Error Propagation

In general a problem has

- several **input values** $x_1, \ldots, x_n$
- several **output values** $y_1, \ldots, y_m$ which depend on the input values: $y_1 = f_1(x_1, \ldots, x_n), \ldots, y_n = f_n(x_1, \ldots, x_n)$

We consider the simplest case: one input value x and one output value

$$\boxed{y = f(x)}$$

Error propagation

For the exact input value x we get the exact output

$$y := f(x)$$

But in practice we have an inexact input value $\hat{x}$ due to measurement error and rounding error: we have a relative error

$$\varepsilon_{\hat{x}} := \frac{\hat{x} - x}{x}$$

Then the best we can do is to compute

$$\hat{y} := f(\hat{x})$$

Let us (unrealistically) assume that we can find this $\hat{y}$ exactly without further errors. This gives a relative error

$$\varepsilon_{\hat{y}} = \frac{\hat{y} - y}{y}$$

This is called "**error propagation**". If we have e.g. $|\varepsilon_{\hat{x}}| \approx 10^{-5}$ can we conclude that the corresponding error $|\varepsilon_{\hat{y}}|$ in the output has the same order of magnitude? Can it be much larger?

Example 1: Consider the function $\boxed{f(x) = \frac{1}{x}}$ at $x = 2$. For the approximate value $\hat{x} = 1.96$ we obtain

$$x = 2 \qquad \hat{x} = 1.96 \qquad \varepsilon_{\hat{x}} = \frac{\hat{x} - x}{x} = \boxed{-.02}$$

$$y = f(x) = \frac{1}{2} = .5 \qquad \hat{y} = f(\hat{x}) = \frac{1}{1.96} \approx .5102 \qquad \varepsilon_{\hat{y}} = \frac{\hat{y} - y}{y} \approx \boxed{.0204}$$

Here we seem to have

$$\varepsilon_{\hat{y}} \approx -\varepsilon_{\hat{x}}$$

For the function $\boxed{f(x) = x^\alpha}$ we obtain the condition number $c_f(x) = \frac{x \cdot \alpha x^{\alpha-1}}{x^\alpha} = \alpha$. This is therefore well conditioned unless $|\alpha|$ is huge.

Example 2: The function $\boxed{f(x) = \ln x}$ has the condition number $c_f(x) = \frac{x \cdot \frac{1}{x}}{\ln x} = \frac{1}{\ln x}$. For $x = 1.01$ the function is ill conditioned: we obtain the condition number

$$c_f(x) = \frac{1}{\ln x} \approx \frac{1}{1 - x} = \frac{1}{.01} = 100$$

using the Taylor approximation $\ln x \approx 0 + 1 \cdot (x-1)$ for x close to 1. E.g., for $x = 1.01$ and $\hat{x} = 1.02$ we obtain $\varepsilon_{\hat{y}} \approx 100\varepsilon_{\hat{x}}$:

$$x = 1.01 \qquad \hat{x} = 1.02 \qquad \varepsilon_{\hat{x}} = \frac{\hat{x}-x}{x} \approx \boxed{.0099}$$

$$y = f(x) \approx .00995 \qquad \hat{y} = f(\hat{x}) \approx .0198 \qquad \varepsilon_{\hat{y}} = \frac{\hat{y}-y}{y} \approx \boxed{.99}$$

Note that $|\varepsilon_{\hat{y}}| \gg |\varepsilon_{\hat{x}}|$. We seem to have

$$\varepsilon_{\hat{y}} \approx 100\,\varepsilon_{\hat{x}}$$

Note that x and $\hat{x}$ are close to a zero of $\ln x$. Since $\ln x \approx x - 1$ for x close to 1 we have $y \approx .01$ and $\hat{y} \approx .02$ which corresponds to a relative error of 1 whereas $\frac{\hat{x}-x}{x} \approx .01$.

Claim: for small error $|\varepsilon_{\hat{x}}|$ in the input we will get an error in the output

$$\varepsilon_{\hat{y}} \approx c \cdot \varepsilon_{\hat{x}}$$

with a "magnification factor" c. This factor depends on the function f and the value x and is called the **condition number** $\boxed{c_f(x)}$.

How to find the condition number $c_f(x)$

We have

$$\varepsilon_{\hat{y}} := \frac{\hat{y}-y}{y} = \frac{f(\hat{x})-f(x)}{f(x)} = \frac{f(\hat{x})-f(x)}{\hat{x}-x} \cdot \frac{x}{f(x)} \cdot \frac{\hat{x}-x}{x} \approx \underbrace{\left(f'(x)\frac{x}{f(x)}\right)}_{c_f(x)} \cdot \frac{\hat{x}-x}{x}$$

If $\tilde{x}$ is close to x we have $\frac{f(\tilde{x})-f(x)}{\tilde{x}-x} \approx f'(x)$.

RESULT: For $\hat{x}$ close to x we obtain

$$\boxed{\varepsilon_{\hat{y}} \approx c_f(x)\varepsilon_{\hat{x}}} \quad \text{with the } \textbf{condition number} \quad \boxed{c_f(x) := \frac{xf'(x)}{f(x)}}$$

is called the **condition number** of the function f at x .

