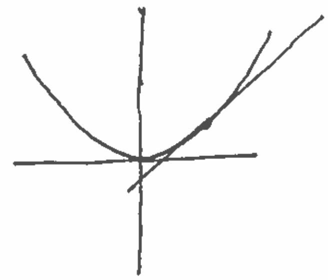
consider $y = x^2$

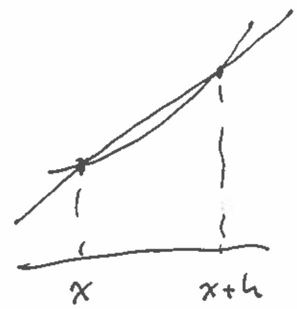
$y'(x) = 2x$



derivative at $x = 2$ is $y'(2) = 2(2) = 4$

what if we couldn't get $y'(x)$ analytically?

Recall $$y'(x) \equiv \lim_{h \to 0} \frac{y(x+h) - y(x)}{h}$$



to approximate $y'(x)$, just use a small but finite h

$h = 0.1$ $$y'(2) \approx \frac{y(2.1) - y(2)}{0.1} = \frac{4.41 - 4}{0.1} = 4.1$$

$h = 0.01$ $$y'(2) \approx \frac{y(2.01) - y(2)}{0.01} = \frac{4.0401 - 4}{0.01} = 4.01$$

→ as h decreases, our approximation improves.
However, making h too small does not work, as we shall see later.

derivative schemes

forward difference $\frac{f(x+h) - f(x)}{h}$

backward difference $\frac{f(x) - f(x-h)}{h}$

where $f(x)$ is some smooth function

centred difference $\frac{f(x+h) - f(x-h)}{2h}$

five-point formula $\frac{1}{12h}\left(f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h)\right)$

Differentiating data

suppose we have data e.g. velocity versus time, where the independent variable is equally spaced between points

	t	v	
t_1	0.1	3.0	v_1
t_2	0.2	3.5	v_2
t_3	0.3	3.8	v_3
t_4	0.4	2.6	v_4
t_5	0.5	2.8	v_5
t_6	0.6	1.4	v_6

here $h = \Delta t = 0.1$ is set by the data. We can't change it.

Using the forward difference scheme, we can approximate the acceleration ($\frac{dv}{dt}$) at, say, $t = 0.2$ as

$$\frac{v(0.2+0.1) - v(0.2)}{0.1}$$

or in terms of our data points

$$\frac{v_3 - v_2}{t_3 - t_2} = \frac{3.8 - 3.5}{0.1} = \frac{0.3}{0.1} = 3$$

(Notice that when we substract ~~to~~ similar numbers, we lose ~~loose~~ precision ~~)~~; ~~the result~~ taking derivatives ~~increases~~ results in noisier output.

In Mathematica, data is often stored in a list

e.g.

vdata = {{0.1, 3.0}, {0.2, 3.5}, {0.3, 3.8},
{0.4, 3.6}, {0.5, 2.8}, {0.6, 1.4}}

the acceleration at t=0.2 would be (using forward differen

$$\frac{\text{vdata[[3,2]]} - \text{vdata[[2,2]]}}{\text{vdata[[3,1]]} - \text{vdata[[2,1]]}}$$

A function that takes an arbitrary list of data and calculates the forward difference at an arbitrary point might look like

$$\text{fd[list\_, i\_]:=} \frac{\text{list[[i+1,2]]} - \text{list[[i,2]]}}{\text{list[[i+1,1]]} - \text{list[[i,1]]}}$$

:= is crucial here (vs =)

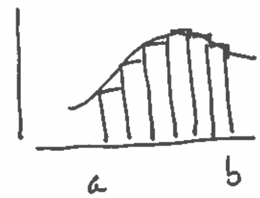
fd[vdata, 2] gives the acceleration at t=0.2

Numerical Integration

basic idea

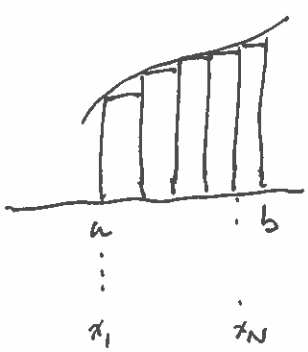
$\int_a^b f(x)\,dx$

$\approx$



Area under the curve is approximated by the area occupied by rectangles (or ~~other~~ simple slices of other shapes). ~~For the~~ We will consider slices of ~~the same~~ equal width

