Chapter 1
Mathematical Preliminaries and Error Analysis

Hung-Yuan Fan (范洪源)

Department of Mathematics,
National Taiwan Normal University, Taiwan

Spring 2016

Section 1.1
Review of Calculus

Def 1.1

A fuction $f\colon X \to \mathbb{R}$ has the limit L at x_0, denoted by

$$\lim_{x \to x_0} f(x) = L,$$

if $\forall\ \varepsilon > 0, \exists\ \delta > 0 s.t.\ x \in X, 0 < |x - x_0| < \delta \Rightarrow |f(x) - L| < \varepsilon$.

Def 1.2

1. A fuction $f: X \to \mathbb{R}$ is continuous (簡寫: conti.) at $x_0 \in X$ if $\lim\limits_{x \to x_0} f(x) = f(x_0)$.
2. f is conti. on X if it is conti. at each point of X.
3. $C(X) = \{f |\, f \text{ is conti. on } X\}$ denotes the set of all conti. functions defined on X.

Note: if $X = [a, b]$, (a, b), $[a, b)$ or $(a, b]$ with $a < b$, write $C[a, b]$, $C(a, b)$, $C[a, b)$ or $C(a, b]$, respectively.

Def 1.3

A sequence (簡寫: seq.) of real numbers $\{x_n\}_{n=1}^{\infty}$ converges (簡寫: conv.) to the limit x, written

$$\lim_{n\to\infty} x_n = x, \text{ or } \quad x_n \to x \text{ as } n \to \infty,$$

if $\forall\ \varepsilon > 0,\ \exists\ N(\varepsilon) \in \mathbb{N}$ s.t. $n > N(\varepsilon) \Rightarrow |x_n - x| < \varepsilon$.

Thm 1.4 (序列與連續性的關係)

Let f be a real-valued function defined on $\emptyset \neq X \subseteq \mathbb{R}$ and $x_0 \in X$. The followings are equivalent:

a. f is conti. at x_0.

b. $\forall$ seq. $\{x_n\}_{n=1}^{\infty} \subseteq X$ with $\lim_{n\to\infty} x_n = x_0$, $\lim_{n\to\infty} f(x_n) = f(x_0)$.

Differentiability (可微分性)

Def 1.5

1. A fuction $f : X \to \mathbb{R}$ is differentiable (簡寫: diffi.) at $x_0 \in X$ if

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

2. f is cdiff. on X if it is cdiffy6t. at each point of X.
3. $C^n(X)$ denotes the set of all functions having n conti. derivatives on X.
4. $C^\infty(X)$ denotes the set of functions having derivatives of all orders on X.

Continuity v.s. Differentiability

Thm 1.6

Let f be a real-valued function defined on X and $x_0 \in X$. Then

$$f \text{ is diff. at } x_0 \Longrightarrow f \text{ is conti. at } x_0.$$

Thm 1.7 (Rolle's Thm)

$f \in C[a, b]$ and f is diff. on (a, b). If $f(a) = f(b)$, then $\exists\ c \in (a, b)$ s.t. $f'(c) = 0$.

Generalized Rolle's Theorem

Thm 1.10 (Generalized Rolle's Thm)

$f \in C[a,b]$ is n times diff. on (a,b). If $f(x_i) = 0$ for some $n+1$ *distinct* numbers $a \leq x_0 < x_1 < \cdots < x_n \leq b$, then $\exists\ c \in (x_0, x_n) \subseteq [a,b]$ s.t. $f^{(n)}(c) = 0$.

Mean Value Theorem (簡寫: MVT, 均值定理)

Thm 1.8 (MVT)

$f \in C[a,b]$ and f is diff. on (a,b). Then $\exists\ c \in (a,b)$ s.t.

$$f'(c) = \frac{f(b)-f(a)}{b-a} \quad \text{or} \quad f(b)-f(a) = f'(c)(b-a).$$

Extreme Value Theorem (簡寫: EVT, 極值定理)

Thm 1.9 (EVT)

If $f \in C[a, b]$, then $\exists\ c_1, c_2 \in [a, b]$ s.t.

$$f(c_1) \leq f(x) \leq f(c_2) \quad \forall\ x \in [a, b].$$

Intermediate Value Theorem (簡寫: IVT, 中間值定理)

Thm 1.11 (IVT)

$f \in C[a,b]$, K is any number between $f(a)$ and $f(b)$
$\Longrightarrow \exists\ c \in (a,b)$ s.t. $f(c) = K$.

