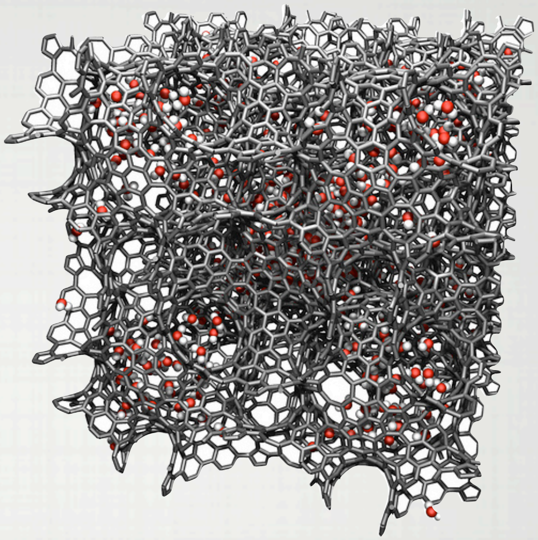


PHY-4810
COMPUTATIONAL PHYSICS

LECTURE 2: NUMERICAL INTEGRATION & DIFFERENTIATION

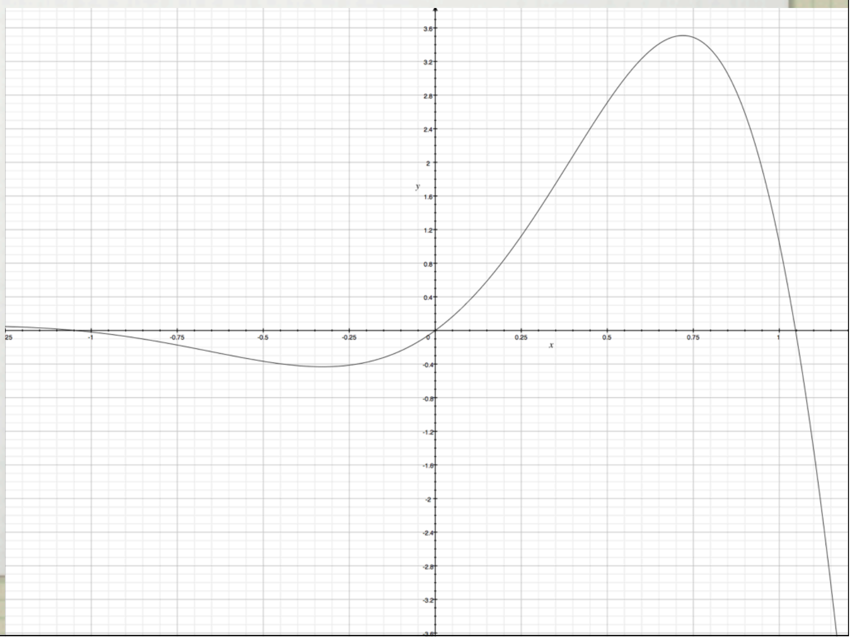
ON THE MENU TODAY

- Part 1: Numerical Integration
  - Trapezoid rule
  - Simpson's rule
  - Gaussian quadrature
- Part 2: Introduction to numerical differentiation
  - Forward difference
  - Central difference
  - Higher order methods, stencil and extrapolated differences

