Corso di Laurea Triennale in Ingegneria e Scienze Informatiche

---

Metodi Numerici **A.A. 2021-2022**

Esercitazione N. 6
Metodo di eliminazione gaussiana

Obiettivo

Preparazione dei codici relativi all'applicazione del metodo di eliminazione gaussiana e sua equivalente fattorizzazione con/senza pivoting.

Codici

Si consideri il sistema lineare $Ax = b$, con $A \in \mathbb{R}^{n \times n}$, $b, x \in \mathbb{R}^n$.

1. Implementare la function

   function `[x,flag]= Lsolve(A,b)`

   che risolve il sistema lineare nel caso di matrice A triangolare inferiore con il metodo della risoluzione forward

   $$x_i = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j \right) / a_{ii}, \qquad i = 1, 2, ..., n.$$

   e restituisce in output

   - la soluzione del sistema
   - una variabile flag=0 se il sistema é risolubile e flag=1 se il sistema non é risolubile (si annulla almeno un elemento diagonale)

2. Implementare la function

   function `[x,flag]= Usolve(A,b)`

   che risolve il sistema lineare nel caso di matrice A triangolare superiore con il metodo della risoluzione backward

   $$x_i = \left(b_i - \sum_{j=i+1}^{n} a_{ij} x_j \right) / a_{ii}, \qquad i = n, n-1, ..., 1.$$

   e restituisce in output

   - la soluzione del sistema

- una variabile flag=0 se il sistema é risolubile e flag=1 se il sistema non é risolubile (si annulla almeno un elemento diagonale)

3. Implementare la function

`function [L,U,flag]=LUnopivot(A),`

che utilizza il metodo di eliminazione gaussiana secondo l'algoritmo base e restituisce in outpute restituisce in output

- i fattori L ed U.
- una variabile flag=0 se per la matrice A esiste la fattorizzazione LU (minori principali fino a quello di ordine n-1 diversi da zero) e flag=1 in caso contrario.

per $k = 1 : n-1$,
    per $i = k+1 : n$,
        $m_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$ (questa scelta fará sí che $a_{ik}^{(k+1)} = 0$)
        per $j = k+1 : n$,
            $a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik} a_{kj}^{(k)}$

o secondo la sua versione vettorizzata

```
per  k=1:n-1,
A(k+1:n,k)=A(k+1:n,k)/A(k,k);
A(k+1:n,k+1:n)= A(k+1:n,k+1:n)-A(k+1:n,k) * A(k,k+1:n);
```

Nota implementativa: memorizzare il moltiplicatore m_{ik} della riga i al passo k nella posizione a_{ik}. In questo modo alla fine dell'algoritmo il fattore L triangolare inferiore é il triangolo inferiore di A (con elementi 1 sulla diagonale) e il fattore U é il triangolo superiore di A. `L=tril(U,-1)+eye(n);  U=triu(U);`

4. Implementare la function

function `[L,U,P,flag]= LUparziale(A)`

una variante della function al punto 3, in cui si applica la tecnica del pivot parziale.

Sia P la matrice identitá di ordine n;

per $k = 1 : n - 1$,
  per $i = k + 1 : n$,
  determinare $a_{rpiv,k}$ t.c. $|a_{rpiv,k}| = \max_{i=k,\ldots,n} |a_{i,k}|$
  if (rpiv != k)
   scambiare riga k della matrice A con la riga rpivot e tenere traccia
   dello scambio effettuato scambiando la riga k e la riga rpivot della matrice P
  $m_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$
   per $j = k + 1 : n$,
    $a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik}a_{kj}^{(k)}$

e si restituiscono in output

- la matrice P ottenuta dalla matrice identitá in cui sono effettuati gli stessi scambi tra righe effettuati durante la fattorizzazione LU con pivoting parziale
- i fattori L ed U.
- la variabile flag=1 se la matrice non é fattorizzabile (si annulla al passo k l'elemento $a_k k$, altrimenti flag=0

5. Implementare la function

```
[x,flag]= function LUsolve(L,U,P,b)
```

che prende in input:

- i fattori L ed U della fattorizzazione $PA = LU$ di una matrice A;
- la matrice P, matrice ottenuta dalla matrice identitá in cui sono effettuati gli stessi scambi tra righe effettuati durante la fattorizzazione LU con pivoting parziale $P \cdot A = LU$,
- il termine noto b del sistema

e restituisce in x la soluzione del sistema lineare nel caso di fattorizzazione $PA = LU$,combinando i metodi di risoluzione forward e backward precedentemente implementati. la variabile flag=0 se il sistema é risolubile, altrimenti flag=0

Nota bene 1: sul termine noto b bisogna effettuare gli stessi scambi che sono stati fatti sulla matrice durante la fase di fattorizzazione. Questo equivale a premoltiplicare il vettore b per P, cioé Pb
Nota bene 2: Nel caso in cui si utilizzi il metodo di Gauss senza pivot allora la matrice P é l'identitá.