Thm 1.14 (Taylor's Thm, 泰勒定理)

$f \in C^n[a,b]$, $f^{(n+1)}$ $\exists$ on $[a,b]$ and $x_0 \in [a,b]$.
$\Rightarrow \forall\ x \in [a,b]$, $\exists\ \xi(x)$ between x_0 and x s.t. $f(x) = P_n(x) + R_n(x)$, where

$$P_n(x) = \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k, \text{(\textbf{the } \textbf{\textit{n}th Taylor poly.} for } f)$$

$$R_n(x) = \frac{f^{(n+1)}(\xi(x))}{(n+1)!}(x - x_0)^{n+1}.$$

(**remainder** or **truncation error** associated with $P_n(x)$)

Remarks

1. If $\lim_{n\to\infty} R_n(x) = 0 \quad \forall\, x \in I$ (I: interval with $x_0 \in I$), then

$$f(x) = \lim_{n\to\infty} P_n(x) = \sum_{k=0}^{\infty} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k \quad \forall\, x \in I.$$

   We say that the **Taylor series for f about** x_0 conv. to f on I.

2. If $x_0 = 0$, the Taylor series is often called the **Maclaurin series**.

Example 3, p. 11

The second (or third) Taylor poly. for $f(x) = \cos x$ about $x_0 = 0$ is $P_2(x) = P_3(x) = 1 - \frac{1}{2}x^2$, but their truncation errors satisfy

$$|R_2(x)| \leq \frac{|\sin \xi(x)|\,|x|^3}{6} \leq \frac{|x|^4}{6} = 0.1\bar{6} \cdot |x|^4$$

$$(\because \ |\sin \xi(x)| \leq |\xi(x)| \leq |x| \quad \forall\ x \in \mathbb{R})$$

$$|R_3| \leq \frac{|\cos \tilde{\xi}(x)|\,|x|^4}{24} \leq \frac{|x|^4}{24} = 0.041\bar{6} \cdot |x|^4.$$

(**Sharper Bound** for $|x| \approx 0$!)

What are the goals of numerical analysis?

Remark

Two objectives of numerical analysis:

1. Find an approximation to the solution of a given problem.
2. Determine a bound for the accuracy of the approximation. Is this error bound tight and sharp?

Def 1.12 (定積分的定義)

1. The (Riemann) definite integral of f on $[a, b]$ is defined by

$$\int_a^b f(x)\,dx = \lim_{\max\limits_{1\le i\le n} \Delta x_i \to 0} \sum_{i=1}^n f(z_i)\Delta x_i,$$

where $P = \{a = x_0 < x_1 < \cdots < x_n = b\}$ is any partition of $[a, b]$, $z_i \in [x_{i-1}, x_i]$ and $\Delta x_i = x_i - x_{i-1}$ for $i = 1, 2, \ldots, n$.

2. f is called (Riemann) integrable over $[a, b]$ if the limit exists.

Note: f is conti. on $[a, b]$ $\Rightarrow$ f is integrable over $[a, b]$.

Remark

f is integrable over $[a, b] \Longrightarrow$

$$\int_a^b f(x)\,dx = \lim_{n\to\infty} \sum_{i=1}^{n} f(z_i)\,\mathbf{x}, \quad \left(\mathbf{x} = \frac{\mathbf{b}-\mathbf{a}}{\mathbf{n}}\right)$$

$$\approx \sum_{i=0}^{n} w_i \cdot f(x_i)\,\mathbf{x}, \quad (w_i\textbf{: weighting coeff.})$$

with $z_i = x_i$ or x_{i-1} for $i = 1, 2, \ldots, n$.

Riemann Sums (黎曼和) with $z_i = x_i \quad \forall\ i$

Thm 1.13 (定積分的權重均值定理)

$f \in C[a, b]$ and g is an integrable function that **does not** change sign on $[a, b]$. Then $\exists\ c \in (a, b)$ s.t.

$$\int_a^b f(x)g(x)\,dx = f(c)\int_a^b g(x)\,dx.$$

Note: When $g(x) \equiv 1$, we have

$$f(c) = \frac{1}{b-a}\int_a^b f(x)\,dx \equiv f_{avg},$$

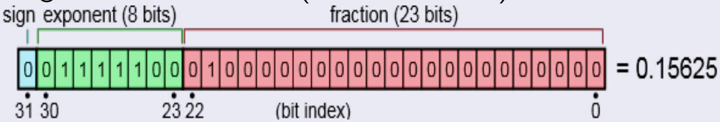
where f_{avg} is the **average value** of f on $[a, b]$.

