12.1 Numerical Differentiation

- In calculus we did differentiation of continuous functions by taking the limit of the following difference quotient:

$$\frac{\Delta y}{\Delta x} = \frac{f(x_i + \Delta x) - f(x_i)}{\Delta x}$$

- To perform this task numerically we simply take a small value for Δx and calculate the quotient without taking the limit.

- Usually differentiation is much easier to do analytically than integration but there are still times where numerical methods are necessary.

- When functions are extremely complex it may be practical to use a simple high accuracy numerical differentiation method. Also when we measure discrete data points we have no choice but to perform numerical differentiation to get the rates of change.

© MCGRAW HILL

Lagrange Error Bound

- Recall from calculus the Taylor series expansion of a function:

$$f(x_1 = x_0 + h) = f(x_0) + f'(x_0)h + \frac{1}{2!}f''(x_0)h^2 + \frac{1}{3!}f^{(3)}(x_0)h^3 + \cdots$$

- If we truncate after the *n*th term we can write the function at the point x_1 as the Taylor polynomial approximation plus the Lagrange error bound:

$$f(x_1) = T_n(x_1) + R_n(x_1)$$

where,

$$R_n(x_1) = \frac{f^{(n+1)}(\xi)}{(n+1)!}h^{n+1}, \quad x_0 \le \xi \le x_1$$

- We will use this to derive numerical differentiation formulas.

© MCGRAW HILL

Derivation of the Lagrange Error Bound

- Much of our numerical error analysis is reliant on Taylor series and the Lagrange error bound which is stated as follows.

***Theorem* (Lagrange Error Bound)**
An analytic function, $f(x)$, can be approximated by a Taylor polynomial, $T_N(x)$, with an associated error $R_N(x)$, such that,

$$f(x) = T_N(x) + R_N(x)$$

with the error term having an upper bound given by,

$$|R_N(x)| \leq \left| \frac{f^{(N+1)}(c)}{(N+1)!} h^{N+1} \right|$$

where, $x_0 \leq c \leq x$ and $x = x_0 + h$.

© MCGRAW HILL

- Before we can prove it we require the following theorem that was proved in calculus.

***Theorem* (Generalised Mean Value Theorem)**
For a function, $f(x)$, that is continuous on $[a, b]$, then there exists a $c \in (a, b)$ such that,

$$\int_a^b f(x)g(x)\,dx = f(c)\int_a^b g(x)\,dx$$

if $g(x)$ doesn't change sign on $[a, b]$.

Proof (Lagrange Error Bound)

- Consider the following application of the product rule:

$$\frac{d}{dt}\left[(x-t)^{n+1}f^{(n+1)}(t)\right] = -(n+1)(x-t)^n f^{(n+1)}(t) + (x-t)^{n+1}f^{(n+2)}(t)$$

- Rearranging the above formula gives:

$$(n+1)(x-t)^n f^{(n+1)}(t) = (x-t)^{n+1} f^{(n+2)}(t) - \frac{d}{dt}\left[(x-t)^{n+1} f^{(n+1)}(t)\right]$$

© MCGRAW HILL

- Let's now take ***n* = 0** in the previous equation *:

$$\int_{x_0}^{x} f'(t) = \int_{x_0}^{x} (x-t) f''(t)\, dt + h f'(x_0)$$

© MCGRAW HILL

- Substituting the last equation into **:

$$*** \quad f(x) = f(x_0) + hf'(x_0) + \frac{h^2}{2} f''(x_0) + \int_{x_0}^{x} \frac{(x-t)^2}{2} f^{(3)}(t)\, dt$$

© MCGRAW HILL

- Continuing in this way in general we have:

$$f(x) = \sum_{k=0}^{N} \frac{h^k}{k!} f^{(k)}(x_0) + \int_{x_0}^{x} \frac{(x-t)^N}{N!} f^{(N+1)}(t)\, dt$$

- Using the **Generalised Mean Value Theorem** we have:

$$R_N(x) = f^{(N+1)}(c) \int_{x_0}^{x} \frac{(x-t)^N}{N!}\, dt$$

EXAMPLE 1 Use the Lagrange error bound to estimate the error in approximating sin(x) with a 3rd order Taylor polynomial at a distance of 0.1 from the expansion point.

