UNIVERSITY OF LUXEMBOURG
Physics and Materials Science
Research Unit (PHYMS)

02 – Numerical Differentiation

Almaz Khabibrakhmanov, Mario Galante, Alexandre Tkatchenko

AA 2021/2022 Computational Methods for MSc in Physics

Lesson Outline

Theory:

- Short reminder of basic Calculus
- Numerical recipes to calculates derivatives
- Estimate of errors

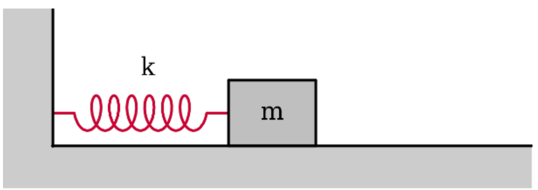
Hands-on session:

- Implementing studied numerical schemes
- Learning basics of coding and plotting in Python
- Analyzing obtained results

Why Should We Care?

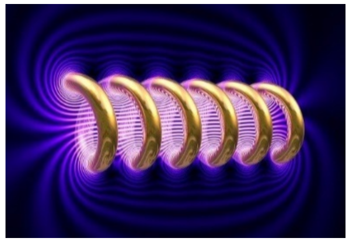
- Differentiation is the basic mathematical operation with a wide range of applications in many areas of Physics

$$\frac{d\boldsymbol{p}}{dt} = \boldsymbol{F}$$



$$\operatorname{div} \boldsymbol{D} = 4\pi\rho$$

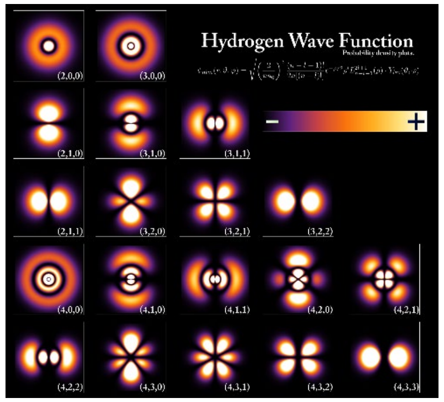
$$\operatorname{div} \boldsymbol{B} = 0$$



$$\operatorname{rot} \boldsymbol{E} = -\frac{1}{c}\frac{\partial \boldsymbol{B}}{\partial t}$$

$$\operatorname{rot} \boldsymbol{H} = \frac{4\pi}{c}\boldsymbol{j} + \frac{1}{c}\frac{\partial \boldsymbol{D}}{\partial t}$$

$$i\hbar\frac{\partial \Psi}{\partial t} = \hat{H}\Psi$$



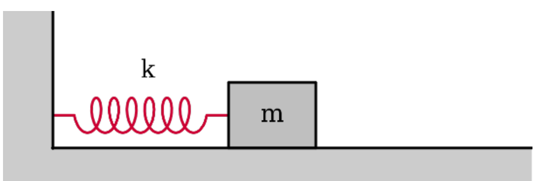
Why Should We Care?

UNIVERSITY OF LUXEMBOURG

- Differentiation is the basic mathematical operation with a wide range of applications in many areas of Physics

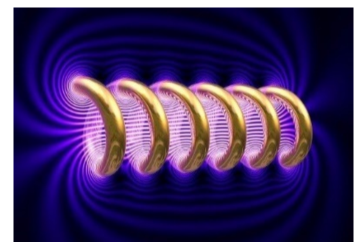
$$i\hbar\frac{\partial\Psi}{\partial t} = \hat{H}\Psi$$

$$\frac{d\boldsymbol{p}}{dt} = \boldsymbol{F}$$



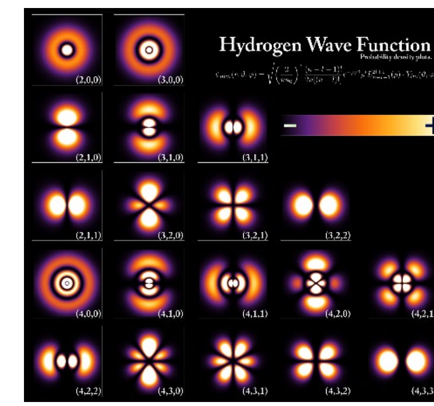
$$\mathrm{div}\,\boldsymbol{D} = 4\pi\rho$$

$$\mathrm{div}\,\boldsymbol{B} = 0$$



$$\mathrm{rot}\,\boldsymbol{E} = -\frac{1}{c}\frac{\partial\boldsymbol{B}}{\partial t}$$

$$\mathrm{rot}\,\boldsymbol{H} = \frac{4\pi}{c}\boldsymbol{j} + \frac{1}{c}\frac{\partial\boldsymbol{D}}{\partial t}$$



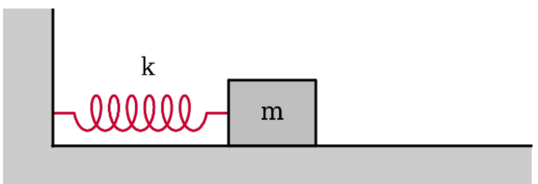
- Derivatives of most of the functions can be computed analytically using rules you studied at school and university
- However, very often in practical applications a function is only known at a few points, and most of the times differential equations cannot be solved analytically.

Why Should We Care?

- Differentiation is the basic mathematical operation with a wide range of applications in many areas of Physics

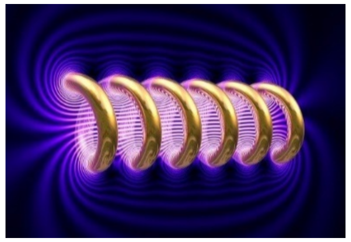
$$i\hbar\frac{\partial\Psi}{\partial t} = \hat{H}\Psi$$

$$\frac{d\boldsymbol{p}}{dt} = \boldsymbol{F}$$



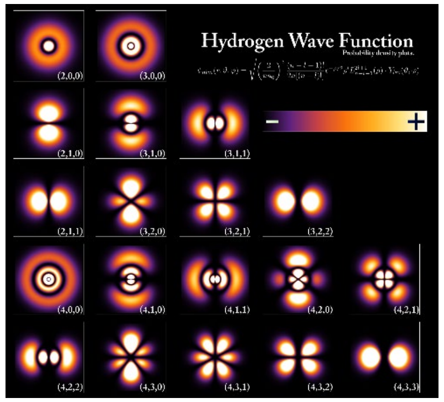
$$\operatorname{div}\boldsymbol{D} = 4\pi\rho$$

$$\operatorname{div}\boldsymbol{B} = 0$$



$$\operatorname{rot}\boldsymbol{E} = -\frac{1}{c}\frac{\partial\boldsymbol{B}}{\partial t}$$

