

ANALYSE NUMÉRIQUE

Corrigés des Travaux Pratiques 2024 – 2025

Séance 4

1. Une possible manière de faire :

```
clear; close; format short
A = [1 -100 0 0; ...
     1 1 -100 0; ...
     0 1 1 -100; ...
     -100 0 1 1];
Asv = A;
% élément 4 est le plus grand en valeur absolue
% sous la diagonale de la colonne 1
% => permutation des lignes 1-4
P1 = eye(4); P1([1,4], :) = P1([4,1], :);
A = P1*A
% construction de L1
L1 = eye(4); L1(2:4,1) = -A(2:4,1)./A(1,1);
A = L1*A
% élément 4 est le plus grand en valeur absolue
% sous la diagonale de la colonne 2
% => permutation des lignes 2-4
P2 = eye(4); P2([2,4], :) = P2([4,2], :);
A = P2*A
% construction de L2
L2 = eye(4); L2(3:4,2) = -A(3:4,2)./A(2,2);
A = L2*A
% élément 4 est le plus grand en valeur absolue
% sous la diagonale de la colonne 3
% => permutation des lignes 3-4
P3 = eye(4); P3([3,4], :) = P3([4,3], :);
A = P3*A
% construction de L3
L3 = eye(4); L3(4,3) = -A(4,3)./A(3,3);
U = L3*A
% on a donc
%          L3*P3*L2*P2*L1*P1*A = U
Lp3 = L3;
Lp2 = P3*L2*P3;
Lp1 = P3*P2*L1*P2*P3;
P    = P3*P2*P1;
% ou de manière équivalente
%          Lp3*Lp2*Lp1*P*A = U
% en utilisant les propriétés 1 & 2 du chapitre 3
% il suffit de copier-coller les colonnes
% correspondantes de Lpi dans L, et d'inverser
% le signe des éléments sous la diagonale principale
L = eye(4);
```

```

L(4,3)    = -Lp3(4,3);
L(3:4,2) = -Lp2(3:4,2);
L(2:4,1) = -Lp1(2:4,1);
    % vérification
[l u p] = lu(Asv);
norm(L-l)
norm(U-u)
norm(P-p)
    % alors?

```

2. Une possible manière de faire :

```

clear; close; format short
A = [1 -100 0 0; ...
     1 1 -100 0; ...
     0 1 1 -100; ...
     -100 0 1 1];
Asv = A;
    % transformation 1
x = A(1:4,1);
x(1) = x(1)+sign(x(1))*norm(x);
v1 = x/norm(x);
A -= v1*(2*(v1'*A));
    % Q1 = eye(4) - 2*v1*v1';
    % A = Q1*A;
    %
    % transformation 2
x = A(2:4,2);
x(1) = x(1)+sign(x(1))*norm(x);
v2 = x/norm(x);
A(2:4,2:4) -= v2*(2*(v2'*A(2:4,2:4)));
    % Q2 = eye(4); Q2(2:4,2:4) = eye(3) - 2*v2*v2';
    % A = Q2*A;
    %
    % transformation 3
x = A(3:4,3);
x(1) = x(1)+sign(x(1))*norm(x);
v3 = x/norm(x);
A(3:4,3:4) -= v3*(2*(v3'*A(3:4,3:4)));
    % Q3 = eye(4); Q3(3:4,3:4) = eye(2) - 2*v3*v3';
    % A = Q3*A;
    %
    % calculer Q
R = A;
Q = eye(4);
Q(3:4,3:4) -= v3*(2*(v3'*Q(3:4,3:4)));
Q(2:4,2:4) -= v2*(2*(v2'*Q(2:4,2:4)));
Q          -= v1*(2*(v1'*Q));
    % vérification
[q r] = qr(Asv)

```

```
% quelle est la différence entre les matrices
% q,r et les matrices Q,R?
% a-t-on une factorisation QR dans les deux cas?
norm(Asv-Q*R)
% alors?
```