PART 1: INTEGRATION

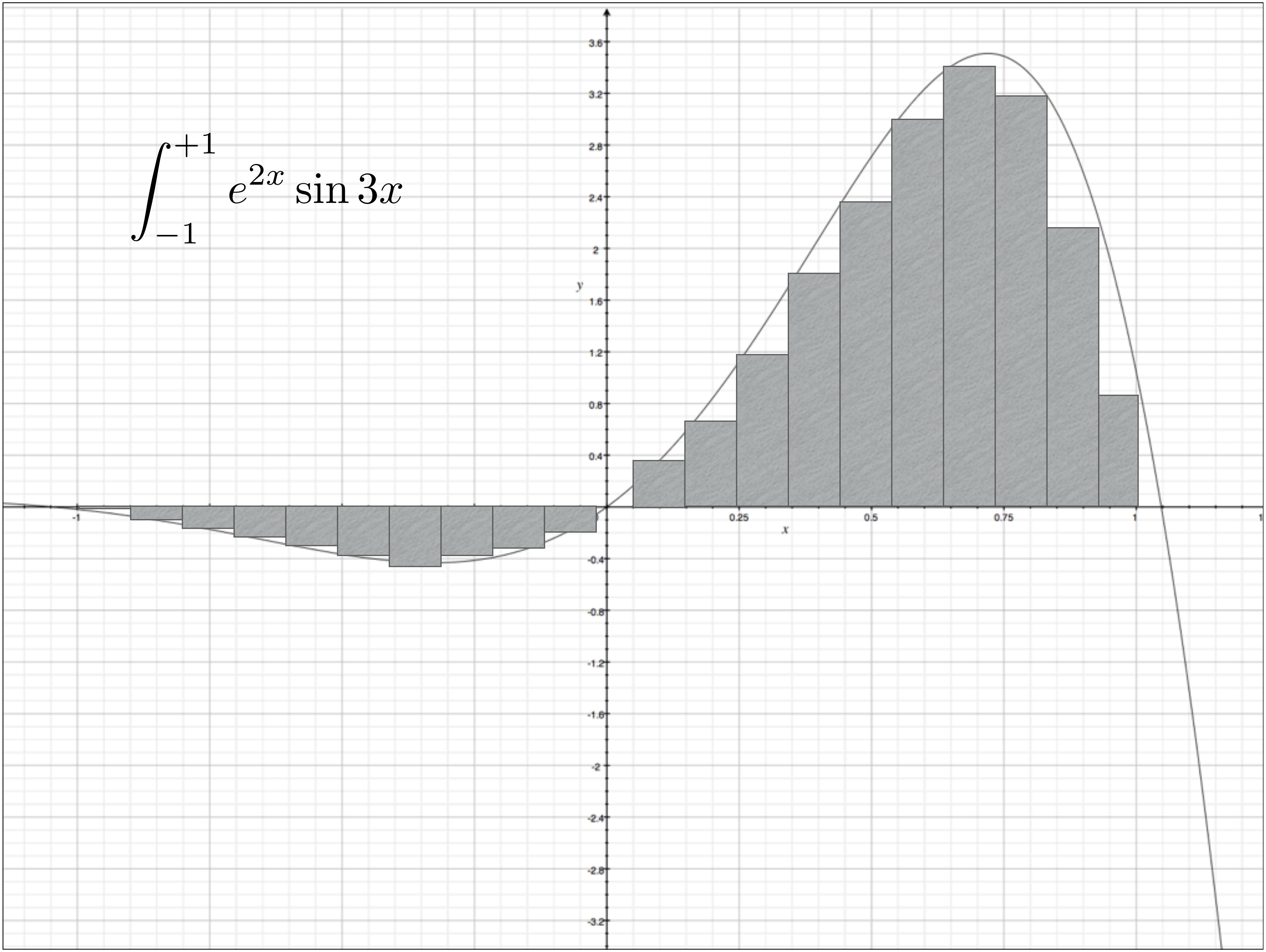
INTEGRATION

$$\int e^{2x}\sin 3x = \frac{e^{2x}(2\sin 3x - 3\cos 3x)}{13} + K$$

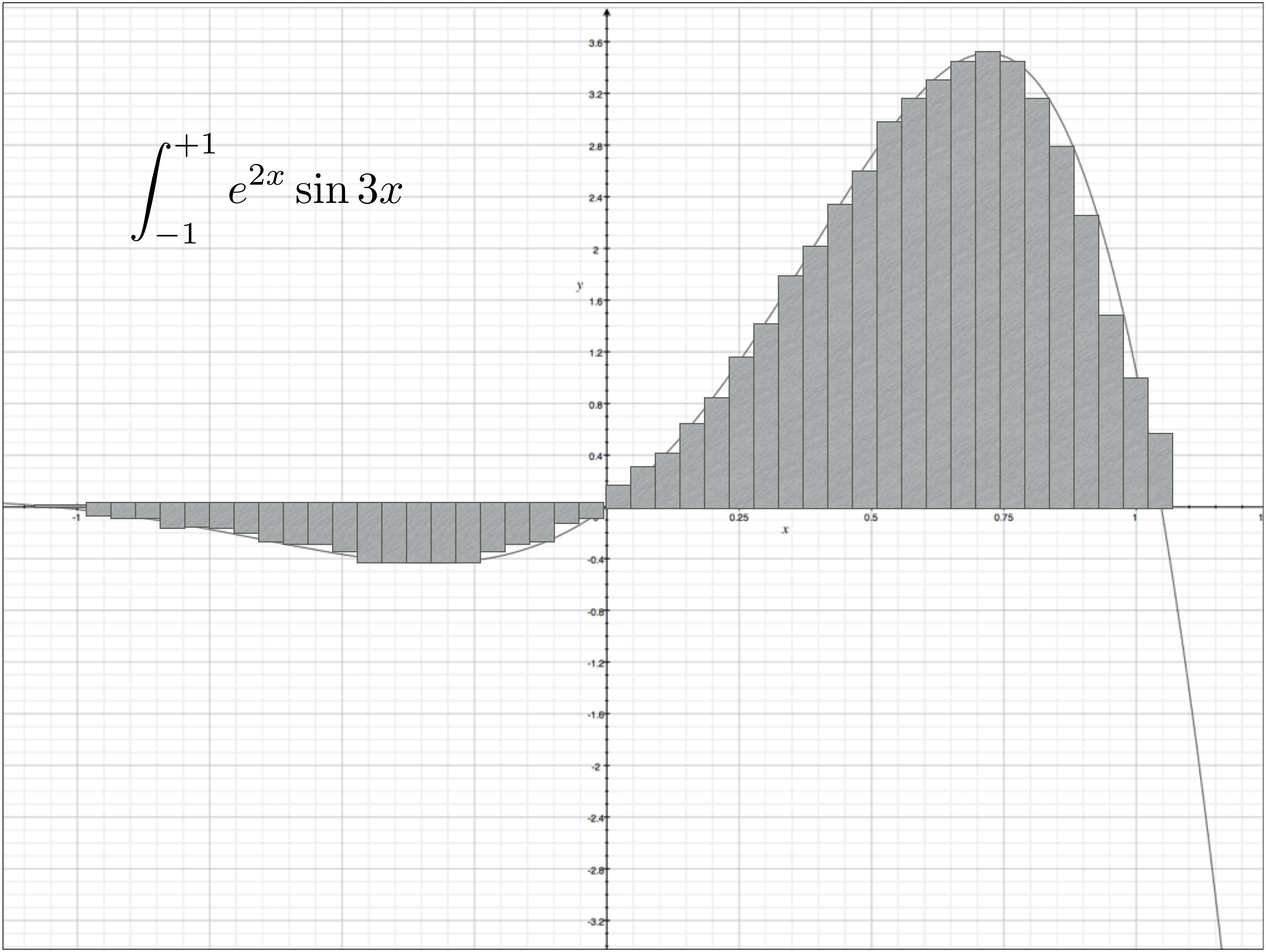


QUADRATURE AS BOX COUNTING

- A traditional way to do numerical integration by hand is to take a piece of graph paper and **count the number of boxes or quadrilaterals lying below a curve of the integrand**.
- For this reason numerical integration is also called **numerical quadrature**, even when it becomes more sophisticated than simple box counting.



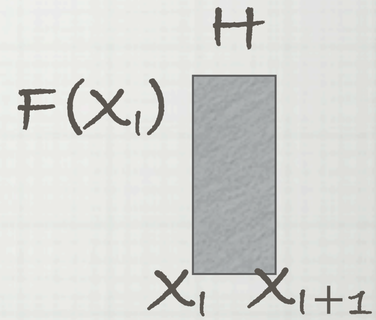
$\int_{-1}^{+1} e^{2x} \sin 3x$
y
x
3.6
3.2
2.8
2.4
2
1.6
1.2
0.8
0.4
-0.4
-0.8
-1.2
-1.6
-2
-2.4
-2.8
-3.2
-1
0.25
0.5
0.75
1



$$\int_{-1}^{+1} e^{2x} \sin 3x$$
y
3.6
3.2
2.8
2.4
2
1.6
1.2
0.8
0.4
-0.4
-0.8
-1.2
-1.6
-2
-2.4
-2.8
-3.2
-1
0.25
0.5
0.75
1
x

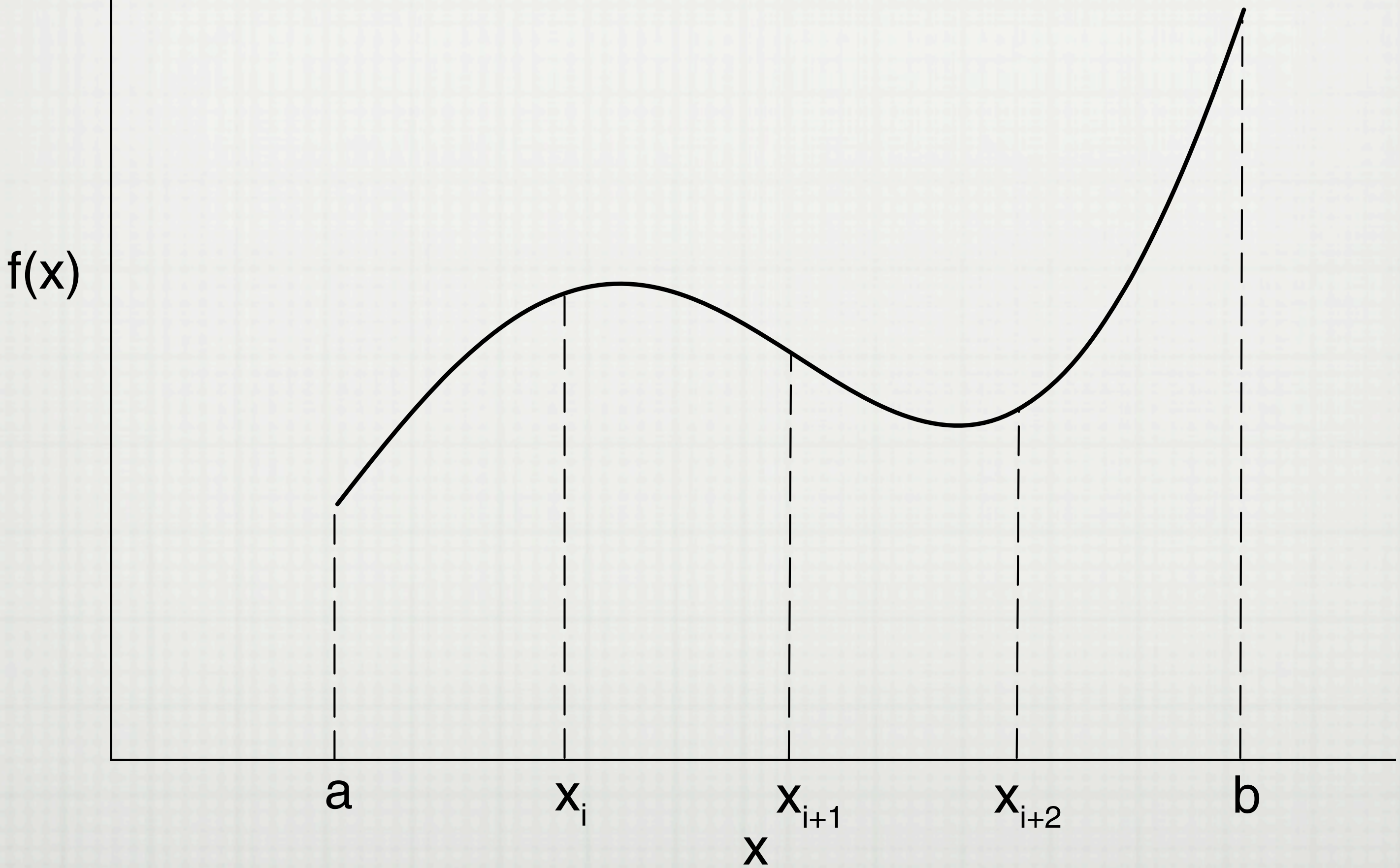
QUADRATURE AS BOX COUNTING : RIEMANN

- The **Riemann definition** of an integral is the limit of the sum over boxes as the width h of the box approaches zero:

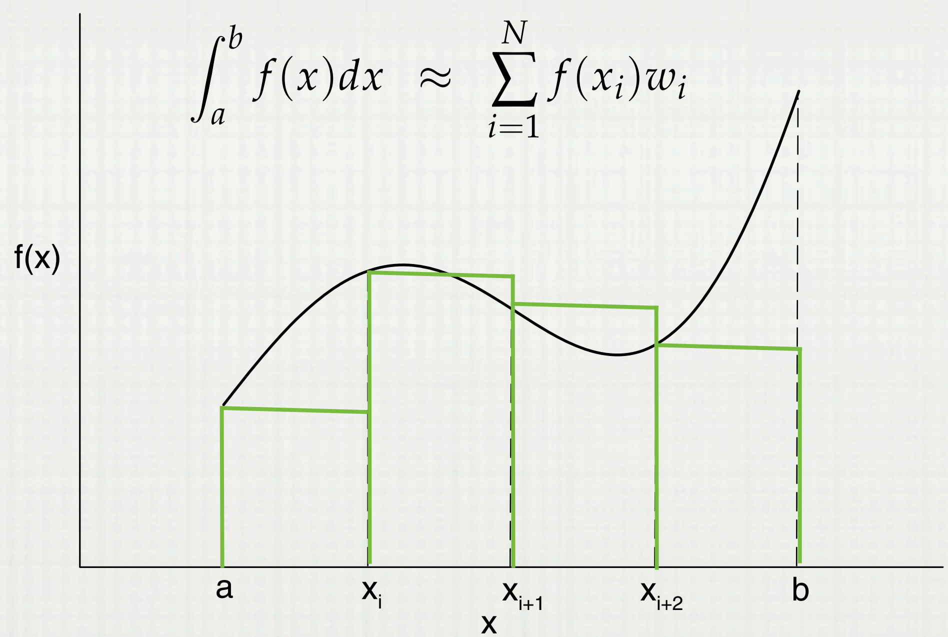


$$\boxed{\int_a^b f(x)dx = \lim_{h \to 0} \left[h \sum_{i=1}^{(b-a)/h} f(x_i) \right]}$$