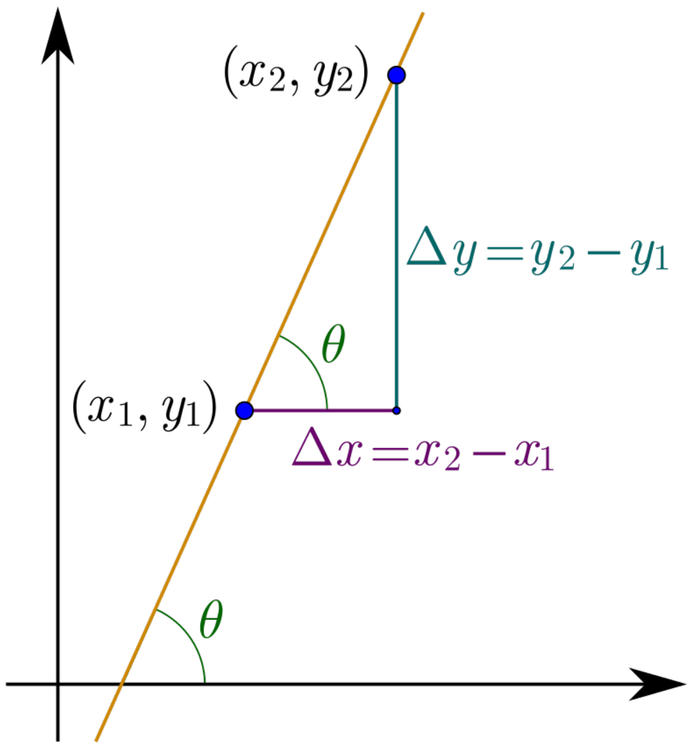
$$\operatorname{rot}\boldsymbol{H} = \frac{4\pi}{c}\boldsymbol{j} + \frac{1}{c}\frac{\partial\boldsymbol{D}}{\partial t}$$



- Derivatives of most of the functions can be computed analytically using rules you studied at school and university
- However, very often in practical applications a function is only known at a few points, and most of the times differential equations cannot be solved analytically.

We have to use approximate numerical methods

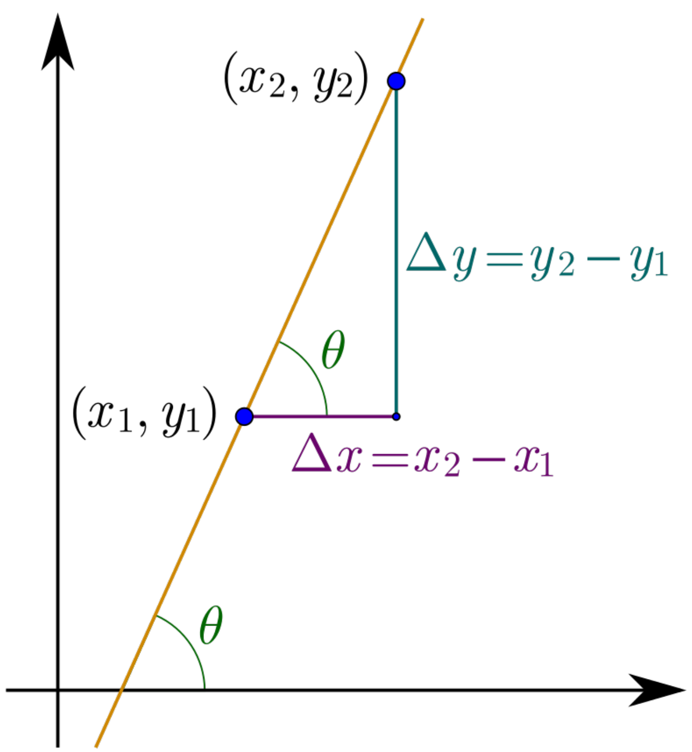
Derivative and Its Meaning



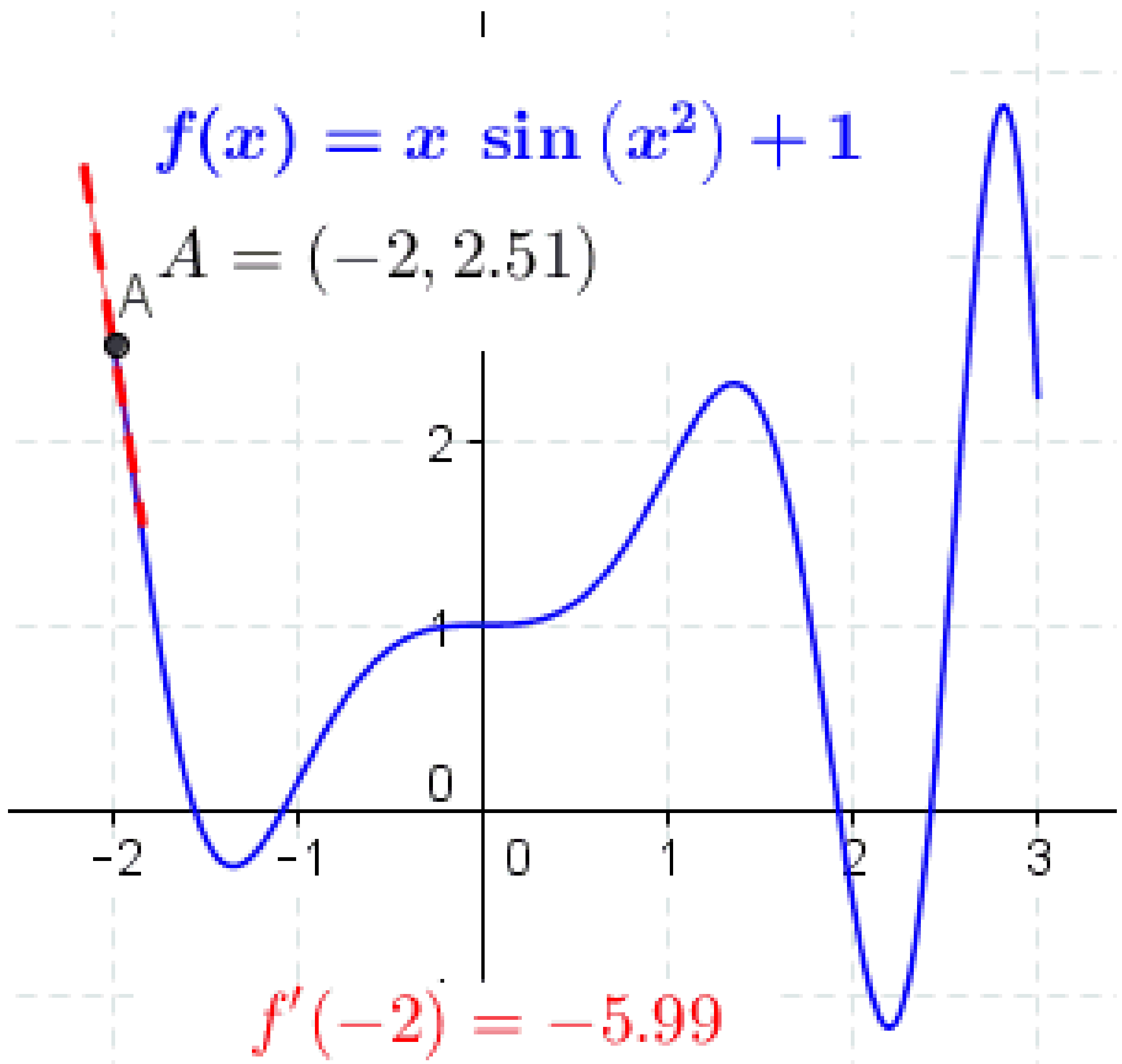
$$y'(x) = \lim_{\Delta x \to 0} \frac{\Delta y}{\Delta x} = \tan\theta$$

