Homework 3 Aaryan Bhagat 862468325 CS210

① $A_1 = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$

To Solve $Ax = b$

$\det(A) = -2 \neq 0$

$\therefore$ Unique Solution

Hence

$x_1 + 2x_2 = b_1 \rightarrow$ (I)

$3x_1 + 4x_2 = b_2 \rightarrow$ (II)

Solving (I) and (II)

$x_1 = b_2 - 2b_1$ $\quad x_2 = \frac{3b_1 - b_2}{2}$

$A_2 = \begin{pmatrix} 1 & 2 & 6 \\ 2 & 4 & 7 \\ 3 & 6 & 5 \end{pmatrix}$

To Solve $Ax = b$

$\det(A) = 0$

$\therefore$ No Solution or infinite solution

Hence

$x_1 + 2x_2 + 6x_3 = b_1$

$2x_1 + 4x_2 + 7x_3 = b_2$

$3x_1 + 6x_2 + 5x_3 = b_3$

Using LU Decomposition

$LU = A$

$A = \begin{bmatrix} 1 & 2 & 6 \\ 2 & 4 & 7 \\ 3 & 6 & 5 \end{bmatrix}$

① $R_2 \rightarrow R_2 - 2R_1$

② $R_3 \rightarrow R_3 - 3R_1$

We get $U = \begin{bmatrix} 1 & 2 & 6 \\ 0 & 0 & -5 \\ 0 & 0 & -13 \end{bmatrix}$

From U and Elementary Matrices we get

$L = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 0 & 1 \end{bmatrix}$

Similarly Reduced Row Echelon Form

(III) $R_2 \Rightarrow R_2 - 2R_1$

(IV) $R_3 \rightarrow R_3 - 3R_1$

(V) $R_2 \rightarrow -R_2/5$

(VI) $R_1 \rightarrow R_1 - 6R_2$

(VII) $R_3 \rightarrow R_3 + 13R_2$

Doing Same Operations

$$\therefore \mathbb{R} = \begin{bmatrix} 1 & 2 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{array}{cc} \begin{bmatrix} (17b_1 - 6b_2)/5 \\ (b_2 - 2b_1)/5 \\ (b_3 - 3b_1) + \frac{13}{5}(b_2 - 2b_1) \end{bmatrix} & \begin{array}{c} B1 \\ B2 \\ B3 \end{array} \end{array}$$

Clearly this system is inconsistent, hence NO Sollution

$\because$ $0x_1 + 0x_2 + 0x_3 = B_3$ [which is not possible]

$$A_3 = \begin{pmatrix} 1 & 3 \\ 2 & 4 \\ 5 & 7 \end{pmatrix}$$

To Solve $Ax = b$

$x_1 + 3x_2 = b_1$
$2x_1 + 4x_2 = b_2$
$5x_1 + 7x_2 = b_3$

For vector space

$$B = \begin{bmatrix} (4\beta - \alpha)/3 \\ \beta \\ \alpha \end{bmatrix} \rightarrow (a)$$

$\rightarrow$ Infinite Solutions of the type

$$\begin{bmatrix} \frac{1}{8}(5\beta - 2\alpha) \\ \frac{1}{6}(4\alpha - 7\beta) \end{bmatrix} = X$$

$\alpha, \beta \in \mathbb{R}$

Otherwise No solution

So for a fixed b_1 and b_2, if they are of the type (a) then its unique solution else ~~or~~ No Solution

$$A_4 = \begin{pmatrix} 1 & 2 & 4 \\ 3 & 6 & 7 \end{pmatrix}$$

To Solve $Ax = B$

$x_1 + 2x_2 + 4x_3 = b_1 \rightarrow$ (I)
$3x_1 + 6x_2 + 7x_3 = b_2 \rightarrow$ (II)

$LU = A$

$$\begin{bmatrix} 1 & 0 \\ 3 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 4 \\ 0 & 0 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

$$\begin{bmatrix}1&0\\3&1\end{bmatrix}\begin{bmatrix}y_1\\y_2\end{bmatrix}=\begin{bmatrix}b_1\\b_2\end{bmatrix}$$

$y_1=b_1$

$3y_1+y_2=b_2$

$\therefore\ y_1=b_1$ and $y_2=b_2-3b_1$

$$\begin{bmatrix}1&2&4\\0&0&-5\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}=\begin{bmatrix}y_1\\y_2\end{bmatrix}$$

$x_1+2x_2+4x_3=y_1$

$-5x_3=y_2$

$x_3=(3b_1-b_2)/5$

$x_1+2x_2=b_1-\frac{4}{5}(3b_1-b_2)$

So for given $\begin{bmatrix}b_1\\b_2\end{bmatrix}$

We have infinite solutions of the form

$$\begin{bmatrix}(4b_2-7b_1)/5-2t\\t\\(3b_1-b_2)/5\end{bmatrix}\quad t\in\mathbb{R}$$

However if we look at equations (1) and (11) we see that we can easily cancel out x_1 and x_2, only x_3 remains with a constant value.

