Uncertainty Analysis December 2024

Error propagation:

model: $y = f(x_1, x_2, \dots)$

N inputs

$$\underbrace{u_{x_i}}_{\text{absolute uncertainty}} = x_i \cdot \underbrace{e_{x_i}}_{\text{relative uncertainty}}$$

experiment: $x_{ie} = x_{i\,true} + \underbrace{b_{x_i} + \varepsilon_{x_i}}_{u_{x_i}}$

bias

random.

Taylor Series Method: Uncertainty propagation
Gauss error formula.

$$u_y^2 = \sum_{i=1}^{N} \left(\frac{\partial y}{\partial x_i}\right)^2 \underbrace{\left(b_{x_i}^2 + \varepsilon_{x_i}^2\right)}_{\text{overall uncertainty } u_{x_i}^2}$$

$$\boxed{\begin{aligned} u_y^2 &= \sum_{i=1}^{N} \left(u_{x_i} \frac{\partial y}{\partial x_i}\right)^2 \\ e_y^2 &= \sum_{i=1}^{N} \left(\frac{x_i}{y} \cdot \frac{\partial y}{\partial x_i}\right)^2 e_{x_i}^2 \end{aligned}}$$

$e_y = \frac{u_y}{y}$

u_{x_i} is unknown and must be estimated

$$u_x^2 \approx b_x^2 + z_\alpha \sigma_x^2$$

z_α: 1.96 for $\alpha = 95\%$ CI

Monte Carlo Method

model: $y = F(x_1, x_2, \dots)$

model: $y = F(x_1, x_2, \ldots)$

$u_{x_i} = x_i \cdot e_{x_i}$

Assume a distribution for u_{x_i}, typically a Gaussian with standard deviation $\sigma \sim$ absolute uncertainty.

Then sample many realisations and measure output distribution

$\boxed{y = 5\sqrt{x}}$ TSM: $u_y = \frac{5}{2\sqrt{x}} u_x$

Input x: $\mu_x, \sigma_x^2 \Rightarrow \mu_y, \sigma_y^2$

$y = 5\sqrt{x}$

2) Errors in Numerics.

Floating point formats: IEEE 754 standard

$x = s \cdot m \cdot b^e$ ← exponent (11)

floating-point number

sign (1)

mantissa (52)

base $b = 2$

$\Rightarrow$ 64 bit

machine epsilon: relative error of a floating point number

$\varepsilon_m = \frac{u_x}{x}$

smallest number such that

$1 + \varepsilon_m \neq 1$

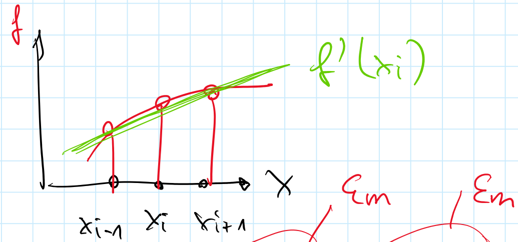
name	Fortran	Python	exponent bits	mantissa bits	total size	ε_m
quadruple	`real*16`		15	112	128	$1.93 \cdot 10^{-34}$

name	Fortran	Python	exponent bits	mantissa bits	total size	ε_m
quadruple	`real*16`		15	112	128	$1.93 \cdot 10^{-34}$
double	`real*8`	`np.float64`	11	52	64	$2.22 \cdot 10^{-15}$
single	`real*4`	`np.float32`	8	23	32	$1.19 \cdot 10^{-7}$
half	`real*3` ?	`np.float16`	5	10	16	$9.77 \cdot 10^{-4}$
bfloat16	`real*2`		8	7	16	$7.8 \cdot 10^{-3}$

Derivatives

error propagation for computations of derivatives

$$\boxed{y = f(x_1, x_2 \ldots)}$$



$$f'_{num}(x_i) = \frac{\overset{\varepsilon_m}{f(x_{i+1})} - \overset{\varepsilon_m}{f(x_{i-1})}}{2\Delta x}$$

error propagation:

$$u_y^2 = \sum \left(u_{x_i} \frac{\partial y}{\partial x_i}\right)^2$$

$$\boxed{\varepsilon_y^2 = \sum \left(\frac{x_i}{y} \frac{\partial y}{\partial x_i}\right)^2 \varepsilon_{x_i}^2}$$

Example: $y = x_1 + x_2$ $\qquad \frac{\partial y}{\partial x_1} = 1 = \frac{\partial y}{\partial x_2}$

$$\Rightarrow \varepsilon_y^2 = \underbrace{\frac{x_1^2 + x_2^2}{(x_1 + x_2)^2}}_{x_1 = x_2 = 1 \Rightarrow \frac{1}{2}} \cdot \varepsilon_m^2$$

Problem: $\boxed{x_1 \approx -x_2}$

$$x_1 + x_2 \approx 0$$
$$x_1^2 + x_2^2 \gg 0$$

large amplification of the relative error possible!