Derivative and Its Meaning

The derivative shows the rate of function change at a point



$$y'(x) = \lim_{\Delta x \to 0} \frac{\Delta y}{\Delta x} = \tan\theta$$

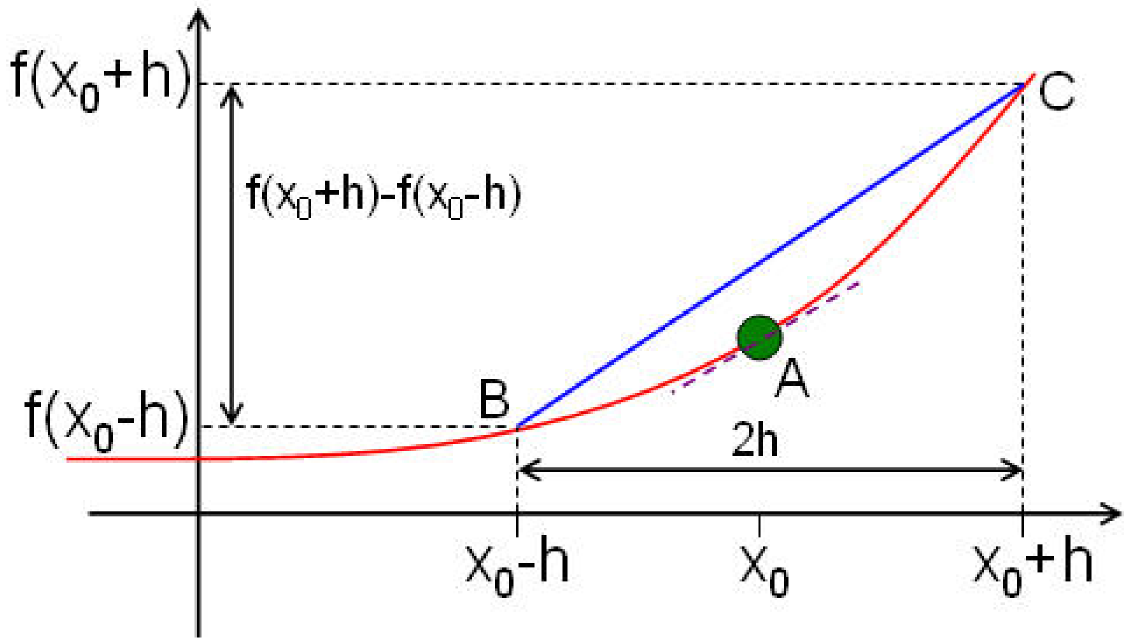


Another interpretation: The derivative is the best LINEAR approximation at a given point of a curve.

Numerical Differentiation

$$f'(x_0) = \lim_{\Delta x \to 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x}$$

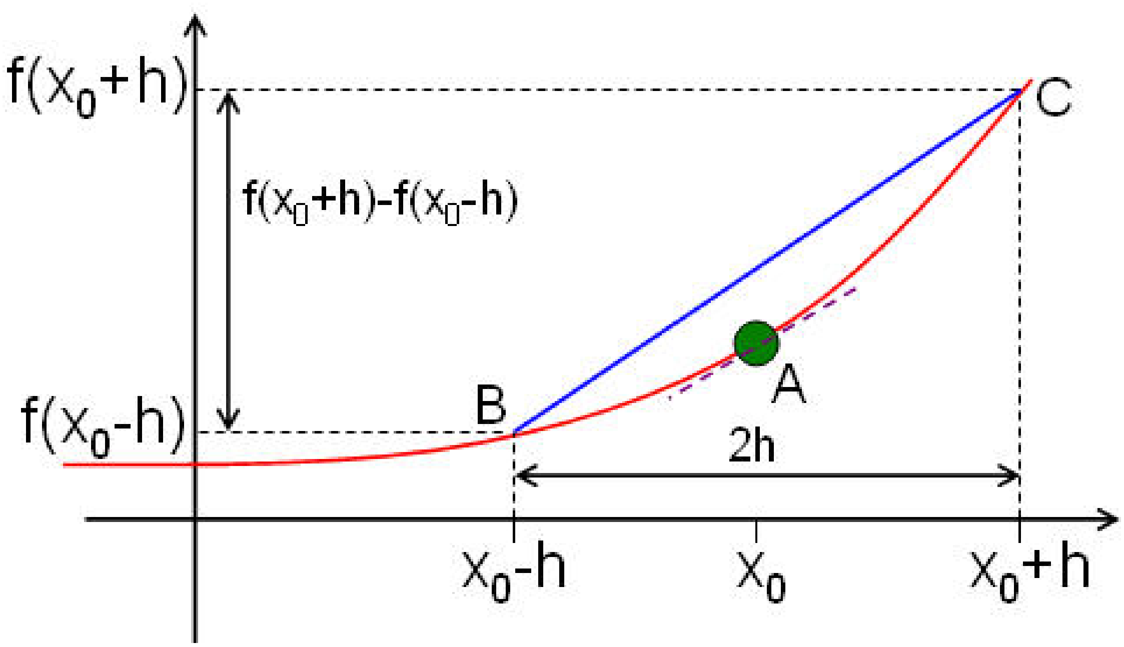
Let's apply some geometrical intuition:



Numerical Differentiation

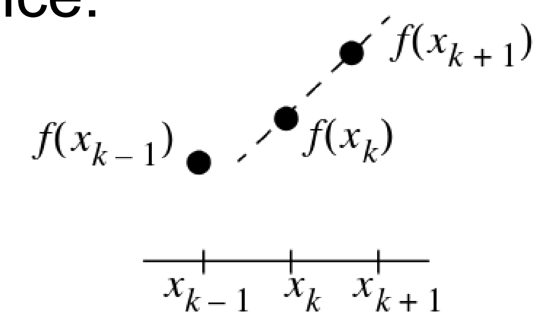
$$f'(x_0) = \lim_{\Delta x \to 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x}$$