Hence the solution actually is immaterial of x_1 and x_2

Q3

(a) Take a triangular matrix = $\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ 0 & \ddots & \\ \vdots & & \\ 0 & & a_{nn} \end{bmatrix}$

Determinant = $a_{11} \begin{vmatrix} a_{22} & & \\ & \ddots & \\ & & a_{nn} \end{vmatrix} - 0 \begin{vmatrix} \quad \end{vmatrix} + 0 \begin{vmatrix} \quad \end{vmatrix}$ — [Using determinant formula across column]

$\Rightarrow a_{11} a_{22} \begin{vmatrix} a_{33} & & \\ & \ddots & \\ & & a_{nn} \end{vmatrix}$

Similarly expanding we get

$$\det(A) = a_{11}\, a_{22} \cdots a_{nn}.$$

So if anyone is 0 whole det is 0 hence matrix is Singular

For ~~Upper~~ Lower Triangular Matrix

$$A = \begin{bmatrix} a_{11} & 0 & 0 & \cdots \\ a_{21} & a_{22} & & \\ \vdots & & \ddots & \\ a_{n1} & \cdots & \cdots & a_{nn} \end{bmatrix}$$

We get the same formula for det(A) by ~~making~~ opening the determinant formula with cc across row

Hence Answer is **TRUE**

(b)

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22} & & \\ \vdots & & \ddots & \\ 0 & & & a_{nn} \end{bmatrix}_{n \times n} \qquad B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ 0 & b_{22} & & \vdots \\ \vdots & & \ddots & \vdots \\ & & & b_{nn} \end{bmatrix}_{n \times n}$$

Output Matrix = C

General Formula $C_{ij} = \sum_{1 \le k \le n} a_{ik} \times b_{kj}$

$a_{ij} = 0 \ \forall \ i > j \qquad b_{ij} = 0 \ \forall \ i > j$

$$C_{ij} = \underbrace{\sum_{1 \le k \le j} (\ )}_{0\ [\because\ a_{ik} = 0]} + \underbrace{\sum_{j+1 \le k \le i} (\ )}_{0\ [\because\ b_{kj} = 0]} + \underbrace{\sum_{i+1 \le k \le n} (\ )}_{0\ [\because\ a_{ik} = 0]}$$

$\therefore C_{ij} = 0 \ \forall \ i > j$

Hence C_{ij} is also upper triangular

Hence <u>TRUE</u>

(c) Given the form $A = LU$ and hence $LUx = b$

(i) We can first solve $L\vec{y} = \vec{b}$ using forward substitution which $O(n^2)$ to obtain $\vec{y}$

(ii) Then we can solve $U\vec{x} = \vec{y}$ by back substitution in $O(n^2)$

Hence whole operation is $O(2n^2) = O(n^2)$

Hence <u>TRUE</u>

③ $A = \begin{pmatrix} 2 & 1 & 0 \\ 6 & 4 & 2 \\ 0 & 3 & 5 \end{pmatrix}$

Doing LU decomposition

(a) $R_2 \Rightarrow R_2 - 3R_1$ $\quad E_1 = \begin{bmatrix} 1 & 0 & 0 \\ -3 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

(b) $R_3 \to R_3 - 3R_2$ $\quad E_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -3 & 1 \end{bmatrix}$

