METODE NUMERICE: Laborator #2
Operaţii cu matrici în Matlab. Rezolvarea recursivă a sistemelor triunghiulare. Inversarea matricelor prin partiţionare.

Titulari curs: *Florin Pop, George-Pantelimon Popescu*

Responsabili Laborator: **Mădălina Hristache**

Obiective Laborator

În urma parcurgerii acestui laborator studentul va fi capabil să:

- factorizeze o matrice folosind una dintre metodele LU: Crout, Doolittle, Cholesky;
- rezolve recursiv un sistem triunghiular;
- inverseze o matrice prin metoda partiționării.

Factorizarea LU

Această metodă presupune descompunerea unei matrici A astfel:

$$A = LU \tag{1}$$

L - inferior triunghiulară, U - superior triunghiulară.

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \dots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \dots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} * \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \dots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix}. \tag{2}$$

În funcție de condițiile impuse, se disting factorizările: Crout ($u_{ii} = 1$), Doolittle ($l_{ii} = 1$) și Cholesky (pentru matrici simetrice și pozitiv-definite). Importanța factorizării LU constă în transformarea unui sistem cu matrice pătratică în două sisteme triunghiulare.

Factorizarea Crout

Sistemul de ecuații din forma generală a facorizării LU este supradererminat. Pentru rezolvarea lui, condiția ce se impune este: $u_{ii} = 1$, **pentru** $i = 1 : n$. Factorizarea Crout este:

pentru $p = 1 : n$

$$l_{ip} = a_{ip} - \sum_{k=1}^{p-1} l_{ik} \cdot u_{kp}, \ \textbf{pentru} \ i = p : n$$

$$u_{pj} = \frac{1}{l_{pp}} \cdot \left(a_{pj} - \sum_{k=1}^{p-1} l_{pk} \cdot u_{kj} \right), \ \textbf{pentru} \ j = p+1 : n$$

Factorizarea Doolittle

În acest caz, condiția ce se impune este : $l_{ii} = 1$, **pentru** $i = 1 : n$, iar factorizarea Doolittle este:

pentru $p = 1 : n$

$$u_{pj} = a_{pj} - \sum_{k=1}^{p-1} l_{pk} \cdot u_{kj}, \ \textbf{pentru} \ j = p : n$$

$$l_{ip} = \frac{1}{u_{pp}} \cdot \left(a_{ip} - \sum_{k=1}^{p-1} l_{ik} \cdot u_{kp} \right), \ \textbf{pentru} \ i = p+1 : n$$

Factorizarea Cholesky

Factorizarea Cholesky, impune ca $U = L^T$. Ea se aplică numai pentru matrici simetrice ($A = A^T$) și pozitiv definite ($x^T Ax > 0, \forall x \in R^n, x \neq 0$).

$$l_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2}, \ \textbf{pentru} \ i = 1 : n$$

$$l_{ij} = \frac{a_{ij} - \sum_{k=1}^{j-1} l_{ik} \cdot l_{jk}}{l_{jj}}, \ \textbf{pentru} \ j = 1 : i-1$$

Rezolvarea sistemelor triunghiulare

	Sistem inferior triunghiular	**Sistem superior triunghiular**
Forma	$\begin{array}{ccccccccc} a_{11}x_1 & & & & & & & = & b_1 \\ a_{21}x_1 & + & a_{22}x_2 & & & & & = & b_2 \\ \vdots & & \vdots & & \ddots & & & \vdots & \vdots \\ a_{n1}x_1 & + & a_{n2}x_2 & + & \cdots & + & a_{nn}x_n & = & b_n \end{array}$	$\begin{array}{ccccccccc} a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1n}x_n & = & b_1 \\ & & a_{22}x_2 & + & \cdots & + & a_{2n}x_n & = & b_2 \\ & & & & \ddots & & \vdots & \vdots & \vdots \\ & & & & & & a_{nn}x_n & = & b_n \end{array}$
Soluția	$x_i = \dfrac{b_i - \sum_{j=1}^{i-1} A_{ij} \cdot x_j}{A_{ii}}$, **pentru** $i = 1 : n$	$x_i = \dfrac{b_i - \sum_{j=i+1}^{n} A_{ij} \cdot x_j}{A_{ii}}$, **pentru** $i = n : 1$