Rectangle rule



using N rectangles, we define

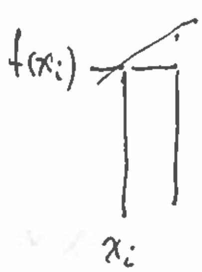
$\Delta x = \frac{b-a}{N}$ (width of a rectangle)

$x_i = a + (i-1)\Delta x$

$$\left(\begin{aligned} x_1 &= a \\ x_N &= a + (N-1)\left(\frac{b-a}{N}\right) = a+b-a - \frac{b-a}{N} \\ &= b - \Delta x \end{aligned} \right)$$

x_i is the location of the (left side of) i^{th} rectangle

area of the i^{th} rectangle is $f(x_i)\,\Delta x$

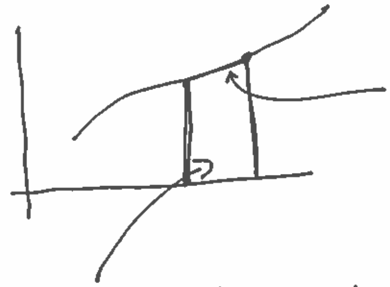


call $f_i \equiv f(x_i)$

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{N} f_i\,\Delta x = \Delta x \sum_{i=1}^{N} f_i$$

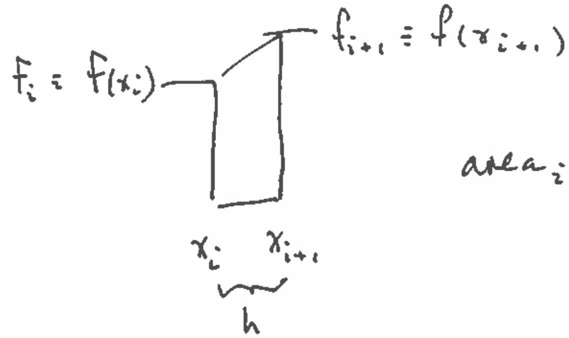
(notice we don't use $x=b$ or $f(b)$)

Trapezoid rule



approximate $f(x)$ with a straight line connecting end points

area of a trapezoid is



$$\text{area}_i = \frac{h}{2}\left(f_i + f_{i+1}\right)$$

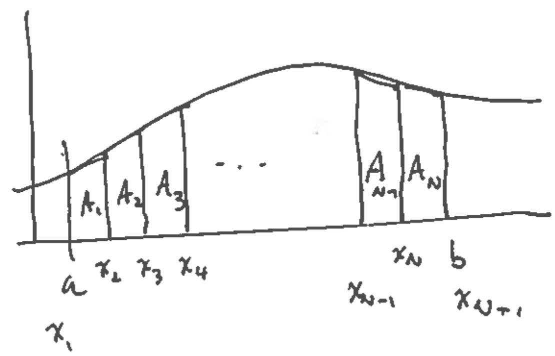
using N trapezoids ($N+1$ points)

$$h = (\Delta x =)\ \frac{b-a}{N}$$

$$x_i = a + (i-1)h$$

$$x_1 = a$$

$$x_{N+1} = b$$



$$\int_a^b f(x)\,dx \doteq A_1 + A_2 + A_3 + \cdots + A_{N-1} + A_N$$

$$= \frac{h}{2}\left(f_1 + f_2\right) + \frac{h}{2}\left(f_2 + f_3\right) + \frac{h}{2}\left(f_3 + f_4\right) + \cdots + \frac{h}{2}\left(f_{N-1} + f_N\right) + \frac{h}{2}\left(f_N + f_{N+1}\right)$$

$$= \frac{h}{2}\left(f_1 + \underbrace{f_2 + f_2} + \underbrace{f_3 + f_3} + \underbrace{f_4 + \cdots} \quad \underbrace{f_{N-1}} + \underbrace{f_N + f_N} + f_{N+1}\right)$$