$U = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 1 & 2 \\ 0 & 0 & -1 \end{bmatrix}$ $\quad \cancel{L = \begin{bmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 0 & 3 & 1 \end{bmatrix}}$

$$E_1 E_2 A = U$$

$$A = (E_2^{-1} E_1^{-1}) U$$

$$\therefore E_2^{-1} E_1^{-1} = L$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{bmatrix} = L$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 0 & 3 & 1 \end{pmatrix} = L$$

$$\therefore L^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ -3 & 1 & 0 \\ 0 & -3 & 1 \end{bmatrix} = E$$

$$\therefore \begin{pmatrix} 2 & 1 & 0 \\ 6 & 4 & 2 \\ 0 & 3 & 5 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 0 & 3 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 0 \\ 0 & 1 & 2 \\ 0 & 0 & -1 \end{pmatrix}$$

$$A = L \quad U$$

(4) $A = \begin{pmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ b & c & 1 \end{pmatrix}$ $E_1 = \begin{pmatrix} 1 & 0 & 0 \\ -a & 1 & 0 \\ -b & 0 & 1 \end{pmatrix}$

$E_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -c & 1 \end{pmatrix}$

(a) $E_2 E_1 = \begin{pmatrix} 1 & 0 & 0 \\ -a & 1 & 0 \\ ac & -c & 1 \end{pmatrix} = E$

$EA = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = I$

(b) $E_1^{-1} E_2^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ b & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & c & 1 \end{pmatrix}$

$= \begin{pmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ b & c & 1 \end{pmatrix} = A = L$

(5) $A_1 = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 2 \\ 0 & 2 & 2 \end{pmatrix}$

(a) $R_2 \to R_2 - R_1$

$$\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 2 \\ 0 & 2 & 2 \end{pmatrix} \qquad E_1 = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

(b) $R_3 \to R_3 - 2R_2$

$$U = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 2 \\ 0 & 0 & -2 \end{pmatrix} \qquad E_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -2 & 1 \end{pmatrix}$$

$$E_2^{-1} E_1^{-1} = L = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix} = L$$

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 2 \\ 0 & 2 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 2 \\ 0 & 0 & -2 \end{pmatrix} \to (1)$$

$$A = LDL^T \Rightarrow \underbrace{[\ \ ]}_{L} \underbrace{\begin{bmatrix} d_1 & 0 & 0 & 0 & \dots \\ 0 & d_2 & 0 & 0 & \dots \\ \vdots & & \ddots & & \\ 0 & \dots & & 0 & d_n \end{bmatrix}}_{D} \underbrace{[\ \ ]}_{L^T}$$

Matrix D will only multiply to diagonal elements of L^T

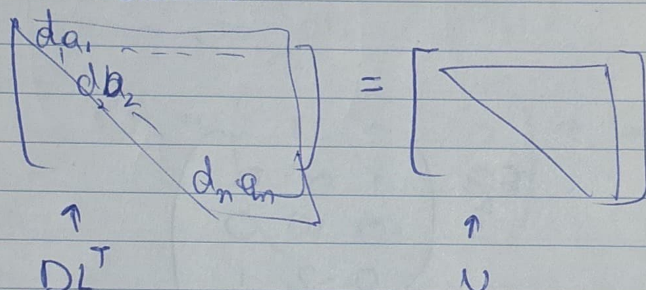
Hence we have

$LB = LDL^T$ Where $B = DL^T$

But We also have $A = LU$

$\therefore$ We can get D from U

$DL^T = U$



So from (1) we know that diagonal elements of L are unity
Hence D = diagonal elements of U

$$\therefore LDL^T = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix}}_{L} \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -2 \end{bmatrix}}_{D} \underbrace{\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix}}_{L^T}$$

$$A_2 = \begin{pmatrix} a & a & 0 \\ a & a+b & b \\ 0 & b & b+c \end{pmatrix}$$

(i) $R_2 \to R_2 - R_1$

$$\begin{pmatrix} a & a & 0 \\ 0 & b & b \\ 0 & b & b+c \end{pmatrix} \qquad E_1 = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

(ii) $R_3 \to R_3 - R_2$

$$\begin{pmatrix} a & a & 0 \\ 0 & b & b \\ 0 & 0 & c \end{pmatrix} = U \qquad E_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix}$$

$\therefore \quad E_2 E_1 A_2 = U$

$A_2 = E_1^{-1} E_2^{-1} U$

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{E_2^{-1}} \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{E_1^{-1}} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} = L$$

$$\therefore \begin{pmatrix} a & a & 0 \\ a & a+b & b \\ 0 & b & b+c \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} a & a & 0 \\ 0 & b & b \\ 0 & 0 & c \end{pmatrix}$$

$$\underset{\substack{\uparrow \\ E_1^{-1}}}{\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}} \underset{\substack{\uparrow \\ E_2^{-1}}}{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}} = \cancel{0}L$$

$$\therefore L = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

$$\therefore \underset{A}{\begin{pmatrix} a & a & 0 \\ a & a+b & b \\ 0 & b & b+c \end{pmatrix}} = \underset{L}{\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}} \underset{U}{\begin{pmatrix} a & a & 0 \\ 0 & b & b \\ 0 & 0 & c \end{pmatrix}}$$