Let's apply some geometrical intuition:



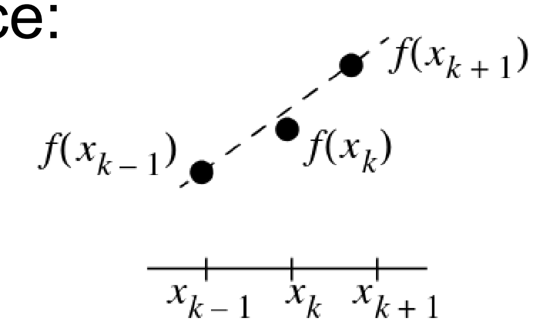
1. Forward difference:

$$f'(x_k) \approx \frac{f(x_{k+1}) - f(x_k)}{x_{k+1} - x_k}$$



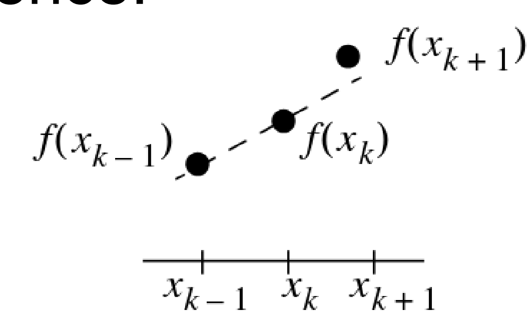
2. Central difference:

$$f'(x_k) \approx \frac{f(x_{k+1}) - f(x_{k-1})}{x_{k+1} - x_{k-1}}$$



3. Backward difference:

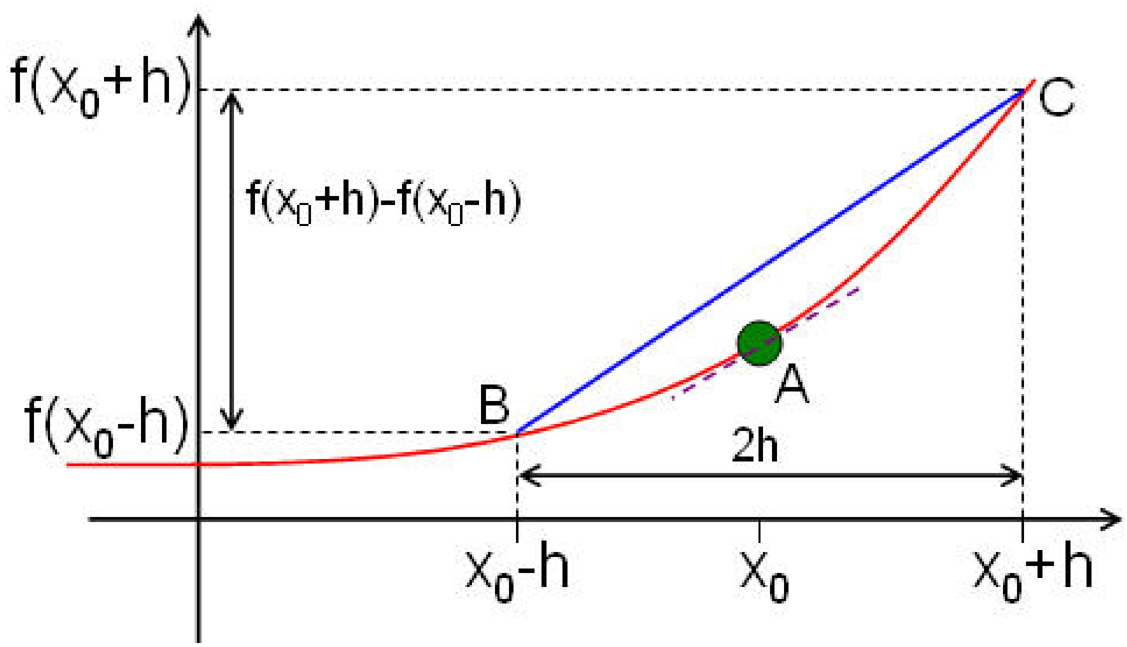
$$f'(x_k) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}$$



Numerical Differentiation

$$f'(x_0) = \lim_{\Delta x \to 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x}$$

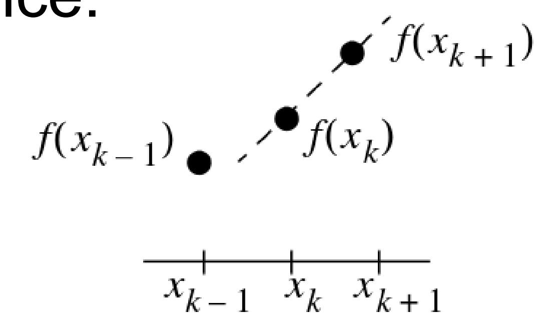
Let's apply some geometrical intuition:



What is the difference between the results obtained via these formulae?

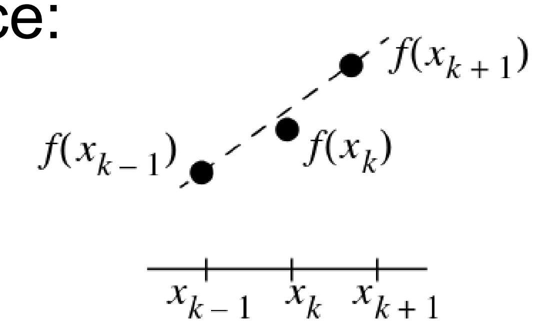
1. Forward difference:

$$f'(x_k) \approx \frac{f(x_{k+1}) - f(x_k)}{x_{k+1} - x_k}$$



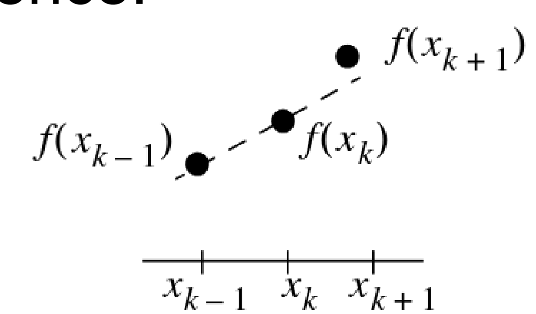
2. Central difference:

$$f'(x_k) \approx \frac{f(x_{k+1}) - f(x_{k-1})}{x_{k+1} - x_{k-1}}$$



3. Backward difference:

$$f'(x_k) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}$$



Taylor Expansion

Any function $f(x)$ which is “smooth enough” in the vicinity of point x_0 can be represented in that vicinity by its **Taylor expansion**:

$$f(x) = f(x_0) + \frac{f'(x_0)}{1!}(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + o\big((x - x_0)^2\big), \qquad x \to x_0 \tag{1}$$

