Condizionamento di un problema e stabilità dell'algoritmo risolutivo

Problema matematico, che indichiamo con *f*, è una descrizione chiara e non ambigua di una connessione funzionale tra i dati del problema e i corrispondenti risultati (detta anche funzione dato-risultato).

Un problema si dice ben posto se la sua soluzione

- esiste
- è unica
- dipende in modo continuo dai dati del problema

E' importante osservare che, se un problema ammette una ed una sola soluzione, esisterà una funzione dato-risultato, detta anche applicazione risolvente, che risulta essere biiettiva. Per garantire la buona posizione del problema dobbiamo richiedere che tale applicazione sia anche continua.

Algoritmo, che indichiamo Ψ, è una sequenza di operazioni di macchina che devono essere eseguite al fine di ottenere, in un numero finito di passi, da un vettore di numeri di macchina $\tilde{x}$, un output $\Psi(\tilde{x}) = \tilde{y}$.

Condizionamento di un problema:

Quando un problema è ben posto (ossia ammette un'unica soluzione che dipende con continuità dai dati) si cerca di dare **una misura quantitativa di come la sua soluzione venga influenzata da una perturbazione dei dati**.

Indichiamo con $\tilde{x} = x + \delta x$ i dati affetti da una perturbazione δx.

Come vengono propagati dal problema $\boldsymbol{f}$ gli errori presenti nei dati, nell'ipotesi ideale che non ci siano errori di calcolo ed indipendentemente dall'algoritmo utilizzato?

Quando a piccole perturbazioni sui dati *x*, corrispondono perturbazioni relative sul risultato *f(x)* dello stesso ordine di grandezza, il problema *y=f(x)* è detto **ben condizionato**, altrimenti è detto **mal condizionato**.

Uno stesso problema può essere mal condizionato per certi dati ma non per altri.

$$\frac{\|f(x) - f(\tilde{x})\|}{\|f(x)\|} \leq K \frac{\|x - \tilde{x}\|}{\|x\|}$$

K è detto indice di condizionamento.

Il condizionamento è legato al problema numerico e non ha alcun legame con gli errori di arrotondamento delle operazioni di macchina, né con il particolare algoritmo utilizzato.

Esempio 1: Studio del **condizionamento della valutazione di una funzione** $f: R \rightarrow R$ (differenziabile) in un punto *x*.

Sia $\tilde{x} = x + \delta x$,

$$f(\tilde{x}) = f(x) + (\tilde{x} - x)f'(x) + O((\tilde{x} - x)^2)$$

$$f(\tilde{x}) - f(x) \approx (\tilde{x} - x)f'(x)$$

$$\frac{f(\tilde{x}) - f(x)}{f(x)} \approx \frac{(\tilde{x} - x)f'(x)}{f(x)}$$

$$\left|\frac{f(\tilde{x}) - f(x)}{f(x)}\right| \approx \left|\frac{f'(x)x}{f(x)}\right| \left|\frac{\tilde{x} - x}{x}\right|$$

Poniamo $K = \left|\frac{f'(x)x}{f(x)}\right|$, allora

$$\boldsymbol{\left|\frac{f(x + \delta x) - f(x)}{f(x)}\right| \approx K \left|\frac{\tilde{x} - x}{x}\right|}$$

Sia $f(x) = cos(x)$, x= 1.57079.

$\cos(1.57079) = 6.32679489 \cdot 10^{-6}$

$$f'(x) = -sin(x); \quad K = \left|\frac{sin(x) \cdot x}{cos(x)}\right| = |x \cdot tan(x)|$$

K= 2.48275·10^5

Ci aspettiamo che il problema sia mal-condizionato per x vicino a $\frac{\pi}{2}$=(**1.57079**6326794897) o a multipli di $\frac{\pi}{2}$

Consideriamo adesso un valore $\tilde{x}$, ottenuto perturbando x, in partiolare

$\tilde{x} = 1.57078$; $\boldsymbol{\delta x = |x - \tilde{x}| \approx 10^{-6}}$.

$$cos(x) = 6.32679489 \cdot 10^{-6}$$

$$cos(\tilde{x}) = 1.632679489 \cdot 10^{-5}$$

$$\left|\frac{f(\tilde{x}) - f(x)}{f(x)}\right| \approx 1.58$$

Definiamo **Errore inerente**,

$$\boldsymbol{E_{in} = \left|\frac{f(\tilde{x}) - f(x)}{f(x)}\right|}$$

N.B. 1) Uno stesso problema può essere mal condizionato per certi dati ma non per altri.

