Introduzione al Calcolo Scientifico

Rossana Vermiglio

CD Lab CDLab
Dipartimento di Scienze Matematiche, Informatiche e Fisiche
Università degli Studi di Udine

Gruppo UMI *Modellistica Socio-Epidemiologica*

- **Materiale didattico**:
  - Slides delle lezioni, temi d'esame, note, esercizi svolti disponibili in rete elearning.uniud.it

- **Materiale didattico**:
  - Slides delle lezioni, temi d'esame, note, esercizi svolti disponibili in rete elearning.uniud.it
- **Testi di consultazione**:
  - Quarteroni, A., Saleri, F.: *Introduzione al Calcolo Scientifico*, Springer, Milano.Terza Edizione 2006.
  - Moler, C.: *Numerical Computing with MATLAB* libro on line http://www.mathworks.com/moler/

- **Materiale didattico**:
  - Slides delle lezioni, temi d'esame, note, esercizi svolti disponibili in rete elearning.uniud.it
- **Testi di consultazione**:
  - Quarteroni, A., Saleri, F.: *Introduzione al Calcolo Scientifico*, Springer, Milano.Terza Edizione 2006.
  - Moler, C.: *Numerical Computing with MATLAB* libro on line http://www.mathworks.com/moler/
- **MATLAB**
  - Moler, C.: *Numerical Computing with MATLAB* libro on line http://www.mathworks.com/moler/intro.pdf

Analisi Numerica $\rightarrow$ Calcolo Scientifico

L'**analisi numerica** si occupa dello sviluppo e dell'analisi di **algoritmi** per la risoluzione di problemi della **matematica del continuo** (problemi che coinvolgono variabili reali e complesse) che si incontrano nelle scienze computazionali e nell'ingegneria. Per questo motivo più recentemente è stato proposto anche il nome **calcolo scientifico**.

Tale disciplina ricerca degli algoritmi **accurati** e **veloci**, che **convergano rapidamente** a una soluzione approssimata del problema.

Gli algoritmi sono implementati su un **calcolatore**. Pertanto anche l'ambiente computazionale (hardware, sistemi operativi, linguaggi) gioca un ruolo non trascurabile nell'accuratezza della risposta finale.

Esempi di applicazioni all'Informatica

- Grafica
- Elaborazione di dati e immagini
- Motori di ricerca
- *Machine Learning*
- ...

Esempio: le radici di un polinomio

In molti problemi della matematica del continuo, l'approssimazione non si può evitare, perchè le quantità coinvolte non sono calcolabili con una formula matematica esplicita, che richiede una sequenza **finita** di operazioni elementari.

Problem

Calcolare gli zeri di un polinomio di grado n:

$$p_n(x) = a_0 + a_1 x + ... + a_n x^n = 0$$

.

È noto che non esiste una formula esplicita per il calcolo delle radici se il grado $n \geqslant 5$ (Ruffini e Abel).

Uno scienziato che desidera risolvere questo problema, utilizza il calcolatore e ottiene rapidamente una risposta con una precisione anche di 15 cifre decimali!! Il calcolatore sicuramente non usa una formula esplicita ma un algoritmo accurato e veloce.

Esempio: sistema lineare di equazioni

Anche quando esiste una formula che in principio può risolvere il problema con una sequenza finita di operazioni elementari, in pratica può non essere utilizzabile sul calcolatore

Problem

Calcolare $x_1, x_2, ..., x_n$ *che sono soluzione del sistema lineare di* n *equazioni*

$$\left\{ \; a_{i,1}x_1 + a_{i,2}x_2 + ... + a_{i,n}x_n = b_i, \quad i = 1, ..., n \right.$$

Esiste una formula risolutiva (**regola di Cramer**) che richiede un numero di operazioni dell'ordine di $n!$. In pratica non è utilizzabile in pratica sul calcolatore quando n risulta dell'ordine del centinaio o migliaio.

La regola di Cramer

La soluzione di un sistema lineare può essere teoricamente ottenuta con la formula

$$x_i = \frac{\det(A_i)}{\det(A)}, \quad i = 1, \dots, n$$

dove A_i è la matrice ottenuta sostituendo la colonna i-esima di A con il temine noto b.

Il **costo computazionale** di tale algoritmo è dell'ordine di $O((n+1)!)$.

