2.8. Elementary Matrices (elmats)

The row operations (ops), or column, that take place in GE/GJ techniques may be conveniently represented as the product of some elementary matrices (elmats) of the form:

$$E = I - xy^T \quad , \quad x \text{ and } y \text{ col vecs.}$$

They are rank-1 modifications of the identity matrix. If $x^Ty \neq 1$, the inverse E^{-1} is guaranteed to exist:

$$E^{-1} = (I - xy^T)^{-1} \overset{\text{M.I.L.}}{=} I + \frac{xy^T}{1 - x^Ty}$$

MATRIX INVERSION LEMMA

When it exists, E^{-1} has the same structure as the original matrix E, apart from a scaling factor $1 - x^Ty$.

There are essencially 3 types of elmats

Type I

row/col exchange

$$E = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$E = I - uu^T$$

$$u = e_1 - e_2$$

$$r_1 \longleftrightarrow r_2$$

or

$$c_1 \longleftrightarrow c_2$$

Type II

row/col scaling

$$E = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$E = I - (1-\alpha)e_2e_2^T$$

$$r_2 \longrightarrow \alpha r_2$$

or

$$c_2 \longrightarrow \alpha c_2$$

Type III

Lin comb rows/cols

$$E = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \alpha & 0 & 1 \end{bmatrix}$$

$$E = I + \alpha e_3 e_1^T$$

$$r_3 \longrightarrow \alpha r_1 + r_3$$

or

$$\boxed{c_3 \longrightarrow \alpha c_1 + c_3}$$

Check for cols E^T (maybe)

1a

4) $E = I - xy^T$, $E^{-1} = (I - xy^T)^{-1} = I + \frac{xy^T}{1 - x^Ty}$

Só está definida a inversa se $x^Ty \neq 1$

Note que: $x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}$, $y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$.

Produto interno = produto escalar (por enquanto)

$$= x^Ty = [x_1\ x_2 \cdots x_N] \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \sum_{k=1}^{N} x_k y_k$$

$$= y^Tx = [y_1\ y_2 \cdots y_N] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} = \sum_{k=1}^{N} y_k x_k \equiv x^Ty.$$

Produto externo = produto matricial quando o primeiro vetor é coluna e o segundo é linha.
O resultado é uma matriz:

$$xy^T = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \begin{array}{c} [y_1\ y_2 \cdots y_N] \\ \begin{bmatrix} x_1y_1 & x_1y_2 & \cdots & x_1y_N \\ x_2y_1 & x_2y_2 & \cdots & x_2y_N \\ & \vdots & & \\ x_Ny_1 & x_Ny_2 & \cdots & x_Ny_N \end{bmatrix} \end{array} \neq \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \begin{array}{c} [x_1\ x_2 \cdots x_N] \\ \begin{bmatrix} y_1x_1 & y_1x_2 & \cdots & y_1x_N \\ y_2x_1 & y_2x_2 & \cdots & y_2x_N \\ & \vdots & & \\ y_Nx_1 & y_Nx_2 & \cdots & y_Nx_N \end{bmatrix} \end{array} = yx^T$$

Isso faz sentido, pois o produto matricial é não-comutativo.

Checando: $E E^{-1} \stackrel{?}{=} I$

O "truque" é usar o fato de que $y^T x = x^T y$ = scalar

$$(I - xy^T)\left(I + \frac{xy^T}{1 - x^Ty}\right) = I + \frac{xy^T}{1-a} - xy^T - \frac{xy^T xy^T}{1-a}$$

$\hat{=} a$ (scalar)

$$= \frac{(1-a)I + xy^T - (1-a)xy^T - x(y^Tx)y^T}{1-a}$$

scalar can be put out

$$= \frac{(1-a)I + xy^T - xy^T + axy^T - \underbrace{\boxed{(y^Tx)}}_{a} xy^T}{1-a}$$

$$= \frac{(1-a)I + \cancel{xy^T} - \cancel{xy^T} + \cancel{axy^T} - \cancel{a xy^T}}{1-a} = \frac{(1-a)I}{(1-a)} = I \;//.$$