N.B. 2) Il condizionamento è legato al problema numerico e non ha alcun legame con gli errori di arrotondamento delle operazioni di macchina né con il particolare algoritmo utilizzato. Si tratta di una caratteristica del problema, che non dipende dal modo in cui la soluzione viene calcolata.

Esempio 2:

Calcolare gli zeri del polinomio $p(x) = (x-1) \cdot (x-2) \cdot \ldots (x-20) =$

$= x^{20} - 210x^{19} + \cdots$ i cui zeri sono **1,2,3,4,….,20**

Sia $\beta = 2, t = 30.$

Perturbiamo adesso il coefficiente di x^{19}

-210----------> -210+2^{-23}

Il polinomio diventa

$$p(x) + 2^{-23}x^{19}$$

Le radici diventano

1.00000 0000	10.09526 6145 ± 0.64350 0904i
2.00000 0000	11.79363 3881 ± 1.65232 9728i
3.00000 0000	13.99235 8137 ± 2.51883 0070i
4.00000 0000	16.73073 7466 ± 2.81262 4894i
4.99999 9928	19.50243 9400 ± 1.94033 0347i
6.00000 6944	
6.99969 7234	
8.00726 7603	
8.91725 0249	
20.84690 8101	

Condizionamento di un sistema lineare

In questa sezione si vuole esaminare come perturbazioni sugli elementi della matrice A e sugli elementi del termine noto b influenzano la soluzione x del sistema lineare. Queste perturbazioni sono tipicamente dovute sia agli errori di approssimazione quando la matrice A ed il termine noto b vengono rappresentati con numeri finiti, sia al fatto che tutte le operazioni dell'algoritmo risolutivo vengono effettuate in aritmetica finita.

Sia $\| \|$ una qualunque norma naturale; sia $A \in M(n \times n)$ a rango massimo e sia δA una matrice di perturbazione e δb il vettore di perturbazione del termine noto.

Sia x la soluzione di Ax=b.

Un metodo numerico fornisce una soluzione approssimata, $x+ \delta x$, (δx *vettore di perturbazione della soluzione*) che verifica un sistema perturbato:

Caso 1: Perturbazione solo sul termine noto

$$A(x+ \delta x)=b+\delta b$$

Vogliamo stimare δx in funzione di δb :

$$Ax+ A\delta x =b+\delta b$$

$$Ax-b+A\delta x = \delta b$$

Ma per ipotesi Ax=b, e quindi Ax-b=0, quindi risulta

$$A\delta x = \delta b \text{ da cui } \quad \delta x = A^{-1}\delta b$$

Passando alle norme di ambo i membri:

$$\|\delta x\| \leq \|A^{-1}\| \ \|\delta b\| \qquad (1)$$

Inoltre da Ax=b segue che

$$\|b\| = \|Ax\| \leq \|A\|\|x\|$$

e quindi

$$\frac{1}{\|x\|} \leq \|A\| \cdot \frac{1}{\|b\|} \qquad (2)$$

Moltiplicando membro a membro (1) e (2) si ha;

$$\frac{\|\delta x\|}{\|x\|} \leq \|A^{-1}\|\|A\| \cdot \frac{\|\delta b\|}{\|b\|}$$

$K(A) = \|A^{-1}\|\|A\|$ rappresenta l'indice di condizionamento del problema del calcolo della soluzione di un sistema lineare.

Caso 2: Perturbazione sia sulla matrice che sul termine noto

$$(A+\delta A)(x+\delta x)=(b+\delta b)$$

Sotto l'ipotesi che $\|A^{-1}\|\|\delta A\| < 1$, (da cui si può dimostrare che $A+\delta A$ è non singolare) vale la seguente relazione:

$$\frac{\|\delta x\|}{\|x\|} \leq \|A^{-1}\| \ \ \|A\| \ \ \frac{\frac{\|\delta A\|}{\|A\|} + \frac{\|\delta b\|}{\|b\|}}{1 - \|A^{-1}\| \ \ \|A\| \ \ \frac{\|\delta A\|}{\|A\|}}$$

L'indice di condizionamento della matrice identità è uguale ad 1, K(I)=1.

In generale l'indice di condizionamento di una matrice, per ogni norma matriciale indotta, è maggiore o uguale ad 1, K(A)≥1.