Taylor Expansion

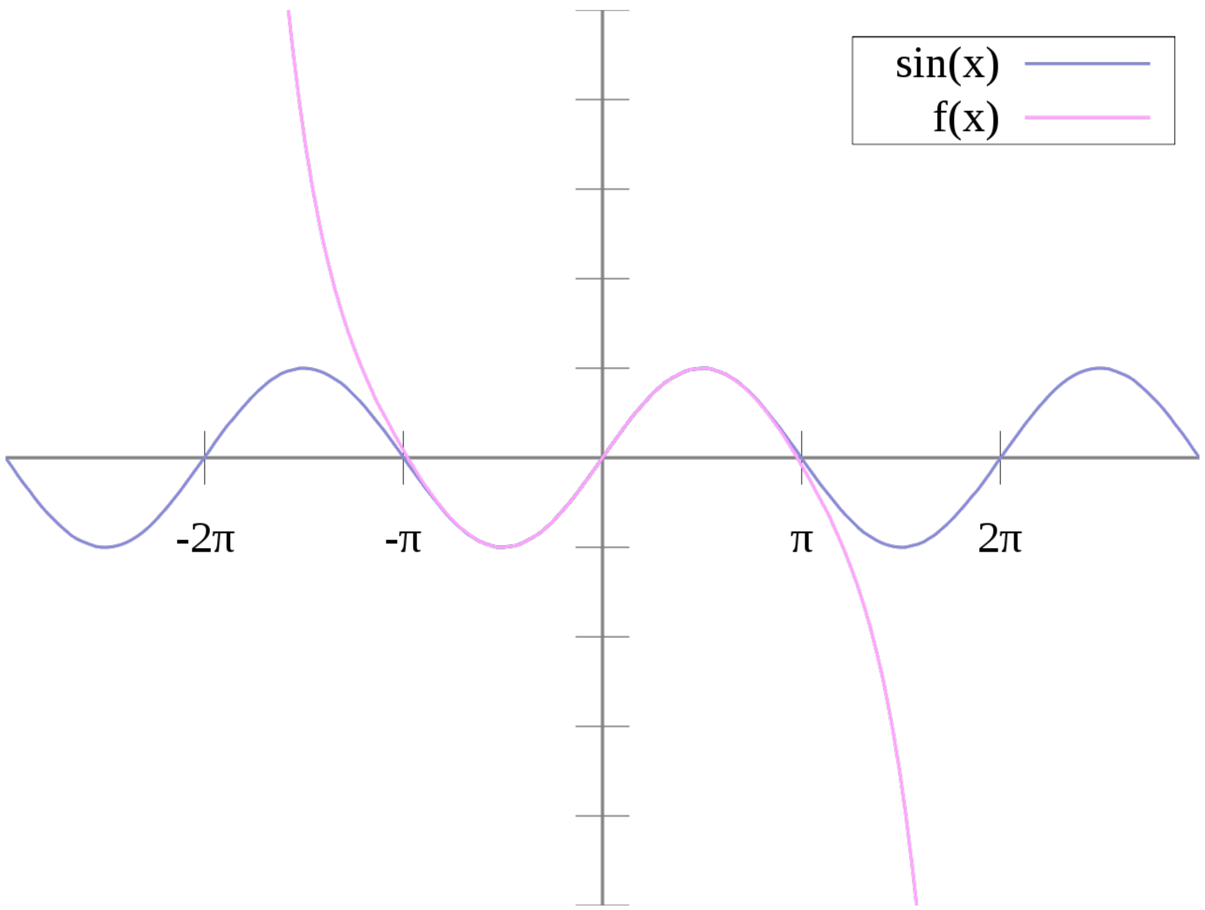
Any function $f(x)$ which is "smooth enough" in the vicinity of point x_0 can be represented in that vicinity by its **Taylor expansion**:

$$f(x) = f(x_0) + \frac{f'(x_0)}{1!}(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + o\big((x - x_0)^2\big), \qquad x \to x_0 \qquad (1)$$

Example:

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + o(x^9), \qquad x \to 0$$

The error of approximation of *sin* by this polynomial is no more than $\frac{|x|^9}{9!}$, i.e. less than 3×10^{-6} for $-1 < x < 1$



Order of Numerical Method

Let us write down Taylor expansions for $f(x_0 + h)$ and $f(x_0 - h)$:

$$f(x_0 + h) = f(x_0) + f'(x_0)h + \frac{1}{2}f''(x_0)h^2 + \frac{1}{6}f'''(x_0)h^3 + \cdots$$

$$f(x_0 - h) = f(x_0) - f'(x_0)h + \frac{1}{2}f''(x_0)h^2 - \frac{1}{6}f'''(x_0)h^3 + \cdots$$

Order of Numerical Method

Let us write down Taylor expansions for $f(x_0+h)$ and $f(x_0-h)$:

$$f(x_0+h) = f(x_0) + f'(x_0)h + \frac{1}{2}f''(x_0)h^2 + \frac{1}{6}f'''(x_0)h^3 + \cdots$$

$$f(x_0-h) = f(x_0) - f'(x_0)h + \frac{1}{2}f''(x_0)h^2 - \frac{1}{6}f'''(x_0)h^3 + \cdots$$

$$f'(x_0) = \begin{cases} \dfrac{f(x_0) - f(x_0-h)}{h} + \dfrac{\boldsymbol{h}}{2}f''(x_0) + \cdots \\ \dfrac{f(x_0+h) - f(x_0-h)}{2h} - \dfrac{\boldsymbol{h^2}}{12}f'''(x_0) + \cdots \\ \dfrac{f(x_0+h) - f(x_0)}{h} - \dfrac{\boldsymbol{h}}{2}f''(x_0) + \cdots \end{cases} = \begin{cases} \dfrac{f(x_0) - f(x_0-h)}{h} + O(\boldsymbol{h}) & \text{forward difference} \\ \dfrac{f(x_0+h) - f(x_0-h)}{2h} + O(\boldsymbol{h^2}) & \text{central difference} \\ \dfrac{f(x_0+h) - f(x_0)}{h} + O(\boldsymbol{h}) & \text{backward difference} \end{cases} \qquad (2)$$