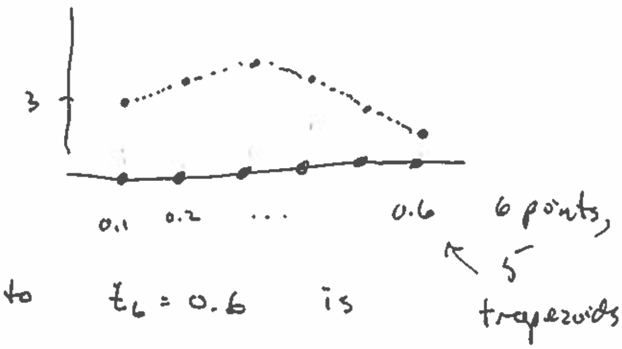
interior points get counted twice

$$= h\left(\frac{f_1}{2} + f_2 + f_3 + \cdots + f_N + \frac{f_{N+1}}{2}\right)$$

$$= h\left(\frac{f_1}{2} + \frac{f_{N+1}}{2} + \sum_{i=2}^{N} f_i\right)$$

Integrating data

Back to our data stored in vdata



recall $t_1 = 0.1$

$\cancel{(f(x) \text{ is } v(t))}$

$t_6 = 0.6$

The displacement from $t_1 = 0.1$ to $t_6 = 0.6$ is

$$\Delta x = \int_{0.1}^{0.6} v(t)\,dt = \Delta t \left(\frac{v_1}{2} + \frac{v_6}{2} + \sum_{i=2}^{5} v_i \right)$$

In Mathematica

v_i is vdata[[i, 2]]

t_i is vdata[[i, 1]]

Δt is vdata[[2, 1]] - vdata[[1, 1]] (equally spaced points, so take the first pair to find Δt)

Δx is

$$\Delta t \left(\frac{\text{vdata[[1,2]]}}{2} + \frac{\text{vdata[[6,2]]}}{2} + \sum_{i=2}^{5} \text{vdata[[i, 2]]} \right)$$

A function that integrates an arbitrary list of data might look like

NITrap[list_] := (Δt = list[[2,1]] - list[[1,1]] ;

n = Length[list] - 1 ;

$$\frac{\Delta t}{2} \left(\text{list[[1,2]]} + \text{list[[n+1,2]]} + 2 \sum_{i=2}^{n} \text{list[[i, 2]]} \right) \Bigg)$$

NITrap[vdata] would give displacement from t=0.1 to t=0.6 as above.

Taylor series

Idea: approximate a smooth function with a polynomial

We "expand" $f(x)$ around $x=a$

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

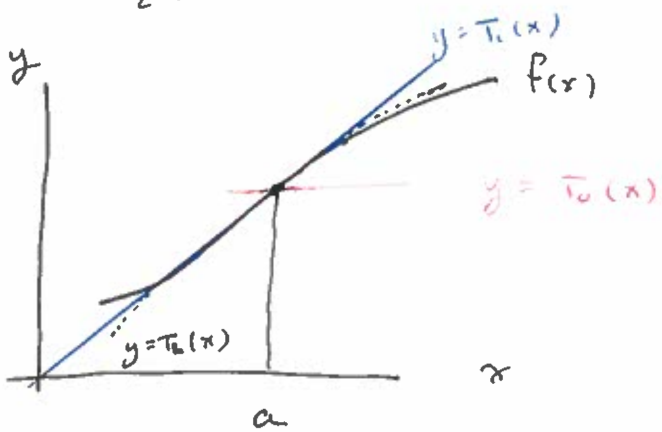
$$= \sum_{n=0}^{\infty} f^{(n)}(a)\frac{(x-a)^n}{n!}$$

Each term in the series makes the polynomial a better approximation to $f(x)$

$T_0(x) = f(a)$ (a constant)

$T_1(x) = f(a) + (x-a)f'(a)$ (a line)

$T_2(x) = f(a) + (x-a)f'(a) + \frac{(x-a)^2}{2}f''(a)$ (parabola)



Series[f[x], {x, a, n}]

The series is designed so that if you take the n^{th} derivative of the series at $x=a$, you get precisely $f^{(n)}(a)$

Eg. Expand $f(x) = \sin x$ around $x=0$ to 3^{rd} order
(expansion around $x=0$ is called a Maclaurin series)