N.B. Se A è ortogonale, cioè se $A^T A = A\,A^T = I$ (cioè la sua trasposta coincide con l'inversa ($A^T = A^{-1}$), allora

$$K_2(A) = ||A||_2 ||A^{-1}||_2 = 1$$

Infatti

$$||A||_2 = \sqrt{\rho(A^T A)} = \sqrt{\rho(I)} = 1$$

$$||A^{-1}||_2 = \ ||A^T||_2 = \sqrt{\rho(AA^T)} = \sqrt{\rho(I)} = 1$$

La risoluzione di Ax = b con A ortogonale è sempre un problema ben condizionato

K(A) piccolo (ordine n^p p=0,1,2,3): **il problema/matrice è ben condizionato**

K(A) grande (ordine 10^n): **Il problema/matrice è mal condizionato**

Proprietà dell'indice di condizionamento in norma 2

Si definisce numero di condizionamento in norma 2 di A il numero reale

$$K_2(A) := \frac{\sqrt{\lambda_{max}(A^T A)}}{\sqrt{\lambda_{min}(A^T A)}}$$

$$K_2(A^T A) = K_2(A)^2$$

Esempi di matrici mal condizionate.

1) Matrice di Vandermonde: dato un vettore $x = (x_0, x_1, ..., x_n)$, la matrice di Vandermonde è una matrice di dimensione (n+1)×(n+1), il cui generico elemento di posto (i,j) è dato da

$$a_{ij} = (x_i)^j \qquad \text{i=0,..}n \quad j\text{=0,..}n$$

cioè:

$$A = \begin{bmatrix} 1 & x_0 & (x_0)^2 & (x_0)^3 \\ 1 & x_1 & (x_1)^2 & (x_1)^3 \\ 1 & x_2 & (x_2)^2 & (x_2)^3 \\ 1 & x_3 & (x_3)^2 & (x_3)^3 \end{bmatrix}$$

2) Matrice di Hilbert

La matrice di Hilbert è così definita:

$$h_{ij} = \frac{1}{i + j - 1} \qquad i, j = 1, ..., n$$

Nel caso n=4

$$H = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \end{bmatrix}$$

L'indice di condizionamente della matrice di Hilbert di ordine 4 è $K(H)=1.55*10^4$.

Osservazione: L'indice di condizionamento di una matrice dipende intrinsecamente dal problema , dalla matrice stessa, cioè non ha nulla a che vedere con l'algoritmo risolutivo; esso ci dice come le inevitabili perturbazioni che si hanno sulla matrice o

sul termine noto si ripercuotono sulla soluzione del sistema, prescindendo da come questa soluzione si ottiene.

Vediamo ora con un esempio banale come un alto indice di condizionamento può falsare i risultati completamente a partire da piccole perturbazioni sui dati originali.