$$\int_a^b f(x)dx = \lim_{h \to 0} \left[h \sum_{i=1}^{(b-a)/h} f(x_i) \right]$$

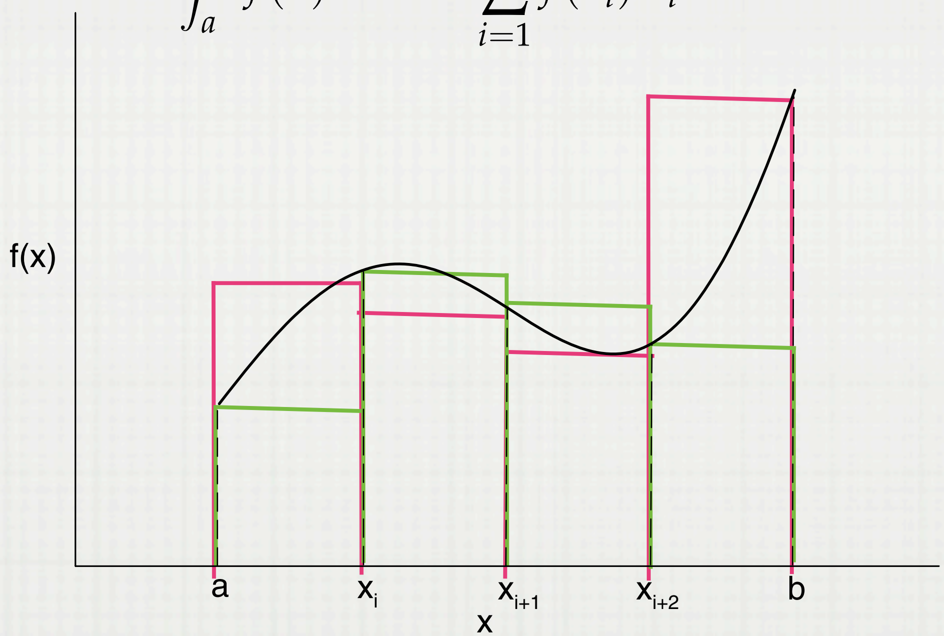


NUMERICAL INTEGRATION: THE ART OF COUNTING BOXES



TOP LEFT CORNER APPROXIMATION

$$\int_a^b f(x)dx \approx \sum_{i=1}^{N} f(x_i)w_i$$

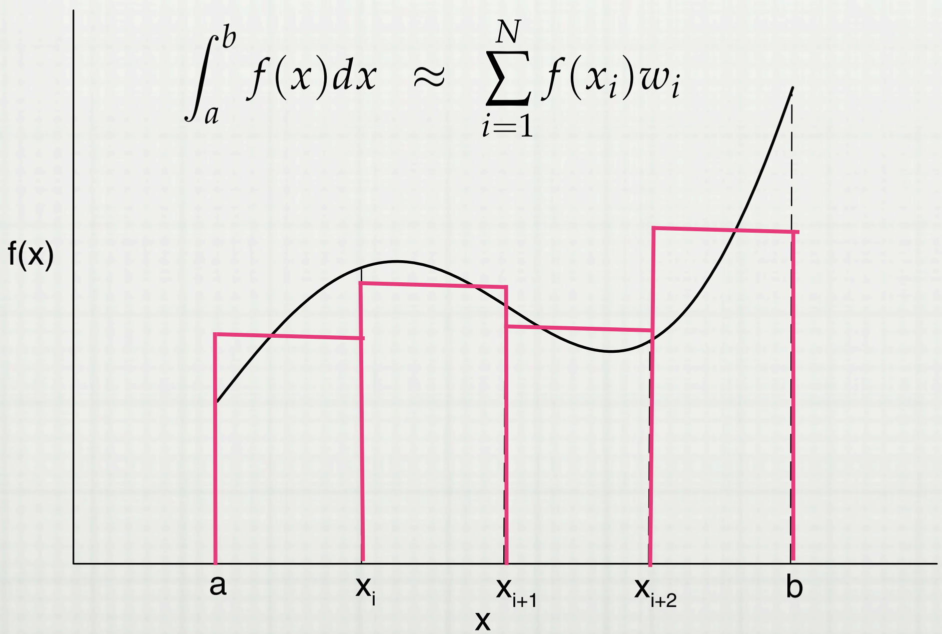


TOP RIGHT CORNER APPROXIMATION

TOP LEFT CORNER APPROXIMATION

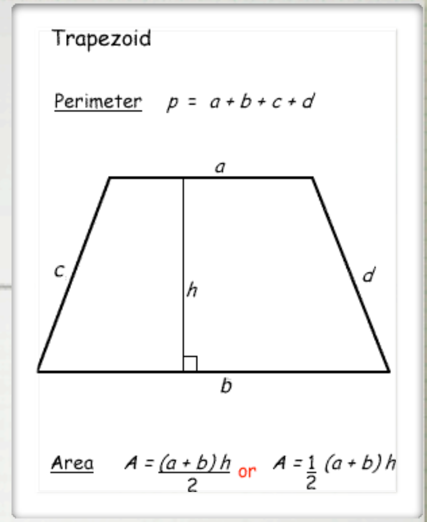
BETTER APPROXIMATION

$$\int_a^b f(x)dx \approx \sum_{i=1}^{N} f(x_i)w_i$$



MID-POINT APPROXIMATION

TRAPEZOID RULE

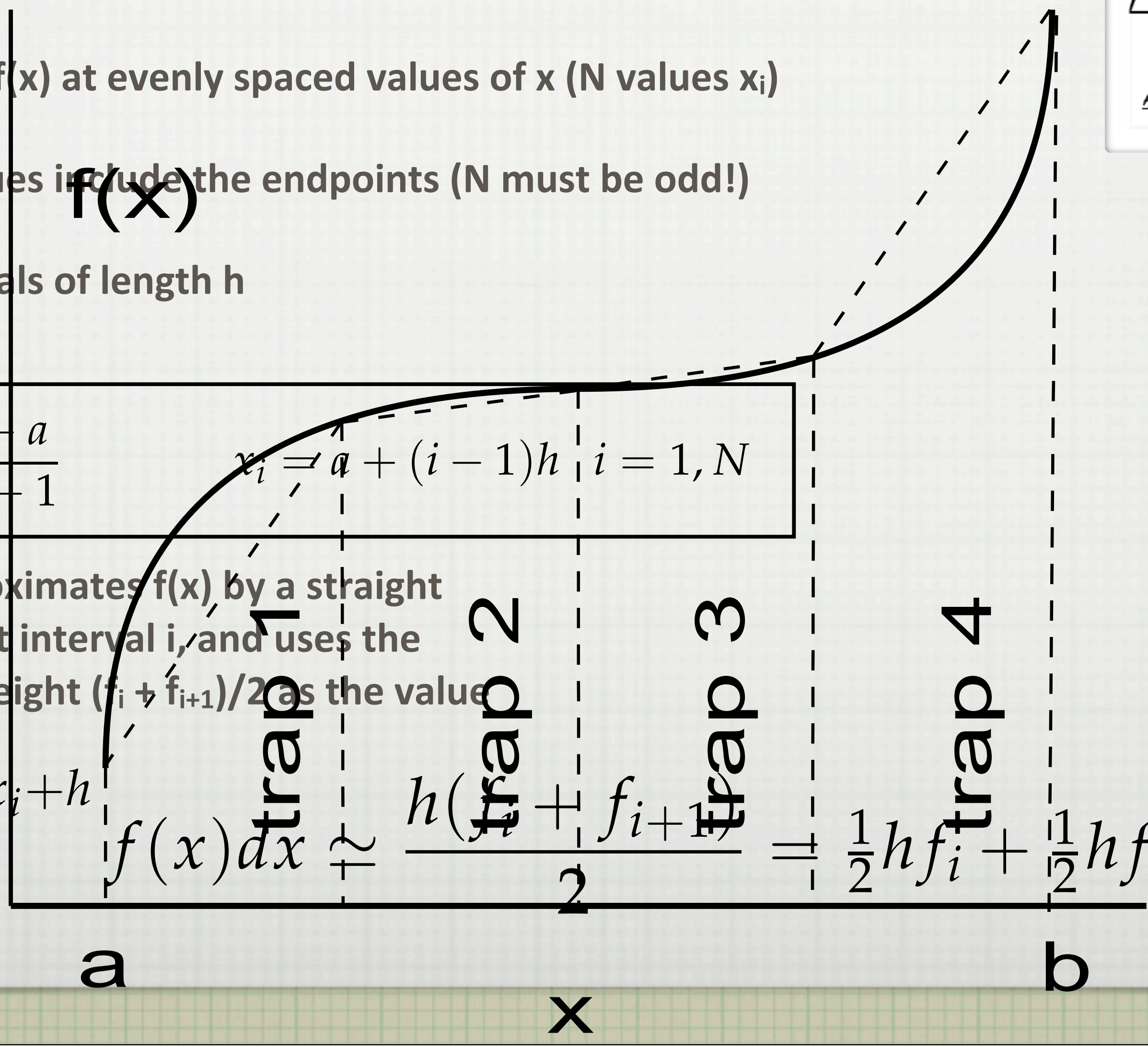


- Values of f(x) at evenly spaced values of x (N values x_i)
- The N values include the endpoints (N must be odd!)
- N-1 intervals of length h

$$h = \frac{b-a}{N-1} \qquad x_i = a + (i-1)h \quad i = 1, N$$

- This approximates f(x) by a straight line in that interval i, and uses the average height $(f_i + f_{i+1})/2$ as the value for f

$$\int_{x_i}^{x_i+h} f(x)dx \simeq \frac{h(f_i + f_{i+1})}{2} = \frac{1}{2}hf_i + \frac{1}{2}hf_{i+1}$$