"cancellation"

- adding up many numbers.

- adding up many numbers.

$$\mu = \frac{1}{N} \sum x_i$$

sort the numbers x_i first

- computation of derivatives

$$\underbrace{f' = \frac{f(x+\Delta x) - f(x-\Delta x)}{2\Delta x}}_{\text{(Taylor series expansion)}} + \underbrace{\frac{\Delta x^2}{6} f''(x) + \mathcal{O}(\Delta x^4)}_{\text{truncation error}}$$

Finite differences 2nd order.

$$y = x_1 - x_2 \quad , \quad x_1 = \frac{f(x+\Delta x)}{2\Delta x} \qquad x_2 = \frac{f(x-\Delta x)}{2\Delta x}$$

$$\frac{\partial y}{\partial x_1} = 1 \quad , \quad \frac{\partial y}{\partial x_2} = -1$$

$$\varepsilon_{x_1} = \varepsilon_{x_2} = \varepsilon_m$$

(machine epsilon $\varepsilon_m \approx 2 \cdot 10^{-15}$)

$$\varepsilon_y^2 = \left[\frac{x_1}{y} \cdot \varepsilon_{x_1} \right]^2 + \left[\frac{x_2}{y} \varepsilon_{x_2} \right]^2$$

$$\varepsilon_y^2 \approx \left(\frac{f(x+\Delta x)}{2\Delta x} \right)^2 \frac{\varepsilon_m^2}{|f'|^2} + \left(\frac{f(x-\Delta x)}{2\Delta x} \right)^2 \frac{\varepsilon_m^2}{|f'|^2}$$

$$\approx 2 \cdot \frac{\varepsilon_m^2}{|f'|^2} \left(\frac{f(x)}{2\Delta x} \right)^2$$

$$\boxed{\varepsilon_y^2 \approx \left(\frac{f(x) \cdot \lambda}{|f'(x)| \Delta x} \right)^2 \cdot \varepsilon_m^2}$$

$\lambda = \frac{\sqrt{2}}{2}$ second order

$\lambda = \sqrt{2}$ first order

propagation error.

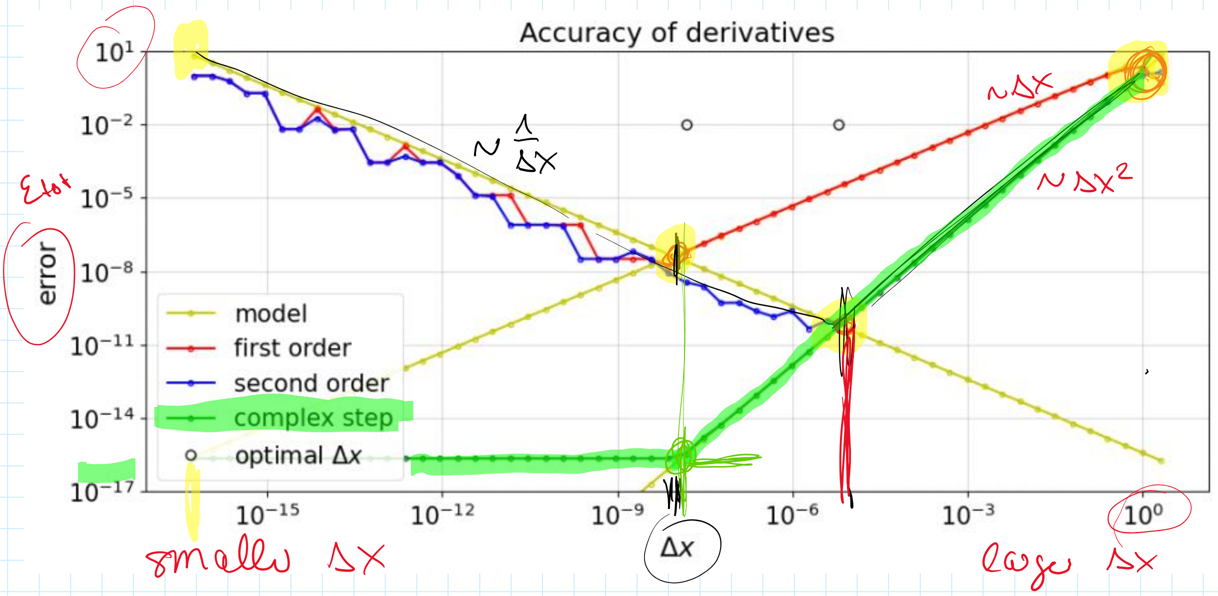
$$\varepsilon_d^2 = \left(\frac{\Delta x^2 f''(x)}{6 f'(x)} \right)^2$$

