4. Operazioni aritmetiche con numeri approssimati

Nei capitoli precedenti si è discussa una delle basi del calcolo numerico: il sistema floating-point di rappresentazione dei numeri reali nel calcolatore.
Il concetto cardine è quello di precisione di macchina, il massimo errore relativo di arrotondamento di un numero reale rappresentabile (ovvero non in overflow o underflow) con il reale-macchina "più vicino".

In questo capitolo inizieremo ad entrare nel cuore del calcolo numerico, trattando innanzitutto le operazioni aritmetiche nel sistema floating-point, cioè quale sia l'effetto degli errori sui dati, sul risultato dell'operazione.
L'analisi che faremo però, sara' più generale e permettera' di studiare la risposta agli errori sui dati, delle operazioni aritmetiche, qualunque sia la fonte di errore (ad esempio errori di misura sperimentale).

Cominciamo col definire il concetto di *operazione-macchina*; sia $\star$ un'operazione aritmetica nei reali, ovvero:

$$\star = \begin{cases} \pm & \text{addizione, sottrazione} \\ \cdot & \text{moltiplicazione} \\ / & \text{divisione} \end{cases}$$

Allora dato un insieme di reali-macchina $\mathbb{F}(b,t,L,U)$, il modo in cui il processore realizza l'operazione tra due reali rappresentabili x, y è

$$\underbrace{x \star y}_{\text{operazione macchina}} \quad = \quad \mathrm{fl}^t\Big(\mathrm{fl}^t(x) \star \mathrm{fl}^t(x)\Big)$$

ovvero il modello è il seguente:

1. i due reali vengono arrotondati;
2. viene fatta l'operazione tra gli arrotondati;
3. il risultato viene a sua volta arrotondato.

E' importante osservare da subito che le operazioni-macchina nel sistema floating-point perdono varie proprieta' delle operazioni aritmetiche teoriche.
Ad esempio, mentre la proprieta' commutativa di addizione e moltiplicazione resta valida, in generale non sono più valide le proprieta' associativa e distributiva.

Facciamo un esempio in cui non vale la proprieta' associativa della moltiplicazione, per problemi di overflow.
Consideriamo $\mathbb{F}(10, 16, -307, 308)$ che come abbiamo visto corrisponde sostanzialmente all'interfaccia di Matlab. Siano $a = 10^{200}, b = 10^{150}, c = 10^{-50}$ in aritmetica reale

$$(a \cdot b) \cdot c = a \cdot (b \cdot c) = 10^{300}$$

ma, con le limitazioni date per gli esponenti

$$(a \odot b) \odot c = \text{overflow, perché } a \odot b = 10^{350}$$

mentre

$$a(b \odot c) = 10^{300}$$

Come vedremo più avanti, la proprieta' associativa (e anche la distributiva) possono non valere piu' anche per effetto degli errori di arrotondamento.

D'altra parte, c'è un'altra proprieta' che non è più valida a causa dell'arrotondamento: l'unicita' dell'elemento neutro dell'addizione.
Per capirlo, consideriamo di nuovo un sistema floating-point "virtuale" $\mathbb{F}(10, 16, -307, 308)$, dove la cosa che conta davvero è il numero di cifre di mantissa.
Si ha $1 \oplus 10^{-16} = 1$ cioè 10^{-16} si comporta come 0 nell'addizione con 1 mentre $1 \oplus 10^{-15} = 1 + 10^{-15}$ ma perché?
Basta riflettere sul fatto che le cifre di mantissa sono 16:

$$1 + 10^{-16} = (0.100\ldots0 \underbrace{1}_{\text{17-esima cifra}}) \cdot 10^1$$

viene arrotondato ad 1 perché la prima cifra trascurata è $1 < \dfrac{b}{2} = 5$.
Invece

$$1 + 10^{-15} = (0.100\ldots0 \underbrace{1}_{\text{16-esima cifra}}) \cdot 10^1$$

e quindi non c'è bisogno di alcun arrotondamento.

D'altra parte, si vede anche subito che $10^{-1} \oplus 10^{-16} = 10^{-1} + 10^{-16}$ cioè 10^{-16} non è elemento neutro per 10^{-1}. In questo contesto si può dare una seconda caratterizzazione della precisione di macchina: $\epsilon_M = \min\{\mu \in \mathbb{F}^+ : 1 \oplus \mu > 1\}$.

Passiamo alla questione chiave: la risposta delle operazioni agli errori sui dati.

