Laboratorio computazionale

Appunti lezione 1

Stefano Franceschina

05/03/2025

1 Introduzione

L'Analisi numerica è una branca della matematica che si occupa di trovare soluzioni approssimate a problemi matematici, in cui gli errori computazionali e di modellazione giocano un ruolo fondamentale. Le principali fonti di errore sono:

- **Errori di rounding**: derivano dall'approssimazione necessaria perché il computer rappresenta i numeri reali con una precisione limitata.
- **Errori di approssimazione**: dipendono dal tipo di problema e dall'algoritmo utilizzato. Ad esempio, nel calcolo dell'esponenziale tramite la somma della sua espansione in serie, è necessario un troncamento, introducendo un errore dovuto al tralasciare gli infiniti termini successivi. Tale errore viene indicato con O.

2 Rappresentazione dell'esponenziale

Un esempio di errore di approssimazione è il calcolo dell'esponenziale tramite la sua espansione in serie di Taylor:

$$\exp(x) = \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \dots$$

Per x molto piccoli, ad esempio per $x \in [0, 1]$ la serie converge e l'errore di approssimazione è trascurabile. Tuttavia, per x grandi, la serie diverge e perciò non è possibile utilizzarla per la rappresentazione dell'esponenziale. Per ovviare a questo problema, si possono adottare diverse strategie, come la riscrittura dell'esponenziale o l'utilizzo di metodi iterativi più stabili. Vediamo come riscrivere l'esponenziale. Si può esprimere $\exp(x)$ come:

$$\exp(x) = \exp(x_1) \cdot \exp(k \cdot \log 2) = \exp(x_1) \cdot 2^k,$$

ovvero, riscrivendo x come $x_1 + k \log 2$, il calcolatore lavora sempre con un x_1 compreso tra 0 e 1, migliorando così la precisione nel calcolo.

3 Rappresentazione dei Numeri

Prima dell'avvento dei computer è stato fondamentale comprendere come rappresentare un numero. Si sceglie una base B e ogni numero viene scritto come una sommatoria di coefficienti (da 0 a $B - 1$) moltiplicati per potenze di B.

$$x = \pm \sum_{i=-n}^{m} b_i \cdot B^i$$

I computer, per motivi hardware, adottano la base binaria. Questo perché è più semplice rappresentare due stati fisici (ad esempio, alto o basso livello di tensione) rispetto a rappresentare dieci stati differenti. Inizialmente si utilizzava la cosiddetta *fixed-point representation*, in cui si fissavano valori di n e m e si rappresentavano i numeri come:

$$x = \pm \sum_{i=-n}^{m} b_i \cdot 2^i$$

Questo sistema ha però un range limitato e non permette di rappresentare numeri molto grandi o molto piccoli, in particolare numeri $|x| \geq B^m$ o $|x| < B^{-n}$.

3.1 Floating Point Representation

Per rappresentare i numeri reali si è passati dalla *fixed-point representation* (con un numero fisso di cifre decimali o binarie prima e dopo la virgola) alla

floating point representation, che consente di rappresentare un range molto più ampio. L'idea è di scrivere un numero come prodotto di tre fattori: segno, mantissa e esponente. In generale ci si basa sulla scomposizione:

$$x = \pm a \times B^b$$

Indicando con b il punto in cui si trova la virgola rispetto ad a. Ad esempio con $a = 0.123$ e $b = -2$ si ha, in base 10, $x = 0.00123$. In base 2, si ha $x = 0.0101 = 1.01 \times 2^{-2}$, e in entrambi i casi $b = -2$ sposta la virgola di due posti verso sinistra.

All'interno di un calcolatore si sfrutta il *sistema a virgola mobile* e perciò x si scrive come:

$$x = (-1)^s(1 + f) \cdot 2^b, \tag{1}$$

dove:

- s rappresenta il segno (0 per numeri positivi e 1 per numeri negativi),
- f è la mantissa, che rappresenta la parte frazionaria,
- b è l'esponente.

Sia f che b vengono rappresentati in base binaria in questo modo:

$$f = \sum_{i=0}^{d} b_i \cdot 2^{-i}, \quad b = \sum_{i=0}^{n-1} a_i \cdot 2^i - b_{offset},$$

dove d e n sono alcuni interi fissi che determinano la precisione della rappresentazione, mentre $b_{offset} = 2^{n-1}-1$ è un offset convenzionale per l'esponente. Nota in particolare che l'esponente b può essere negativo. Ricordati che con quella rappresentazione sia f che b rappresentabili come delle sequenze di 0 e 1 e che la posizione che occupa la cifra è determinata dalla potenza di 2 corrispondente.