In pratica non è utilizzabile. Su un calcolatore in grado di eseguire

- 1 mega-flops (10^6) $\rightarrow 1.6 \times 10^6$ **anni** per risolvere un sistema con $n = 20$
- 10^{15} flops$\rightarrow$ 20 **anni** per risolvere un sistema con $n = 24$

Anche aumentando la potenza di calcolo l'algoritmo di Cramer è inadeguato alla risoluzione di un sistema lineare e pertanto tale problema matematico va affrontato sviluppando altri tecniche di approssimazione.

Complessità computazionale

L'analisi di un algoritmo per la risoluzione di un problema richiede lo studio non solo della sua **accuratezza**, ma anche della sua **complessità computazionale**.

Tale parametro misura il numero delle operazioni aritmetiche richieste in funzione della dimensione del problema considerato e fornisce una stima del tempo di esecuzione dell'algoritmo.

Parametri valutazione qualità del software matematico

- **affidabilità** il codice svolge in maniera soddisfacente le funzioni richieste;

Parametri valutazione qualità del software matematico

- **affidabilità** il codice svolge in maniera soddisfacente le funzioni richieste;
- **accuratezza** il codice produce risultati accurati in relazione al problema e ai dati di ingresso e fornisce una stima dell'accuratezza della risposta;

Parametri valutazione qualità del software matematico

- **affidabilità** il codice svolge in maniera soddisfacente le funzioni richieste;
- **accuratezza** il codice produce risultati accurati in relazione al problema e ai dati di ingresso e fornisce una stima dell'accuratezza della risposta;
- **efficienza** il codice gestisce al meglio e senza sprechi le risorse di memoria e minimizza il tempo di esecuzione;

Parametri valutazione qualità del software matematico

- **affidabilità** il codice svolge in maniera soddisfacente le funzioni richieste;
- **accuratezza** il codice produce risultati accurati in relazione al problema e ai dati di ingresso e fornisce una stima dell'accuratezza della risposta;
- **efficienza** il codice gestisce al meglio e senza sprechi le risorse di memoria e minimizza il tempo di esecuzione;
- **flessibilità-applicabilità** il codice permette la risoluzione di un'ampia classe di problemi;

Parametri valutazione qualità del software matematico

- **affidabilità** il codice svolge in maniera soddisfacente le funzioni richieste;
- **accuratezza** il codice produce risultati accurati in relazione al problema e ai dati di ingresso e fornisce una stima dell'accuratezza della risposta;
- **efficienza** il codice gestisce al meglio e senza sprechi le risorse di memoria e minimizza il tempo di esecuzione;
- **flessibilità-applicabilità** il codice permette la risoluzione di un'ampia classe di problemi;
- **mantenibilità** il codice è facilmente modificabile;

Parametri valutazione qualità del software matematico

- **affidabilità** il codice svolge in maniera soddisfacente le funzioni richieste;
- **accuratezza** il codice produce risultati accurati in relazione al problema e ai dati di ingresso e fornisce una stima dell'accuratezza della risposta;
- **efficienza** il codice gestisce al meglio e senza sprechi le risorse di memoria e minimizza il tempo di esecuzione;
- **flessibilità-applicabilità** il codice permette la risoluzione di un'ampia classe di problemi;
- **mantenibilità** il codice è facilmente modificabile;
- **portabilità** il codice si adatta facilmente a diversi ambienti di calcolo.

Parametri valutazione qualità del software matematico

- **affidabilità** il codice svolge in maniera soddisfacente le funzioni richieste;
- **accuratezza** il codice produce risultati accurati in relazione al problema e ai dati di ingresso e fornisce una stima dell'accuratezza della risposta;
- **efficienza** il codice gestisce al meglio e senza sprechi le risorse di memoria e minimizza il tempo di esecuzione;
- **flessibilità-applicabilità** il codice permette la risoluzione di un'ampia classe di problemi;
- **mantenibilità** il codice è facilmente modificabile;
- **portabilità** il codice si adatta facilmente a diversi ambienti di calcolo.
- **robustezza** il codice tratta problemi di difficile soluzione, verificando se i dati in ingresso sono consistenti con le specifiche, analizzando e segnalando gli errori in caso di fallimento;