Possiamo osservare fin da subito che l'arrotondamento finale del risultato ha un errore che non può superare la precisione di macchina $\epsilon_M = \dfrac{b^{1-t}}{2}$ e quindi è ininfluente ai fini dell'analisi dell'effetto degli errori sui dati.
Invece, l'errore chiave da stimare è l'errore relativo:

$$\epsilon_{x \star y} = \frac{\left| \Big(x \star y \Big) - \Big(\mathrm{fl}^t(x) \star \mathrm{fl}^t(y) \Big) \right|}{\left| \Big(x \star y \Big) \right|}$$

purché $x \star y \neq 0$, in funzione degli errori relativi sui dati:

$$
\begin{aligned}
_x &= \frac{|x - \mathrm{fl}^t(x)|}{|x|} \qquad x \neq 0 \\
\epsilon_y &= \frac{|y - \mathrm{fl}^t(y)|}{|y|} \qquad y \neq 0
\end{aligned}
$$

Più in generale, dati due numeri reali $x, y \neq 0$ e due loro approssimazioni $\tilde{x} \approx x, \tilde{y} \approx y$, dove supponiamo di conoscere i loro errori relativi, andremo a stimare l'errore relativo sul risultato dell'operazione $\star$, commesso utilizzando i dati approssimati invece dei dati esatti, in pratica andremo a calcolare:

$$
\epsilon_{x \star y} = \frac{\big|(x \star y) - (\tilde{x} \star \tilde{y})\big|}{\big|(x \star y)\big|}, x \star y \neq 0
$$

in funzione di ϵ_x, ϵ_y.

> Nel caso delle operazioni-macchina si ha $\tilde{x} = \mathrm{fl}^t(x), \tilde{y} = \mathrm{fl}^t(y), \epsilon_x, \epsilon_y \leq \epsilon_M$.

Chiameremo *stabile* un'operazione aritmetica per cui l'errore sul risultato ha lo stesso ordine di grandezza dell'errore massimo sui dati.

Analisi della moltiplicazione

> Indicheremo il prodotto cola notazione standard xy, sapendo che nei linguaggi di calcolo il simbolo della moltiplicazione è $*$.

Iniziamo definendo quindi:

$$
\epsilon_{xy} = \frac{\big|(xy) - (\tilde{x}\tilde{y})\big|}{\big|(xy)\big|}, \quad x, y \neq 0
$$

Usiamo la stessa tecnica di stima che si usa per dimostrare che il limite del prodotto di due successioni o funzioni è il prodotto dei limiti, aggiungendo e togliendo a numeratore ad esempio $\tilde{x}y$:

$$
\begin{aligned}
\epsilon_{xy} &= \frac{\big|xy - \tilde{x}y + \tilde{x}y - \tilde{x}\tilde{y}\big|}{\big|xy\big|} \\
&= \frac{\big|\overbrace{y(x - \tilde{x})}^{=a} + \overbrace{\tilde{x}(y - \tilde{y})}^{=b}\big|}{\big|xy\big|} \\
&\leq \frac{\big|y(x - \tilde{x})\big| + \big|\tilde{x}(y - \tilde{y})\big|}{\big|xy\big|}
\end{aligned}
$$

dove abbiamo usato la *disuguaglianza triangolare*, che è uno strumento chiave per fare le stime.
Ricordandoci ora che il modulo del prodotto è il prodotto dei moduli otteniamo:

$$\epsilon_{xy} \leq \frac{|y||x-\tilde{x}|}{|x||y|} + \frac{|\tilde{x}||y-\tilde{y}|}{|x||y|}$$
$$\text{con } \epsilon_x = \frac{|x-\tilde{x}|}{|x|}, \epsilon_y = \frac{|y-\tilde{y}|}{|y|}$$
$$\text{quindi } \epsilon_{xy} \leq \frac{\cancel{|y|}\,|x-\tilde{x}|}{|x|\,\cancel{|y|}} + \frac{|\tilde{x}||y-\tilde{y}|}{|x||y|} = \epsilon_x + \frac{|\tilde{x}|}{|x|}\epsilon_y$$

Ora, siccome $\tilde{x} \approx x$ possiamo dire almento qualitativamente che $\frac{|\tilde{x}|}{|x|} \approx 1$ e quindi

$$\epsilon_{xy} \leq \epsilon_x + \frac{|\tilde{x}|}{|x|}\epsilon_y \approx \epsilon_x + \epsilon_y$$

cioè l'operazione di moltiplicazione è *stabile*, perchè l'errore relativo sul risultato è maggiorato da una quantita che è dell'ordine dell'errore sui dati.

