Calcolo numerico

calcolo numerico → lo usiamo per risolvere problemi di tipo matematico.
usa:
- derivate
- integrali
- vettori
- matrici

Manipoliamo numeri facendo uso di una macchina che esegue per noi conti lunghi.

Osservazione di un fenomeno naturale → modellazione del fenomeno → uso del calcolo numerico con la matematica

A seconda di come organizziamo le operazioni con Matlab e risultato cambia; qual'è allora il modo migliore per organizzare le operazioni?

L'aritmetica della macchina

MASTER COPY
Tel. 388/9837745

- La macchina rappresenta i numeri in base 2

BASE = numero naturale > 1

(la macchina considera i numeri complessi come una coppia di numeri $\mathbb{R}$)

Supponiamo di avere un numero reale $x \in \mathbb{R}$

- Se $x \neq 0$ possiamo rappresentare il numero x in questo modo

$$x = \pm B^p (0.d_1 \dots d_t)_B = \pm B^p (d_1 B^{-1} + d_2 B^{-2} + \dots + d_t B^{-t})_B \quad p \in \mathbb{Z}$$

es: $10^{1} \cdot 0,0125$ — B indica la base

esempio: 0,125 in base 10 lo scriviamo come

$0,125 = 1 \cdot 10^{-1} + 2 \cdot 10^{-2} + 5 \cdot 10^{-3} = 10^{1} \cdot 0,0125$

→ posso rappresentarlo in varie maniere;

Per avere invece l'unicità della rappresentazione, impongo come condizione che la prima cifra sia non nulla, cioè $d_1 \neq 0$.
Stiamo considerando una RAPPRESENTAZIONE NORMALIZZATA.

A questo punto la rappresentazione non è unica, ma possiamo ottenere sostanzialmente lo sia.
I numeri periodici, ad esempio hanno una rappresentazione infinita

$0,\overline{9} = 1$
↓ → $10^{1}(0,1)$
$10^{0}(0,99999\dots)$

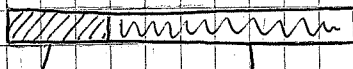
Entrambe le rappresentazioni vanno bene per i nostri scopi perché la prima cifra è $\neq 0$

In macchina per rappresentare numeri abbiamo dei REGISTRI di LUNGHEZZA FINITA.
Rappresentare numeri di lunghezza infinita come i periodici diventa impossibile!
Passando alla rappresentazione in macchina è allora inevitabile introdurre delle approssimazioni.

COSTRUIAMO UNA RAPPRESENTAZIONE IN MACCHINA.

Supponiamo di lavorare in base 2 e di avere un registro lungo 32 bit per rappresentare il numero.

$p \in \mathbb{Z}$ (è un intero con segno) → perché se ad esempio voglio rappresentare il numero 0,0125 ho bisogno di scrivere $10^{-1} \cdot (0,125)$



un pezzo lo usiamo x l'esponente 8 bit

un pezzo lo usiamo per rappresentare la sequenza di cifre (mantissa) 24 bit (in ogni bit ci va una cifra)

MASTER COPY
Tel. 388/9837745

8 bit → il primo bit mi serve per il segno
→ gli altri 7 mi avanzano per rappresentare l'esponente.
$p = 1 \cdot 2^0 + 1 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3 + 1 \cdot 2^4 + 1 \cdot 2^5 + 1 \cdot 2^6 = 2^7 - 1 = 127$
L'esponente p deve essere compreso tra -127 e 127.

24 bit → il primo bit mi serve per il segno
→ gli altri 23;

Posso rappresentare quindi in macchina i numeri

$$\tilde{x} = \pm 2^p \sum_{i=1}^{23} d_i \cdot 2^{-i}$$ dove $-127 \leq p \leq 127$

$d_i \in \{0, 1\}$

e $d_1 = 1$ perché voglio avere una rappresentazione normalizzata.