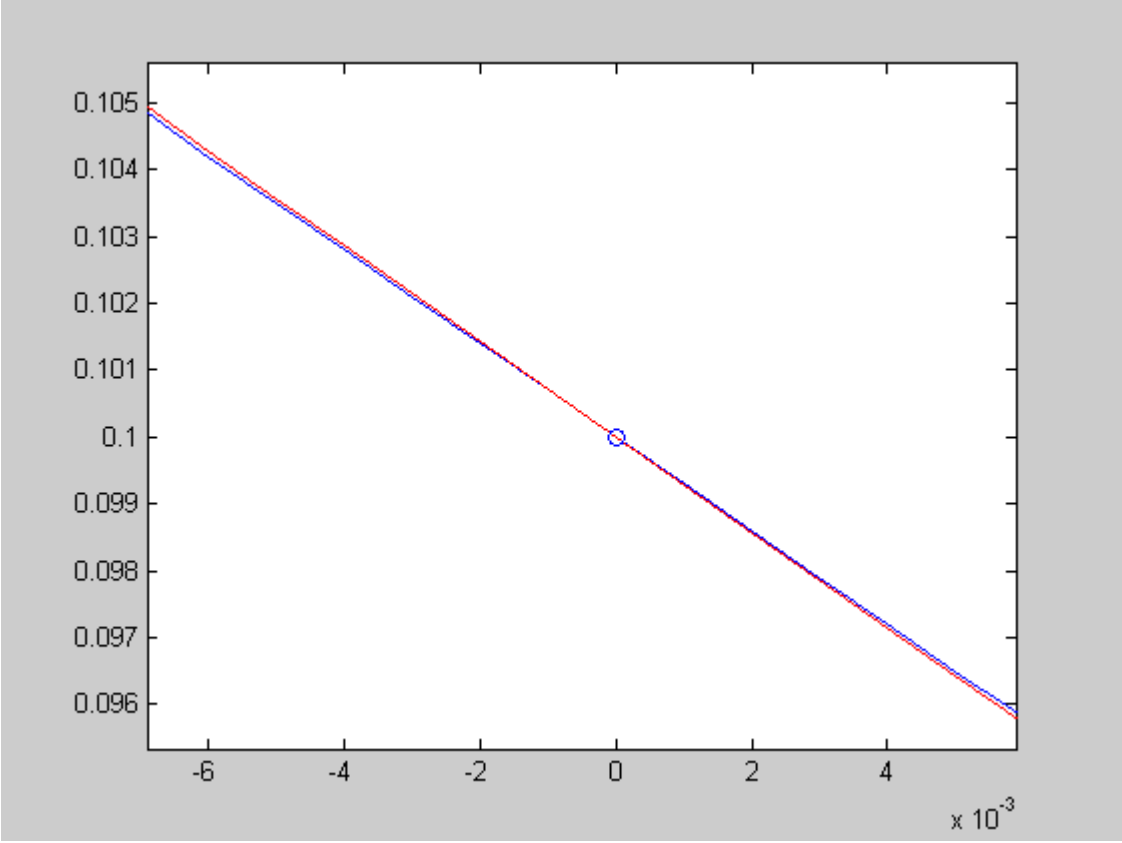
Consideriamo il sistema lineare Ax=b dove

$$A=\begin{bmatrix}7 & 10\\ 5 & 7\end{bmatrix} \quad b=\begin{bmatrix}1\\ 0.7\end{bmatrix}$$

La soluzione esatta di questo sistema lineare è x= $\begin{bmatrix}0\\ 0.1\end{bmatrix}$

Questo punto di R^2 il punto di intersezione delle due rette

$$7x_1+10x_2=1 \quad e \quad 5x_1+7x_2=0.7$$



Sia $\delta b = \begin{bmatrix} 0.01 \\ -0.01 \end{bmatrix}$ una perturbazione del termine noto.

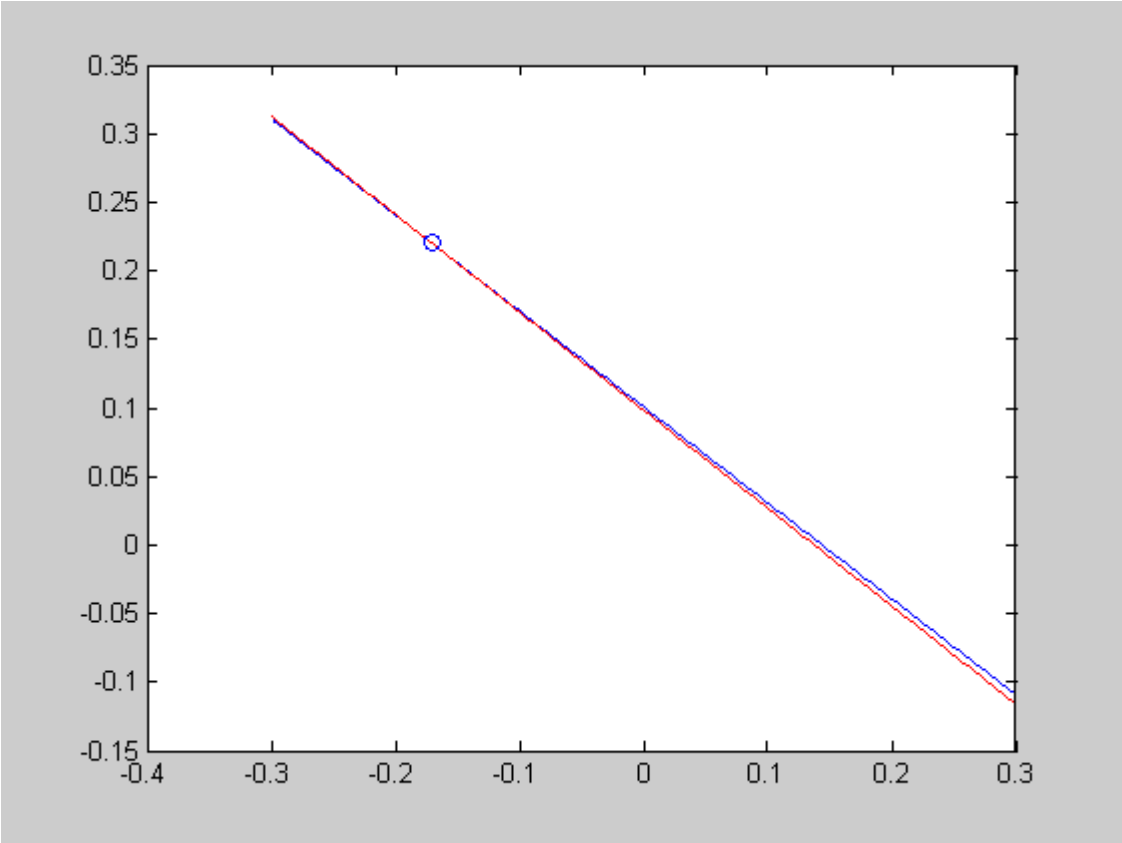
Risolviamo lo stesso sistema lineare con termine noto perturbato:

$$b' = b + \delta b = \begin{bmatrix} 1.01 \\ 0.69 \end{bmatrix}$$

La soluzione diventa $x' = \begin{bmatrix} -0.17 \\ 0.22 \end{bmatrix}$

Cioè le rette

$7x_1 + 10x_2 = 1.01$ *e* $5x_1 + 7x_2 = 0.69$ si intersecano nel punto $x' = \begin{bmatrix} -0.17 \\ 0.22 \end{bmatrix}$



Calcoliamo l'indice di condizionamento della matrice A:

$A = \begin{bmatrix} 7 & 10 \\ 5 & 7 \end{bmatrix}$ $A^{-1} = \begin{bmatrix} -7 & 10 \\ 5 & -7 \end{bmatrix}$

K(A)=17x17=289

$\frac{\|\delta b\|_\infty}{\|b\|_\infty} = 0.01$ (errore dell'1% sul termine noto).

$\delta x = x - x' = \begin{bmatrix} 0.17 \\ -0.12 \end{bmatrix}$

$\frac{\|\delta x\|_\infty}{\|x\|_\infty} = \frac{0.17}{0.1} = 1.7$

Cioè un'errore relativo sui dati dell'ordine dell'1% comporta un errore relativo sulla soluzione maggiore del 100% (precisamente del 170%).

Risolvere il sistema lineare:

$$\begin{cases} x + y = 2 \\ 1001x + 1000y = 2001 \end{cases}$$

Le soluzioni sono x=1, y=1;

Perturbiamo il coefficiente della x dell'1%:

$$\begin{cases} \left(1 + \frac{1}{100}\right)x + y = 2 \\ 1001x + 1000y = 2001 \end{cases}$$

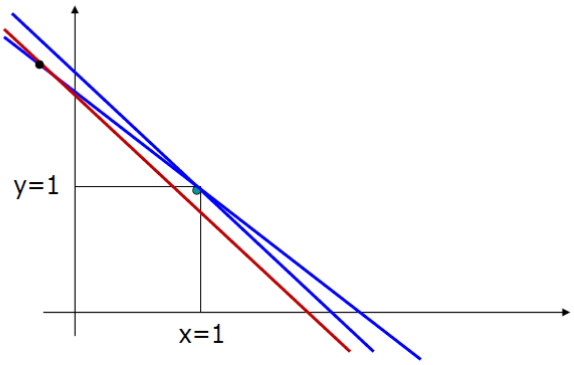
$x = -\frac{1}{9} = -0.1111$

$y = \frac{1901}{900} = 2.1122$

Indice di condizionamento di A= 2.002002999900244e+06

Errore sulla soluzione del 110%.

Interpretazione geometrica



Nel caso in cui si abbia dinanzi un problema mal posto, si possono seguire le seguenti strade:

1) Cambiare la formulazione del problema, per superare l'ostacolo.
2) Usare la precisione multipla nei calcoli
3) Usare tecniche di regolarizzazione, che sostituiscono al problema di partenza un problema leggermente modificato, ma che risulta ben condizionato.

Algoritmi e stabilità

Studiamo adesso una nuova proprietà degli algoritmi: la **stabilità** che esprime il comportamento dell'algoritmo considerato rispetto alla propagazione degli errori.