Order of Numerical Method

Let us write down Taylor expansions for $f(x_0+h)$ and $f(x_0-h)$:

$$f(x_0+h) = f(x_0) + f'(x_0)h + \frac{1}{2}f''(x_0)h^2 + \frac{1}{6}f'''(x_0)h^3 + \cdots$$

$$f(x_0-h) = f(x_0) - f'(x_0)h + \frac{1}{2}f''(x_0)h^2 - \frac{1}{6}f'''(x_0)h^3 + \cdots$$

$$f'(x_0) = \begin{cases} \dfrac{f(x_0)-f(x_0-h)}{h} + \dfrac{\boldsymbol{h}}{2}f''(x_0) + \cdots \\ \dfrac{f(x_0+h)-f(x_0-h)}{2h} - \dfrac{\boldsymbol{h^2}}{12}f'''(x_0) + \cdots \\ \dfrac{f(x_0+h)-f(x_0)}{h} - \dfrac{\boldsymbol{h}}{2}f''(x_0) + \cdots \end{cases} = \begin{cases} \dfrac{f(x_0)-f(x_0-h)}{h} + O(\boldsymbol{h}) & \text{forward difference} \\ \dfrac{f(x_0+h)-f(x_0-h)}{2h} + O(\boldsymbol{h^2}) & \text{central difference} \\ \dfrac{f(x_0+h)-f(x_0)}{h} + O(\boldsymbol{h}) & \text{backward difference} \end{cases} \quad (2)$$

The lowest power of *h* in the remainder of Taylor formula determines the *order of the method*

$O(h)$ – method of the 1st order

$O(h^2)$ – method of the 2nd order

Order of Numerical Method

Let us write down Taylor expansions for $f(x_0+h)$ and $f(x_0-h)$:

$$f(x_0+h) = f(x_0) + f'(x_0)h + \frac{1}{2}f''(x_0)h^2 + \frac{1}{6}f'''(x_0)h^3 + \cdots$$

$$f(x_0-h) = f(x_0) - f'(x_0)h + \frac{1}{2}f''(x_0)h^2 - \frac{1}{6}f'''(x_0)h^3 + \cdots$$

$$f'(x_0) = \begin{cases} \dfrac{f(x_0)-f(x_0-h)}{h} + \dfrac{\boldsymbol{h}}{2}f''(x_0) + \cdots \\ \dfrac{f(x_0+h)-f(x_0-h)}{2h} - \dfrac{\boldsymbol{h^2}}{12}f'''(x_0) + \cdots \\ \dfrac{f(x_0+h)-f(x_0)}{h} - \dfrac{\boldsymbol{h}}{2}f''(x_0) + \cdots \end{cases} = \begin{cases} \dfrac{f(x_0)-f(x_0-h)}{h} + O(\boldsymbol{h}) & \text{forward difference} \\ \dfrac{f(x_0+h)-f(x_0-h)}{2h} + O(\boldsymbol{h^2}) & \text{central difference} \\ \dfrac{f(x_0+h)-f(x_0)}{h} + O(\boldsymbol{h}) & \text{backward difference} \end{cases} \quad (2)$$

The lowest power of h in the remainder of Taylor formula determines the *order of the method*

$O(h)$ – method of the 1st order

$O(h^2)$ – method of the 2nd order

Analogously, one can obtain the numerical differentiation scheme of the 4th order:

$$f'(x_0) = \frac{f(x_0-2h) - 8f(x_0-h) + 8f(x_0+h) - f(x_0+2h)}{12h} + O(\boldsymbol{h^4}) \quad (3)$$

Order of Numerical Method

Let us write down Taylor expansions for $f(x_0+h)$ and $f(x_0-h)$:

$$f(x_0+h) = f(x_0) + f'(x_0)h + \frac{1}{2}f''(x_0)h^2 + \frac{1}{6}f'''(x_0)h^3 + \cdots$$

$$f(x_0-h) = f(x_0) - f'(x_0)h + \frac{1}{2}f''(x_0)h^2 - \frac{1}{6}f'''(x_0)h^3 + \cdots$$

$$f'(x_0) = \begin{cases} \dfrac{f(x_0)-f(x_0-h)}{h} + \dfrac{\boldsymbol{h}}{2}f''(x_0) + \cdots \\ \dfrac{f(x_0+h)-f(x_0-h)}{2h} - \dfrac{\boldsymbol{h^2}}{12}f'''(x_0) + \cdots \\ \dfrac{f(x_0+h)-f(x_0)}{h} - \dfrac{\boldsymbol{h}}{2}f''(x_0) + \cdots \end{cases} = \begin{cases} \dfrac{f(x_0)-f(x_0-h)}{h} + O(\boldsymbol{h}) & \text{forward difference} \\ \dfrac{f(x_0+h)-f(x_0-h)}{2h} + O(\boldsymbol{h^2}) & \text{central difference} \\ \dfrac{f(x_0+h)-f(x_0)}{h} + O(\boldsymbol{h}) & \text{backward difference} \end{cases} \quad (2)$$

The lowest power of *h* in the remainder of Taylor formula determines the *order of the method*

$O(h)$ – method of the 1st order

$O(h^2)$ – method of the 2nd order

Analogously, one can obtain the numerical differentiation scheme of the 4th order:

$$f'(x_0) = \frac{f(x_0-2h) - 8f(x_0-h) + 8f(x_0+h) - f(x_0+2h)}{12h} + O(\boldsymbol{h^4}) \quad (3)$$

Or the scheme for second derivative:

$$f''(x_0) = \frac{f(x_0-h) - 2f(x_0) + f(x_0+h)}{h^2} + O(\boldsymbol{h^2}) \quad (4)$$

Truncation Error

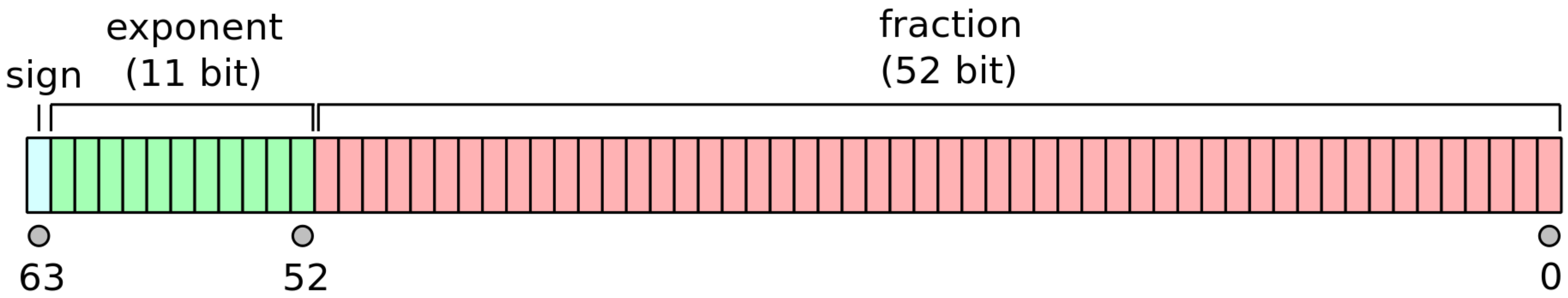
- Choosing the differentiation scheme of order *N* is equivalent to the truncation of Taylor series at (*N*+1)-th term. In fact, we substitute the function by a polynomial as its approximation and then calculate the derivative of this polynomial.