Sono tutti i numeri che posso rappresentare con la macchina

Se il numero da rappresentare non è tra questi non può essere rappresentato

---

Lo zero non può essere rappresentato in questo modo perché nella rappresentazione normalizzata abbiamo bisogno di $d_1 \neq 0$ e quindi lo zero non lo prendiamo mai!
Ma tutti i calcolatori assumono di poter rappresentare lo zero ma questa è una rappresentazione interna, non di questo tipo.

L'INSIEME DEI NUMERI DI MACCHINA:

$$F(\underbrace{2}_{\text{base}}, m, M, \overbrace{t}^{\text{cifre della mantissa che riesco a rappresentare}}) = \left\{ x \in \mathbb{R},\ x = \pm 2^p \sum_{i=1}^{t} d_i 2^{-i},\ d_1 = 1,\ d_i \in \{0,1\} \right\} \cup \{0\}$$

$m = -127$
$M = 127$

↑ tutti i numeri rappresentabili

in generale abbiamo:

$$F(B, m, M, t) = \left\{ x \in \mathbb{R} : x = B^p \sum_{i=1}^{t} d_i B^{-i},\ d_1 \neq 0,\ \underbrace{0 \leq d_i \leq B-1}_{\text{perché le cifre in una base B possono essere tutte fino a B esclusa!}} \right\} \cup \{0\}$$

$-m \leq p \leq M$

- L'insieme dei numeri di macchina è un insieme finito.
- IL PIÙ GRANDE E IL PIÙ PICCOLO NUMERO RAPPRESENTABILE

$$\Omega = B^M \sum_{i=1}^{t} (B-1) \cdot B^{-i}$$

ho dato all'esponente il valore più grande e alle cifre il valore più grande

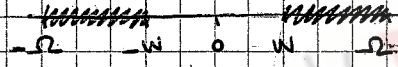
$$w = B^{-m} \cdot B^{-1}$$

nota: non possiamo prendere lo zero perché vogliamo una rapp. normalizzata

esponente più piccola e il valore più piccolo per le cifre che è 1

è il più piccolo numero rappresentabile, se escludiamo lo zero

NUMERI RAPPRESENTABILI IN MACCHINA:



MASTER COPY

Tel. 388/9837745

Come faccio ad associare a un numero reale x la sua rappresentazione in macchina $\tilde{x}$, che cosa succede?

$x \in \mathbb{R} \rightsquigarrow \tilde{x} \in F(B, m, M, t)$ CHE TIPO DI ERRORI SI VANNO A GENERARE?

Le singole operazioni vengono fatte su registri più lunghi di quelli che vengono usati per la memorizzazione.
La macchina si domanda ① $x = 0$? Se la risposta è NO →

② $|x| \in [w, \Omega]$? SI → la macchina comincia a lavorare *

NO → la macchina fornisce un messaggio di errore

- Se $|x| < w$ messaggio di UNDERFLOW:
  (hai inserito un numero troppo piccolo)
  di fronte a un messaggio di underflow:
  - alcuni sistemi si bloccano
  - altri associano a tale numero troppo piccolo $\ddot{y}$ e continuano a lavorare
- Se $|x| > \Omega$ messaggio di OVERFLOW:
  (le numero è troppo grande, non riesco a maneggiarlo)
  - il programma termina in modo non corretto.

* Il fatto che stia nell'insieme significa che il numero reale ha una rappresentazione in macchina $\tilde{x}$

$$x = \pm B^p \sum_{i=1}^{\infty} d_i B^{-i} \rightsquigarrow \tilde{x} = \pm B^p \sum_{i=1}^{t} d_i B^{-i}$$

$-m \leq p \leq M$

∞ perché la somma è infinita se ho un numero non finito.

Se la mantissa ha troppe cifre al numero x associo $\tilde{x}$ e $\tilde{x}$ lo ottengo per **TRONCAMENTO**.
Ovviamente, nel momento in cui faccio questa approssimazione, commetto un errore, $\tilde{x}$ è un dato perturbato rispetto a quello inserito x.

Commetto un **ERRORE di RAPPRESENTAZIONE in MACCHINA**:
(se $x \neq 0$)

$$\varepsilon_x = \frac{\tilde{x} - x}{x}$$

l'errore considerato è un errore relativo, non ci interessa in realtà quanto dista $\tilde{x}$ da x.