$f(x) = \sin x$

$f(0) = 0$

$f'(0) = \cos x|_{x=0} = 1$

$f''(0) = -\sin x|_{x=0} = 0$

$f'''(0) = -\cos x|_{x=0} = -1$

$$f(x) = f(0) + xf'(0) + \frac{x^2}{2}f''(0) + \frac{x^3}{6}f'''(0)$$

$$\sin x \approx x - \frac{x^3}{6}$$ (next term would be $\sim x^5$)

or we can write

$$\sin x = x - \frac{x^3}{6} + O(x^5)$$

of all the other terms we are missing, the ~~largest~~ leading one varies as x^5

i.e. for x small $x^5 \gg x^7 \gg x^9$ etc

---

we can rewrite the Taylor series as

expand around x
$x + h - x = h$

(1) $$f(x+h) = f(x) + h f'(x) + \frac{h^2}{2} f''(x) + \frac{h^3}{6} f'''(x) + \ldots$$

we are expanding around x and h is a small step away from x

Solve (1) for $f'(x)$:

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

$$= \underbrace{\frac{f(x+h) - f(x)}{h}}_{\text{forward difference approximation}} + O(h)$$

the error is of order h
(size of error term decreases linearly with h)

$h \to -h$

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

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

backward difference

same order error as forward difference

(1) − (2) $$f(x+h) - f(x-h) = 2h f'(x) + \frac{h^3}{3} f'''(x) + \ldots$$

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

error term $\sim h^2$

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

solve for

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

centred difference approximation for the second derivative of a f$^{\underline{n}}$.

= errors example

$$y = \sin x \quad \text{at} \quad x = \pi/4$$

$$y' = \cos x \qquad y'(\pi/4) = \cos(\pi/4) = \sqrt{2}/2$$
$$= 0.70710678 1$$

using forward diff. $$y'(x) \approx \frac{y(x+h) - y(x)}{h}$$

$h = 0.1$ $$y'(\pi/4) \approx \frac{\sin(\pi/4 + 0.1) - \sin(\pi/4)}{0.1}$$

$$= \cdots = 0.67060297$$ error ≈ 0.0366

$h = 0.01$ $$y'(\pi/4) \approx \frac{\sin(\pi/4 + 0.01) - \sin(\pi/4)}{0.01} = \cdots$$

error = 0.00355

decreasing h by a factor of 10
decreases error by a factor of 10

using the centred difference

$$y'(x) \approx \frac{y(x+h) - y(x-h)}{2h}$$

$h = 0.1$

$$y'(\pi/4) \doteq \frac{\sin(\pi/4 + 0.1) - \sin(\pi/4 - 0.1)}{2(0.1)}$$

$$= 0.705928956$$

$$\text{error} \approx 0.00118 \qquad \sim 10^{-3}$$

(comparable to f.d. with $h=0.01$)

$h = 0.01$

$$y'(\pi/4) \doteq \frac{\sin(\pi/4 + 0.01) - \sin(\pi/4 - 0.01)}{2(0.01)}$$

$$= 0.70709491$$

$$\text{error} \approx 0.0000118 \qquad \sim 10^{-5}$$

decreasing h by a factor of 10 decreases error by a factor of $10^2 = 100$

These errors arise essentially from truncating the Taylor series at a certain order of the expansion (truncation error).

There is another source of error, roundoff error, arising from the finite precision used to store numbers on the computer.

e.g. $\left.\frac{d}{dx}\sin x\right|_{\pi/4}$ with forward difference

$h = 10^{-7}$

$$\frac{\sin(\pi/4 + 10^{-7}) - \sin(\pi/4)}{10^{-7}} = 0.70711\ldots$$

error $\sim 3\times10^{-6}$

$h = 10^{-9}$

$$\frac{\sin(\pi/4 + 10^{-9}) - \sin(\pi/4)}{10^{-9}}$$

error $\sim 2\times10^{-3}$

error increases even though h got smaller

$h = 10^{-12}$

$$\frac{\sin(\pi/4 + 10^{-12}) - \sin(\pi/4)}{10^{-12}} = 0$$

(on my calculator)

not enough digits are stored to calculate these differences accurately

idea:

$$\begin{array}{r} 0.12345678\,13 \\ 0.12345677\,61 \\ \hline 0.00000000\,52 \end{array}$$

"double precision"