Dato che f per definizione è una somma di potenze negative di 2, sappiamo che è un numero compreso tra 0 e 1. Ciò implica che $1 + f$ sia un numero compreso tra 1 e 2. Se riscriviamo f in questa maniera:

$$f = \sum_{i=0}^{d} b_i \cdot 2^{-i} = 2^{-d} \sum_{i=0}^{d} b_i \cdot 2^{d-i} = 2^{-d} z, \text{ con } z = [0, 1, 2, ..., 2^d - 1].$$

ci accorgiamo che le uniche possibilità per z di mantenere che $1 + f$ sia compreso tra 1 e 2 sono $z = 0, 1, ..., 2^d - 1$. In tal modo vediamo che gli z ammessi sono esattamente 2^d, cioè $1 + f$ ammette un numero finito di valori, pari a 2^d. Inserendo tale risultato in (1) otteniamo che le x corrono tra $[2^b, 2^b + 1)$, estremi ottenuti inserendo i valori estremi di z, e cioè di f. In più otteniamo solamente certi valori, cioè 2^d valori, perchè gli z sono finiti, per di più equispaziati.

Il valore 2^d viene definito come *precisione macchina* (ϵ_{mach}) e rappresenta il numero di valori rappresentati in un intervallo $[2^b, 2^b + 1)$.

3.2 Mappa $\mathbb{R} \to \mathbb{F}$

Scopo dell'informatica è quindi stabilire una mappa tra i numeri reali (insieme $\mathbb{R}$) e i numeri macchina (insieme $\mathbb{F}$), in modo da rappresentare il maggior numero possibile di numeri reali con un numero finito di numeri macchina. La mappa tra i numeri reali e i numeri macchina è una funzione che associa a ogni numero reale un numero macchina. Questa mappa è una funzione a gradini, in quanto i numeri macchina sono finiti e quindi la mappa non è iniettiva. Inoltre, la mappa non è suriettiva, in quanto i numeri reali sono infiniti e i numeri macchina sono finiti. Questo porta a un errore di rappresentazione, che è l'errore di approssimazione. Di certo dovremo chiedere che, definendo la funzione $fl : \mathbb{R} \to \mathbb{F}$ per un numero $x \notin \mathbb{F}$ si abbia:

$$|fl(x) - x| \leq |fl(y) - g| \quad \forall g \in \mathbb{F}.$$

In tal modo la distanza tra l'approssimazione $fl(x)$ e il numero reale x è più piccola di quella tra x stesso e qualsiasi altro numero macchina g. Sapendo che in un intervallo $[2^b, 2^b + 1)$ ci sono 2^d numeri macchina, possiamo dire che la distanza tra due numeri macchina successivi è 2^{b-d} e quindi qualunque numero reale x in $[2^b, 2^b + 1)$ avrà una distanza da $fl(x)$ al massimo di $\frac{2^{b-d}}{2}$. Da qui si possono trarre conclusioni sull'approssimazione. Guarda le dispense per ulteriori dettagli..

3.3 Standard di Precisione e IEEE 754

Gli standard più comuni sono:

- **Double Precision**: 64 bit totali, suddivisi in 1 bit per il segno, 11 bit per l'esponente e 52 bit per la mantissa. La precisione macchina, ovvero il più piccolo incremento rappresentabile, è 2^{-52}.

- **Single Precision**: 32 bit totali, con 1 bit per il segno, 8 bit per l'esponente e 23 bit per la mantissa, e una precisione macchina pari a 2^{-23}.

Lo standard IEEE 754 definisce non solo la rappresentazione ma anche le regole di arrotondamento (tipicamente "round to nearest") e il comportamento in presenza di eccezioni, come i numeri denormalizzati (che permettono di rappresentare numeri estremamente piccoli in modo graduale), infiniti e NaN (Not a Number). Le dispense offrono un'analisi approfondita di questi concetti, evidenziando come il design di IEEE 754 contribuisca a ridurre gli errori cumulativi nei calcoli numerici.

4 Conclusioni

Questa lezione offre una panoramica della rappresentazione numerica nei computer e degli errori che ne derivano. Le dispense, che approfondiscono gli aspetti teorici e pratici, rappresentano un ottimo strumento per approfondire la conoscenza degli algoritmi numerici e per capire come ridurre e gestire gli errori computazionali. Studi approfonditi permettono di affrontare problemi complessi con strumenti matematici e computazionali affidabili. La cosa più importante da tenere a mente è la relazione

$$\frac{|fl(x) - x|}{|(|x)} \leq \frac{\epsilon_{mach}}{2}.$$

e cioè che la precisione relativa è sempre la stessa, limitata dalla precisione macchina. L'altra relazione importante è

$$fl(x) = x(1 + \epsilon_x), \quad \epsilon_x = \frac{f' - f}{1 + f}$$