

## TD 3 - Algèbre linéaire : méthode du pivot de Gauss

On considère une matrice carrée  $A$  de taille  $n \times n$ .

On suppose que la matrice  $A$  est inversible.

On souhaite associer à la matrice  $A$  une matrice échelonnée obtenue par opérations élémentaires sur les lignes de la matrice  $A$ .

On applique la méthode du pivot de Gauss.

Choix du pivot dans la méthode : pivot partiel.

Pour  $k$  entier allant de 1 à  $n - 1$ , on détermine, dans la colonne  $k$ , l'élément de plus grande valeur absolue parmi l'ensemble des coefficients situés sur ou sous la diagonale de la matrice  $A^{(k)}$  (coefficients s'écrivant  $a_{l,k}$  avec  $l \geq k$ ).

On convient que cet élément, non nul puisque  $A^{(k)}$  est inversible, est le  $k$ -ième pivot choisi pour la méthode de Gauss.

### Travail demandé :

Programmer une fonction qui prend en argument une matrice  $A$  et qui renvoie la matrice échelonnée obtenue par la méthode du pivot de Gauss (avec choix du pivot partiel).

Par exemple, la matrice échelonnée associée à :  $A = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$  est  $\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix}$

1. On demande à l'utilisateur de donner la dimension  $n$  de la matrice carrée  $A$  puis de rentrer  $A$ . Ceci peut se faire avec la commande `x_matrix('A :', A)`. L'argument  $A$  de cette commande est une matrice de taille  $n \times n$  considérée par défaut. On peut prendre  $A = \text{eye}(n, n)$  ou  $A = \text{toeplitz}[1, 1, \text{zeros}(1, n-2)]$ .
2. Programmer la recherche du pivot, pour une colonne de la matrice courante  $A^{(k)}$ . La commande Scilab `max` prend en argument une matrice  $M$  et renvoie la valeur du max de  $M$  ainsi qu'un vecteur ligne contenant les numéros de ligne et de colonne de cet élément max.
3. Programmer l'échange de deux lignes de la matrice courante puis les combinaisons de lignes permettant d'annuler les coefficients au-dessous du pivot pour cette matrice.
4. Afficher la matrice échelonnée associée à la matrice  $A$ .

## Script Scilab

```
// méthode du pivot de Gauss
// -----
M=evstr(x_matrix('entrer une matrice 3x3 :',rand(3,3)))
// -----
[m,n]=size(M); // m=n=3 pour ce script
// -----
for k=1:n-1
    [pivot_k, numlgn]=max(abs(M(k:n,k)));
    // numlgn contient la position du pivot
    // dans le vecteur colonne extrait ce qui explique
    // la ligne suivante
    numlgnpivot=numlgn+k-1;
    // on effectue la permutation des lignes
    L=[];
    L=M(numlgnpivot,:);
    M(numlgnpivot,:)=M(k,:);
    M(k,:)=L;
    // On annule ensuite les coefficients de M situés sous le pivot
    // dans la colonne k, par combinaisons de lignes
    for i=k+1:n
        alpha_k=M(i,k)/M(k,k); //abs(M(k,k))=pivot_k
        M(i,k:n)=M(i,k:n)-alpha_k.*M(k,k:n);
    end
end
// affichage de la matrice échelonnée
disp(M);
```