TRAPEZOID RULE (CONTINUED)

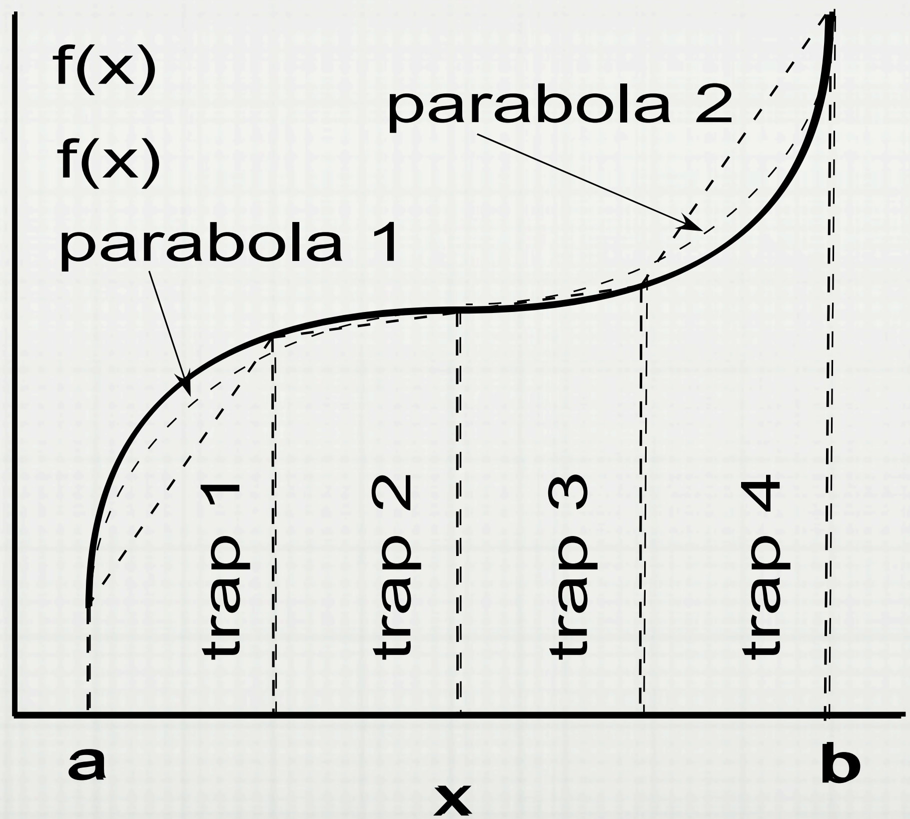
- **Finally:**

$$\boxed{\int_a^b f(x)dx \approx \frac{h}{2}f_1 + hf_2 + hf_3 + \cdots hf_{N-1} + \frac{h}{2}f_N}$$

- **We end up with a simple formula, compared to the original approximation! Just the weight at the end points have changed.**

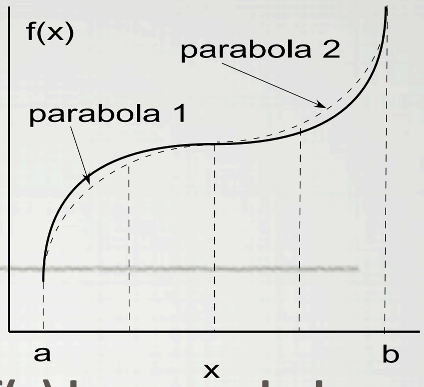
$$w_i = \left\{\frac{h}{2}, h, \ldots, h, \frac{h}{2}\right\}$$

SEEKING AN EVEN BETTER APPROXIMATION



f(x)
parabola 2
f(x)
parabola 1
trap 1
trap 2
trap 3
trap 4
a
b
x

SIMPSON RULE



- For each interval, Simpson's rule approximates the integrand f(x) by a parabola

$$f(x) \approx \alpha x^2 + \beta x + \gamma$$

- The area of each section is then this integral of this parabola

$$\int_{x_i}^{x_i+h} (\alpha x^2 + \beta x + \gamma) dx = \frac{\alpha x^3}{3} + \frac{\beta x^2}{2} + \gamma x \Big|_{x_i}^{x_i+h}$$

- This is equivalent to integrating the Taylor series up to the quadratic term.

SIMPSON RULE (2)

$$f(x) \approx \alpha x^2 + \beta x + \gamma$$

- In order to relate the parameters α, β, and γ to the function, we consider an interval from −1 to +1, in which case

$$\int_{-1}^{1} (\alpha x^2 + \beta x + \gamma) dx = \frac{2\alpha}{3} + 2\gamma$$

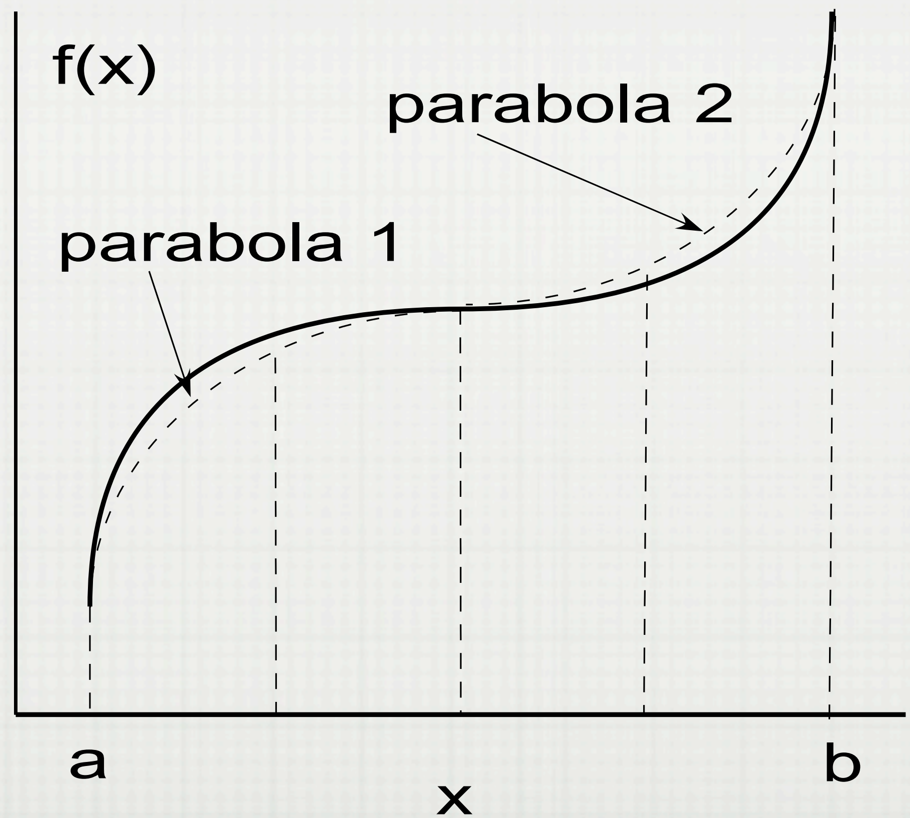
- We also see that:

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

$$\Rightarrow \quad \alpha = \tfrac{f(1)+f(-1)}{2} - f(0) \quad \beta = \tfrac{f(1)-f(-1)}{2} \qquad \gamma = f(0)$$

- In this way we can express the integral as the weighted sum over the values of the function at three points:

$$\int_{-1}^{1} (\alpha x^2 + \beta x + \gamma) dx = \frac{f(-1)}{3} + \frac{4f(0)}{3} + \frac{f(1)}{3}$$



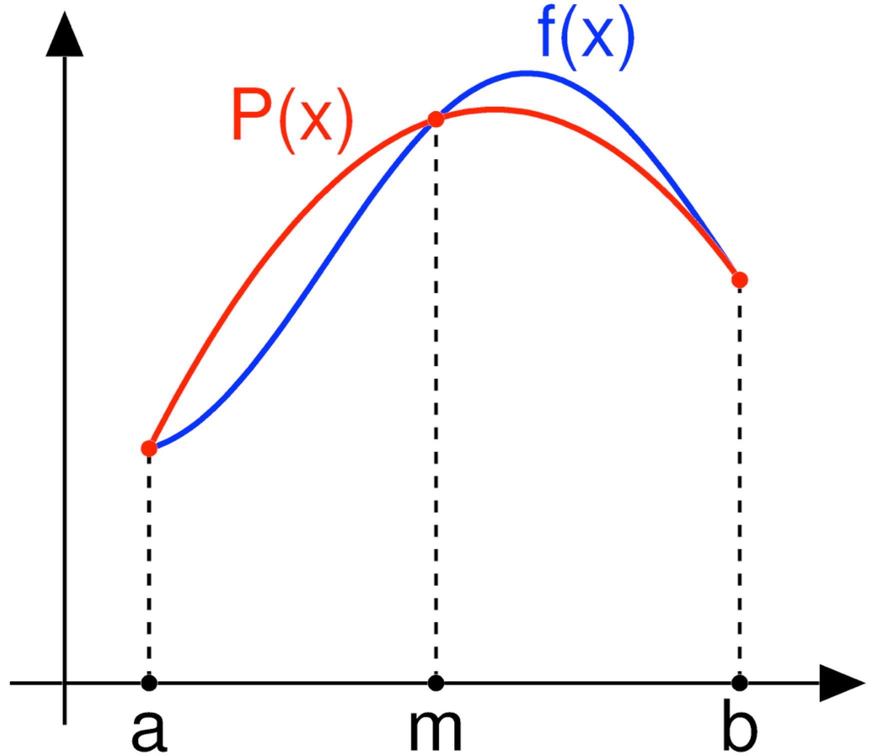
[PICTURE FROM BOOK:] WHAT'S WRONG WITH THIS PICTURE?