discretisation / truncation error.

$\varepsilon_{tot}^2 = \varepsilon_g^2 + \varepsilon_d^2$ total error.

- truncation error goes down as Δx^2
  as expected from a second order scheme.

- propagation error goes up with Δx^{-1}



propagation

truncation error

- optimal Δx? Simple geometric arguments

$\Delta x_{opt.} \approx \varepsilon_m^{1/2}$ for first-order scheme
$\approx 10^{-8}$

$\Delta x_{opt} \approx \varepsilon_m^{1/3}$ for second-order scheme
$\approx 10^{-6}$

Recommendation Newton-Krylov methods.

- complex step method:

$f(x+\Delta x) = \ldots$ FD methods

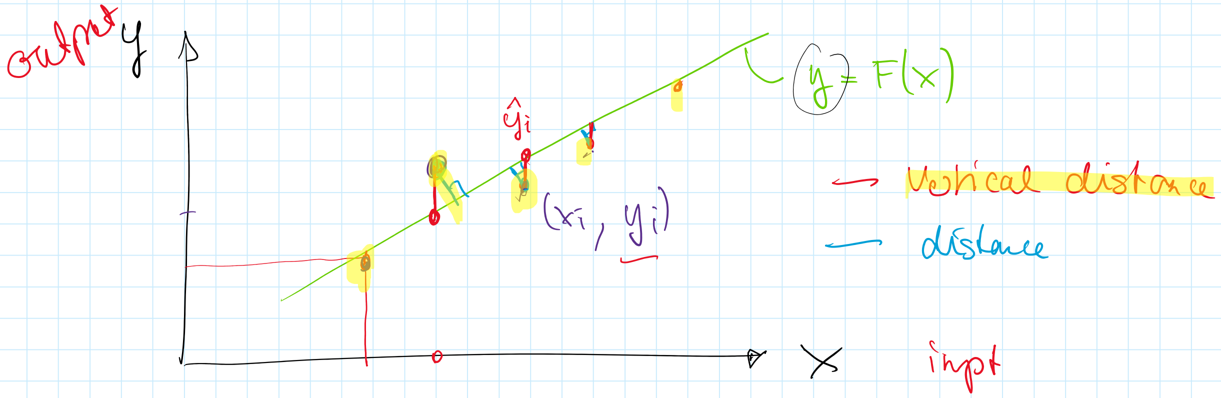
$$f(x+i\Delta x) = f(x) + i\Delta x\, f'(x) - \frac{\Delta x^2}{2} f''(x) + O(\Delta x^3)$$

$$f'(x)_{num} = \mathrm{Im}\left\{\frac{f(x+i\Delta x)}{\Delta x}\right\}$$

propagation error Em independent of Δx.

best method for validating numerical methods.