- The error raising due to this truncation is called **truncation error**. This is a *fundamental* mathematical limitation restricting the accuracy of numerical differentiation methods.

- However, this is not the only source of errors, since the numbers in computer memory can be represented only with a *finite* precision. This fact leads to **round-off errors**.

Real Numbers in Computer Memory

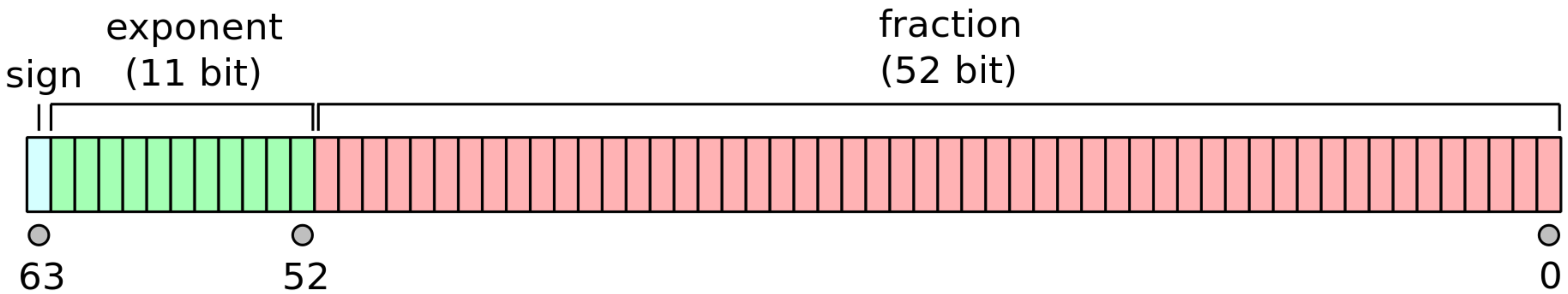
The IEEE Standard for Floating-Point Arithmetic was established in 1985.



Any decimal number a can be represented as: $a = b \times 10^{c}$, e.g. $175 = 1.75 \times 10^{2}$

Real Numbers in Computer Memory

The IEEE Standard for Floating-Point Arithmetic was established in 1985.



Any decimal number a can be represented as: $a = b \times 10^{c}$, e.g. $175 = 1.75 \times 10^{2}$

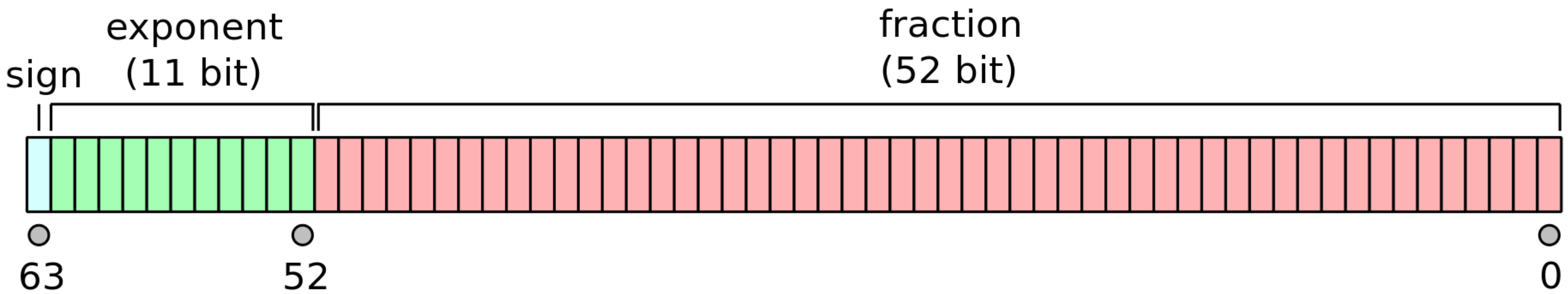
Machine epsilon ε_m is the largest number for which the following equation holds true in machine arithmetic:

$$1 + \varepsilon_m = 1$$

For 64-bit float-point numbers, $\varepsilon_m = 1/2 \cdot 2^{-52} \approx \mathbf{10^{-16}}$ (5)

Real Numbers in Computer Memory

The IEEE Standard for Floating-Point Arithmetic was established in 1985.



Any decimal number a can be represented as: $a = b \times 10^{c}$, e.g. $175 = 1.75 \times 10^{2}$

Machine epsilon ε_m is the largest number for which the following equation holds true in machine arithmetic:

$$1 + \varepsilon_m = 1$$

For 64-bit float-point numbers, $\varepsilon_m = 1/2 \cdot 2^{-52} \approx \mathbf{10^{-16}}$ (5)

Basically, this means that any number smaller than ε_m will be considered as zero by a machine when adding/subtracting it

Error of Numerical Differentiation

Let us denote the real value of function as $f(x_0)$, and the corresponding number in a computer memory as $\overline{f(x_0)}$. Then, we can write:

$$\overline{f(x_0)} = f(x_0)(1+\varepsilon_1), \qquad \overline{f(x_0+h)} = f(x_0+h)(1+\varepsilon_2), \qquad |\varepsilon_{1,2}| \le \varepsilon_m$$

Error of Numerical Differentiation

Let us denote the real value of function as $f(x_0)$, and the corresponding number in a computer memory as $\overline{f(x_0)}$. Then, we can write:

$$\overline{f(x_0)} = f(x_0)(1+\varepsilon_1), \qquad \overline{f(x_0+h)} = f(x_0+h)(1+\varepsilon_2), \qquad |\varepsilon_{1,2}| \le \varepsilon_m$$

Using Taylor formula in the Lagrange form, we also know that:

$$\left| f'(x_0) - \frac{f(x_0+h) - f(x_0)}{h} \right| = \frac{h}{2} |f''(\xi)|$$

Error of Numerical Differentiation

Let us denote the real value of function as $f(x_0)$, and the corresponding number in a computer memory as $\overline{f(x_0)}$. Then, we can write:

$$\overline{f(x_0)} = f(x_0)(1+\varepsilon_1), \qquad \overline{f(x_0+h)} = f(x_0+h)(1+\varepsilon_2), \qquad \left|\varepsilon_{1,2}\right| \le \varepsilon_m$$

Using Taylor formula in the Lagrange form, we also know that:

$$\left| f'(x_0) - \frac{f(x_0+h) - f(x_0)}{h} \right| = \frac{h}{2}\left|f''(\xi)\right|$$