The Average Value of a Function

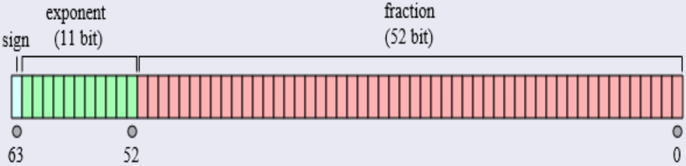
Section 1.2
Round-off Errors and Computer Arithmetic
(捨入誤差與電腦算術)

IEEE 754-1985 Standard (updated version: IEEE 754-2008)

1. Single Precision Format (32 bits; 單精度)



2. Double Precision Format (64 bits; 雙精度)



3. Extended Precision Format (80 bits; 擴充精度)
   sign: 1 bit, exponent: 16 bits, fraction: 63 bits

64-bit Floating-Point Representation

- 64-bit representation is used for a real number.
- Each **binary** floating-point number (浮點數) has at least 16 **decimal digits** of precision.
- 1-bit **sign (符號)** s is followed by 11-bit **exponent (指數)** c (**characteristic**, $0 \leq c \leq 2^{11} - 1 = 2047$) and 52-bit binary **fraction** f (**mantissa**: 尾數).

The Normalized Forms (正規化形式或標準化形式)

- Normalized *binary* floating-pint form of $x \in \mathbb{R}$ is

$$fl(x) = (-1)^s 2^{c-1023}(1+f)_2 = (-1)^s \left(1 + \sum_{i=1}^{k} b_i 2^{-i}\right)_{10} 2^{c-1023},$$

where $f = (0.b_1 b_2 \cdots b_k)_2$.

- $\mathfrak{F} = \{fl(y) \mid y \in \mathbb{R}\}$ is a **finite** (and **proper**) subset of $\mathbb{R}$.
- The difference between two **adjacent** (相鄰的) 64-bit floating-point numbers is $\varepsilon_M = 2^{-52} \approx 2.22 \times 10^{-16}$.

Note: the **machine precision** (or **epsilon**) is $\varepsilon_M = 2^{-23} \approx 1.19 \times 10^{-7}$ for the single precision format.

1. Since

$$27.56640625_{10} = \mathbf{11011.10010001_2}$$
$$= 1.\mathbf{101110010001_2} \times 2^4, \text{ (Normalized Form)}$$

we have $s = 0$, $c = 4 + 1023 = 1027_{10} = \mathbf{10000000011_2}$ and mantissa $f = 0.\mathbf{101110010001_2}$. Using IEEE 754 format ⇒

$$\mathbf{0}\ \mathbf{10000000011}\ \mathbf{101110010001}00\cdots0 \text{ (補 40 個零!)}$$

2. Note that

$$0.1_{10} = 0.0\,\overline{\mathbf{0011}}_2 = 1.1\,\overline{\mathbf{0011}}_2 \times 2^{-4}.$$

How to store 0.1_{10} by using IEEE 754 format?

Remarks on IEEE 754 Format

1. The smallest positive floating-point number (with $s = 0$, $c = 1$ and $f = 0$) is
$$fl_{\min} = 2^{-1022}(1+0) \approx 2.2 \times 10^{-308}.$$
2. The largest one (with $s = 0$, $c = 2046$ and $f = 1 - 2^{-52}$) is
$$fl_{max} = 2^{1023}(2 - 2^{-52}) \approx 1.8 \times 10^{308}.$$
3. $|fl(x)| > fl_{\max} \Rightarrow$ **overflow (上溢位)** and $|fl(x)| < fl_{\min} \Rightarrow$ **underflow (下溢位)** and **reset** $\mathbf{x} = \mathbf{0}$.
4. Two zeros $+\mathbf{0}$ (with $s = 0, c = 0, f = 0$) and $-\mathbf{0}$ (with $s = 1, c = 0, f = 0$) exist!

