Sistemi di equazioni lineari

$$
\begin{aligned}
a_{00}\ x_0 + a_{01}\ x_1 + a_{02}\ x_2 &= b_0 \\
a_{10}\ x_0 + a_{11}\ x_1 + a_{12}\ x_2 &= b_1 \\
a_{20}\ x_0 + a_{21}\ x_1 + a_{22}\ x_2 &= b_2
\end{aligned}
$$

Per N equazioni

$$\sum_{j=0}^{N-1} a_{ij}\ x_j = b_i \qquad i = 0, N-1$$

sono equivalenti ad una equazione matriciale

$$
\begin{pmatrix} a_{0,0} & \dots & a_{0N-1} \\ \dots & \dots & \dots \\ a_{N-1,0} & \dots & a_{N-1,N-1} \end{pmatrix}
\begin{pmatrix} x_0 \\ \dots \\ x_{N-1} \end{pmatrix}
=
\begin{pmatrix} b_0 \\ \dots \\ b_{N-1} \end{pmatrix}
$$

la soluzione è unica se $\det(A) \neq 0$ e vale

$$\vec{x} = A^{-1}\ \vec{b}$$

Matrici triangolari

U è triangolare superiore se:
$u_{ij} = 0$ per $i > j$.

$$U = \begin{pmatrix} u_{00} & u_{01} & u_{02} \\ 0 & u_{11} & u_{12} \\ 0 & 0 & u_{22} \end{pmatrix}$$

$U\vec{x} = \vec{b}$ è molto facile da risolvere. Analogamente L è triangolare inferiore se $l_{ij} = 0$ per $i < j$.

$$L = \begin{pmatrix} l_{00} & 0 & 0 \\ l_{10} & l_{11} & 0 \\ l_{20} & l_{21} & l_{22} \end{pmatrix}$$

Tornando all'equazione corrispondente a una matrice triangolare superiore

$$\begin{aligned} u_{00}\,x_0 + u_{01}\,x_1 + u_{02}\,x_2 &= b_0 \\ u_{11}\,x_1 + u_{12}\,x_2 &= b_1 \\ u_{22}\,x_2 &= b_2 \end{aligned}$$

la cui soluzione è data da

$$\begin{aligned} x_2 &= b_2/u_{22} \\ x_1 &= \left(b_1 - u_{12}x_2\right)/u_{11} \\ x_0 &= \left(b_0 - u_{01}x_1 - u_{02}x_2\right)/u_{00} \end{aligned}$$

ogni x_i viene usato solo dopo essere stato calcolato.

Per un sistema di N equazioni in N incognite la soluzione sarà:

$$x_i = \left(b_i - \sum_{k=i+1}^{N-1} u_{ik}x_k\right)/u_{ii}$$

che va applicata calcolando gli x_i a partire dall'ultimo all'indietro.

Data l'efficacia di questo metodo, cerco di ridurre ogni sistema di equazioni $A\vec{x} = \vec{b}$ ad un sistema triangolare $U\vec{x} = \vec{b'}$ con opportuni U e $\vec{b'}$.

Eliminazione gaussiana

Prendo un sistema di tre equazioni in tre incognite

$$\begin{aligned} a_{00}\, x_0 + a_{01}\, x_1 + a_{02}\, x_2 &= b_0 \\ a_{10}\, x_0 + a_{11}\, x_1 + a_{12}\, x_2 &= b_1 \\ a_{20}\, x_0 + a_{21}\, x_1 + a_{22}\, x_2 &= b_2 \end{aligned}$$

sommando ad un'equazione il multiplo di un'altra la soluzione non cambia. Sottraggo allora alla seconda equazione la prima moltiplicata per λ e ottengo

$$(a_{10} - \lambda a_{00})\, x_0 + (a_{11} - \lambda a_{01})\, x_1 + (a_{12} - \lambda a_{02})\, x_2 = b_1 - \lambda b_0$$

voglio eliminare il termine che moltiplica x_1, cosa che posso fare se scelgo

$$\lambda = a_{10}/a_{00}$$

ottenendo in questo modo che $a'_{10} = 0$

Posso ora ripetere il procedimento per la terza riga sottraendo la prima moltiplicata per a_{20}/a_{00} e ottengo

$$\begin{aligned} a_{00}\ x_0 + a_{01}\ x_1 + a_{02}\ x_2 &= b_0 \\ a'_{11}\ x_1 + a'_{12}\ x_2 &= b'_1 \\ a'_{21}\ x_1 + a'_{22}\ x_2 &= b'_2 \end{aligned}$$

con

$$a'_{11} = a_{11} - \tfrac{a_{10}}{a_{00}}\, a_{01} \qquad a'_{12} = a_{12} - \frac{a_{10}}{a_{00}}\, a_{02}$$

$$a'_{21} = a_{21} - \tfrac{a_{20}}{a_{00}}\, a_{01} \qquad a'_{22} = a_{22} - \frac{a_{20}}{a_{00}}\, a_{02}$$

$$b'_1 = b_1 - \tfrac{a_{10}}{a_{00}}\, b_0 \qquad b'_2 = b_2 - \frac{a_{20}}{a_{00}}\, b_0$$

Sottraggo ora alla terza riga la seconda moltiplicata per a'_{21}/a'_{11} e ottengo

$$\begin{aligned} a_{00}\ x_0 + a_{01}\ x_1 + a_{02}\ x_2 &= b_0 \\ a'_{11}\ x_1 + a'_{12}\ x_2 &= b'_1 \\ a''_{22}\ x_2 &= b''_2 \end{aligned}$$

dove

$$a''_{22} = a'_{22} - \frac{a'_{21}}{a'_{11}} a'_{12} \quad \text{e} \quad b''_2 = b'_2 - \frac{a'_{21}}{a'_{11}} b'_1$$

$A\vec{x} = \vec{b}$ è stata quindi trasformata in $U\vec{x} = \vec{b}$ e il sistema può ora essere risolto per backsubstitution.

