $O(h^n)$

$$\begin{array}{cl} Better & = more + (1/(2^n - 1)) * (more - less) \\ estimate & accurate \end{array} \tag{4}$$

where more and less in the last term are the two estimates at h_1 and $h_2 = h_1/2$. *More accurate* is the estimate at the smaller value of h and n is the power of h in the order of the errors.

- As example, apply this to values from Table 1 which were from forward-difference approximations. Here the errors are $O(h)$.

h	Approximation
0.05	4.05010
0.025	4.10955

Using Eq. 4, we have

$$\begin{aligned} Better\ estimate &= 4.10955 + (1/(2^1 - 1))(4.10955 - 4.05010) \\ &= 4.16900 \end{aligned}$$

which shows considerable improvement but not as good as from the central differences.

1.2 Numerical Integration - The Trapezoidal Rule

- Given the function, $f(x)$, the antiderivative is a function $F(x)$ such that $F'(x) = f(x)$. The definite integral

$$\int_a^b f(x)dx = F(b) - F(a)$$

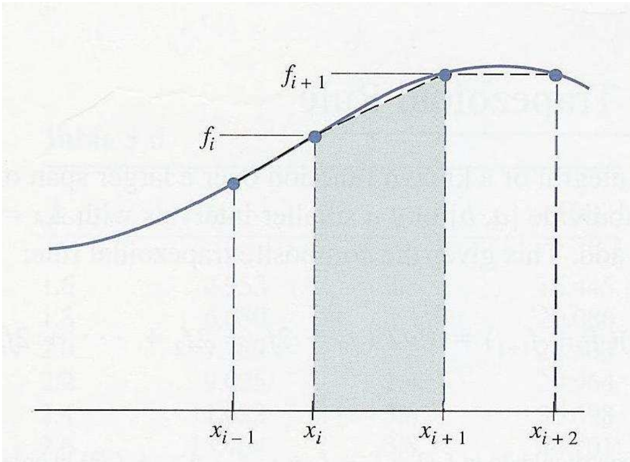

Figure 1: The trapezoidal rule.

can be evaluated from the antiderivative. Still, there are functions that do not have an antiderivative expressible in terms of ordinary functions.

- Is there any way that the definite integral can be found when the antiderivative is unknown? We can do it numerically.

- The definite integral is the area between the curve of $f(x)$ and the x-axis. That is the principle behind all numerical integration-we divide the distance from $x = a$ to $x = b$ into vertical strips and add the areas of these strips (the strips are often made equal in widths but that is not always required).

1.2.1 The Trapezoidal Rule

- Approximate the curve with a sequence of straight lines; in effect, we slope the top of the strips to match with the curve as best we can.

- We are approximating the curve with interpolating polynomials of degree-1. The gives us the *trapezoidal rule.* Figure 1 illustrates this.

- It is clear that the area of the strip from x_i to x_{i+1} gives an approximation to the area under the curve:

$$\int_{x_i}^{x_{i+1}} f(x)dx \approx \frac{f_i + f_{i+1}}{2}(x_{i+1} - x_i)$$

We will usually write $h = (x_{i+1} - x_i)$ for the width of the interval.

- Error term for the trapezoidal integration is

$$Error = -(1/12)h^3 f'(\xi) = O(h^3)$$

1.2.2 The Composite Trapezoidal Rule

- If we are getting the integral of a known function over a larger span of x-values, say, from $x = a$ to $x = b$, we subdivide [a,b] into n smaller intervals with $\Delta x = h$, apply the rule to each subinterval, and add. This gives the composite trapezoidal rule;

$$\int_a^b \approx \sum_{i=0}^{n-1} (h/2)(f_i + f_{i+1}) = (h/2)(f_0 + 2f_1 + 2f_2 + \ldots + 2f_{n-1} + f_n)$$

The error is not the local error $O(h^3)$ but the global error, the sum of n local errors;

$$Global\ error = (-1/12)h^3[f''(\xi_1) + f''(\xi_2) + \ldots + f''(\xi_n)]$$

In this equation, each of the ξ_i is somewhere within each subinterval. If $f''(x)$ is continuous in [a, b], there is some point within [a,b] at which the sum of the $f''(\xi_i)$ is equal to $f''(\xi)$, where ξ in [a, b]. We then see that, because $nh = (b - a)$,

$$Global\ error = (-1/12)h^3 f''(\xi) = \frac{-(b-a)}{12} h^2 f''(\xi) = O(h^2)$$

An Algorithm for Integration by the Composite Trapezoidal Rule:

```
Given at function f(x)
(Get user inputs)
Input
a, b = end points of interval
n=number of intervals
(Do the integration)
Set h = (b − a)/n.
Set sum = 0
For i = 1 to n − 1 Step 1 Do
Set x = a + h ∗ i
Set sum = sum + 2 ∗ f(x)
End Do (For i)
Set sum = sum + f(a) + f(b)
Set ans = sum ∗ h/2
The value of the integral is given by answer
```

x	$f(x)$	x	$f(x)$
1.6	4.953	2.8	16.445
1.8	6.050	3.0	20.086
2.0	7.389	3.2	24.533
2.2	9.025	3.4	29.964
2.4	11.023	3.6	36.598
2.6	13.464	3.8	44.701

Table 3: Example for the trapezoidal rule.

- **Example:** Given the values for x and $f(x)$ in Table3, use the trapezoidal rule to estimate the integral from $x = 1.8$ to $x = 3.4$. Applying the trapezoidal rule:

$$\begin{aligned}\int_{1.8}^{3.4} f(x)dx &\approx \tfrac{0.2}{2}[6.050 + 2(7.389) + 2(9.025) + 2(11.023) + 2(13.464) \\ &+2(16.445) + 2(20.086) + 2(24.533) + 29.964] \\ &= 23.9944\end{aligned}$$

The data in Table 3 are for $f(x) = e^x$ and the true value is $e^{3.4} - e^{l.8} = 23.9144$. The trapezoidal rule value is off by 0.08; there are three digits of accuracy. How does this compare to the estimated error?

$$\begin{aligned} Error &= -\tfrac{1}{12}h^3 n f''(\xi),\ 1.8 \le \xi \le 3.4 \\ &= -\tfrac{1}{12}(0.2)^3(8) * \left\{ \begin{array}{ll} e^{1.8} & (max) \\ e^{3.4} & (min) \end{array} \right\} = \left\{ \begin{array}{ll} -0.0323 & (max) \\ -0.1598 & (min) \end{array} \right\}\end{aligned}$$

Alternatively,

$$Error = -\tfrac{1}{12}(0.2)^2(3.4 - 1.8) * \left\{ \begin{array}{ll} e^{1.8} & (max) \\ e^{3.4} & (min) \end{array} \right\} = \left\{ \begin{array}{ll} -0.0323 & (max) \\ -0.1598 & (min) \end{array} \right\}$$

The actual error was -0.080.