SUMMARY OF SIMPSON'S RULE

SIMPSON'S RULE CAN BE DERIVED BY APPROXIMATING THE INTEGRAND F (X) (IN BLUE) BY THE QUADRATIC INTERPOLANT P (X) (IN RED).

$$\int_a^b f(x)\,dx \approx \frac{b-a}{6}\left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right]$$

ERROR: $\frac{(b-a)^5}{180n^4}\left|f^{(4)}(\xi)\right|$



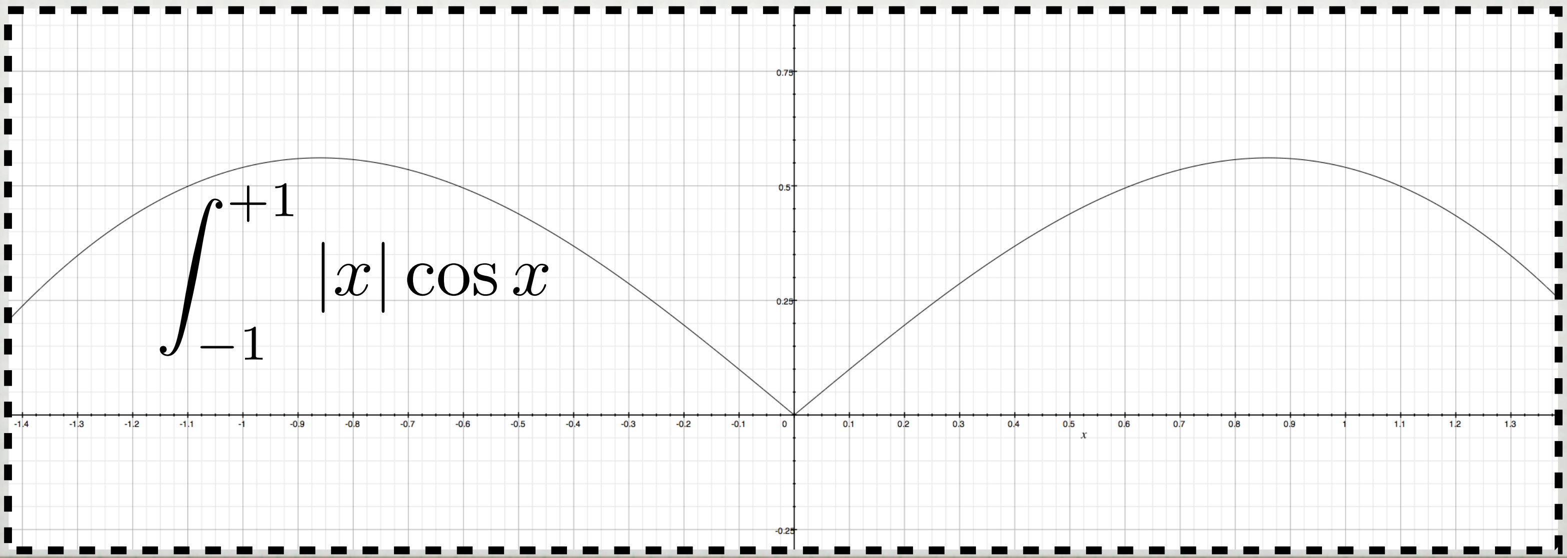
SMOOTH OR SINGULAR FUNCTION

- Singularity: You may be able to remove the singularity, by breaking the interval down into several subintervals, so the singularity is at an endpoint where a Gauss point never falls, or by a change of variable.

  - Example:

$$\int_{-1}^{1} |x| f(x) dx = \int_{-1}^{0} f(-x) dx + \int_{0}^{1} f(x) dx$$

- Smooth region: Increase the step locally and create a non-uniform grid.



GAUSSIAN QUADRATURE

- **Fundamental Theorem:**
  - Seeks to obtain the best numerical estimate of an integral by picking optimal abscissas at which to evaluate the function f(x).
- *The fundamental theorem of Gaussian quadrature states that the optimal abscissas x_i of the m-point Gaussian quadrature formulas are precisely the roots of the orthogonal polynomial for the same interval and weighting function.*
  - Gaussian quadrature is optimal because it fits all polynomials up to degree *n* exactly. (m-pointGC -> exact for 2m-1 degrees)

INTRODUCTION TO GAUSSIAN QUADRATURE: TWO-POINT GAUSS QUADRATURE RULE

- **Trapezoid Rule:** The approximation is exact for a straight line

$$\int_a^b f(x)dx \approx c_1 f(a) + c_2 f(b)$$

- **Gaussian Quadrature:** The two-point Gauss quadrature rule is an extension of the trapezoidal rule approximation where the arguments of the function are not predetermined as a and b, but as unknowns x_1 and x_2 .

$$I = \int_a^b f(x)dx$$

$$\approx c_1 f(x_1) + c_2 f(x_2)$$

  - There are four unknowns x_1 , x_2 , c_1 and c_2 .

  - These are found by assuming that the formula gives exact results for integrating a general third order polynomial, $f(x)=a_0 + a_1x + a_2x^2 + a_3x^3$.

$$c_1 = \frac{b-a}{2}$$

$$c_2 = \frac{b-a}{2}$$

$$x_1 = \left(\frac{b-a}{2}\right)\left(-\frac{1}{\sqrt{3}}\right) + \frac{b+a}{2}$$

$$x_2 = \left(\frac{b-a}{2}\right)\left(\frac{1}{\sqrt{3}}\right) + \frac{b+a}{2}$$

$$\int_a^b f(x)dx \approx \frac{b-a}{2} f\left(\frac{b-a}{2}\left(-\frac{1}{\sqrt{3}}\right) + \frac{b+a}{2}\right) + \frac{b-a}{2} f\left(\frac{b-a}{2}\left(\frac{1}{\sqrt{3}}\right) + \frac{b+a}{2}\right)$$

INTRODUCTION TO GAUSSIAN QUADRATURE: THREE-POINT GAUSS QUADRATURE RULE

- **Three points:**

$$\int_a^b f(x)dx \approx c_1 f(x_1) + c_2 f(x_2) + c_3 f(x_3)$$

- **Weights and abscissas chosen for the approximation to be exact for a five order polynomial:**

$$\int_a^b \left(a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + a_5 x^5\right)dx \,.$$

IT CAN BE SHOWN THAT THE EVALUATION POINTS ARE JUST THE ROOTS OF A POLYNOMIAL BELONGING TO A CLASS OF ORTHOGONAL POLYNOMIALS. (LEGENDRE POLYNOMIAL)

N-POINT GAUSS QUADRATURE RULE

- **General formulation:**

$$\int_{-1}^{1} g(x)dx \approx \sum_{i=1}^{n} c_i g(x_i)$$

- **Problem:**

So if the table is given for $\int_{-1}^{1} g(x)dx$ integrals, how does one solve $\int_{a}^{b} f(x)dx$?

- **Solution:**

$$\int_{a}^{b} f(x)dx = \int_{-1}^{1} f\left(\frac{b-a}{2}x + \frac{b+a}{2}\right)\frac{b-a}{2}dx$$

Points	Weighting Factors	Function Arguments
2	$c_1 = 1.000000000$	$x_1 = -0.577350269$
	$c_2 = 1.000000000$	$x_2 = 0.577350269$
3	$c_1 = 0.555555556$	$x_1 = -0.774596669$
	$c_2 = 0.888888889$	$x_2 = 0.000000000$
	$c_3 = 0.555555556$	$x_3 = 0.774596669$
4	$c_1 = 0.347854845$	$x_1 = -0.861136312$
	$c_2 = 0.652145155$	$x_2 = -0.339981044$
	$c_3 = 0.652145155$	$x_3 = 0.339981044$
	$c_4 = 0.347854845$	$x_4 = 0.861136312$
5	$c_1 = 0.236926885$	$x_1 = -0.906179846$
	$c_2 = 0.478628670$	$x_2 = -0.538469310$
	$c_3 = 0.568888889$	$x_3 = 0.000000000$
	$c_4 = 0.478628670$	$x_4 = 0.538469310$
	$c_5 = 0.236926885$	$x_5 = 0.906179846$
6	$c_1 = 0.171324492$	$x_1 = -0.932469514$
	$c_2 = 0.360761573$	$x_2 = -0.661209386$
	$c_3 = 0.467913935$	$x_3 = -0.238619186$
	$c_4 = 0.467913935$	$x_4 = 0.238619186$

GC - REMARKS

- If the function being integrated is not smooth (for instance, if it contains noise), then using a higher order method such as Gaussian quadrature may well lead to lower accuracy.
- Sometimes the function may not be smooth because it has different behaviors in different regions. In these cases it makes sense to integrate each region separately and then add the answers together.
- In fact, some of the “smart” integration subroutines will decide for themselves how many intervals to use and what rule to use in each interval.