$$|R_N(x)| \leq \left| \frac{f^{(4)}(c)}{(3+1)!} h^4 \right|$$

$$= \left| \frac{\sin(c)}{4!} 0.1^4 \right| = 4.1667e - 6$$

Note we take the value of *c* that maximises the 4th derivative here

12.2 Finite Differences

- Taking the first order Taylor polynomial we have:

$$f(x_1) = f(x_0) + f'(x_0)h + \frac{f''(c)}{2!}h^2$$

- We use Big 'O' notation to write:

$$f(x_1) = f(x_0) + f'(x_0)h + O(h^2)$$

where $O(h^2)$ means "terms of order h^2 or higher". In other words constants multiplied by h^2, h^3, h^4 etc.

- Rearranging the top equation gives:

$$f'(x_0) = \frac{f(x_1) - f(x_0)}{h} - \frac{f''(c)}{2!}\frac{h^2}{h} = \frac{f(x_1) - f(x_0)}{h} - \frac{f''(c)}{2!}h$$

© MCGRAW HILL

- In Big 'O' notation it is:

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

- This equation, known as a finite difference approximation, gives us an approximation for the first derivative at x_0 plus an error of order h.

- When the step size, h, gets smaller the error gets smaller and the finite difference approaches the true derivative (think of the limit of the difference quotient).

- However, by using Taylor series we now have a way of expressing the error.

© MCGRAW HILL

Forward, Backwards, Centred Differences

- The previous formula is known as a **forward difference** since the formula calculates the derivative of f at x_0 using x_0 and x_1.

- We also have a **backward difference** and **centred difference** approximation.

Forward difference

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{h} + O(h)$$

Backward difference

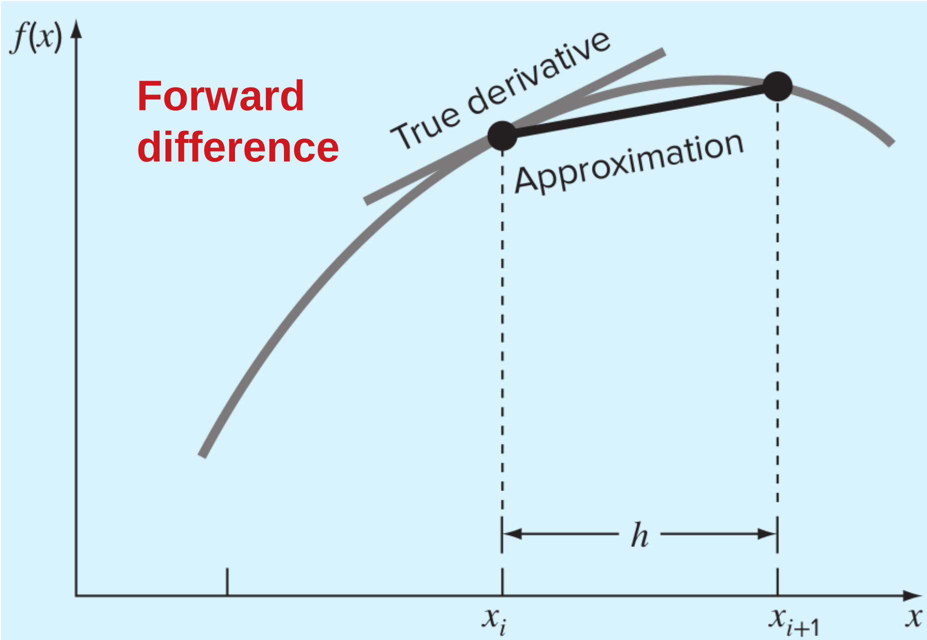
$$f'(x_i) = \frac{f(x_i) - f(x_{i-1})}{h} + O(h)$$