Parametri valutazione qualità del software matematico

- **affidabilità** il codice svolge in maniera soddisfacente le funzioni richieste;
- **accuratezza** il codice produce risultati accurati in relazione al problema e ai dati di ingresso e fornisce una stima dell'accuratezza della risposta;
- **efficienza** il codice gestisce al meglio e senza sprechi le risorse di memoria e minimizza il tempo di esecuzione;
- **flessibilità-applicabilità** il codice permette la risoluzione di un'ampia classe di problemi;
- **mantenibilità** il codice è facilmente modificabile;
- **portabilità** il codice si adatta facilmente a diversi ambienti di calcolo.
- **robustezza** il codice tratta problemi di difficile soluzione, verificando se i dati in ingresso sono consistenti con le specifiche, analizzando e segnalando gli errori in caso di fallimento;
- **testabilità** la correttezza e l'efficienza del codice sono verificabili su esempi test;

Parametri valutazione qualità del software matematico

- **affidabilità** il codice svolge in maniera soddisfacente le funzioni richieste;
- **accuratezza** il codice produce risultati accurati in relazione al problema e ai dati di ingresso e fornisce una stima dell'accuratezza della risposta;
- **efficienza** il codice gestisce al meglio e senza sprechi le risorse di memoria e minimizza il tempo di esecuzione;
- **flessibilità-applicabilità** il codice permette la risoluzione di un'ampia classe di problemi;
- **mantenibilità** il codice è facilmente modificabile;
- **portabilità** il codice si adatta facilmente a diversi ambienti di calcolo.
- **robustezza** il codice tratta problemi di difficile soluzione, verificando se i dati in ingresso sono consistenti con le specifiche, analizzando e segnalando gli errori in caso di fallimento;
- **testabilità** la correttezza e l'efficienza del codice sono verificabili su esempi test;
- **usabilità** il codice ha un'interfaccia utente di facile comprensione e ben documentata.

Linee guida

Una strategia generale per risolvere un problema matematico consiste nel sostituire al problema **difficile** uno più **semplice**, che sappiamo risolvere e che fornisce una soluzione *"vicina"* a quella originale secondo lo schema:

- non lineare $\rightsquigarrow$ lineare
- processo infinito $\rightsquigarrow$ processo finito
- funzioni *"complicate"* $\rightsquigarrow$ funzioni *"semplici"*
- matrici generali $\rightsquigarrow$ matrici con semplice struttura

Poiché la soluzione ottenuta è un'approssimazione di quella originale, è essenziale **analizzare gli errori** introdotti.

Il procedimento risolutivo richiede diverse fasi, ognuna delle quali introduce delle semplificazioni e quindi degli errori. Pertanto per valutare l'accuratezza della risposta finale (**attendibilità del risultato**), bisogna analizzare le "sorgenti d'errore" o "fonti d'incertezza" e i relativi errori.

Fonti di incertezza

Prima della computazione

- formulazione del modello matematico = insieme di leggi di natura matematica che descrivono il fenomeno da analizzare $\rightsquigarrow$ **adeguatezza del modello**
- formulazione del modello numerico = algoritmo per approssimare il modello matematico $\rightsquigarrow$ **errore analitico o di troncamento**
- misurazioni sperimentali o computazioni precedenti $\rightsquigarrow$ **errore nei dati di input**

Fonti di incertezza

Prima della computazione

- formulazione del modello matematico = insieme di leggi di natura matematica che descrivono il fenomeno da analizzare $\rightsquigarrow$ **adeguatezza del modello**
- formulazione del modello numerico = algoritmo per approssimare il modello matematico $\rightsquigarrow$ **errore analitico o di troncamento**
- misurazioni sperimentali o computazioni precedenti $\rightsquigarrow$ **errore nei dati di input**

Durante la computazione:

- strumento di calcolo $\rightsquigarrow$

$$\textbf{errori di arrontondamento} = \begin{cases} \text{errori di } \textbf{rappresentazione} \text{ numeri reali} \\ \text{errori nelle } \textbf{operazioni} \text{ aritmetiche} \end{cases}$$

Fonti di incertezza

Prima della computazione

- formulazione del modello matematico = insieme di leggi di natura matematica che descrivono il fenomeno da analizzare $\rightsquigarrow$ **adeguatezza del modello**
- formulazione del modello numerico = algoritmo per approssimare il modello matematico $\rightsquigarrow$ **errore analitico o di troncamento**
- misurazioni sperimentali o computazioni precedenti $\rightsquigarrow$ **errore nei dati di input**