Decimal Machine Numbers (十進位機器數字)

- Normalized *decimal* floating-point form of $y \in \mathbb{R}$ is

$$fl(y) = \pm 0.d_1 d_2 \cdots d_k \times 10^n,$$

where $\mathbf{1 \leq d_1 \leq 9}$, $\mathbf{0 \leq d_i \leq 9}$ $\mathbf{(i = 2, \ldots, k)}$ and $n \in \mathbb{Z}$. In this case, $fl(y)$: **k-digit decimal machine number**.

- The k-digit $fl(y)$ of a **normalized** real number

$$y = \pm 0.d_1 d_2 \cdots d_k d_{k+1} \cdots \times 10^n$$

can be obtained by **terminating** the mantissa of y at k decimal digits.

1. **Chopping:** (直接捨去法)

$$fl(y) = \pm 0.\mathbf{d_1 d_2 \cdots d_k} \times 10^n,$$

i.e. simply chop off the digits $d_{k+1} d_{k+2} \cdots$.

2. **Rounding:** (四捨五入法)

$$fl(y) = \begin{cases} \pm(0.d_1 d_2 \cdots d_k + \mathbf{10^{-k}}) \times 10^n, & d_{k+1} \geq 5 \textbf{ (Round Up)} \\ \pm 0.\mathbf{d_1 d_2 \cdots d_k} \times 10^n, & d_{k+1} < 5 \textbf{ (Round Down)} \end{cases}$$

$$\equiv \pm 0.\delta_1 \delta_2 \cdots \delta_k \times 10^n \text{ after chopping.}$$

Example 1, p. 20

Determine the 5-digit (a) chopping and (b) rounding values of

$$\pi = 0.\mathbf{31415}926\cdots \times 10^1.$$

Sol:

(a) $fl(\pi) = 0.31415 \times 10^1$ by chopping.

(b) $fl(\pi) = (0.31415 + 10^{-5}) \times 10^1 = 0.3141\mathbf{6} \times 10^1$ by rounding.

Def 1.15

If p^* is an approximation to p, then

1. the **absolute error** is $AE(p^*) = |p^* - p|$.
2. the **relative error** is

$$RE(p^*) = \frac{|p^* - p|}{|p|}, \text{ provided that } p \neq 0.$$

Note: the relative error is independent of the magnitude of p, but the absolute error might vary widely!

Example 2, p. 21

Find the abs. and rel. errors when approximating p by p^*.

(a) $p = 0.3000 \times \mathbf{10^1}$ and $p^* = 0.3100 \times \mathbf{10^1}$.

(b) $p = 0.3000 \times \mathbf{10^{-3}}$ and $p^* = 0.3100 \times \mathbf{10^{-3}}$.

(c) $p = 0.3000 \times \mathbf{10^4}$ and $p^* = 0.3100 \times \mathbf{10^4}$.

Sol:

(a) $AE(p^*) = \mathbf{0.1}$ and $RE(p^*) = \mathbf{0.3333 \times 10^{-1}}$.

(b) $AE(p^*) = \mathbf{0.1 \times 10^{-4}}$ and $RE(p^*) = \mathbf{0.3333 \times 10^{-1}}$.

(c) $AE(p^*) = \mathbf{0.1 \times 10^3}$ and $RE(p^*) = \mathbf{0.3333 \times 10^{-1}}$.

(相對誤差都一樣, 但是絕對誤差變化很大!)

Def 1.16

p^* approximate $p \neq 0$ to t **significant digits** (or **figures**) if $\exists$ largest $t \in \mathbb{N} \cup \{0\}$ satisfying

$$RE(p^*) = \frac{|p^* - p|}{|p|} \leq 5 \times 10^{-t}.$$

Note: for any normalized $y = 0.d_1 d_2 \cdots \times 10^n \in \mathbb{R}$, its k-digit decimal representation satisfies

$$RE(fl(y)) \leq 10^{-k+1} = \mathbf{10^{-(k-1)}}$$

by using **chopping** (see the textbook), and

$$RE(fl(y)) \leq 0.5 \times 10^{-k+1} = \mathbf{5 \times 10^{-k}}$$

by using **rounding**. (See Ex. 24, p. 31!)