More accurate

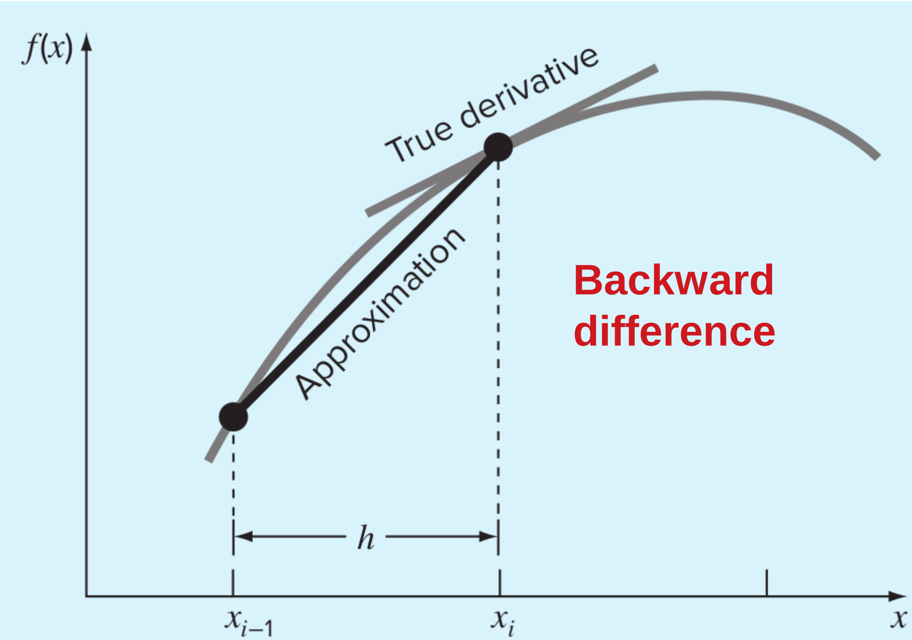
Centred difference

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

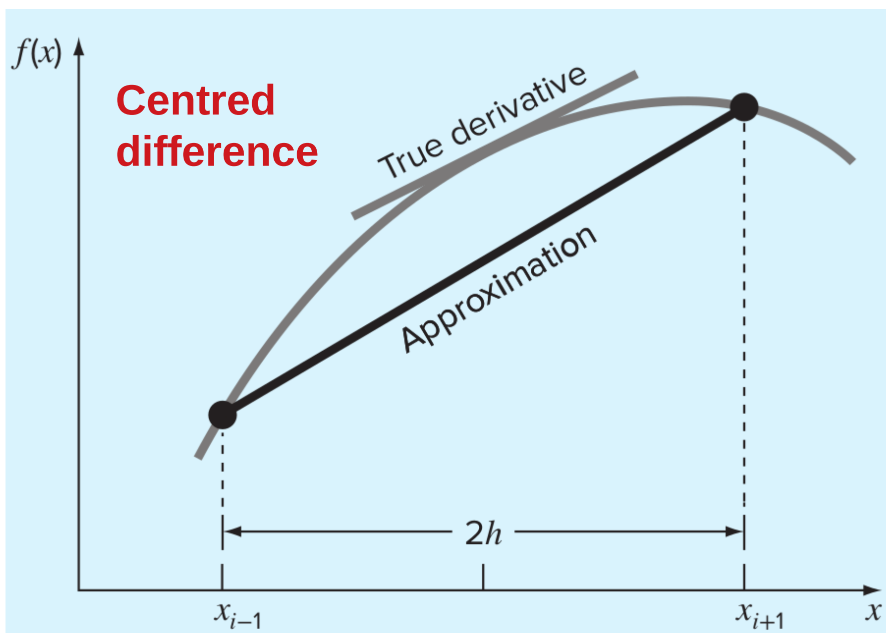
© MCGRAW HILL



$f(x)$
Forward difference
True derivative
Approximation
h
x_i
x_{i+1}
x



$f(x)$
True derivative
Approximation
Backward difference
h
x_{i-1}
x_i
x



$f(x)$
Centred difference
True derivative
Approximation
$2h$
x_{i-1}
x_{i+1}
x

© MCGRAW HILL

EXAMPLE 2 Use forward and backward difference approximations of $O(h)$ and a centred difference approximation of $O(h^2)$ to estimate the first derivative of the following function at $x = 0.5$ using step sizes of $h = 0.5$ and $h = 0.25$.
Compare with the true value.