Inversarea matricelor prin metoda partiționării

În unele cazuri (de exmplu când anumite zone ale matricei conțin elemente nule), se poate diviza matricea de inversat în patru submatrice A_1, A_2, A_3 și A_4, astfel încât matricele de pe diagonala principală A_1 și A_4

să fie pătratice:

$$A = \begin{bmatrix} A_1 & A_3 \\ A_2 & A_4 \end{bmatrix} \tag{3}$$

Dacă se notează inversa matricei A

$$X = A^{-1} = \begin{bmatrix} X_1 & X_3 \\ X_2 & X_4 \end{bmatrix} \tag{4}$$

este valabilă ecuația matriceală:

$$A \cdot A^{-1} = \begin{bmatrix} A_1 & A_3 \\ A_2 & A_4 \end{bmatrix} \cdot \begin{bmatrix} X_1 & X_3 \\ X_2 & X_4 \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}. \tag{5}$$

În final rezultă:

$$\begin{aligned}
X_1 &= \left(A_1 - A_3 \cdot A_4^{-1} \cdot A_2\right)^{-1} \\
X_2 &= -A_4^{-1} \cdot A_2 \cdot X_1 \\
X_3 &= -A_1^{-1} \cdot A_3 \cdot X_4 \\
X_4 &= \left(A_4 - A_2 \cdot A_1^{-1} \cdot A_3\right)^{-1}
\end{aligned}$$

Problema 1

Calculați inversa prin metoda partiționării pentru matricea:

$$A = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 1 & 2 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

Problema 2

Fiind dată matricea $A \in \mathbb{R}^{n \times n}$ şi un vector coloană $b \in \mathbb{R}$:

a) Să se implementeze o funcţie MATLAB care aduce matricea A în forma superior triunghiulară prin eliminare Gaussiană.

```
function [A, b] = Elim_Gauss(A,b)
```

b) Să se realizeze o funcţie MATLAB care rezolvă un sistem superior triunghiular de ecuaţii liniare. Folosindu-vă de funcţia `Elim_Gauss(A, b)` să scrie o funcţie de test care rezolvă sistemul de ecuaţii liniare dat prin A şi b iniţial.

```
function x = SST(A, b)
```

```
function x = test(A, b)
```

Problema 3

Rulați exemplul de cod pentru factorizarea Crout și faceți modificările necesare pentru a o transforma în factorizare Doolittle.

```
function [L U] = crout(A)
  [n n] = size(A);
  L = zeros(n);
  U = eye(n);
  L(1 : n, 1) = A(1 : n, 1);
  for k = 1 : n
    for i = k : n
        % --------------------------------
        s = 0;
        for m = 1 : k-1
            s = s + L(i, m) * U(m, k);
        endfor
        % --------------------------------
        % echivalent pentru calculul sumei
        % s = L(i, 1 : k-1) * U(1 : k-1, k);
        L(i, k) = A(i, k) - s;
    endfor
    for j = k+1 : n
        % --------------------------------
        s = 0;
        for m = 1 : k-1
            s =  s + L(k, m) * U(m, j);
        endfor
        % ---------------------------------
        % echivalent pentru calculul sumei
        % s = L(k, 1 : k-1) * U(1 : k-1, j);
        U(k, j) = (A(k, j) - s) / L(k, k);
    endfor
  endfor
endfunction
```

Listing 1: Factorizare Crout

Problema 4

Fie $T \in \mathbb{R}^{n \times n}$ o matrice simetrică, tridiagonală şi pozitiv definită. Scrieţi un algoritm de calcul al factorizării Cholesky $T = L \cdot L^T$, cu L inferior triunghiulară.

Hint: Din cazul general de factorizare $L \cdot U$ se deduce o formulă de recurenţă.

```
function [L, U] = choT(A)
```