3. Regression



- being able to interpolate (data-driven method)
- identifying physical relation. between x and y.

calibration

⇒ Regression error

model $y = F(x)$

Parametrize $F(x)$: $\hat{y} = a_0 + a_1 \cdot x$

estimate   intercept c   slope m

linear regression

Minimize the vertical distance

distance: $H(a_0, a_1) = \sum_{i=1}^{N} \left(\underbrace{a_0 + a_1 x_i}_{\hat{y}_i} - y_i\right)^2$

∂H ∂H

$$\hat{y}_i$$

optimise: $\frac{\partial H}{\partial a_0} = 0 \ , \quad \frac{\partial H}{\partial a_1} = 0$

$$\frac{\partial H}{\partial a_0} = 2 \sum \left(a_0 + a_1 x_i - y_i \right) = 0$$

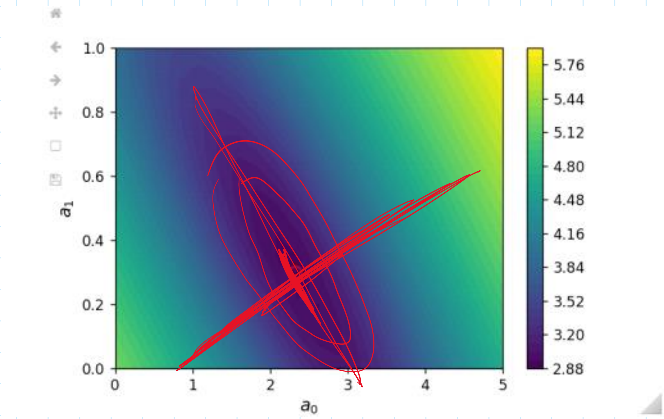
$$\frac{\partial H}{\partial a_1} = 2 \sum \left(a_0 x_i + a_1 x_i^2 - x_i y_i \right) = 0$$

$$a_0 = \frac{\sum x_i^2 \sum y_i - \sum x_i \sum x_i y_i}{N \sum x_i^2 - \left(\sum x_i\right)^2} \qquad a_1 = \frac{N \sum x_i y_i - \sum x_i \sum y_i}{N \sum x_i^2 - \left(\sum x_i\right)^2}$$

classical least square fit.

linear regression: $\boxed{y = A \cdot p}$

$$y = \begin{bmatrix} \vdots \\ y_i \\ \vdots \end{bmatrix}, \quad A = \begin{bmatrix} \vdots & \vdots \\ 1 & x_i \\ \vdots & \vdots \end{bmatrix}, \quad p = \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} c \\ m \end{bmatrix}$$



Generalisation: $\hat{y} = \sum_{i=1}^{N} \left[\sum_{j=0}^{k-1} a_j f_j(x_i) \right]^2$

$\underbrace{1, x, x^2, \sin(x)}_{\text{independent of } a_j}$

coefficient $a_0, a_1, \ldots$

Linear regression.

$\boxed{y = A \cdot p}$ $\quad [\ \vdots\ ]$

$$\boxed{y = A \cdot p}$$

$\begin{bmatrix} \vdots \\ y_i \\ \vdots \end{bmatrix}$

$p = \begin{bmatrix} \vdots \\ a_j \\ \vdots \end{bmatrix}$ k coefficients

$\overset{\longleftarrow k \longrightarrow}{\begin{bmatrix} \vdots & \vdots & \vdots \\ f_0(x_i) & f_1(x_i) & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix}}$ N data pairs.

non-square (rectangular)

$\Rightarrow$ overdetermined system.

$$p = \text{"}A^{-1}\text{"} \cdot y$$

pseudo inverse

$$\boxed{A^+ = (A^T A)^{-1} \cdot A^T}$$

$$Ax = b$$

$$b - Ax = r$$

minimize residual

$$r^T r = (b - Ax)^T (b - Ax)$$

$$= \underbrace{x^T A^T A x}_{} - \underbrace{x^T A^T b - b^T A x}_{\text{scalars}} + \underbrace{b^T b}_{\text{irrelevant, because independent of } x}$$

$$= 2 \underbrace{\left(\frac{1}{2} x^T \underbrace{A^T A}_{\text{"}A\text{"}} x - x^T \underbrace{A^T b}_{\text{"}b\text{"}} \right)}_{} + \underbrace{b^T b}_{\text{does not matter}}$$