Therefore, for the total error of numerical differentiation $p(h)$ we obtain:

$$p(h) = \left| f'(x_0) - \frac{\overline{f(x_0+h)} - \overline{f(x_0)}}{h} \right| \le \boldsymbol{\frac{h}{2} M_1 + \frac{2\varepsilon_m}{h} M_2}$$

$$M_1 = \max_{x\in[x_0,\, x_0+h]} \left|f''(x)\right|, \qquad M_2 = \max_{x\in[x_0,\, x_0+h]} \left|f(x)\right|$$

Optimal Choice of *h*

Function $p(h)$ describing the error is not monotonous, so it could be minimized:

$$p(h) \approx \frac{h}{2}|f''(x_0)| + \frac{2\varepsilon_m}{h}|f(x_0)| \qquad (6)$$

$$p'(h) \approx \frac{|f''(x_0)|}{2} - \frac{2\varepsilon_m}{h^2}|f(x_0)| = 0$$

Optimal Choice of *h*

Function $p(h)$ describing the error is not monotonous, so it could be minimized:

$$p(h) \approx \frac{h}{2}|f''(x_0)| + \frac{2\varepsilon_m}{h}|f(x_0)| \qquad (6)$$

$$p'(h) \approx \frac{|f''(x_0)|}{2} - \frac{2\varepsilon_m}{h^2}|f(x_0)| = 0$$

Solving the last equation, we obtain the approximate optimal value h^*:

$$h^* \approx 2\frac{\sqrt{\varepsilon_m |f(x_0)|}}{\sqrt{|f''(x_0)|}}$$

Optimal Choice of h

Function $p(h)$ describing the error is not monotonous, so it could be minimized:

$$p(h) \approx \frac{h}{2}|f''(x_0)| + \frac{2\varepsilon_m}{h}|f(x_0)| \qquad (6)$$

$$p'(h) \approx \frac{|f''(x_0)|}{2} - \frac{2\varepsilon_m}{h^2}|f(x_0)| = 0$$

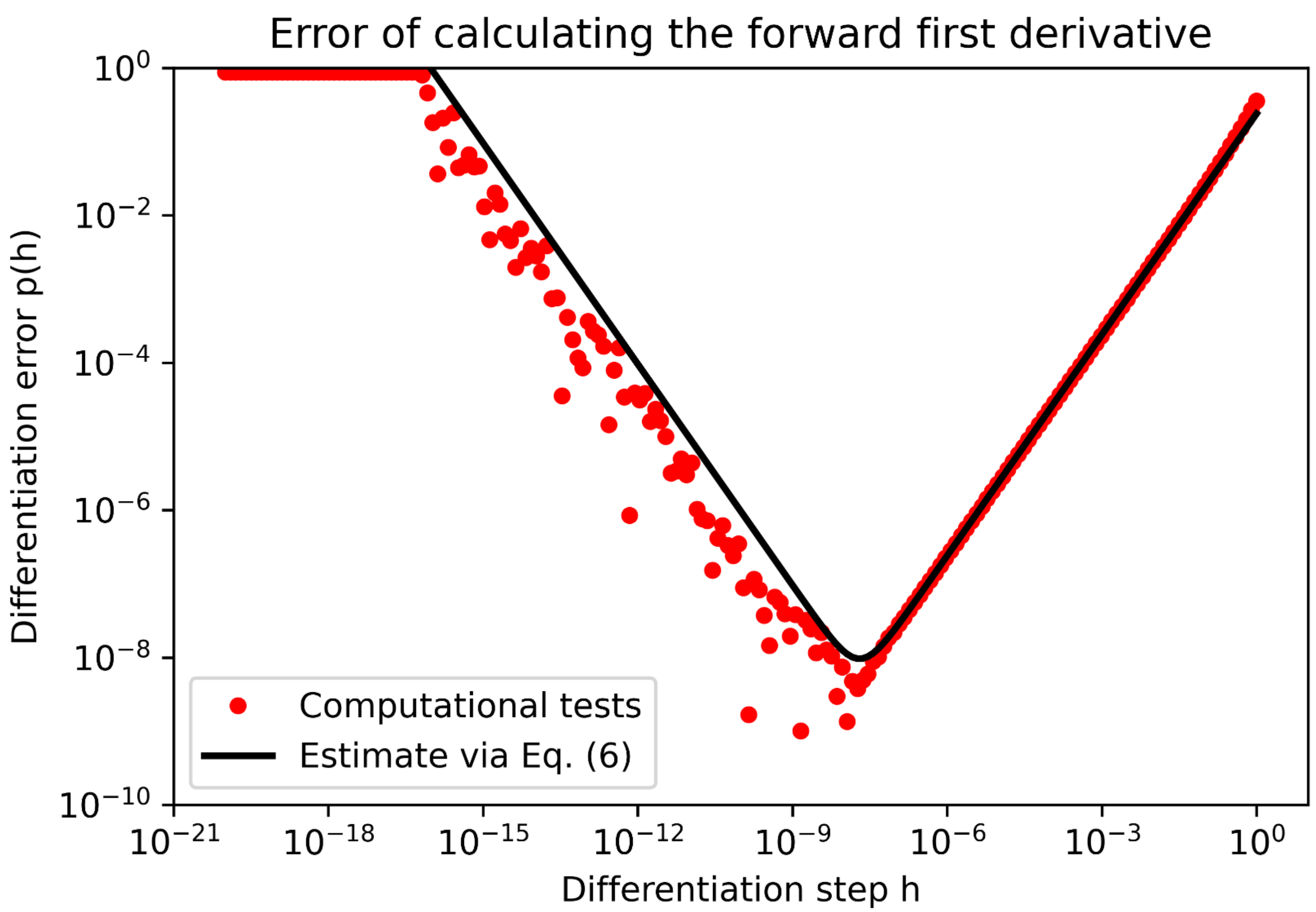
Solving the last equation, we obtain the approximate optimal value h^*:

$$h^* \approx 2\frac{\sqrt{\varepsilon_m|f(x_0)|}}{\sqrt{|f''(x_0)|}}$$

Example:

$f(x) = \sin x$ and $x_0 = 0.5$ gives:

$h^* \approx 2\sqrt{\varepsilon_m} \approx \mathbf{2 \times 10^{-8}}$



Hands-on session

Take-Home Messages

- The idea of numerical differentiation is based on the Taylor formula
- The higher is the order of differentiation method, the quicker is the convergence with changing step *h*
- Numerical differentiation is susceptible to round-off errors
- The higher is the derivative order, the stronger is effect of round-off errors
- Mathematically, numerical differentiation is not a *well-posed problem*, since the total error may increase despite $h \to 0$