The condition number determines how sensitive a problem is to small perturbations of input values.

If $|c_f|$ is not much larger than 1 we call the problem **well-conditioned**, in the case of $|c_f| \gg 1$ we call the problem **ill-conditioned**.

Example 1: Consider the function $\boxed{f(x) = \frac{1}{x}}$ has the condition number $c_f(x) = \frac{x \cdot (-x^{-2})}{x^{-1}} = -1$. Therefore we have

$$\varepsilon_{\hat{y}} \approx -\varepsilon_{\hat{x}}$$

and the problem is **well conditioned**.

Example 1': For the function $\boxed{f(x) = x^{\alpha}}$ with $\alpha \in \mathbb{R}$ we obtain the condition number $c_f(x) = \frac{x \cdot \alpha x^{\alpha-1}}{x^{\alpha}} = \alpha$.

This is **well conditioned** unless $|\alpha| \gg 1$.

Example 2: The function $\boxed{f(x) = \ln x}$ has the condition number

$$c_f(x) = \frac{x \cdot \frac{1}{x}}{\ln x} = \frac{1}{\ln x}$$

For $x = 1.01$ we obtain the condition number

$$c_f(x) = \frac{1}{\ln x} \approx \frac{1}{1-x} = \frac{1}{.01} = 100$$

using the Taylor approximation $\ln x \approx 0 + 1 \cdot (x-1)$ for x close to 1. Therefore we have

$$\varepsilon_{\hat{y}} \approx 100\, \varepsilon_{\hat{x}}$$

and the problem is **ill conditioned**.

Unavoidable error

We want to solve the problem $y = f(x)$

- on a computer with machine epsilon ε_M
- we have measurement error $\left|\frac{\tilde{x}-x}{x}\right| \leq \varepsilon_{\text{meas}}$ for the input value
- our program will output a machine number $\hat{y}$

QUESTION: What amount of error $|\varepsilon_{\hat{y}}| = \left|\frac{\hat{y}-y}{y}\right|$ can we expect for a "good algorithm"?

We start with the exact input value x. We obtain a measured value $\tilde{x}$ with the **measurement error**

$$\left|\frac{\tilde{x}-x}{x}\right| \leq \varepsilon_{\text{meas}}$$

The computer will use instead of $\tilde{x}$ the closest machine number $\hat{x} := fl(\tilde{x})$ with the **rounding error**

$$\left|\frac{\hat{x}-\tilde{x}}{\tilde{x}}\right| \leq \varepsilon_M \quad \text{"machine epsilon"}$$

Hence we have a combined relative error

$$|\varepsilon_{\hat{x}}| = \left|\frac{\hat{x}-x}{x}\right| \underset{\approx}{\leq} \varepsilon_{\text{meas}} + \varepsilon_M$$

The **ideal algorithm** would use this value $\hat{x}$ and then compute

$$\tilde{y} := f(\hat{x})$$

exactly (or with e.g. 100 digits accuracy). Due to error propagation we have

$$\varepsilon_{\tilde{y}} \approx c_f(x)\varepsilon_{\tilde{x}} = c_f(x)\,(\varepsilon_{\text{meas}} + \varepsilon_M) \quad \text{with the condition number } c_f(x)$$

But the computer has to represent the output value as a machine number, so we use the closest machine number

$$\hat{y} := fl(\tilde{y})$$

where we have the **rounding error**

$$\left|\frac{\hat{y}-y}{y}\right| \leq \varepsilon_M$$

Therefore we have the combined error

$$|\varepsilon_{\hat{y}}| \leq |\varepsilon_{\tilde{y}}| + \varepsilon_M = \boxed{c_f(x)\,(\varepsilon_{\text{meas}} + \varepsilon_M) + \varepsilon_M} \quad \text{"}\textbf{unavoidable error}\text{"}$$

RESULT: Consider the case **without a measurement error**:

- the "ideal algorithm" we give an error $|\varepsilon_{\hat{y}}|$ as large as the **unavoidable error** $\boxed{c_f(x)\varepsilon_M + \varepsilon_M}$

Our actual program produces an output value $\hat{y}$. Let us pretend that we know the error

$$|\varepsilon_{\hat{y}}| = \left|\frac{\hat{y}-y}{y}\right|$$

- if we have $\boxed{|\varepsilon_{\hat{y}}| \gg \text{unavoidable error}}$ our algorithm is "suboptimal". In this case we call the algorithm "**numerically unstable**".
- otherwise our algorithm is (essentially) "optimal". In this case we call the algorithm "**numerically stable**".

Usually we don't know the exact value y, so we don't know $\varepsilon_{\hat{y}}$.

If our computation uses single precision machine numbers we can proceed as follows:

- run the algorithm in single precision, this gives a computed result $\hat{y}$
- run the algorithm in double precision, this gives a computed result $\hat{y}_d$
- approximate the error using

$$\frac{\hat{y}-y}{y} \approx \frac{\hat{y}-\hat{y}_d}{\hat{y}_d}$$