Elementary Floating-Pont Arithmetic

For floating-point representations $fl(x)$ and $fl(y)$ of real numbers x and y, assume that

$$x \oplus y = fl(fl(x) + fl(y)), \quad x \otimes y = fl(fl(x) \times fl(y)),$$
$$x \ominus y = fl(fl(x) - fl(y)), \quad x \oslash y = fl(fl(x) \div fl(y)).$$

Note: in practical computation, we usually have

$$fl(x \,\mathbf{op}\, y) = (x \,\mathbf{op}\, y)(1+\delta) \text{ with } |\delta| \leq \varepsilon_M,$$

where $\mathbf{op} = +, -, \times, \div$, and ε_M is the machine precision.

Cancellation of Significant Digits

If $x, y \in \mathbb{R}$ $(x > y)$ have the k-digit decimal representations

$$fl(x) = 0.\mathbf{d_1 d_2 \cdots d_p} \alpha_{p+1}\alpha_{p+2}\cdots\alpha_k \times 10^n,$$
$$fl(y) = 0.\mathbf{d_1 d_2 \cdots d_p} \beta_{p+1}\beta_{p+2}\cdots\beta_k \times 10^n,$$

then

$$fl(x) - fl(y) = (0.\alpha_{p+1}\alpha_{p+2}\cdots\alpha_k - 0.\beta_{p+1}\beta_{p+2}\cdots\beta_k) \times \mathbf{10^{n-p}}$$
$$\equiv 0.\sigma_{\mathbf{p+1}}\sigma_{\mathbf{p+2}}\cdots\sigma_{\mathbf{k}} \times \mathbf{10^{n-p}},$$

i.e. $x \ominus y = fl(fl(x) - fl(y))$ has **at most** $k - p$ significant digits, with the last p digits being either 0 or randomly assigned.

Remark

Suppose that $fl(z) = z + \delta$ with $|\delta|$ being the absolute error. If $\varepsilon = 10^{-n}$ with $n \in \mathbb{N}$ is a number of **small magnitude**, then

$$\frac{fl(z)}{fl(\varepsilon)} \approx (z + \delta) \times 10^n = \frac{z}{\varepsilon} + 10^n \delta.$$

So, the absolute error in computing z/ε is

$$\left| \frac{fl(z)}{fl(\varepsilon)} - \frac{z}{\varepsilon} \right| \approx 10^n \cdot |\delta| = |\delta|/\varepsilon.$$

Example 4, pp. 23–24 (1/2)

Given four real numbers

$$x = \frac{5}{7} = 0.\overline{714285}, \quad u = 0.714251$$
$$v = 98765.9, \quad w = 0.111111 \times 10^{-4}.$$

Find 5-digit chopping values of $x \ominus u$, $(x \ominus u) \oslash w$, $(x \ominus u) \otimes v$ and $u \oplus v$.

Sol: The absolute error for $x \ominus u$ is

$$\begin{aligned}
&|(x-u)-(x\ominus u)| = |(x-u) - fl(fl(x) - fl(u))| \\
=&|(\frac{5}{7} - 0.714251) - fl(0.\mathbf{71428} \times 10^0 - 0.\mathbf{71425} \times 10^0)| \\
=&|\mathbf{0.347143 \times 10^{-4} - 0.30000 \times 10^{-4}}| \\
=&\mathbf{0.47143 \times 10^{-5}}.
\end{aligned}$$

The relative error for $x \ominus u$ is given by

$$RE(x \ominus u) = \left| \frac{0.47143 \times 10^{-5}}{0.347143 \times 10^{-4}} \right| = 0.1358 \leq \mathbf{0.136}.$$

Some Tricks

1. Reformulation of the calculations to avoid the subtraction of two nearly equal numbers.
(改變計算公式以避免相近數字相減)
2. Rearrangement of the calculations by the nested arithmetic.
(利用巢狀算術技巧以減少四則運算數量)

The lesson: Think before you compute!

- Distinct real roots of $ax^2 + bx + c = 0$ with $a \neq 0$ and $b^2 - 4ac > 0$ are

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}, \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}.$$

- If $b > 0$ and $4ac \ll b^2$, then
  - $-b + \sqrt{b^2 - 4ac} \approx 0 \Rightarrow$ **Loss of accuracy for computing x_1!**
  - Rewrite the formula for x_1 by rationalization (有理化)

$$x_1 = \frac{-2c}{\mathbf{b} + \sqrt{\mathbf{b}^2 - 4\mathbf{ac}}}.$$ **(分母不是相近數相減!)**

  - Use $x_1 x_2 = \frac{c}{a} \Rightarrow x_2 = \frac{c}{a x_1} = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$.