$$f(x) = -0.1x^4 - 0.15x^3 - 0.5x^2 - 0.25x + 1.2$$

True value

$$f'(x) = -0.4x^3 - 0.45x^2 - 1.0x - 0.25 \longrightarrow f'(0.5) = -0.9125$$

© MCGRAW HILL

***h* = 0.5 (backward)**

$$f'(0.5) \cong \frac{0.925 - 1.2}{0.5} = -0.55 \qquad |\varepsilon_t| = 39.7\%$$

© MCGRAW HILL

- Notice that the forward and backward difference approximations are roughly the same size but the central difference is a significant improvement since it is predicted to be $O(h^2)$ by Taylor series.

- Also by halving the step size for the forward and backward difference approximations we approximately halved the error.

- However for the centred difference we when we halved the error we improved the error by nearly a factor of 4.

 © MCGRAW HILL

12.3 Higher Order derivatives

- Taylor expanding at the point 2 steps up from x_0 gives:

$$
\begin{aligned}
f(x_{i+2} = x_i + 2h) &= f(x_i) + f'(x_i)(2h) + \frac{f''(x_i)}{2!}(2h)^2 + \frac{f'''(x_i)}{3!}(2h)^3 + \cdots \\
&= f(x_i) + 2f'(x_i)h + 2f''(x_i)h^2 + O(h^3)
\end{aligned}
$$

© MCGRAW HILL

Backward difference for 2nd derivative

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

Centred difference for 2nd derivative

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

$$= \frac{\frac{f(x_{i+1})-f(x_i)}{h} - \frac{f(x_i)-f(x_{i-1})}{h}}{h} + O(h^2)$$

- The 2nd derivative finite differences are just finite differences of the first derivative approximations (similarly to how the second derivative in calculus is the derivative of the first derivative).

© MCGRAW HILL

12.4 Higher Accuracy Formulas

- Taking more terms in the Taylor series leads to higher accuracy.

$$f(x_{i+1}) = f(x_i) + f'(x_i)h + \frac{f''(x_i)}{2!}h^2 + O(h^3)$$

- Rearranging for the first derivative:

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{h} - \frac{f''(x_i)}{2!}h + O(h^2)$$

- Substituting the forward difference approximation for the second derivative into the above we obtain:

$$f'(x_i) = \frac{4f(x_{i+1}) - 3f(x_i) - f(x_{i+2})}{2h} + O(h^2)$$

- Similarly backward and centred differences can be improved.

© MCGRAW HILL

First Derivative | Error

Forward differences (Standard vs. improved)

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{h} \qquad O(h)$$

$$f'(x_i) = \frac{-f(x_{i+2}) + 4f(x_{i+1}) - 3f(x_i)}{2h} \qquad O(h^2)$$

Second Derivative

$$f''(x_i) = \frac{f(x_{i+2}) - 2f(x_{i+1}) + f(x_i)}{h^2} \qquad O(h)$$

$$f''(x_i) = \frac{-f(x_{i+3}) + 4f(x_{i+2}) - 5f(x_{i+1}) + 2f(x_i)}{h^2} \qquad O(h^2)$$

Third Derivative

$$f'''(x_i) = \frac{f(x_{i+3}) - 3f(x_{i+2}) + 3f(x_{i+1}) - f(x_i)}{h^3} \qquad O(h)$$

$$f'''(x_i) = \frac{-3f(x_{i+4}) + 14f(x_{i+3}) - 24f(x_{i+2}) + 18f(x_{i+1}) - 5f(x_i)}{2h^3} \qquad O(h^2)$$

Fourth Derivative

$$f''''(x_i) = \frac{f(x_{i+4}) - 4f(x_{i+3}) + 6f(x_{i+2}) - 4f(x_{i+1}) + f(x_i)}{h^4} \qquad O(h)$$

$$f''''(x_i) = \frac{-2f(x_{i+5}) + 11f(x_{i+4}) - 24f(x_{i+3}) + 26f(x_{i+2}) - 14f(x_{i+1}) + 3f(x_i)}{h^4} \qquad O(h^2)$$