Now Since I have proved D = diagonal elements of U in the previous question, I will just apply it here

<u>Hence</u> [Diagonal elements of L here also are unity]

$$A = LDL^T$$

$$\begin{pmatrix} a & a & 0 \\ a & a+b & b \\ 0 & b & b+c \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

(6) $P_1 A P_2 = LU \Rightarrow \begin{pmatrix} 1 & 0 \\ 1/2 & 1 \end{pmatrix} \begin{pmatrix} 4 & 3 \\ 0 & -1/2 \end{pmatrix} = \begin{pmatrix} 4 & 3 \\ 2 & 1 \end{pmatrix}$

Steps are

$\text{New } A = \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}$

→ Finding Pivot
→ Dividing by first pivot
→ Removing $l_1 u_1^*$ from A

Pivot for A is $a_{22} = 4$ $\quad i = j = 2$

Dividing by pivot and permuting we get

$$\underset{P_1}{\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}} \underset{A}{\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}} \underset{Q_1}{\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}}$$

Using multiplication matrix $M_1 = \begin{bmatrix} 1 & 0 \\ -3/4 & 1 \end{bmatrix}$

Hence we get

$M_1 P_1 A Q_1 = U = \begin{bmatrix} 4 & 2 \\ 0 & -1/2 \end{bmatrix}$

Equivalent to

$$\begin{bmatrix} 3/4 \\ 1 \end{bmatrix} [2, 4] + \begin{bmatrix} -1/2 & 0 \\ 0 & 0 \end{bmatrix} = l_1 u_1^* + l_2 u_2^*$$

After solving for L and permuting, we get

$$P_1 A P_2 = \begin{bmatrix} 1 & 0 \\ 3/4 & 1 \end{bmatrix} \begin{bmatrix} 4 & 2 \\ 0 & -1/2 \end{bmatrix}$$

- I have tried the inbuilt Matlab Function and stored the outputs in L1 and U1, each variable of L1 and U1 is printed individually to show their output, followed by multiplication of L1 and U1 to see if the original A gets back or not.

```
>> A = [1e-20 1; 1 1]

A =

    0.0000    1.0000
    1.0000    1.0000
>> [L1, U1] = lu(A)

L1 =

    0.0000    1.0000
    1.0000         0


U1 =

     1     1
     0     1
>> L(0,0)
Incorrect number or types of inputs or outputs for function L.

>> L1(0,0)
Index in position 1 is invalid. Array indices must be positive integers or logical values.

>> L1(1, 1)

ans =

   1.0000e-20
>> L1(1, 2)

ans =

     1
>> L1(2, 1)

ans =

     1
>> L1(2, 2)

ans =

     0
>> U1(1, 1)

ans =

     1
>> U1(1, 2)

ans =

     1
>> U1(2, 1)

ans =

     0
>> U1(2, 2)
```

```
ans =

     1
>> C1 = L1 * U1

C1 =

    0.0000    1.0000
    1.0000    1.0000
>> C1(1, 1)

ans =

   1.0000e-20
>> C1(1,2)

ans =

     1
>> C1(2,1)

ans =

     1
>> C2(2,2)

ans =

     1
>>
```

- I have used the lu_cs210() function and stored the output in L0 and U0, I have then followed the same procedure as in the above step and stored L0 * U0 in C0.

```
>> [L0, U0] = lu_cs210(A)

n =

     2


L =

     0     0
     0     0


U =

     0     0
     0     0


A2 =

    0.0000    1.0000
    1.0000    1.0000


L0 =

   1.0e+20 *

    0.0000         0
    1.0000    0.0000
```

```
U0 =

   1.0e+20 *

    0.0000    0.0000
         0   -1.0000
>> L0(1,1)

ans =

     1
>> L0(1,2)

ans =

     0
>> L0(2,1)

ans =

   1.0000e+20
>> L0(2,2)

ans =

     1
>> U0(1,1)

ans =

   1.0000e-20
>> U0(1,2)

ans =

     1
>> U0(2,1)

ans =

     0
>> U0(2,2)

ans =

  -1.0000e+20
>>
```