SUMMARY: GAUSSIAN QUADRATURE

- In the Gaussian quadrature approach to integration, the N points and weights are chosen to make the approximation **error actually vanish for *g(x)*, a *2N – 1* degree polynomial.** To obtain this incredible optimization, the points x_i end up having a very specific distribution over *[a, b]*.

$$\int_a^b f(x)dx \equiv \int_a^b W(x)g(x)dx \approx \sum_{i=1}^{N} w_i g(x_i)$$

- In general, if g(x) is smooth, or can be made smooth by factoring out some W(x), Gaussian algorithms produce higher accuracy than lower order ones, or conversely, the same accuracy with a fewer number of points.

$$w_i = \frac{2}{(1-x_i^2)\,[P_n'(x_i)]^2}$$

For the integration problem stated above, the associated polynomials are Legendre polynomials, $P_n(x)$. With the nth polynomial normalized to give $P_n(1) = 1$, the ith Gauss node, x_i, is the ith root of P_n; its weight is given by (Abramowitz & Stegun 1972, p. 887)

```fortran
SUBROUTINE gauleg(x1,x2,x,w,n)
  use kinds
  integer, intent(in) ::  n
  real(dbl), intent(in) ::  x1,x2
  real(dbl), dimension(n),  intent(out) :: x, w
  real(dbl), parameter :: EPS=3.0D-14
!  real(dbl), parameter :: EPS=3.D-14
  integer i,j,m
  real(dbl) :: p1,p2,p3,pp,xl,xm,z,z1,pi
  m=(n+1)/2
  xm=0.5d0*(x2+x1)
  xl=0.5d0*(x2-x1)
  pi=4.0D0*atan(1.0D0)
  do i=1,m
     z=cos(pi*(i-.25d0)/(n+.5d0))
1    continue
     p1=1.0d0
     p2=0.0d0
     do j=1,n
        p3=p2
        p2=p1
        p1=((2.0d0*j-1.0d0)*z*p2-(j-1.0d0)*p3)/j
     end do
     pp=n*(z*p1-p2)/(z*z-1.0d0)
     z1=z
     z=z1-p1/pp
     if(abs(z-z1).gt.EPS) goto 1
     x(i)=xm-xl*z
     x(n+1-i)=xm+xl*z
     w(i)=2.d0*xl/((1.0d0-z*z)*pp*pp)
     w(n+1-i)=w(i)
     end do
     return
 END SUBROUTINE gauleg
```

FORTRAN

```cpp
#include <cmath>
#include "nr.h"
using namespace std;

void NR::gauleg(const DP x1, const DP x2, Vec_O_DP &x, Vec_O_DP &w)
{
        const DP EPS=1.0e-14;
        int m,j,i;
        DP z1,z,xm,xl,pp,p3,p2,p1;

        int n=x.size();
        m=(n+1)/2;
        xm=0.5*(x2+x1);
        xl=0.5*(x2-x1);
        for (i=0;i<m;i++) {
                z=cos(3.141592654*(i+0.75)/(n+0.5));
                do {
                        p1=1.0;
                        p2=0.0;
                        for (j=0;j<n;j++) {
                                p3=p2;
                                p2=p1;
                                p1=((2.0*j+1.0)*z*p2-j*p3)/(j+1);
                        }
                        pp=n*(z*p1-p2)/(z*z-1.0);
                        z1=z;
                        z=z1-p1/pp;
                } while (fabs(z-z1) > EPS);
                x[i]=xm-xl*z;
                x[n-1-i]=xm+xl*z;
                w[i]=2.0*xl/((1.0-z*z)*pp*pp);
                w[n-1-i]=w[i];
        }
}
```

C++

PART 2: DIFFERENTIATION

OUR QUEST:

$$\boxed{\frac{df(x)}{dx} \stackrel{\text{def}}{=} \lim_{h\to 0} \frac{f(x+h) - f(x)}{h}}$$

OUR STARTING POINT (TAYLOR):

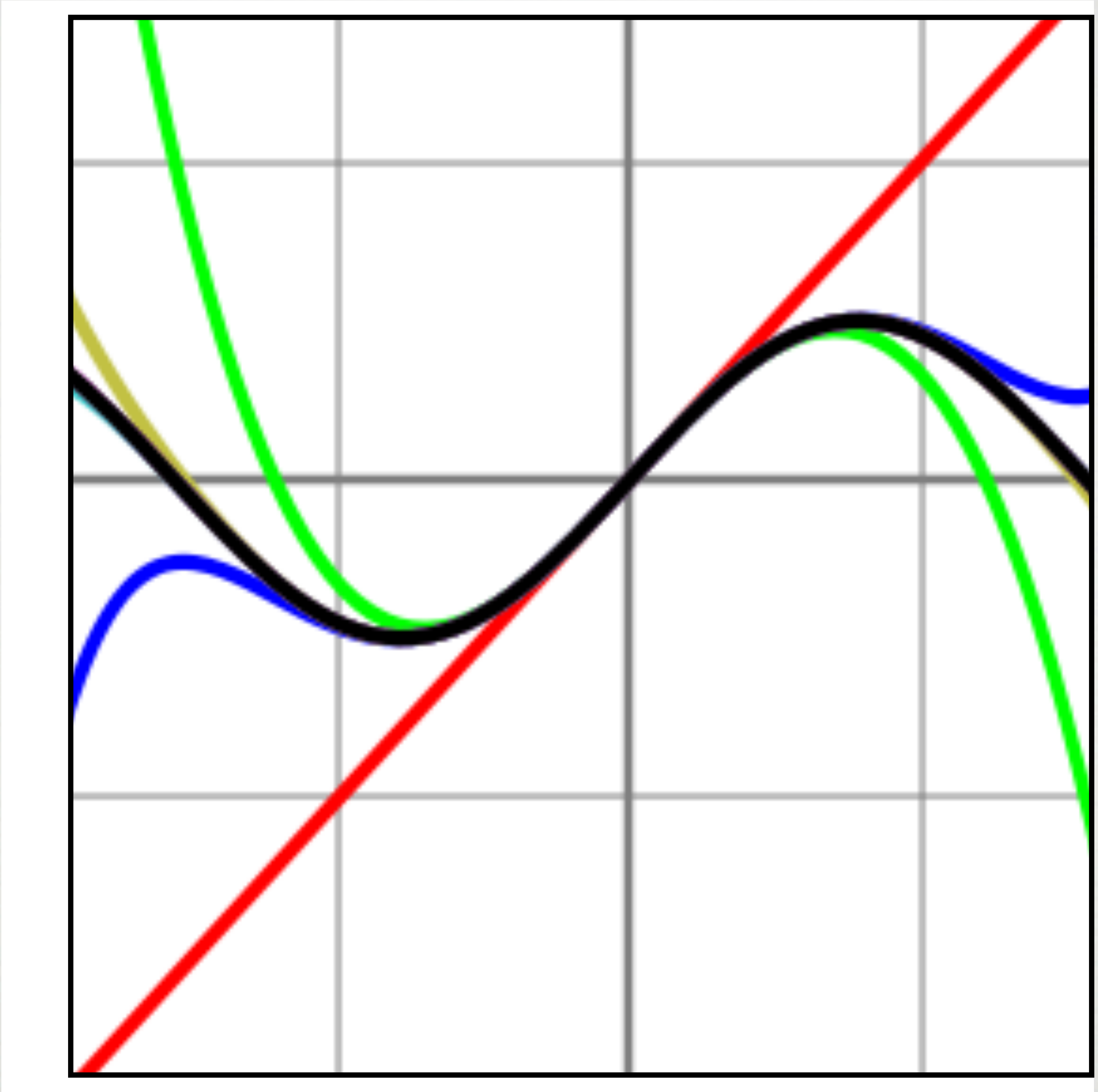
$$\boxed{f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f^{(3)}(x) + \cdots}$$

EXAMPLE: SINE (X) AROUND X=0

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

- f(x)=sin(x) =0 at x=0
- f'(x)=cos(x) = 1 at x =0
- f''(x)=-sin(x) = 0 at x=1
- f'''(x)=-1 at x=0
- ...