Durante la computazione:

- strumento di calcolo $\rightsquigarrow$

$$\textbf{errori di arrontondamento} = \begin{cases} \text{errori di } \textbf{rappresentazione} \text{ numeri reali} \\ \text{errori nelle } \textbf{operazioni} \text{ aritmetiche} \end{cases}$$

L'accuratezza del risultato finale dipende dalla combinazione degli errori introdotti nei vari passi, che possono essere amplificati dalla natura del problema trattato e/o dall'algoritmo scelto.

Esempio: calcolo superficie della terra

Il calcolo della superficie S della terra mediante la formula

$$S = 4 * \pi * r^2$$

coinvolge le seguenti approssimazioni

- la terra viene modellata come una sfera, che idealizza la sua vera forma $\rightsquigarrow$ adeguatezza del modello matematico
- il valore del raggio terrestre r si basa su misurazioni empiriche e su altre computazioni $(6378160 - 6356778 \text{ Km})$ $\rightsquigarrow$ errore nel dato
- il valore di π richiede una tecnica di approssimazione e il troncamento di un processo infinito $\rightsquigarrow$ errore nel dato
- i dati e i risultati delle operazioni aritmetiche sono arrotondati sul calcolatore $\rightsquigarrow$ errori di arrotondamento

Errore assoluto e relativo

Come misurare gli errori?

Sia $\tilde{x}$ un'aprossimazione del numero reale x.

Definition

$$e_x = |\tilde{x} - x| \quad \textbf{errore assoluto}$$

mentre

$$\epsilon_x = \frac{e_x}{|x|}, \quad x \neq 0 \quad \textbf{errore relativo}$$

Una definizione equivalente di errore relativo è la seguente

$$\tilde{x} = x(1 + \delta), \quad \epsilon_x = |\delta|.$$

Nelle approssimazioni scientifiche, dove le quantità in gioco possono variare molto in grandezza, è opportuno misurare l'errore relativo, perchè non dipende dall'ordine di grandezza del dato.

Esempio

Consideriamo $x = 0.12345 \times 10^e$ e sia $\tilde{x} = 0.1234\mathbf{6} \times 10^e$. Otteniamo

$$e_x = |\tilde{x} - x| = 0.00001 \times 10^e = 0.1 \times 10^{e-4}$$

mentre

$$\epsilon_x = \frac{e_x}{|x|} = \frac{0.1 \times 10^{e-4}}{0.12345 \times 10^e} \approx 0.81 \times 10^{-4}.$$

Errore assoluto e relativo: vettori

Se $x = (x_1, \ldots, x_n)$ e $\tilde{x} = (\tilde{x}_1, \ldots, \tilde{x}_n)$, gli errori assoluti e relativi si definiscono usando le **norme vettoriali** $\|\cdot\|$

Definition

$$e_x = \|\tilde{x} - x\| \quad \textbf{errore assoluto}$$

mentre

$$\epsilon_x = \frac{e_x}{\|x\|}, \quad x \neq 0 \quad \textbf{errore relativo}$$

In alcuni contesti può essere utile stimare **l'errore relativo componente per componente**, valutando $\max_{i=1,\ldots,n} \epsilon_{x_i}$.

Remark

Solitamente il valore esatto x è sconosciuto, allora si cerca una stima o una limitazione superiore dell'errore assoluto. Per la stessa ragione, l'errore relativo è spesso valutato rispetto al valore approssimato $\tilde{x}$ invece di quello esatto x.

Cifre significative

Spesso nella valutazione di accuratezza di un risultato, si parla di **cifre significative esatte**. Per tale concetto **intuitivo** è problematico fornire una definizione rigorosa!

Le **cifre significative** di un numero x sono le prime cifre diverse da zero e tutte le successive

$$0.0\mathbf{12302},\ \mathbf{121.000567}$$

La nozione di **cifre significative esatte** è legata all'errore relativo. Consideriamo

$$\begin{array}{lll} x = 1.00000, & \tilde{x} = 1.00345, & \epsilon_x = 3.45 \times 10^{-3} \\ y = 9.0000, & \tilde{y} = 8.99899, & \epsilon_y = 1.12 \times 10^{-4} \end{array}$$

x, y concordano fino a **tre** cifre significative con le loro rispettive approssimazioni $\tilde{x}, \tilde{y}$.