Pivoting

Il metodo fin qui descritto non è molto stabile, cioè gli errori si propagano amplificandosi, e possono portare a soluzioni molto sbagliate.

Si vede con la pratica che l'eliminazione gaussiana è molto più stabile se gli elementi di matrice sulla diagonale principale sono più grandi degli altri in valore assoluto.

Ciò si ottiene scambiando tra loro righe della matrice (pivoting parziale) oppure righe e colonne (full pivoting).

Se $|a_{20}| = \max(|a_{00}||a_{10}||a_{20}|)$ il pivoting parziale richiede lo scambio della prima e terza riga. Guardo ora la seconda e terza riga: se $|a_{21}| > |a_{11}|$ scambio la seconda e la terza riga.

Come risultato su ogni diagonale sono piazzati gli elementi di massimo modulo tra quelli che stanno sotto l'elemento diagonale.

Scambiando le righe gli x_i sono invariati. Se scambio le colonne i e j, anche x_i viene scambiato con x_j.

Inversa di una matrice

$$AA^{-1} = A^{-1}A = 1$$

Se chiamo c_{ij} gli elementi di matrice di A^{-1}

$$\sum_j a_{ij}c_{jk} = \delta_{ik}$$

Sono N sistemi di N equazioni. Per una matrice di ordine 3 ad esempio

$$\begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} c_{00} & c_{01} & c_{02} \\ c_{10} & c_{11} & c_{12} \\ c_{20} & c_{21} & c_{22} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Che può essere spezzato in 3 sistemi di 3 equazioni lineari ciascuno il primo dei quali è

$$\begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} c_{00} \\ c_{10} \\ c_{20} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Fisso un valore di k e chiamo $\vec{b}^{(i)}$ il vettore che ha la i-esima componente uguale a uno, e nulle tutte le altre.

$$\sum_j a_{ij} c_{jk} = b_k^{(i)}$$

- Risolvere uno dei sistemi di equazioni vuole dire trovare gli elementi di matrice di una colonna della matrice inversa.

- Ripetendo per tutti i valori di i trovo la matrice intera.

- Non è necessario fare N volte l'eliminazione gaussiana, perché questa si può fare contemporaneamente per tutte le equazioni.

Rappresentazione delle matrici

- nel linguaggio C tutti gli array sono monodimensionali
- una espressione del tipo $a[i][j]$ indica un array monodimensionale di array monodimensionali
- l'allocazione della memoria è inutilmente complicata e l'accesso può essere lento lento
- conviene definire array lineari di dimensione $N \times N$ e stabilire se si memorizza per riga o per colonna
- in questo modo è più facile allocare dinamicamente la memoria
- una scelta possibile è che $a[0] \ldots a[N-1]$ sia la riga zero, $a[k \cdot N] \ldots a[k \cdot N + N - 1]$ la k-esima riga
- l'elemento con indice di riga uno più di $a[j]$ è $a[j+1]$
- l'elemento con indice colonna uno più di $a[j]$ è $a[j + N]$
- se $a[j]$ è su una certa riga e colonna, $a[j + 1]$ è sulla stessa riga (se $j \neq N-1$) e ha l'indice colonna maggiore di uno
- se $a[j]$ è su una certa riga e colonna, $a[j+N]$ è sulla stessa colonna e ha l'indice riga maggiore di uno

Tempo di calcolo

- La backsubstitution comporta
  1. 1 divisione per l'ultima riga
  2. 1 divisione + 1 prodotto + 1 sottrazione per la penultima riga
  3. 1 divisione + k prodotti + k sottrazioni per la riga $N-k$
  4. $O(N^2)$ operazioni

- L'eliminazione gaussiana comporta
  1. Per azzerare la prima colonna $N+1$ prodotti e sottrazioni più una divisione da ripetere per $N-1$ righe; quindi è $O(N^2)$
  2. bisogna ripetere per $N-2, N-3, \ldots, 1$
  3. si ottiene un procedimento $O(N^3)$

Determinante

- Il determinante di una matrice non cambia aggiungengo ad una riga il multiplo di un'altra

- l'eliminazione gaussiana non cambia il determinante

- dopo l'eliminazione tutti gli elementi sotto la diagonale principale sono nulli

- il determinante, a questo punto, è dato dal prodotto degli elementi diagonali

Esercizio

Considero la matrice

$$\begin{pmatrix} 1 & 1 & 1 \\ a & b & c \\ a^2 & b^2 & c^2 \end{pmatrix}$$

Calcolo il determinante in funzione di a tenendo fissi b e c e faccio un grafico di quello che ottengo. Poi faccio la stessa operazione con una matrice 4×4 della forma

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ a & b & c & d \\ a^2 & b^2 & c^2 & d^2 \\ a^3 & b^3 & c^3 & d^3 \end{pmatrix}$$

e cerco di trovare una formula generale per una matrice $N \times N$ di questo tipo