Example, pp. 25–26

Use 4-digit rounding arithmetic to determine the first root x_1 of $f(x) = x^2 + 62.10x + 1 = 0$.

Sol: Two real roots of $f(x) = 0$ are approximately

$$x_1 = -\mathbf{0.01610723}, \quad x_2 = -62.08390.$$

Use 4-digit rounding $\Rightarrow$

$$fl(\sqrt{b^2 - 4ac}) = fl(\sqrt{(62,10)^2 - (4.000)(1.000)(1.000)}) = \mathbf{62.06},$$

$$f(x_1) = \frac{-\mathbf{62.10} + \mathbf{62.06}}{2.000} = -\mathbf{0.02000},$$

with the relative error being

$$RE(fl(x_1)) = \frac{|-0.01611 + 0.02000|}{|-0.01611|} = \mathbf{2.4} \times \mathbf{10^{-1}}.$$

In addition, if we use the reformulation for x_1, then

$$fl(x_1) = fl\Big(\frac{fl(-2c)}{fl(b+\sqrt{b^2-4ac})}\Big) = fl\Big(\frac{-2.000}{62.10+62.06}\Big) = \mathbf{-0.01610},$$

which has the small relative error $\mathbf{6.2 \times 10^{-4}}$.

Note: **近似零根 x_1 的精度提升至 3 個有效位數!**

Example 6, pp. 26–27

Evaluate the 3-digit chopping and rounding values of a poly. $f(x) = x^3 - 6.1x^2 + 3.2x + 1.5$ at $x = 4.71$.

Sol: The actual value is $y = f(4.71) = -\mathbf{14.263899}$. Using 3-digit chopping/rounding arithmetic, we have. The 3-digit approx. values of y are

$$fl(y) = fl\Big(((104. - 134.) + 15.0) + 1.5\Big) = -\mathbf{13.5}. \text{ (Chopping)}$$

$$fl(y) = fl\Big(((105. - 135.) + 15.1) + 1.5\Big) = -\mathbf{13.4}. \text{ (Rounding)}$$

An Example of Polynomial Evaluation (2/2)

Hence, the relative errors in computing $fl(y)$ are

$$RE(fl(y)) = \left| \frac{-14.263899 + 13.5}{-14.263899} \right| \approx \mathbf{5.36 \times 10^{-2}}, \ \textbf{(Chopping)}$$

$$RE(fl(y)) = \left| \frac{-14.263899 + 13.4}{-14.263899} \right| \approx \mathbf{6.06 \times 10^{-2}}. \ \textbf{(Rounding)}$$

$\Longrightarrow$ Only one significant digit for both chopping and rounding values of $y = f(4.71)$!

Rearrangement of Poly. Evaluation

- Direct Computation: (**4** multiplications and 3 additions)

$$f(x) = x \cdot (x \cdot x) - 6.1 \cdot (x \cdot x) + 3.2 \cdot x + 1.5$$

- Nested Computation: (**2** multiplications and 3 additions)

$$f(x) = \Big((x - 6.1) \cdot x + 3.2 \Big) \cdot x + 1.5$$

Again, using **3-digit** arithmetic with the **nested form** $\Longrightarrow$

$$RE(fl(y)) = \left| \frac{-14.263899 + \mathbf{14.2}}{-14.263899} \right| \approx \mathbf{4.5} \times \mathbf{10^{-3}}, \ \textbf{(Chopping)}$$

$$RE(fl(y)) = \left| \frac{-14.263899 + \mathbf{14.3}}{-14.263899} \right| \approx \mathbf{2.5} \times \mathbf{10^{-3}}. \ \textbf{(Rounding)}$$

Useful Suggestion

The accuracy of an approximation can be improved if we reduce the number of arithmetic operations.
(減少四則運算的數量可以改進計算解的精度!)