Se x esempio introduco

0,1 e la macchina fa un errore assoluto di 0,1 → (0,1 ± 0,1) faccio un errore relativo del 100%.!
Se invece inserisco 10^9 e la macchina commette un errore assoluto di 0,1 → $(10^9 \pm 0{,}1)$, l'errore relativo è piccolissimo.

I numeri in macchina non sono distribuiti in modo uniforme negli intervalli: sono più radi vicino a Ω e più frequenti vicino a ω.
Su numeri grandi sono disposta ad avere errori grandi, su numeri piccoli voglio errori piccoli: Per questo motivo consideriamo l'errore relativo.

Cerchiamo di stimare l'errore, lo vogliamo maggiorare:

$$\varepsilon_x = \left| \frac{\tilde{x} - x}{x} \right|$$

voglio maggiorare questa frazione: posso aumentare il numeratore o diminuire il denominatore. Farò entrambe le cose.

DEN

$$|x| = B^p \cdot \sum_{i=1}^{\infty} d_i B^{-i}$$

questo oggetto è $\geq B^p \cdot B^{-1}$ (il valore più piccolo che d_1 può assumere è 1)

NUM

$$|\tilde{x} - x| = \left| B^p \left(\sum_{i=1}^{t} d_i B^{-i} - \sum_{i=1}^{\infty} d_i B^{-i} \right) \right| = B^p \cdot \sum_{i=t+1}^{\infty} d_i B^{-i}$$

esempio:

$$10^2 (0{,}1 - 0{,}12345\ldots) = 10^2 (0{,}02345\ldots)$$

($0{,}1 = \tilde{x}$, $0{,}12345\ldots = x$)

MASTER COPY
Tel. 388/9837745

$$= B^p \cdot B^{-t} \sum_{i=1}^{\infty} d_{t+1} B^{-i} \leq B^{p-t}$$

$d_1 \cdot B^{-1} + d_2 \cdot B^{-2} + d_3 \cdot B^{-3}$ è un numero minore di 1

metto in evidenza un B^{-t} e ho cambiato l'indice delle cifre

PRECISIONE DI MACCHINA

$$|\varepsilon_x| = \left|\frac{\tilde{x} - x}{x}\right| \leq \frac{B^{p-t}}{B^p \cdot B^{-1}} = B^{1-t} = u \quad \text{(in matlab è eps)}$$

(precisione di macchina di una macchina che opera in base B con t cifre della mantissa operando per troncamento)

non dipende da x ma dalle caratteristiche della macchina (base scelta e n° di cifre usato)

MASTER COPY

Tel. 388/9837745

$$\frac{\tilde{x} - x}{x} = \varepsilon_x \iff \tilde{x} = x(1 + \varepsilon_x) \qquad |\varepsilon_x| \leq u$$

moltiplico per x

RELAZIONE FONDAMENTALE vale per ogni numero di macchina

Come fare le operazioni?

$+, -, \times, :$

la macchina riesce a fare solo le operazioni aritmetiche ed eventualmente i confronti.
Le altre operazioni (funzioni) sono approssimazioni delle operazioni aritmetiche.

Abbiamo due numeri di macchina x e y:

$x \in F(B, m, M, t)$

$y \in F(B, m, M, t)$

Vogliamo definire l'operazione $x + y$ in macchina. (vale la solita cosa anche per le altre 3 operazioni aritmetiche)
La macchina non ci restituisce il valore $x + y$ perché $x + y$ potrebbe non essere un numero di macchina
L'aritmetica di macchina è stata allora progettata in modo tale che non fornisca il valore di $x + y$ ma il suo troncamento.
La macchina sputa il troncamento del valore esatto

$$z = \underbrace{trn(x+y)}_{\text{è il risultato fornito dalla macchina}} = (x+y)(1+\varepsilon) \quad \text{dove } |\varepsilon| \leq u$$

errore locale dell'operazione è minore uguale della precisione di macchina