Per esprimere questo fatto possiamo usare la notazione $\epsilon_{xy} \leq\approx \epsilon_x + \epsilon_y$ dove $\leq\approx$ non è una disuguaglianza esatta ma va intesa nel senso indicato sopra(*"minore o uguale ad una quantita' prossima a"*); in realta' possiamo dare anche una stima quantitativa osservando che per la disuguaglianza triangolare:

$$\frac{|\tilde{x}|}{|x|} = \frac{|\overbrace{x}^{=a} + \overbrace{\tilde{x}-x}^{=b}|}{|x|} \leq \frac{|x| + |\tilde{x}-x|}{|x|} = 1 + \epsilon_x \implies \epsilon_{xy} \leq \epsilon_x + (1+\epsilon_x)\epsilon_y$$

Nel caso della moltiplicazione-macchina in precisione doppia $\epsilon \leq \epsilon_M \approx 10^{-16}$ e quindi $1+\epsilon_x$ è vicinissimo a 1. Ma anche con $\epsilon_x \approx 10^{-1}$, ad esempio un errore di misura del 10%, che è un errore sperimentale grande, avremmo $1+\epsilon_x \approx 1.1$ e quindi la sostanza della stabilita' non cambia.

Analisi della divisione

Siccome la divisione $\frac{x}{y}, y \neq 0$ è la moltiplicazione per il reciproco, $\frac{x}{y} = x \cdot \frac{1}{y}$, ci basta analizzare la stabilita' dell'operazione di reciproco:

$$\epsilon_{\frac{1}{y}} = \frac{\left|\frac{1}{y} - \frac{1}{\tilde{y}}\right|}{\left|\frac{1}{y}\right|} = \left|\frac{1}{y} - \frac{1}{\tilde{y}}\right||y| = \frac{|y-\tilde{y}|}{|y\tilde{y}|}|y| = \frac{|y-\tilde{y}|}{|y|} \cdot \frac{|y|}{|\tilde{y}|} \approx \epsilon_y$$

con l'ipotesi qualitativa che $|y| \approx |\tilde{y}|$ e quindi $\frac{|y|}{|\tilde{y}|} \approx 1$.
Ne deduciamo che anche la divisione è un'operazione stabile, perché il reciproco e' stabile e la moltiplicazione è stabile. Anche in questo caso possiamo però quantificare, stimando meglio $\frac{|y|}{|\tilde{y}|}$.

Assumiamo $\epsilon_y = \dfrac{|y-\tilde{y}|}{|y|} < 1$, cioè che l'errore relativo sia $< 100\%$ (vero in tutte le situazioni ragionevoli), allora $|\tilde{y}| = |y + \tilde{y} - y| = |y| \cdot \left|1 + \dfrac{(\tilde{y}-y)}{y}\right|$; usando la stima da sotto nella disuguaglianza triangolare $|a+b| \geq \Big||a| - |b|\Big|$:

$$\left|\overbrace{1}^{=a} + \frac{\overbrace{(\tilde{y}-y)}^{=b}}{y}\right| \geq \left||1| - \left|\frac{\tilde{y}-y}{y}\right|\right| = |1-\epsilon_y| = 1-\epsilon_y \iff \epsilon_y < 1$$

da cui si ottiene $|\tilde{y}| \geq |y|(1-\epsilon_y)$ e quindi

$$\frac{|y|}{|\tilde{y}|} \leq \frac{\cancel{|y|}}{\cancel{|y|}(1-\epsilon_y)} = \frac{1+\epsilon_y}{(1+\epsilon_y)(1-\epsilon_y)} = \frac{1+\epsilon_y}{1-\epsilon_y^2} \approx 1+\epsilon_y$$

perché $\epsilon_y^2 \ll \epsilon_y < 1$.
Alla fine otteniamo

$$\epsilon_{\frac{1}{y}} = \epsilon_y \frac{|y|}{|\tilde{y}|} \leq\approx \epsilon_y(1+\epsilon_y) \approx \epsilon_y$$

cioè abbiamo quantificato in modo più preciso la stima qualitativa di prima.

Analisi della somma algebrica

Quello che ci importa per distinguere una addizione da una sottrazione in una somma algebrica non è tanto il segno del risultato, ma quanto più i segni dei due operandi x e y: se hanno lo *stesso segno* si tratta di una **addizione**, se hanno *segno opposto* allora è una **sottrazione**.

$$\begin{aligned}
\epsilon_{x+y} &= \frac{|(x+y)-(\tilde{x}+\tilde{y})|}{|x+y|}, \quad x+y \neq 0 \\
&= \frac{|\overbrace{x-\tilde{x}}^{=a} + \overbrace{y+\tilde{y}}^{=b}|}{|x+y|} \leq \frac{|x-\tilde{x}|}{|x+y|} + \frac{|y-\tilde{y}|}{|x+y|} = \\
&= \frac{|x|}{|x+y|}\frac{|x-\tilde{x}|}{|x|} + \frac{|y|}{|x+y|}\frac{|y-\tilde{y}|}{|y|} = w_1\epsilon_x + w_2\epsilon_y
\end{aligned}$$

dove $w_1 = \dfrac{|x|}{|x+y|}$, $w_2 = \dfrac{|y|}{|x+y|}$.
Abbiamo quindi maggiorato ϵ_{x+y} con una *somma pesata* degli errori sui dati, con pesi w_1 e w_2.
Si noti che questi pesi dipendono da x e da y, ma **non** dipendono dagli errori.