HW of Sec 1.2:

$\surd$ **24,**

Section 1.3
Algorithms and Convergence
(演算法與收斂性)

Algorithms and Pseudocodes (虛擬碼)

- An algorithm is a **procedure** that describes **a finite sequence of steps** to be performed in a **specified order**.
- The objective of an algorithm is to implement a procedure for **solving a problem** or **approximating a solution to the problem**.
(演算法目標是求解問題或是得到該問題的數值近似解)
- Pseudocode is an **informal environment-independent** description of the key principles of an algorithm.
- It uses structural conventions of a programming language, but is intended for **human reading** rather than machine reading.

An Example of Pseudocode

To solve the root-finding problem

$$f(x) = ax^2 + bx + c = 0 \quad \text{with} \quad a \neq 0.$$

INPUT coefficients a, b, c.

OUTPUT approximate root x.

Step 1 Compute the discriminant $D = b^2 - 4ac$.

Step 2 Compute approximate root x to $f(x) = 0$ using D.

Step 3 OUTPUT(x); **STOP**.

An Illustration of Algorithm

INPUT $N, x_1, x_2, \ldots, x_n$.

OUTPUT $SUM = \sum_{i=1}^{N} x_i$.

Step 1 Set $SUM = 0$. (累加器初始化)

Step 2 For $i = 1, 2, \ldots, N$ do

set $SUM = SUM + x_i$. (加入下一項)

Step 3 OUTPUT (SUM);

STOP.

Example 1, p. 33

The Nth Taylor poly. of $f(x) = \ln x$ about $x_0 = 1$ is

$$P_N(x) = \sum_{i=1}^{N} \frac{(-1)^{i+1}}{i}(x-1)^i.$$

Construct an algorithm to determine the **minimal** value of N s.t.

$$|\ln(1.5) - P_N(1.5)| < 10^{-5}.$$

Note: From the Alternating Series Thm $\Longrightarrow$

$$|\ln x - P_N(x)| \le \left|\frac{(-1)^{N+1}}{N+1}(x-1)^{N+1}\right|.$$

So, the **stopping criterion** (停止準則) should be

$$|a_{N+1}| = \left|\frac{(-1)^{N+1}}{N+1}(x-1)^{N+1}\right| < TOL,$$

where TOL denotes the tolerance. (容許誤差)

INPUT x 的值，容許誤差 TOL，最大迭代數 M。

OUTPUT 多項式次數 N 或錯誤訊息。

Step 1 Set $N = 1$;
$y = x - 1$;
$SUM = 0$;
$POWER = y$;
$TERM = y$;
$SIGN = -1$. (用以改變正負號)

Step 2 While $N \le M$ do Steps 3–5.

Step 3 Set $SIGN = -SIGN$; (符號交換)
$SUM = SUM + SIGN \cdot TERM$; (累加各項)
$POWER = POWER \cdot y$;
$TERM = POWER/(N + 1)$. (計算下一項)

Step 4 If $|TERM| < TOL$ then (檢驗精度)
OUTPUT (N);
STOP. (計算成功)

Step 5 Set $N = N + 1$. (準備下一次迭代)

Step 6 OUTPUT ('Method Failed'); (計算不成功)
STOP.

Definition

- An algorithm is called **stable** if it satisfies the property that **small** changes in the initial data produce correspondingly **small** changes in the final results.
  (初始資料的微小變動 $\Longrightarrow$ 計算結果也是微小變化)

- Otherwise, the algorithm is called **unstable**, i.e. **small** changes in the initial data produce **large** changes in the final results.
  (初始資料的微小變動 $\Longrightarrow$ 計算結果產生大幅變化)

Def 1.17 (誤差的線性與指數成長)

$E_0 > 0$: the magnitude of error at some stage in the calculations,
E_n: the magnitude of error after n subsequent operations.

1. The growth of error is called **linear** if $E_n \approx CnE_0$, where the constant $C > 0$ is independent of n.
2. The growth of error is called **exponential** if $E_n \approx C^n E_0$ for some $C > 1$.