Original Code (which was in the homework used for part a and part b above)

```
function [L,U] = lu_cs210(A)
%UNTITLED2 Summary of this function goes here
%   Detailed explanation goes here
n = size(A, 1)
L = zeros(size(A))
U = zeros(size(A))
A2 = A
for k = 1:n
   if A2(k, k) == 0
       'Encountered 0 pivot. Stopping'
       return
```

```
end
for i = 1:n
    L(i, k) = A2(i, k)/A2(k, k);
    U(k, i) = A2(k, i);
end
for i = 1:n
    for j = 1:n
        A2(i, j) = A2(i, j) - L(i, k) * U(k, j);
    end
end
```

Modified Code in Python with partial pivoting

```
import numpy as np

def LU_pivot(A):
    '''
    Our algorithm will modify the matrix A to convert it into
    an upper triangular matrix then from that onwards we will find L

    Algorithm is not an in place algorithm
    P_hat, L_hat are the extra variables where ultimately the value will be
stored
    '''
    row, col = A.shape
    P = np.identity(row) # Permutation matrix
    L = np.identity(row) # Lower Triangular matrix
    U = A.copy() # Upper Triangular matrix initialized as copy of original
matrix A
    P_hat = np.identity(row) # initialized
    L_hat = np.zeros((row, row)) # initialized
    for elem in range(0, row-1):
        i = np.argmax(abs(U[elem:, elem])) #finding index
        i = i + elem
        if i != elem:
            P = np.identity(row)
            P[[i, elem], elem:row] = P[[elem, i], elem:row]
            U[[i, elem], elem:row] = U[[elem, i], elem:row]
            P_hat = np.dot(P, P_hat) # new permutation matrix
            L_hat = np.dot(P, L_hat) # Updated value of L into L_hat
        L = np.identity(row)
        for l in range(elem+1, row):
            L[l, elem] = -(U[l, elem] / U[elem, elem]) # dividing the
entire column
```

```
            L_hat[l, elem] = (U[l, elem] / U[elem, elem]) # storing the values in L_hat
        U = np.dot(L, U)
    np.fill_diagonal(L_hat, 1) # Because by default choice we make the Lower Triangular matrix having all the unity diagonals
    return P_hat, L_hat, U

A = [[1,2,3], [3,2,1], [1, -1, 0]]
A = np.array(A)
P1, L1, U1 = LU_pivot(A)
print(P1)
print(L1)
print(U1)
```

- Output for A = [[1,2,3], [3,2,1], [1, -1, 0]]
    - Python one

```
P1
[[0. 1. 0.]
 [0. 0. 1.]
 [1. 0. 0.]]
L1
[[ 1.          0.          0.        ]
 [ 0.33333333  1.          0.        ]
 [ 0.33333333 -0.8         1.        ]]
U1
[[ 3.          2.          1.        ]
 [ 0.         -1.66666667 -0.33333333]
 [ 0.          0.          2.4       ]]
```

    -
    - Matlab One

```
>> A = [1 2 3; 3 2 1; 1 -1 0]

A =

     1     2     3
     3     2     1
     1    -1     0

>> [L, U, P] = lu(A2)
Unrecognized function or variable 'A2'

>> [L, U, P] = lu(A)

L =

    1.0000         0         0
    0.3333    1.0000         0
    0.3333   -0.8000    1.0000


U =

    3.0000    2.0000    1.0000
         0   -1.6667   -0.3333
         0         0    2.4000


P =

     0     1     0
     0     0     1
     1     0     0
```

    -
- Output for A = [10^-20 1; 1 1]

- Matlab Output

```
>> A = [1e-20 1; 1 1]

A =

    0.0000    1.0000
    1.0000    1.0000

>>
>> [L, U, P] = lu(A)

L =

    1.0000         0
    0.0000    1.0000


U =

     1     1
     0     1


P =

     0     1
     1     0

>> L(1,1)

ans =

     1

>> L(1,2)

ans =

     0

>> L(2,1)

ans =

   1.0000e-20

>> L(2,2)

ans =

     1
```

-
- Python Output

```
P1
[[0. 1.]
 [1. 0.]]
L1
[[1.e+00 0.e+00]
 [1.e-20 1.e+00]]
U1
[[1. 1.]
 [0. 1.]]
```

-