2.4 The $PA = LU$ Factorization

Recall that from 2.3 we discovered that "naive" Gaussian elimination encounters difficulties with "swamping."
To avoid such complications we revise the LU factorization technique through the introduction of row swapping, or so-called partial pivoting.

The first step of the partial pivoting procedure asks that we select the row with the largest absolute value entry in column 1, so:

$$\boxed{|a_{p1}| \geq |a_{i1}|} \quad \text{for all } 1 \leq i \leq n$$

and then we swap this row with the first row, so that we can then use this largest value as the pivot in column 1.
We then use the multipliers:

$$\boxed{m_{i1} = \frac{a_{i1}}{a_{11}}} \qquad \text{Note: } |m_{i1}| \leq 1 \quad *$$

To eliminate entry a_{i1}.
We continue this procedure with "pivoting" (i.e. row swapping) for each subsequent column until the conclusion of Gaussian elimination.
We note that at each step, the multiplier will be no greater than 1, which assures us that we avoid the problem of swamping.

2.4 The $PA = LU$ Factorization

Recall that from 2.3 we discovered that "naive" Gaussian elimination encounters difficulties with "swamping."
To avoid such complications we revise the LU factorization technique through the introduction of row swapping, or so-called partial pivoting.

The first step of the partial pivoting procedure asks that we select the row with the largest absolute value entry in column 1, so:

$$\boxed{|a_{p1}| \geq |a_{i1}|} \quad \text{for all} \quad 1 \leq i \leq n$$

and then we swap this row with the first row, so that we can then use this largest value as the pivot in column 1.
We then use the multipliers:

$$\boxed{m_{i1} = \frac{a_{i1}}{a_{11}}} \qquad \text{Note: } |m_{i1}| \leq 1 \quad *$$

to eliminate entry a_{i1}.
We continue this procedure with "pivoting" (i.e. row swapping) for each subsequent column until the conclusion of Gaussian elimination.
We note that at each step, the multiplier will be no greater than 1, which assures us that we avoid the problem of swamping.

Ex. Use Gaussian elimination with partial pivoting.

$$\left[\begin{array}{cc|c} 1 & 1 & 3 \\ 3 & -4 & 2 \end{array}\right] \quad \text{Note: } |a_{21}| = 3 > |a_{11}| = 1$$

so <u>swap</u>.

$$\sim \left[\begin{array}{cc|c} 3 & -4 & 2 \\ 1 & 1 & 3 \end{array}\right] \sim r_2^* \to r_2 - \tfrac{1}{3} \cdot r_1$$

$$\left[\begin{array}{cc|c} 3 & -4 & 2 \\ 0 & \frac{7}{3} & \frac{7}{3} \end{array}\right] \to x_1 = 2,\ x_2 = 1$$

(multiplier was $\underline{\frac{1}{3}}$)

Ex. Use Gaussian elimination with partial pivoting.

$$\left.\begin{aligned} x_1 - x_2 + 3x_3 &= -3 \\ -x_1 \quad\quad - 2x_3 &= 1 \\ 2x_1 + 2x_2 + 4x_3 &= 0 \end{aligned}\right\} \to \left[\begin{array}{ccc|c} 1 & -1 & 3 & -3 \\ -1 & 0 & -2 & 1 \\ 2 & 2 & 4 & 0 \end{array}\right] \text{swap}$$

max in col. #1

$$\sim \left[\begin{array}{ccc|c} 2 & 2 & 4 & 0 \\ -1 & 0 & -2 & 1 \\ 1 & -1 & 3 & -3 \end{array}\right] \quad r_2^* \to r_2 - \left(\tfrac{-1}{2}\right) \cdot r_1$$

$$\sim \left[\begin{array}{ccc|c} 2 & 2 & 4 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & -1 & 3 & -3 \end{array}\right] \quad r_3^* \to r_3 - \tfrac{1}{2} \cdot r_1 \sim \left[\begin{array}{ccc|c} 2 & 2 & 4 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & -2 & 1 & -3 \end{array}\right] \text{swap}$$

$|a_{32}| > |a_{22}|$

$$\sim \left[\begin{array}{ccc|c} 2 & 2 & 4 & 0 \\ 0 & -2 & 1 & -3 \\ 0 & 1 & 0 & 1 \end{array}\right] \sim r_3^* \to r_3 - \tfrac{-1}{2} \cdot r_2$$

$$\sim \left[\begin{array}{ccc|c} 2 & 2 & 4 & 0 \\ 0 & -2 & 1 & -3 \\ 0 & 0 & \frac{1}{2} & -\frac{1}{2} \end{array}\right] \to \text{Back-sub} \to \vec{x} = \langle 1, 1, -1 \rangle.$$

Prior to formalizing the PA=LU (partial pivoting) algorithm, we first need to review several key properties of permutation matrices.

Def. A permutation matrix is a nxn matrix resulting from a sequence of (row) permutations applied to I_n, the identity matrix. (each row/col contains one "1", rest are zeros).

Exs. $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, $\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$, etc. (There are n! nxn perm matrices - why?)

Thm Fundamental Theorem of Permutation Matrices

Let P be the nxn matrix formed by applying a particular sequence of row exchanges to I_n; then $\underline{PA}$ is the matrix obtained by applying the same set of row exchanges to A.

Ex. $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \rightarrow$ formed: Take I_3, swap row 2 $\leftrightarrow$ row 3

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} = \begin{bmatrix} a & b & c \\ g & h & i \\ d & e & f \end{bmatrix}.$$

Note,

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$ } row 2/row 3 swapped in I_3.