Dato l'algoritmo Ψ, sequenza di operazioni di macchina, che traduce in operazioni di macchina il problema *f*, vogliamo vedere come la sequenza di operazioni di macchina eseguite sui numeri di macchina $\tilde{x}$, $\Psi(\tilde{x}) = \tilde{y}$, propaga l'errore iniziale.
La stabilità di un algoritmo valuta la reazione dell'algoritmo all'introduzione di perturbazioni nei dati iniziali.
Vogliamo confrontare la risposta dell'algoritmo con la risposta della funzione *f* su dati perturbati, per vedere l'effetto delle operazioni di macchina sul risultato finale.
Definiamo **l'Errore Algoritmico**:

$$E_{alg} = \left|\frac{\Psi(\tilde{x}) - f(\tilde{x})}{f(\tilde{x})}\right|$$

L'errore algoritmico dipende solo da come il risultato viene calcolato.
Pertanto, influiscono sull'errore algoritmico:

- ▶ il numero di operazioni eseguite
- ▶ l'ordine in cui le operazioni vengono eseguite
- ▶ il tipo di operazioni eseguite

Errore totale:

L'accuratezza della soluzione (di quanto si discosta la soluzione calcolata da quella esatta) , che indichiamo con

$$\boldsymbol{E_{tot} = \left|\frac{\Psi(\tilde{x}) - f(x)}{f(x)}\right|}$$

dipende sia dal condizionamento del problema che dalla stabilità algoritmica.
Adesso giustifichiamo la precedente affermazione.

Consideriamo $\frac{\Psi(\tilde{x})-f(x)}{f(x)} = \frac{\Psi(\tilde{x})}{f(x)} - 1$

$$= \frac{\Psi(\tilde{x})}{f(\tilde{x})}\frac{f(\tilde{x})}{f(x)} - 1.$$

$$= \frac{\Psi(\tilde{x})-f(\tilde{x})+f(\tilde{x})}{f(\tilde{x})} \cdot \frac{f(\tilde{x})-f(x)+f(x)}{f(x)} - 1$$

$$= \left(\frac{\Psi(\tilde{x})-f(\tilde{x})}{f(\tilde{x})} + 1\right)\left(\frac{f(\tilde{x})-f(x)}{f(x)} + 1\right) \text{-1}$$

$$\frac{\Psi(\tilde{x}) - f(x)}{f(x)} = \frac{\Psi(\tilde{x}) - f(\tilde{x})}{f(\tilde{x})}\frac{f(\tilde{x}) - f(x)}{f(x)} + \frac{\Psi(\tilde{x}) - f(\tilde{x})}{f(\tilde{x})} + \frac{f(\tilde{x}) - f(x)}{f(x)} + 1 - 1$$

,

quindi $p$assando ai valori assoluti

$$\left|\frac{\Psi(\tilde{x}) - f(x)}{f(x)}\right| \leq \left|\frac{f(\tilde{x}) - f(x)}{f(x)}\right| + \left|\frac{\Psi(\tilde{x}) - f(\tilde{x})}{f(\tilde{x})}\right| + \left|\frac{f(\tilde{x}) - f(x)}{f(x)}\right|\left|\frac{\Psi(\tilde{x}) - f(\tilde{x})}{f(\tilde{x})}\right|$$

$$E_{tot} \leq E_{in} + E_{alg} + E_{in} \cdot E_{alg}$$

e, trascurando i prodotti degli errori relativi, si ha

$$\boldsymbol{E_{tot} \approx E_{in} + E_{alg}}$$

La bassa accuratezza dei risultati prodotti da un processo numerico può essere imputabile a:

alto **mal condizionamento** intrinseco del problema

oppure

all'**instabilità** dell'algoritmo utilizzato per produrlo.

La stabilità dell'algoritmo non garantisce che il risultato calcolato sia accurato.

Per un problema mal condizionato la distinzione tra algoritmo stabile e instabile non è molto significativa in quanto l'errore totale risulta dominato dall'errore inerente.

Quindi per un problema mal condizionato è opportuna, in generale, una sua riformulazione.

Si parla di **stabilità o instabilità numerica** intendendo che gli errori sui dati non sono (o sono) amplificati durante lo sviluppo dell'algoritmo:

$$\left|E_{alg}\right| \approx g(n) \cdot eps$$

n=numero di operazioni effettuare

$g(n)=c \cdot n$, $c>0$ **crescita dell'errore lineare**

$g(n)=c^n$, $c>1$, **crescita dell'errore esponenziale**.