© MCGRAW HILL

Backward differences (Standard vs. improved)

First Derivative	Error
$f'(x_i) = \dfrac{f(x_i) - f(x_{i-1})}{h}$	$O(h)$
$f'(x_i) = \dfrac{3f(x_i) - 4f(x_{i-1}) + f(x_{i-2})}{2h}$	$O(h^2)$
Second Derivative	
$f''(x_i) = \dfrac{f(x_i) - 2f(x_{i-1}) + f(x_{i-2})}{h^2}$	$O(h)$
$f''(x_i) = \dfrac{2f(x_i) - 5f(x_{i-1}) + 4f(x_{i-2}) - f(x_{i-3})}{h^2}$	$O(h^2)$
Third Derivative	
$f'''(x_i) = \dfrac{f(x_i) - 3f(x_{i-1}) + 3f(x_{i-2}) - f(x_{i-3})}{h^3}$	$O(h)$
$f'''(x_i) = \dfrac{5f(x_i) - 18f(x_{i-1}) + 24f(x_{i-2}) - 14f(x_{i-3}) + 3f(x_{i-4})}{2h^3}$	$O(h^2)$
Fourth Derivative	
$f''''(x_i) = \dfrac{f(x_i) - 4f(x_{i-1}) + 6f(x_{i-2}) - 4f(x_{i-3}) + f(x_{i-4})}{h^4}$	$O(h)$
$f''''(x_i) = \dfrac{3f(x_i) - 14f(x_{i-1}) + 26f(x_{i-2}) - 24f(x_{i-3}) + 11f(x_{i-4}) - 2f(x_{i-5})}{h^4}$	$O(h^2)$

© MCGRAW HILL

First Derivative | Error

Centred differences (Standard vs. improved)

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1})}{2h} \qquad O(h^2)$$

$$f'(x_i) = \frac{-f(x_{i+2}) + 8f(x_{i+1}) - 8f(x_{i-1}) + f(x_{i-2})}{12h} \qquad O(h^4)$$

Second Derivative

$$f''(x_i) = \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1})}{h^2} \qquad O(h^2)$$

$$f''(x_i) = \frac{-f(x_{i+2}) + 16f(x_{i+1}) - 30f(x_i) + 16f(x_{i-1}) - f(x_{i-2})}{12h^2} \qquad O(h^4)$$

Third Derivative

$$f'''(x_i) = \frac{f(x_{i+2}) - 2f(x_{i+1}) + 2f(x_{i-1}) - f(x_{i-2})}{2h^3} \qquad O(h^2)$$

$$f'''(x_i) = \frac{-f(x_{i+3}) + 8f(x_{i+2}) - 13f(x_{i+1}) + 13f(x_{i-1}) - 8f(x_{i-2}) + f(x_{i-3})}{8h^3} \qquad O(h^4)$$

Fourth Derivative

$$f''''(x_i) = \frac{f(x_{i+2}) - 4f(x_{i+1}) + 6f(x_i) - 4f(x_{i-1}) + f(x_{i-2})}{h^4} \qquad O(h^2)$$

$$f''''(x_i) = \frac{-f(x_{i+3}) + 12f(x_{i+2}) - 39f(x_{i+1}) + 56f(x_i) - 39f(x_{i-1}) + 12f(x_{i-2}) - f(x_{i-3})}{6h^4} \qquad O(h^4)$$

© MCGRAW HILL

EXAMPLE 3 Repeat **Example 2** using the improved formulas given in the previous slides and a step size of *h* = 0.25.

$$f(x) = -0.1x^4 - 0.15x^3 - 0.5x^2 - 0.25x + 1.2$$

h = 0.25

$$x_{i-2} = 0 \qquad f(x_{i-2}) = 1.2$$

$$x_{i-1} = 0.25 \qquad f(x_{i-1}) = 1.1035156$$

$$x_i = 0.5 \qquad f(x_i) = 0.925$$

$$x_{i+1} = 0.75 \qquad f(x_{i+1}) = 0.6363281$$

$$x_{i+2} = 1 \qquad f(x_{i+2}) = 0.2$$

Forward

$$f'(0.5) = \frac{-0.2 + 4(0.6363281) - 3(0.925)}{2(0.25)} = -0.859375 \qquad \varepsilon_t = 5.82\%$$