Optimize a scalar square function:

$$f(x) = \frac{1}{2} a x^2 - b x$$

$$f'(x) = ax - b = 0 \Rightarrow ax = b \Rightarrow \underline{x = a^{-1} b}$$

with Matrices: $f(x) = \underbrace{\frac{1}{2} x^T A x - x^T b}_{}$

$$Ax = b \Rightarrow x = A^{-1} b$$

$$\boxed{x = \underbrace{(A^T A)^{-1} A^T}_{A^+} b}$$

We can now solve the overdetermined system

$$y = A\,p$$

using the formal pseudo-inverse $p = A^{+} y$

$$A^{+} = (A^{T}A)^{-1}A^{T}.$$

Error analysis

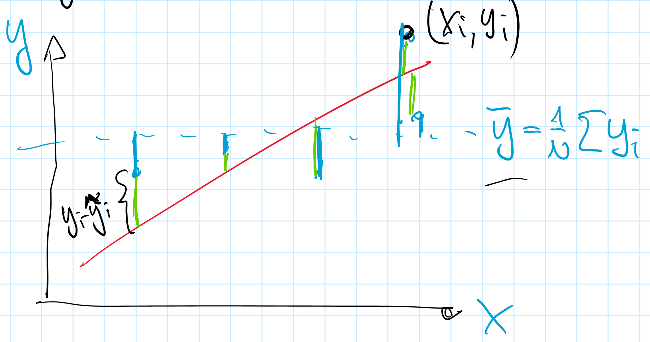
- regression error:

standard deviation

residual sum squared error SSres

$$s_y^2 = \frac{\sum (y_i - m x_i - c)^2}{N-2}$$

remove k=2 degrees of freedom.



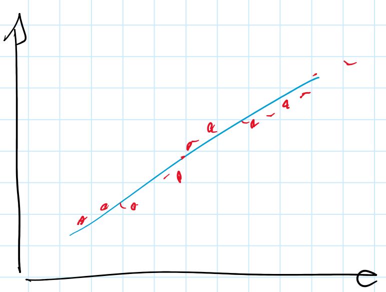
- How good is my regression:

$$SS_{res} = \sum (y_i - \hat{y}_i)^2$$

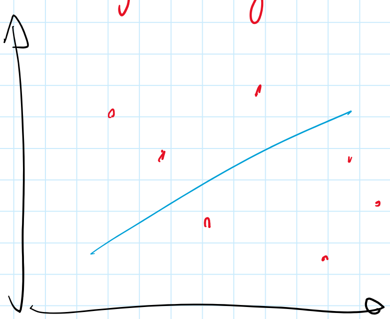
$$SS_{tot} = \sum (y_i - \bar{y})^2$$

$$\boxed{R^2 = 1 - \frac{SS_{res}}{SS_{tot}}}$$ Coefficient of determination

$0 \leq R^2 \leq 1$ Proportion of the total variation explained by the regression.



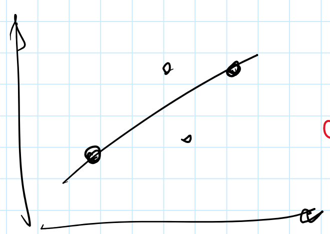
high R^2



low R^2

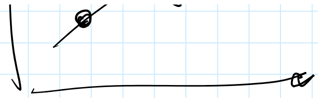
degrees of freedom

dof:



dof = 1

dof:



$$\text{dof} = n - k - 1 = 4 - 2 - 1$$

variables

points

adjusted R^2 : $\bar{R}^2 = 1 - (1-R^2)\frac{N-1}{N-k-1}$

Sensitivity of a_i:

$u_{a_j}^2$: — TSM

$$a_j = a_j\left(x_1, x_2, \ldots, x_N, y_1, y_2, \ldots, y_N\right).$$

$$u_{a_j}^2 = \sum_{i=1}^{N}\left[\left(\frac{\partial a_j}{\partial x_i}\right)^2 u_{x_i}^2 + \left(\frac{\partial a_j}{\partial y_i}\right)^2 u_{y_i}^2\right] + 2\sum_{i=1}^{N-1}\sum_{k=i+1}^{N}\left[\left(\frac{\partial a_j}{\partial x_i}\right)\left(\frac{\partial a_j}{\partial x_k}\right) u_{x_i x_k} + \left(\frac{\partial a_j}{\partial y_i}\right)\left(\frac{\partial a_j}{\partial y_k}\right) u_{y_i y_k}\right] + 2\sum_{i=1}^{N}\sum_{k=i}^{N}\left(\frac{\partial a_j}{\partial x_i}\right)\left(\frac{\partial a_j}{\partial y_k}\right) u_{x_i y_k} \quad (6.10)$$

MCM: perturb x_j, y_j using estimated uncertainty.