Un algoritmo numerico è detto stabile se $g(n)$ è lineare, cioè l'errore algoritmico è dell'ordine di grandezza della precisione di macchina, instabile altrimenti.

Esempio

Valutare la funzione $y = f(x) = \dfrac{(1+x)-1}{x}$

mediante l'algoritmo y=((1+x)-1)/x in x=1e-15

Valutiamo l'espressione in Python otteniamo: **y=1.11022302462516** invece di **y=1**

Il problema è ben condizionato perché:

$K = \left|\dfrac{f'(x)}{f(x)}x\right| = 0$ poiché $f'(x) = 0 \quad \forall x \neq 0$

Adesso verifichiamo se l'algoritmo y=((1+x)-1)/x è stabile o no.

fl(1+x)=(1+fl(x))(1+ε) $|\varepsilon| \leq u$

fl(1+x)-1=(fl(1+x)-1) (1+ε) = [(1+fl(x))(1+ε)-1] (1+ε) =

=[(1+x(1+ε))(1+ε)-1] (1+ε) =

(1+ε+x(1+2ε+ε^2)-1)(1+ε)≈ (x(1+2ε)+ ε)(1+ε)≈ **x(1+3ε)+ ε**

$$\tilde{y} = \frac{fl((1+x)-1)}{fl(x)} \approx \frac{x(1+3\varepsilon)+\varepsilon}{x(1+\varepsilon)} \cdot (1+\varepsilon) \approx \frac{x(1+3\varepsilon)+\varepsilon}{x}$$

Il valore in aritmetica reale di y è 1.

Calcoliamo quindi l'errore relativo

$$\left|\frac{\tilde{y}-y}{y}\right| = \frac{\frac{x(1+3\varepsilon)+\varepsilon}{x}-1}{1} = \frac{x+3\varepsilon x+\varepsilon-x}{x} = 3\varepsilon + \frac{\varepsilon}{x}$$

Se x è piccolo l'errore relativo dell'algoritmo potrebbe essere grande.

L'algoritmo è instabile per valori di x più piccoli dell'unità di arrotondamento.

Stabilità dell'algoritmo della somma di n numeri finiti

Studieremo ora qual è l'errore da cui è affetta la somma "finita" (cioè ottenuta in aritmetica finita) di *n* numeri finiti.

Siano $x_1, x_2, \ldots . x_n$ n numeri finiti da sommare.

L'algoritmo somma è il seguente:

S:=x_1

for i=2,…,n

S:=S+x_i

endfor

I passi eseguiti dall'algoritmo sono:

$S_2 = fl(S_1 + x_2)$

$S_3 = fl(S_2 + x_3)$

.

.

$S_n = fl(S_{n-1} + x_n)$

Quindi il risultato finale S_n differisce dal risultato teorico S.

L'errore relativo è $\left|\frac{S-S_n}{S}\right|$ è dato da:

$$\left|\frac{S - S_n}{S}\right| \leq \frac{(|x_1|n + |x_2|(n-1) + \cdots + |x_n|)}{|S|} \cdot 1.01 \cdot \varepsilon$$

dove $\varepsilon \leq u$

Da qui si vede che ogni addendo è moltiplicato per un peso fisso. Tale formula mostra che, assegnati i numeri finiti $x_1, x_2, \ldots . x_n$, la maggiorazione **dell'errore relativo della loro somma "finita" è minima se si sommano questi numeri in modo che i loro valori assoluti siano in ordine crescente**, cioè: $|x_1| \leq |x_2| \leq \ldots \leq |x_n|$. Infatti, così operando,ai pesi (1.01 ε)n, (1.01 ε)(n-1) più grandi si associano i numeri $|x_1|, |x_2|, \ldots, |x_n|$ più piccoli. Ne segue che l'errore pesa meno e si avranno perciò risultati più attendibili.

La formula può essere riscritta come:

$$\left|\frac{S - S_n}{S}\right| \leq \frac{|x_{max}|}{S} 1.01 \cdot \varepsilon \, \frac{n(n+1)}{2}$$

Se $|x_{\max}|$ è elevato ma S è piccola, come può avvenire se si sommano addendi di segno opposto e modulo simile

$$\Rightarrow \frac{|x_{\max}|}{|S|} \text{ è grande}$$

cioè l'errore **relativo può crescere molto.**