Backward

$$f'(0.5) = \frac{3(0.925) - 4(1.1035156) + 1.2}{2(0.25)} = -0.878125 \qquad \varepsilon_t = 3.77\%$$

Centred

$$f'(0.5) = \frac{-0.2 + 8(0.6363281) - 8(1.1035156) + 1.2}{12(0.25)} = -0.9125 \qquad \varepsilon_t = 0\%$$

- Notice how the improved centred difference approximation of the first derivative is 0 here.

- This is because the function we are trying to integrate is quartic and since this improved formula has error $O(h^4)$ it comes from the 4th order Taylor expansion. The 4th order Taylor expansion for a quartic polynomial has 0 error.

© MCGRAW HILL

12.5 Richardson Extrapolation

- Similarly to how we did for integrals, we can use the approximate errors to improve our estimates following the Richardson extrapolation method to obtain:

$$D = \frac{4}{3} D(h_2) - \frac{1}{3} D(h_1)$$

  for the first level of extrapolation.

- Here D is the true derivative and $D(h_i)$ are the centred finite difference approximations at the step sizes h_i.

- It can be shown that the error here improves from $O(h^2)$ to $O(h^4)$ in an analogous fashion to the Richardson extrapolation for integrals.

© MCGRAW HILL

- Applying the first Richardson extrapolation formula to the centred differences for step sizes *h* and 2*h* we have:

$$\frac{4}{3}D(h)-\frac{1}{3}D(2h)=\frac{1}{3h}\left\{4\left[\frac{f(x_{i+1})-f(x_{i-1})}{2}\right]-\left[\frac{f(x_{i+2})-f(x_{i-2})}{4}\right]\right\}$$

$$=\frac{1}{3h}\left\{\frac{-f(x_{i+2})+8f(x_{i+1})-8f(x_{i-1})+f(x_{i-2})}{4}\right\}$$

- The above is the formula we obtained for the improved centred difference of the first derivative which had error $O(h^4)$.

© MCGRAW HILL

EXAMPLE 4 Repeat **Example 2** using the first level of Richardson extrapolation with initial step size h = 0.5 and refined step size h/2.

***h* = 0.5**

$$D(0.5) = \frac{0.2 - 1.2}{1} = -1.0 \qquad \varepsilon_t = -9.6\%$$

***h* = 0.25**

$$D(0.25) = \frac{0.6363281 - 1.103516}{0.5} = -0.934375 \qquad \varepsilon_t = -2.4\%$$

Richardson extrapolation

$$D = \frac{4}{3}(-0.934375) - \frac{1}{3}(-1) = -0.9125$$

- This can be continued to higher levels of extrapolation in a Romberg style algorithm.

© MCGRAW HILL

12.6 Unequally Spaced Data

- When we make measurements and need the derivative we can interpolate the data with a Lagrange polynomial (for example) then differentiate it.

- For 3 data points we had the Lagrange quadratic polynomial as:

$$f(x) = \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)}f(x_1) + \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)}f(x_2) + \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}f(x_3)$$

- Differentiating we have:

$$f'(x) = \frac{2x-x_2-x_3}{(x_1-x_2)(x_1-x_3)}f(x_1) + \frac{2x-x_1-x_3}{(x_2-x_1)(x_2-x_3)}f(x_2) + \frac{2x-x_1-x_2}{(x_3-x_1)(x_3-x_2)}f(x_3)$$

- This can be done for any order Lagrange polynomial.

© MCGRAW HILL

- Letting $x_2 - x_1 = x_3 - x_2 = h$ in the previous formula we obtain the improved first forward difference approximation.

- Therefore the improved formulas correspond with a polynomial passing through the points x_i.