→ f(h)=h-h³/6



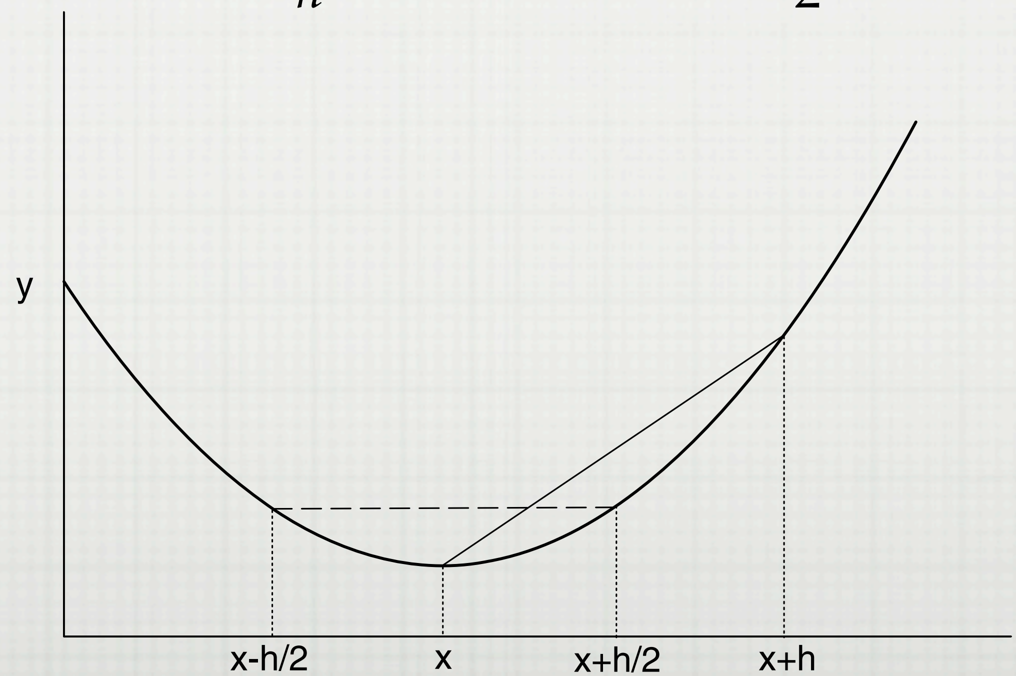
$f(h)=h-h^3/6$

TAYLOR SERIES OF SINE (X)

FORWARD DIFFERENCE

$$f'_{fd}(x) \stackrel{\text{def}}{=} \frac{f(x+h)-f(x)}{h}, \simeq f'(x) + \frac{h}{2} f''(x) + \cdots$$

- What is f' at x?



FORWARD DIFFERENCE: ERROR ESTIMATION

- $f(x+h)=f(x)+h\,f'(x) + O(h^2)$
- $f(x+h)-f(x+h)=h\,f'(x) + O(h^2)$
- $f'(x) = (f(x+h)-f(x))/h \;\; + O(h)$

EXAMPLE:

$$\boxed{f(x) \;=\; a + bx^2}$$

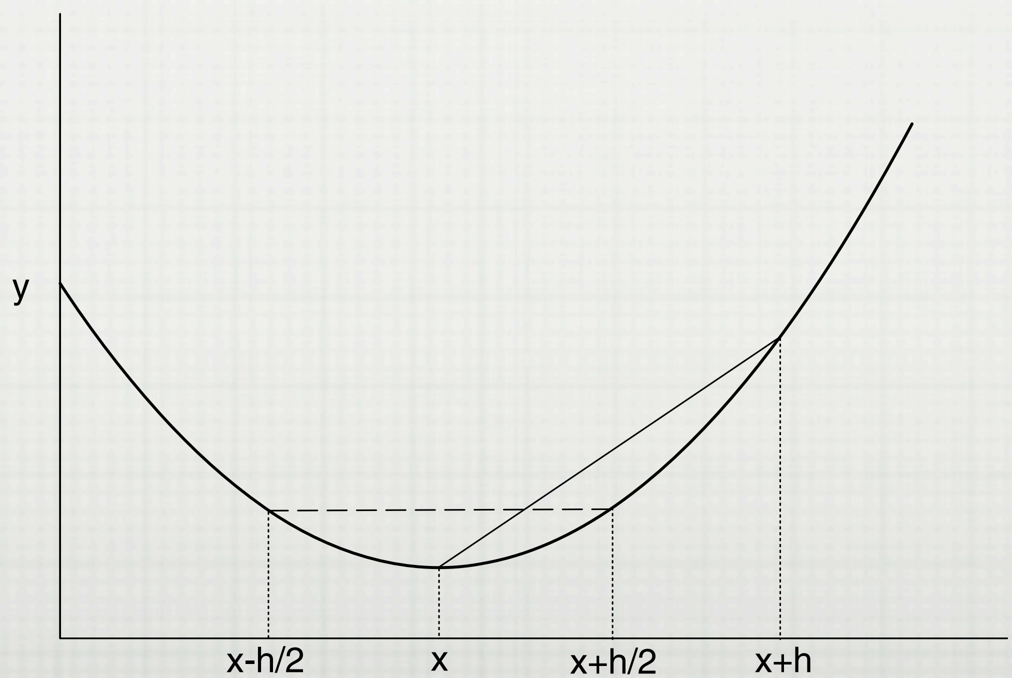
ANALYTICALLY: $f' = 2bx$

NUMERICALLY: $f'_{fd}(x) \;\approx\; \dfrac{f(x+h) - f(x)}{h} = 2bx + bh$

This clearly becomes a good approximation only for small h (h ≪ 2x).

CENTRAL DIFFERENCE

$$f'_{cd}(x) \stackrel{\text{def}}{=} \frac{f(x+h/2) - f(x-h/2)}{h} = D_{cd}f(x,h)$$



CENTRAL DIFFERENCE: ERROR ESTIMATION

- $f(x+h/2)=f(x)+h/2\ f'(x) + h^2/8\ f''(x) + O(h^3)$
- $f(x-h/2)=f(x)-h/2 f'(x) + h^2/8\ f''(x) + O(h^3)$
- $f(x+h/2)-f(x-h/2)=h\ f'(x) + O(h^3)$
- $f'(x) = (f(x+h/2)-f(x-h/2))/h + O(h^2)$

CENTRAL DIFFERENCE

$$\boxed{f'_{cd}(x) \quad \stackrel{\text{def}}{=} \quad \frac{f(x+h/2) - f(x-h/2)}{h} = D_{cd}f(x,h)}$$

- ☐ **The important difference with FD is that when f (x – h/2) is subtracted from f(x+h/2), all terms containing an odd power of h in the Taylor series cancel. Therefore, the central-difference algorithm becomes accurate to one order higher in h, that is, h^2.**
- ☐ **If the function is well behaved, that is, if $f(3)h^2/24 \ll f(2)h/2$, then you can expect the error with the central-difference method to be smaller than with the forward difference**

EXAMPLE:

$$f(x) = a + bx^2$$

ANALYTICALLY: $f' = 2bx$

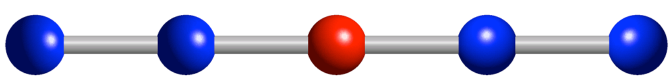
NUMERICALLY: $f'_{cd}(x) \approx \dfrac{f(x+h/2) - f(x-h/2)}{h} = 2bx$

THE CENTRAL DIFFERENCE IS DEFINITELY AN IMPROVEMENT!

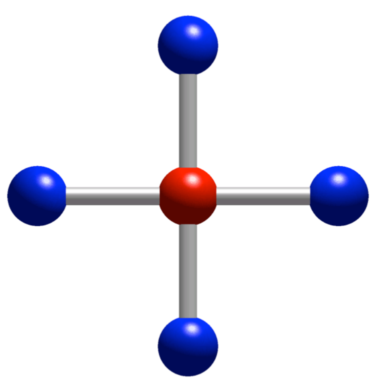
HIGHER ORDER METHODS (EXTRAPOLATED DIFFERENCES)

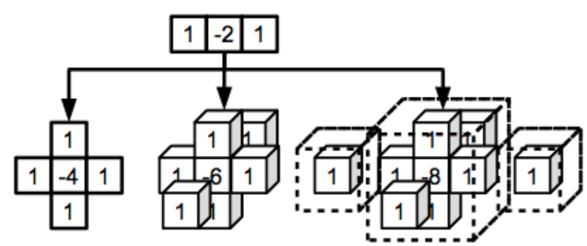
- 5 point method: (five-point stencil)

$$\{x-2h, x-h, x, x+h, x+2h\}.$$



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





PROOF OF THE 5-POINT STENCIL FORMULA

- **Outline of the proof:**

$$f(x \pm h) = f(x) \pm h f'(x) + \frac{h^2}{2} f''(x) \pm \frac{h^3}{6} f^{(3)}(x) + O_{1\pm}(h^4). \qquad (E_{1\pm}).$$

$$f(x+h) - f(x-h) = 2hf'(x) + \frac{h^3}{3} f^{(3)}(x) + O_1(h^4). \qquad (E_1).$$

$$f(x \pm 2h) = f(x) \pm 2hf'(x) + 2h^2 f''(x) \pm \frac{4h^3}{3} f^{(3)}(x) + O_{2\pm}(h^4). \qquad (E_{2\pm})$$

$$8f(x+h) - 8f(x-h) - f(x+2h) + f(x-2h) = 12hf'(x) + O(h^4)$$

- **Error estimation:**

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

MORE ON STENCILS: 7 AND 9 POINT

- **Website of Pavel Holoborodko**
  - **http://www.holoborodko.com/pavel/numerical-methods/numerical-derivative/central-differences/**