L'errore relativo è però diverso nei due casi $\rightsquigarrow$ l'errore relativo fornisce una stima più attendibile e fine dell'accuratezza dell'approssimazione !!

L'algoritmo di Archimede per il calcolo di π

L'algoritmo di Archimede approssima $\pi = 3.141592653589793\ldots$ mediante la lunghezza del semiperimetro p_i dei poligoni regolari con 2^{i+1} lati inscritti in una circonferenza di raggio unitario, per $i = 1, 2, \ldots$.
Dopo semplici calcoli, si ottengono le seguenti relazioni

$$
\begin{aligned}
& l_1 = \sqrt{2} \\
& i = 1, 2, \ldots \\
& p_i = l_i \times 2^i \\
& l_{i+1} = \sqrt{2 - \sqrt{4 - l_i^2}}
\end{aligned}
$$

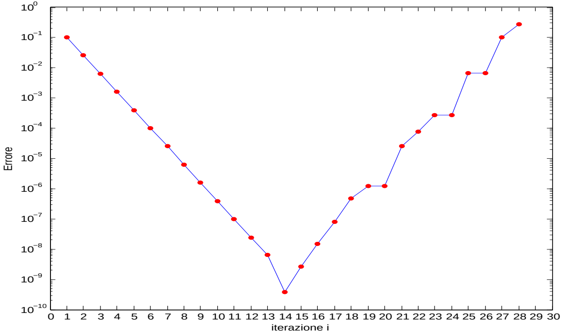
La successione $p_i \to \pi$ per $i \to \infty$ e quindi possiamo controllare l'errore analitico $\epsilon_i = \frac{|\pi - p_i|}{\pi}$ pur di considerare un numero sufficientemente alto di iterazioni.

Risultati numerici per l'algoritmo di Archimede

MATLAB $eps = 2.22 \times 10^{-16}$

i	p_i	err_{ass}	err_{rel}
1	2.828427124746190	3.13e-01	9.97e-02
2	3.061467458920719	8.01e-02	2.55e-02
3	3.121445152258053	2.01e-02	6.41e-03
4	3.136548490545941	5.04e-03	1.60e-03
5	3.140331156954739	1.26e-03	4.01e-04
6	3.141277250932757	3.15e-04	1.00e-04
$\vdots$	$\vdots$	$\vdots$	$\vdots$
11	3.141592345611077	3.07e-07	9.80e-08
12	3.141592576545004	7.70e-08	2.45e-08
13	3.141592633463248	2.01e-08	6.40e-09
14	3.141592654807589	1.22e-09	3.88e-10
15	3.141592645321215	8.27e-09	2.63e-09
16	3.141592607375720	4.62e-08	1.47e-08
17	3.141592910939673	2.57e-07	8.19e-08
$\vdots$	$\vdots$	$\vdots$	$\vdots$
27	3.464101615137754	3.23e-01	1.02e-01
28	4.000000000000000	8.58e-01	2.73e-01

Grafico errore algoritmo di Archimede



Per calcolare la funzione esponenziale $e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!}$, si può considerare il troncamento della serie $g_n(x) = \sum_{k=0}^{n} \frac{x^k}{k!}$ con n determinato in modo che l'ulteriore addendo non migliora l'approssimazione ottenuta.

x	n	$g_n(x)$	err_{rel}
0.5	16	1.648721270700128	2.69e-16
1	19	2.718281828459046	1.63e-16
20	68	4.851651954097902e+08	1.22e-16
40	103	2.353852668370200e+17	1.35e-16
100	192	2.688117141816133e+43	9.21e-16
-0.5	16	6.065306597126333e-01	1.83e-16
-1	20	3.678794411714424e-01	3.01e-16
-20	96	5.621884472130418e-09	1.72e+00
-40	138	-3.165731894063124e+00	7.45e+17
-100	246	-2.913755646891533e+25	7.83e+68