- Considering again **Example 3**, we use the following centred difference improved formula:

$$f'(x_i) = \frac{-f(x_{i+2}) + 8f(x_{i+1}) - 8f(x_{i-1}) + f(x_{i-2})}{12h}$$

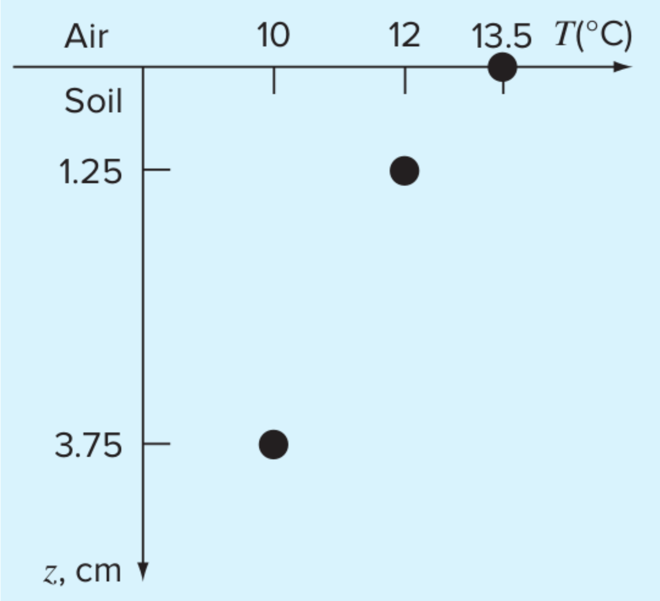
- We now know this corresponds with a 4th order Lagrange polynomial passing through the points x_{i-2}, x_{i-1}, x_{i+1}, x_{i+2} which is why the error was 0 (the function we were differentiating numerically was originally a 4th order polynomial).

© MCGRAW HILL

EXAMPLE 5

The data on the right measures temperature at different distances from soil on the ground. Estimate the heat flux into the ground (z = 0) given thermal conductivity of k = 0.5 W/(m.K) for soil.