Example of an Unstable Algorithm (1/2)

An Unstable Procedure

The sequence $\{p_n\}_{n=0}^{\infty}$ defined by

$$p_n = c_1(\frac{1}{3})^n + c_2 3^n$$

is the general solution to the recursive equation (遞迴方程式) $p_n = \frac{10}{3}p_{n-1} - p_{n-2}, \quad n = 2, 3, \ldots.$

- $\mathbf{p_0 = 1, p_1 = \frac{1}{3} \Rightarrow c_1 = 1, c_2 = 0}$. The solution is

$$p_n = (\frac{1}{3})^n.$$

- Use 5-digit rounding $\Rightarrow \mathbf{\widehat{p}_0 = 1.0000, \widehat{p}_1 = 0.33333}$ and hence $\mathbf{\widehat{c}_1 = 1.0000, \widehat{c}_2 = -0.12500 \times 10^{-5}}$. The solution is

$$\hat{p}_n = 1.0000(\frac{1}{3})^n - 0.12500 \times 10^{-5}(3^n).$$

Example of an Unstable Algorithm (2/2)

The absolute error in computing $\hat{p}_n$ is

$$AE(\hat{p}_n) = p_n - \hat{p}_n = 0.12500 \times 10^{-5}(3^n).$$

$\Longrightarrow$ An **unstable** procedure with **exponential** growth of errors!

Def 1.18

Suppose that $\{\alpha_n\}_{n=1}^{\infty}$ and $\{\beta_n\}_{n=1}^{\infty}$ are two sequences with $\lim_{n\to\infty} \alpha_n = \alpha$ and $\lim_{n\to\infty} \beta_n = 0$. If $\exists\ K > 0$ and $n_0 \in \mathbb{N}$ s.t.

$$|\alpha_n - \alpha| \le K|\beta_n| \quad \forall\ n \ge n_0,$$

then we say that $\{\alpha_n\}_{n=1}^{\infty}$ conv. to α with **rate (or order) of convergence $\mathbf{O}(\beta_\mathbf{n})$**, and write

$$\alpha_n = \alpha + O(\beta_n). \quad (\text{as } n \to \infty)$$

Note: seq. $\{\alpha_n\}_{n=1}^{\infty}$ is often generated by some **iterative method** (迭代法), and it is often compared with $\beta_n = \dfrac{1}{n^p}$ for $p > 0$.

Example 2, p. 37

For $n \geq 1$, consider two sequences of real numbers

$$\alpha_n = \frac{n+1}{n^2} \quad \text{and} \quad \hat{\alpha}_n = \frac{n+3}{n^3}.$$

Determine their rates of convergence.

Sol: Since

$$|\alpha_n - 0| = \frac{n+1}{n^2} \leq \frac{n+n}{n^2} = \mathbf{2} \cdot \frac{1}{n} \equiv \mathbf{2}\beta_n,$$
$$|\hat{\alpha}_n - 0| = \frac{n+3}{n^3} \leq \frac{n+3n}{n^3} = \mathbf{4} \cdot \frac{1}{n^2} \equiv \mathbf{4}\hat{\beta}_n$$

for all $\mathbf{n} \geq \mathbf{1}$, it follows that

$$\alpha_n = 0 + \mathbf{O}(\frac{\mathbf{1}}{\mathbf{n}}), \quad \hat{\alpha}_n = 0 + \mathbf{O}(\frac{\mathbf{1}}{\mathbf{n^2}}).$$

Def 1.19

Suppose that $\lim_{h\to 0} F(h) = L$ and $\lim_{h\to 0} G(h) = 0$. If $\exists\ K > 0$ and $\delta > 0$ s.t.

$$|F(h) - L| \leq K|G(h)| \quad \text{for } 0 < |h| < \delta,$$

then we write

$$F(h) = L + \mathbf{O(G(h))}. \quad (\text{as } h \to 0)$$

Note: In practice, we often choose $G(h) = h^p$ for $p > 0$, and the largest value of p is expected.

Example 3, p. 38

Show that $\cos h + \frac{1}{2}h^2 = 1 + O(h^4)$.

pf: From Taylor's Thm, $\exists\ \xi(h)$ between 0 and h s.t.

$$\cos h = 1 - \frac{1}{2}h^2 + \frac{\cos\xi(h)}{24}h^4 \quad \text{for } h \neq 0.$$

Hence, we see that

$$\left|(\cos h + \frac{1}{2}h^2) - 1\right| = \frac{|\cos\xi(h)|}{24}|h^4| \leq \frac{\mathbf{1}}{\mathbf{24}}|\mathbf{h^4}| \quad \text{for } h \neq 0,$$

which gives the desired result by Def.

Thank you for your attention!