In realta' anche con la moltiplicazione e la divisione siamo arrivati in sostanza a una stima del tipo $\epsilon_{x\star y} \leq w_1\epsilon_x + w_2\epsilon_y$ dove $w_1, w_2 \approx 1$.
Vedremo ora che questa è anche la situazione con l'addizione, mentre le cose possono cambiare radicalmente con la sottrazione.

Addizione ($\text{sgn}(x) = \text{sgn}(y)$)

In questo caso $|x+y| \geq |x|, |y|$ si pensi per semplicita' al caso $x, y > 0$, è chiaro che $x + y > x$ e $x + y > y$.
Quindi

$$w_1 = \frac{|x|}{|x+y|} \leq 1,\ w_2 = \frac{|y|}{|x+y|} \leq 1$$

cioè $\epsilon_{x+y} \leq \epsilon_x + \epsilon_y$ ovver l'addizione è stabile, infatti l'errore relativo sul risultato è maggiorato da una quantita' che è dell'ordine degli errori sui dati.

Sottrazione ($\text{sgn}(x) \neq \text{sgn}(y)$)

In questo caso $|x+y| < |x|$ oppure $|x+y| < |y|$, quindi $\max\{w_1, w_2\} > 1$. Questo ci dice che la sottrazione può far perdere precisione rispetto agli errori sui dati.
Per capire quanto basta analizzare il caso in cui $|x|$ e $|y|$ siano molto vicini in termini relativi, cioè che $|x+y| \ll |x|, |y|$, in queste situazioni $w_1, w_2 \gg 1$ e la sottrazione diventa **instabile**.
Si noti infatti che w_1 e w_2 possono essere arbitrariamente grandi, in quanto dipendenti dai dati.

E' importante comunque osservare che si tratta di un problema di "vicinanza" relativa, non assoluta, tra le dua quantita' che vengono sottratte.
Cioè i casi instabili non sono quelli in cui $|x+y|$ è piccolo, ma quelli in cui è piccolo rispetto a $|x|, |y|$.

Ad esempio sono analoghi i seguenti

$$w_1, w_2 \approx 10^6 \leftarrow \begin{cases} |x|, |y| \approx 1, |x+y| \approx 10^{-6} \\ |x|, |y| \approx 10^6, |x+y| \approx 1 \end{cases}$$

Detto a parole, due numeri dell'ordine delle unita' che distano tra loro qualche milionesimo, sono altrettanto vicini, in termini relativi, di due numeri dell'ordine del milione che distano qualche unita' tra di loro.
In entrambi i casi i pesi w_1, w_2 sono fattori di amplificazione dell'errore dell'ordine di 10^6.
Possiamo sintetizzare che la sottrazione è *potenzialmente instabile*, infatti se $|x|$ e $|y|$ sono distanti in termini relativi, la sottrazione perdera' poca precisione.
Se invece sono vicini perdera' molta precisione, tanta più quanto più sono vicini.

Questo fenomento, che si chiama anche *cancellazione numerica*, è il primo e importante esempio di possibile instabilita' di un algoritmo. E' un fenomeno che generalmente si può affrontare in 2 modi:

1. cercando di riscrivere le espressioni e gli algoritmi in modo da evitare sottrazioni instabili, che si vede ad esempio con la formula risolutiva per le equazioni di secondo grado);

2. aumentando la precisione, cioè diminuendo l'errore sui dati, in funzione della grandezza dei pesi.

In campo sperimentale, questo significa aumentare la precisione dello strumento di misura. Nel caso dell'arrotondamento, questo significa andare in un sistema floating-point a precisione estesa, se ad esempio i pesi sono così grandi da mettere in crisi un sistema a precisione doppia.
Come gia' ossevato però, usare precisioni estese può avere un costo computazionale molto elevato in termini di tempo di calcolo e di occupazione di memoria.

Per fissare le idee, concludiamo il capitolo con un esempio concreto di come la sottrazione può portare ad una perdita significativa di precisione:

$$f(x) = \frac{(1+x)-1}{x}, \quad x \neq 0$$

è evidente che $f(x) = 1$. Proviamo però a calcolarlo usando *Matlab* ad esempio, ricordando che esso lavora con un sistema $\mathbb{F}(10, 16, -307, 308)$:

$$f(10^{-15}) = 1.11\ldots$$

cioè l'errore relativo sul risultato è $> 11\%$, che è un errore enorme considerando che abbiamo una precisione macchina $\epsilon_M \approx 10^{-16}$.
Vedremo che la spiegazione di questo fatto sta nella sottrazione a numeratore, visto che $1 + 10^{-15} \approx 1$.
Invece si può notare come $f(2^{-50}) = 1$, pur essendo $2^{-50} \approx 10^{-15}$.