Approssimazione della derivata con il rapporto incrementale

MATLAB $eps = 2.22 \times 10^{-16}$

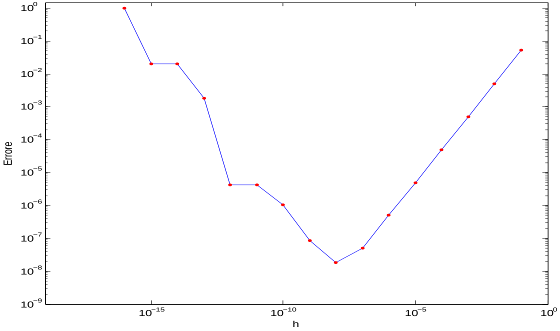
$$\lim_{h \to 0} \frac{f(x+h) - f(x)}{h} = f'(x)$$

Il rapporto incrementale, per h sufficientemente piccolo, fornisce un'approssimazione accurata di $f'(x)$ in un punto x. Per $f(x) = e^x$ e $x = 1$ abbiamo

h	err_{rel}
1.e-01	5.1709e-02
1.e-02	5.0167e-03
1.e-03	5.0017e-04
⋮	⋮
1.e-07	5.1484e-08
1.e-08	2.4290e-09
1.e-09	7.9257e-08
1.e-10	5.6937e-07
⋮	⋮
1.e-14	3.4351e-03
1.e-15	1.4360e-01

Grafico errore approssimazione della derivata con il rapporto incrementale

MATLAB $eps = 2.22 \times 10^{-16}$



Approssimazione del numero di Nepero e

MATLAB $eps = 2.22 \times 10^{-16}$

Per stimare

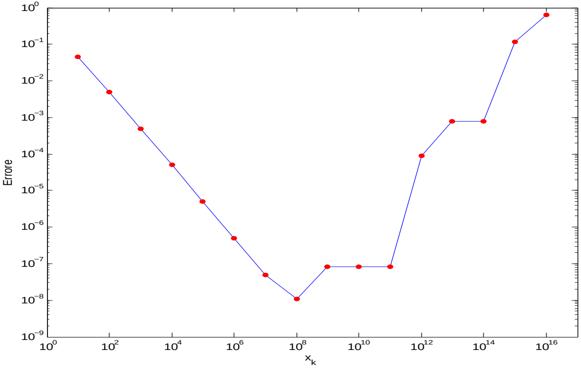
$$\lim_{x\to\infty}\left(1+\frac{1}{x}\right)^x = e = 2.71828182845905\ldots \quad ,$$

si calcola $f_k = (1+\frac{1}{x_k})^{x_k}$ per $x_k = 10^k, k = 1, 2, \ldots, 16$.

k	f_k	err_{rel}
1	2.593742460100002	4.5815e-02
2	2.704813829421528	4.9546e-03
3	2.716923932235594	4.9954e-04
$\vdots$	$\vdots$	$\vdots$
7	2.718281694132082	4.9416e-08
8	2.718281798347358	1.1077e-08
9	2.718282052011560	8.2240e-08
10	2.718282053234788	8.2690e-08
$\vdots$	$\vdots$	$\vdots$
14	2.716110034087023	7.9896e-04
15	3.035035206549262	1.1653e-01
16	1.000000000000000	6.3212e-01

Grafico errore approssimazione numero di Nepero e

MATLAB $eps = 2.22 \times 10^{-16}$



Scegliendo i valori $x_k = 2^k, k = 1, 2, \ldots, 20$, si ottiene

k	f_k	err_{rel}
1	2.565784513950348	5.6101e-02
2	2.697344952565099	7.7022e-03
3	2.715632000168991	9.7482e-04
4	2.717950081189666	1.2204e-04
5	2.718240351930294	1.5258e-05
⋮	⋮	⋮
11	2.718281828300821	5.8208e-11
12	2.718281828439267	7.2759e-12
13	2.718281828456573	9.0949e-13
14	2.718281828458736	1.1371e-13
15	2.718281828459006	1.4213e-14
16	2.718281828459040	1.7971e-15
17	2.718281828459045	1.6337e-16
18	1.000000000000000	6.3212e-01

Conclusioni

Per capire la causa dell'inaccuratezza dei risultati numerici ottenuti, dobbiamo analizzare il ruolo svolto dal **calcolatore**.

In particolare dobbiamo studiare

- **la rappresentazione dei numeri reali**
- **l'aritmetica di macchina**