→ ~15 digits of precision

Simpson's Rule for Integration

Idea

fit parabola through 3 points
→ work with consecutive triplets
→ need an odd # of points

Consider $\int_{-1}^{1} f(x)\,dx$

approximate $f(x) \approx \alpha x^2 + \beta x + \gamma$ (like 2nd order Taylor series)

$$\int_{-1}^{1} f(x)\,dx \approx \int_{-1}^{1} (\alpha x^2 + \beta x + \gamma)\,dx$$

$$= \frac{2}{3}\alpha + 2\gamma$$

Now,

$$f(-1) = \alpha - \beta + \gamma$$
$$f(0) = \gamma \longrightarrow \gamma = f(0)$$
$$f(1) = \alpha + \beta + \gamma$$

solve for α

$$f(-1) + f(1) = 2\alpha + 2\gamma$$

$$\alpha = \frac{f(-1) + f(1)}{2} - f(0)$$

so $$\int_{-1}^{1} f(x)\,dx \approx \frac{2}{3}\alpha + 2\gamma = \frac{2}{3}\left(\frac{f_{-1} + f_1}{2} - f_0\right) + 2f_0$$

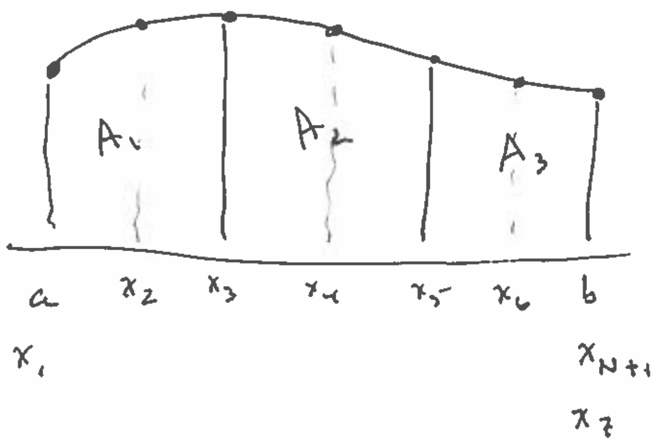
$$= \frac{f_{-1}}{3} + \frac{4}{3}f_0 + \frac{f_1}{3}$$

approximating the integral with 3 f$^{\underline{n}}$ calls with these weights is equivalent to approximating curve with a parabola.

This generalizes to

$$\int_{x_i-h}^{x_i+h} f(x)\,dx \simeq \frac{h}{3} f_{i-1} + \frac{4}{3} h f_i + \frac{h}{3} f_{i+1} \qquad \begin{cases} f_i = f(x_i) \\ f_{i\pm1} = f(x_{i\pm1}) \\ \qquad = f(x_i \pm h) \end{cases}$$

ex



7 points
$N = 6$ slabs
3 areas

$$\int_a^b f(x)\,dx = A_1 + A_2 + A_3$$

$$= \int_{x_2-h}^{x_2+h} f(x)\,dx + \int_{x_4-h}^{x_4+h} f(x)\,dx + \int_{x_6-h}^{x_6+h} f(x)\,dx$$

$$= \left(\frac{h}{3} f_1 + \frac{4}{3} h f_2 + \frac{h}{3} f_3\right) + \left(\frac{h f_3}{3} + \frac{4h}{3} f_4 + \frac{h}{3} f_5\right) + \left(\frac{h}{3} f_5 + \frac{4}{3} h f_6 + \frac{h}{3} f_7\right)$$

$$= \frac{h}{3}\left(f_1 + 4 f_2 + 2 f_3 + 4 f_4 + 2 f_5 + 4 f_6 + f_7\right)$$

In general with $N+1$ points (N even)

$$\int_a^b f(x)\,dx \simeq \frac{h}{3}\left(f_1 + 4f_2 + 2f_3 + \dots + 2f_{N-1} + 4f_N + f_{N+1}\right)$$

odd interior points multiplied by 2

even points multiplied by 4

$\frac{N}{2}$ terms

$\frac{N}{2} - 1$ terms

$$= \frac{h}{3}\left(f_1 + 4 \sum_{i \text{ sum over even points}} f_i + 2 \sum_{i \text{ sum over odd interior points}} f_i + f_{N+1}\right)$$