# Recap of Last lecture

(0)

→ Floating pt. representation of a no.
Scientific notation

→ Numerical approx$^n$ of a no. (b/c of finite memory of a m/c)

Chopping ↙ ↘ Rounding

→ Error → Absolute error
→ Relative error

→ Significant digits (rules)

# Error comparison w/ different numerical strategies.

* In the following example, we will demonstrate that different numerical strategies give different errors.

Q) Let us say we want to compute $y = (a^2 - b^2)$ using 4-digit arithmetic. $a = 0.3237$; $b = 0.3134$

**Ans:-**

**Algo (1)**

$n_1 = a \times a$

$n_2 = b \times b$

$y = n_1 - n_2$

Nxt. pg.

$1.048 \times 10^{-1}$
$-0.9822$
$= 0.6580 \times 10^{-2}$

**Algo (2)**

$n_1 = a + b$

$n_2 = a - b$

$y = n_1 \times n_2$

## Algo (1)

$$n_1 = 0.3237 \times 0.3237$$
$$= 0.1048 \quad \text{(rounding)}$$

$$n_2 = 0.3134 \times 0.3134$$
$$= 0.09822$$
$$= 0.9822 \times 10^{-1} \quad \text{(rounding)}$$

$$y^* = n_1 - n_2$$
$$= 0.6580 \times 10^{-2}$$

## Algo (2)

$$n_1 = a + b = 0.6371$$

$$n_2 = a - b = 0.103 \times 10^{-1}$$

$$y^* = n_1 \times n_2$$
$$= 0.6562 \times 10^{-2}$$

Algo (2) is <u>more</u> accurate

Exact value: $a^2 - b^2 = 0.656213 \times 10^{-2} = y$

**Relative error:-**

Algo (1): $\dfrac{|y - y^*|}{|y|} = 0.002723$
$$= 0.2723 \times 10^{-2}$$

Algo (2): $\dfrac{|y - y^*|}{|y|}$
$$= 0.1981 \times 10^{-4}$$

# Reading assignment !

Q) In the previous example, is algorithm 2 always more accurate irrespective of the values of a and b ?

**Hint** :- Try when a, b is s.t. the following is **NOT** true

$$\frac{1}{3} < \left|\frac{a}{b}\right|^2 < 3 \quad \text{——— (i)}$$

* Why is this the case? Hint: Equivalence of (i) and $3|a^2-b^2| \leq a^2+b^2+|a^2-b^2|$

# Some more remarks on "error analysis"

## Forward error vs Backward error.

* Suppose we wish to find $x$ s.t. $f(x) = 0$ where $f : \mathbb{R} \to \mathbb{R}$ is some known $f^n$.

* We also know (let us say) that there is such an $x$ (say $x_0$) s.t. $f(x_0) = 0$; but if we knew exactly what $x_0$ is then we would not require a numerical algorithm!

* Our algo. gives us $X_{est}$ s.t. $f(X_{est}) = \epsilon$; $|\epsilon| \ll 1$

* We cannot calculate the real error $(X_0 - X_{est})$ b/c $x_0$ is unknown to us

forward errors

backward error

* But we can derive some sense of the error $f(X_{est}) - f(X_0) = f(X_{est})$

b/c $f(x_0) = 0$!

Q) Let us say we seek an algorithm for computing $\tilde{f}(x) = \sqrt{x}$.

When we plug in $x = 2$; this algo gives 1.4

Calculate the fwd. error, & bkwd error?

Ans :- fwd. err : $|(x_{est} - x_0)| = |1.4 - 1.41421\ldots| \cong 0.0142$

difficult to know!

Bkwd. err : $|f(x_{est}) - f(x_0)|$

$= |1.4^2 - 2| = |1.96 - 2| = 0.04$

much easier to calculate!

* In numerical analysis, backward error is closely associated w/ the notion of residual!