Esercizio 1
Calcolare la soluzione dei seguenti sistemi lineari $Ax = b$ usando il metodo di fattorizzazione LU senza pivoting.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 0 & 2 \\ 1 & 6 & 2 \end{bmatrix}, \qquad b = \begin{bmatrix} 6 \\ 6 \\ 9 \end{bmatrix}$$

e

$$A = \begin{bmatrix} 2 & 1 & 3 & 4 \\ 1 & 0 & 3 & 4 \\ 1 & 5 & 0 & 6 \\ 2 & 1 & 3 & 0 \end{bmatrix}, \qquad b = \begin{bmatrix} 10 \\ 8 \\ 12 \\ 6 \end{bmatrix}.$$

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 0 & 2 \\ 1 & 6 & 0 \end{bmatrix}, \qquad b = \begin{bmatrix} 6 \\ 6 \\ 7 \end{bmatrix}$$

In tutti i casi la soluzione esatta ha componenti $x_i = 1$ per ogni $i = 1, ..., n$.

Esercizio 2
Calcolare la soluzione dei seguenti sistemi lineari $Ax = b$ usando il metodo di fattorizzazione LU senza pivoting e con pivoting parziale.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 0 & 1 \\ 1 & 3 & 5 \end{bmatrix}, \qquad b = \begin{bmatrix} 6 \\ 1 \\ 9 \end{bmatrix}$$

e

$$A = \begin{bmatrix} 1 & 1 & 0 & 3 \\ 2 & 1 & -1 & 1 \\ -1 & 2 & 3 & -1 \\ 3 & -1 & -1 & 2 \end{bmatrix}, \qquad b = \begin{bmatrix} 5 \\ 3 \\ 3 \\ 3 \end{bmatrix}.$$

In tutti i casi la soluzione esatta ha componenti $x_i = 1$ per ogni $i = 1, ..., n$. Come è possibile giustificare a priori in tutti i casi l'insuccesso del metodo di fattorizzazione LU in assenza di pivot?

Esercizio 3
Costruire una function per il calcolo della soluzione di una generale equazione $AX = B$, con X, B matrici, che usi la fattorizzazione LU con pivoting parziale. Utilizzarla poi per il calcolo dell'inversa delle matrici non singolari

$$A = \begin{bmatrix} 3 & 5 & 7 \\ 2 & 3 & 4 \\ 5 & 9 & 11 \end{bmatrix}, \qquad A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & -4 & 6 & 8 \\ -1 & -2 & -3 & -1 \\ 5 & 7 & 0 & 1 \end{bmatrix},$$

confrontando i risultati ottenuti con l'output della funzione Matlab `inv(A)`.
Ripetere l'esercizio utilizzando nella function per il calcolo di X la fattorizzazione LU senza pivoting. Che cosa si osserva?

Esercizio 4 - Stabilità del metodo di eliminazione gaussiana.
Al variare di $n = 100, \ldots, 200$ costruire la matrice `A=gallery('orthog',n,1)` di

dimensione $n \times n$, definire la soluzione `xesatta = (1:n)'` e calcolare il termine noto come `b=A*xesatta`. Utilizzando le due diverse function implementate calcolare `xnopivot`, `xparziale`, e confrontarli con `xesatta` usando i grafici in scala semilogaritmica dell'errore relativo al variare di n. Confrontare pure `xesatta` con `xmatlab=A\b`.

Esercizio 5 - Numero di operazioni floating point.
Per verificare che il numero di operazioni è proporzionale a n^3, con n dimensione del sistema, riportare in un grafico in scala semilogaritmica il tempo impiegato a risolvere il sistema al variare di n. Il grafico dovrebbe essere (asintoticamente) una retta con pendenza 3. Per verificarne la pendenza, disegnare contemporaneamente anche la curva n^3 e controllare che siano parallele. La strategia di pivot non dovrebbe influenzare il risultato.