N	N-point stencil Central Differences	
3	$\dfrac{f_1 - f_{-1}}{2h}$	$O(h^2)$
5	$\dfrac{f_{-2} - 8f_{-1} + 8f_1 - f_2}{12h}$	$O(h^4)$
7	$\dfrac{-f_{-3} + 9f_{-2} - 45f_{-1} + 45f_1 - 9f_2 + f_3}{60h}$	$O(h^6)$
9	$\dfrac{3f_{-4} - 32f_{-3} + 168f_{-2} - 672f_{-1} + 672f_1 - 168f_2 + 32f_3 - 3f_4}{840h}$	

HIGHER DERIVATIVES

$$\begin{aligned}
f''(x) &\approx \frac{-f(x+2h)+16f(x+h)-30f(x)+16f(x-h)-f(x-2h)}{12h^2},\\
f^{(3)}(x) &\approx \frac{f(x+2h)-2f(x+h)+2f(x-h)-f(x-2h)}{2h^3},\\
f^{(4)}(x) &\approx \frac{f(x+2h)-4f(x+h)+6f(x)-4f(x-h)+f(x-2h)}{h^4}.
\end{aligned}$$

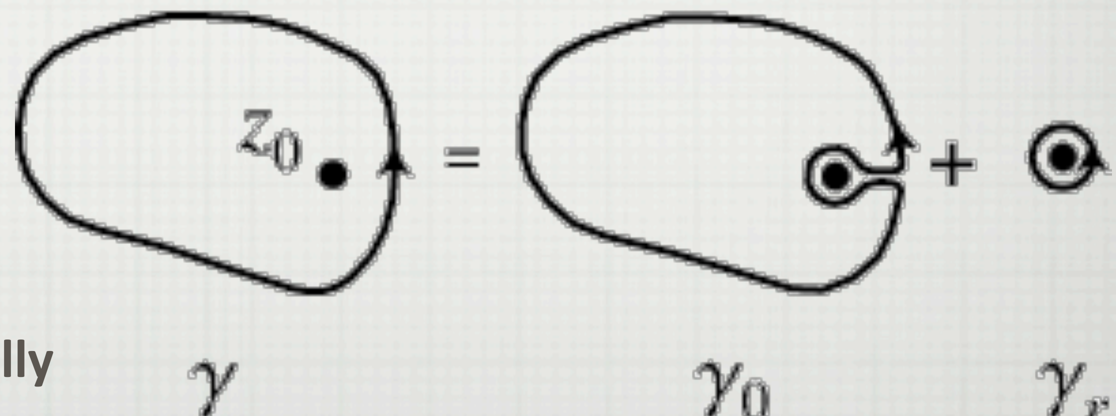
SECOND DERIVATIVES USING STENCILS

N	Second order Central Differences	
3	$\dfrac{f_{-1} - 2f_0 + f_1}{h^2}$	$O(h^2)$
5	$\dfrac{-f_{-2} + 16f_{-1} - 30f_0 + 16f_1 - f_2}{12h^2}$	$O(h^4)$
7	$\dfrac{2f_{-3} - 27f_{-2} + 270f_{-1} - 490f_0 + 270f_1 - 27f_2 + 2f_3}{180h^2}$	$O(h^6)$

N	Third order Central Differences	Error
5	$\dfrac{-f_{-2} + 2f_{-1} - 2f_1 + f_2}{2h^3}$	$O(h^2)$
7	$\dfrac{f_{-3} - 8f_{-2} + 13f_{-1} - 13f_1 + 8f_2 - f_3}{8h^3}$	$O(h^4)$

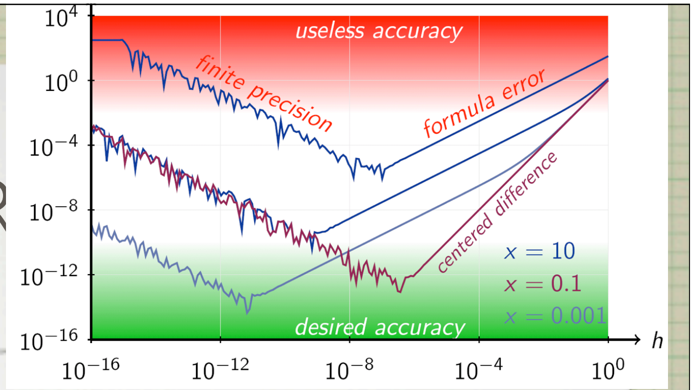
DERIVATIVE USING CAUCHY'S INTEGRAL FORMULA

$$f^{(n)}(a) = \frac{n!}{2\pi i} \oint_\gamma \frac{f(z)}{(z-a)^{n+1}} dz$$



- egral is done numerically

PRACTICAL CONSIDER



$$f'_{cd}(x) \approx \frac{f(x+h/2)-f(x-h/2)}{h} = 2bx$$

- An important consideration in practice when the function is approximated using floating point arithmetic is how small a value of h to choose.
  - If chosen too small, the subtraction will yield a large rounding error and due to cancellation will produce a value of zero if h is small enough.
  - If too large, the calculation of the slope of the secant line will be more accurate, but the estimate of the slope of the tangent by using the secant could be worse.

"MACHINE EPSILON"

- Definition:
  - The largest number that, when added to another number, does not change it.

$$x = x + \epsilon$$

PRACTICAL CONSIDERATIONS

- A good choice for h is:

$$h = x \times \sqrt{\epsilon}$$

- Where the machine epsilon (DP) is typically of the order $2.2e^{-16}$.

- Another important consideration is to make sure that h and x+h are representable in floating point precision so that the difference between x+h and x is exactly h.

- This can be accomplished by placing their values into and out of memory as follows:

$$h = x \times \sqrt{\epsilon}$$
$$temp = x + h$$
$$h = temp - x$$

MORE ON ERRORS

- The approximation errors in numerical differentiation
  - Decrease with decreasing step size h, while
- Roundoff errors increase with decreasing step size
- The least overall approximation occurs for an h that makes the total error $\varepsilon_{approx} + \varepsilon_{ro}$ a minimum, and that a rough guide this occurs when $\varepsilon_{ro} \approx \varepsilon_{approx}$.
- Because differentiation subtracts two numbers close in value, we will assume that the roundoff error for differentiation is machine precision:

$$f' \approx \frac{f(x+h) - f(x)}{h} \approx \frac{\epsilon_m}{h}$$

$$\Rightarrow \qquad \epsilon_{\mathrm{ro}} \approx \frac{\epsilon_m}{h}$$

MORE ON ERRORS (CONTINUED)

- **The approximation error with the forward-difference algorithm is O(h), while that with the central-difference algorithm is O(h^2):**

$$\epsilon_{\text{approx}}^{\text{fd}} \approx \frac{f^{(2)}h}{2}, \qquad \epsilon_{\text{approx}}^{\text{cd}} \approx \frac{f^{(3)}h^2}{24}$$

- **Roundoff and Approximation errors become equal when**

$$\frac{\epsilon_m}{h} \approx \epsilon_{\text{approx}}^{\text{fd}} = \frac{f^{(2)}h}{2} \qquad \frac{\epsilon_m}{h} \approx \epsilon_{\text{approx}}^{\text{cd}} = \frac{f^{(3)}h^2}{24}$$

$$\Rightarrow \quad h_{\text{fd}}^2 = \frac{2\epsilon_m}{f^{(2)}} \qquad \Rightarrow \quad h_{\text{cd}}^3 = \frac{24\epsilon_m}{f^{(3)}}$$

$$\boxed{\begin{aligned} & f' \approx \frac{f(x+h)-f(x)}{h} \approx \frac{\epsilon_m}{h} \\ \Rightarrow \quad & \epsilon_{\text{ro}} \approx \frac{\epsilon_m}{h} \end{aligned}}$$

MORE ON ERRORS (CONTINUED)

- **We take $f' \approx f^{(2)} \approx f^{(3)}$ (which may be crude in general, though not bad for exp or cos x), and assume double precision, $\varepsilon_m \approx 10^{-15}$:**

$$\boxed{\begin{array}{llll} & h_{\rm fd} \;\approx\; 4\times 10^{-8} & & h_{\rm cd} \;\approx\; 3\times 10^{-5} \\ \Rightarrow & \epsilon_{\rm fd} \simeq \frac{\epsilon_m}{h_{\rm cd}} \simeq 3\times 10^{-8}, & \Rightarrow & \epsilon_{\rm cd} \simeq \frac{\epsilon_m}{h_{\rm cd}} \simeq 3\times 10^{-11} \end{array}}$$

- **This may seem backward because the better algorithm leads to a larger h value. It is not. The ability to use a larger h means that the error in the central- difference method is some 1000 times smaller than the error in the forward- difference method here.**

LECTURE 2: SUMMARY

- Integration can be performed in a number of different ways. In each case, the approximation is known to be exact for a limited class of functions.
  - Beware of noise!
- Differentiation formula are derived from Taylor's expression and can be obtained for smaller and smaller error wrt step size.
  - Beware of using a too small h! Error in algorithm and error intrinsic to the machine compete...