PA=LU factorization

(Partial pivoting)

$$\underbrace{P}_{\text{perm. Matrix}} \cdot \overset{\text{coeff. matrix}}{A} = \underbrace{L}_{\text{lower-}\Delta \text{ (contains multipliers)}} \cdot \overset{\text{upper-}\Delta}{U}$$

Ex. Find the PA=LU factorization

$$A = \begin{bmatrix} 2 & 1 & 5 \\ 4 & 4 & -4 \\ 1 & 3 & 1 \end{bmatrix} \quad \text{row}_1 \leftrightarrow \text{row}_2 \; ; \quad \text{"}P\text{"} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\sim \begin{bmatrix} 4 & 4 & -4 \\ 2 & 1 & 5 \\ 1 & 3 & 1 \end{bmatrix} \sim r_2^* \to r_2 - \underbrace{\tfrac{1}{2}}_{\text{multiplier}} r_1 \sim \begin{bmatrix} 4 & 4 & -4 \\ \frac{1}{2} & -1 & 7 \\ 1 & 3 & 1 \end{bmatrix}$$

store multiplier

$$\sim r_3^* \to r_3 - \underline{\tfrac{1}{4}} r_1 \sim \begin{bmatrix} 4 & 4 & -4 \\ \frac{1}{2} & -1 & 7 \\ \frac{1}{4} & 2 & 2 \end{bmatrix} \text{ swap}$$

store. need row swap!

update P $\to \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \sim \text{"}P\text{"} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$

swap rows!

$$\sim \begin{bmatrix} 4 & 4 & -4 \\ \frac{1}{4} & 2 & 2 \\ \frac{1}{2} & -1 & 7 \end{bmatrix} \sim r_3^* \to r_3 - \underline{\left(-\tfrac{1}{2}\right)} r_2 \sim \begin{bmatrix} 4 & 4 & -4 \\ 0 & 2 & 2 \\ 0 & 0 & 8 \end{bmatrix} = \underline{U}.$$

So,
$$\underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}}_{P}\underbrace{\begin{bmatrix} 2 & 1 & 5 \\ 4 & 4 & -4 \\ 1 & 3 & 1 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 1 \end{bmatrix}}_{L} \cdot \underbrace{\begin{bmatrix} 4 & 4 & -4 \\ 0 & 2 & 2 \\ 0 & 0 & 8 \end{bmatrix}}_{U}$$

(matrix of multipliers

Q: Once we have the factorization: $PA = LU$, how do we efficiently perform back-sub? Two steps:

$$A\vec{x} = \vec{b} \longrightarrow \underbrace{PA}\vec{x} = P\vec{b}$$

$$\boxed{LU\vec{x} = P\vec{b}}$$ } "Expensive" part of algorithm

① Solve $L\vec{c} = P\vec{b}$ for $\vec{c}$

② Solve $U\vec{x} = \vec{c}$ for $\vec{x}$.

} "cheap" part

Ex. Solve the system $A\vec{x} = \vec{b}$ with A as in the previous example.

$$A = \begin{bmatrix} 2 & 1 & 5 \\ 4 & 4 & -4 \\ 1 & 3 & 1 \end{bmatrix}, \quad \vec{b} = \begin{bmatrix} 5 \\ 0 \\ 6 \end{bmatrix}$$

① Solve $L\vec{c} = P\vec{b}$ for $\vec{c}$ →

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 1 \end{bmatrix}}_{L}\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}}_{P}\underbrace{\begin{bmatrix} 5 \\ 0 \\ 6 \end{bmatrix}}_{b} = \begin{bmatrix} 0 \\ 6 \\ 5 \end{bmatrix} \longrightarrow$$

$\boxed{c_1 = 0,\ c_2 = 6,\ c_3 = 8}$

(2) Solve $U\vec{x} = \vec{c}$ for $\vec{x}$:

$$\underbrace{\begin{bmatrix} 4 & 4 & -4 \\ 0 & 2 & 2 \\ 0 & 0 & 8 \end{bmatrix}}_{U} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 \\ 6 \\ 8 \end{bmatrix}}_{\vec{c}}$$

$\to \vec{x} = \langle -1, 2, 1 \rangle.$

$\boxed{\text{Ex.}}$ Solve with $PA = LU$ factorization.

$$\begin{bmatrix} 2 & 3 \\ 3 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 3 \\ 3 & 2 \end{bmatrix} \sim \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix} \qquad P = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$\sim r_2^* \to r_2 - \underline{\tfrac{2}{3}} \cdot r_1 \sim \begin{bmatrix} 3 & 2 \\ \frac{2}{3} & \frac{5}{3} \end{bmatrix}$$

$$\to \underset{P}{\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}} \underset{A}{\begin{bmatrix} 2 & 3 \\ 3 & 2 \end{bmatrix}} = \underset{L}{\begin{bmatrix} 1 & 0 \\ \frac{2}{3} & 1 \end{bmatrix}} \underset{U}{\begin{bmatrix} 3 & 2 \\ 0 & \frac{5}{3} \end{bmatrix}}$$

(1) Solve: $L\vec{c} = P\vec{b} \to \vec{c} = \langle 1, 4 \rangle$

(2) Solve $U\vec{x} = \vec{c} \to \vec{x} = \langle -1, 2 \rangle.$

<u>Note that</u> <u>every</u> $n \times n$ matrix has a $PA = LU$ factorization.