3 data points so use formula from first page of Section 12.6



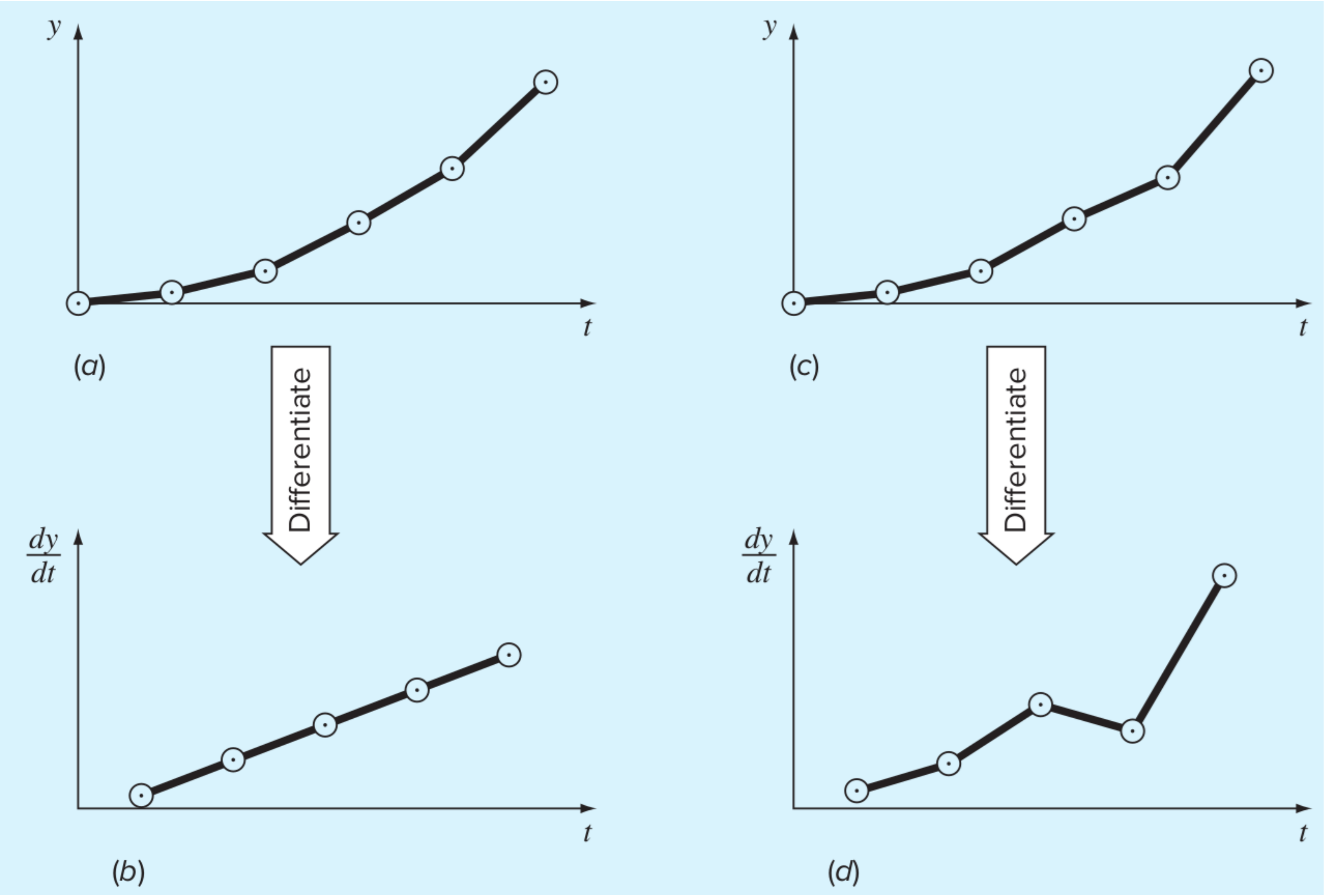
12.7 Data Error Magnification

- Given data with random errors, some errors may be positive, others may be negative. When we integrate this data the positive and negative errors have a tendency to cancel out somewhat, meaning that integration is more tolerant of data errors.

- Numerical differentiation on the other hand has a tendency to amplify errors at each individual point (see the figure on the next page).

- When confronted with data where errors are significant, a best fit polynomial should be obtained using some form of regression.

© MCGRAW HILL


y
t
(a)
Differentiate
dy/dt
t
(b)
y
t
(c)
Differentiate
dy/dt
t
(d)

12.8 Partial Derivatives

- We apply the difference quotient as before for the following definitions:

Centred differences:

$$\frac{\partial f}{\partial x} = \frac{f(x+\Delta x, y) - f(x-\Delta x, y)}{2\Delta x}$$

$$\frac{\partial f}{\partial y} = \frac{f(x, y+\Delta y) - f(x, y-\Delta y)}{2\Delta y}$$

Mixed partial derivatives:

$$\frac{\partial^2 f}{\partial x \partial y} = \frac{\dfrac{f(x+\Delta x, y+\Delta y) - f(x+\Delta x, y-\Delta y)}{2\Delta y} - \dfrac{f(x-\Delta x, y+\Delta y) - f(x-\Delta x, y-\Delta y)}{2\Delta y}}{2\Delta x}$$

The above differentiates with respect to *y* first then *x* using centred finite differences

© MCGRAW HILL

12.9 Matlab Built-In Functions

- We have already seen the **diff** function but we also have the **gradient** function.

- Both functions take vector inputs of length *n*, but return different calculations.

- The **diff** function returns a vector of length (*n* – 1) according to the following formula:

$$z = \text{diff}(x) \rightarrow z(i) = [x(i+1) - x(i)]$$

- The **gradient** function returns a vector of length *n* according to the following formula:

$$z = \text{gradient}(x) \rightarrow z(i) = [x(2) - x(1), \ldots, \frac{x(i+1) - x(i-1)}{2}, \ldots x(n) - x(n-1)]$$

© MCGRAW HILL

EXAMPLE 6

```
>> x = 0:0.1:0.5;
>> y = randi([-10,10],[1,length(x)])

Y =

     1    -9    -8    -8     4     0

>> diff(y)

ans =

   -10     1     0    12    -4

>> gradient(y)

ans =

  -10.0000   -4.5000    0.5000    6.0000    4.0000   -4.0000
```


© MCGRAW HILL