Mechanics of Elimination via EMATS

Using partitioning, it is convenient to write A as a stack of rows or cols

$$A_{M\times N} = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_M \end{bmatrix} \quad \text{or} \quad A_{M\times N} = [c_1 \; c_2 \; \cdots \; c_N]$$

Not the same c_k as in G_k's c_k

where $r_k = [a_{k1} \; a_{k2} \; \cdots \; a_{kN}]$ and $c_k = \begin{bmatrix} a_{1k} \\ a_{2k} \\ \vdots \\ a_{Mk} \end{bmatrix}$

Let us consider a matrix $A_{3\times 3}$. Assume we want to eliminate an element in row 2 using row 1; that is

$$\begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} \longrightarrow \begin{bmatrix} r_1 \\ r_2+\alpha r_1 \\ r_3 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} + \alpha \begin{bmatrix} 0 \\ r_1 \\ 0 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} + \alpha \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

$$= \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} + \alpha \left(\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} [1 \;\; 0 \;\; 0] \right) \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} = \left(I + \alpha e_2 e_1^T \right) \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

Back

$$= E \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} = EA.$$ Pre-multiplying A by E has the effect of altering <u>only</u> row 2 of A, replacing row 2 (r_2) by $r_2 + \alpha r_1$. By properly selecting α, we can eliminate <u>any</u> element in row 2 using row 1. For instance, to eliminate a_{23} we proceed as follows (using row 1)

$$\begin{bmatrix} a_{11} & a_{12} & \boxed{a_{13}} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad \alpha = -\frac{a_{23}}{a_{13}}, \quad \begin{bmatrix} 1 & 0 & 0 \\ -\frac{a_{23}}{a_{13}} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} \cancel{a_{11}} & \cancel{a_{12}} & \cancel{a_{13}} \\ \cancel{-\frac{a_{23}}{a_{13}} a_{11}} & & 0 \\ & & \end{bmatrix}$$

$$I + \alpha e_2 e_1^T = \begin{bmatrix} 1 & 0 & 0 \\ \alpha & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ -\frac{a_{23}}{a_{13}} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} - \frac{a_{11}a_{23}}{a_{13}} & a_{22} - \frac{a_{12}a_{23}}{a_{13}} & 0 \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

1d

$$= \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21}^{(1)} & a_{22}^{(1)} & 0 \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad a_{21}^{(1)} = a_{21} - \frac{a_{23}}{a_{13}} a_{11}, \quad a_{22}^{(1)} = a_{22} - \frac{a_{23}}{a_{13}} a_{12}.$$

In the same vein, if we wanted to ~~elimin~~ eliminate a_{21} from A (which is the first step in GE), then $\alpha = -\frac{a_{21}}{a_{11}}$, using again $e_2 e_1^T$ because the "receiving" row is again r_2 and the "sending" row is r_1:

$$\begin{bmatrix} 1 & 0 & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad a_{22}^{(1)} = a_{22} - \frac{a_{21}a_{12}}{a_{11}}, \quad a_{23}^{(1)} = a_{23} - \frac{a_{21}}{a_{11}} a_{13}.$$

$a_{22}^{(1)}$ and $a_{23}^{(1)}$ indicates such elements have been modified once. The next step in GE would be to eliminate a_{31}; since $a_{21}^{(1)} = 0$, we need to use row 1:

$$\left(I - \frac{a_{31}}{a_{11}} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \right) A = \left(I - \frac{a_{31}}{a_{11}} e_{\underset{\text{receiving row}}{3}} e^T_{\underset{\text{sending row}}{1}} \right) A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} \end{bmatrix}$$

with $a_{32}^{(1)} = a_{32} - \frac{a_{31}}{a_{11}} a_{12}$, $\quad a_{33}^{(1)} = a_{33} - \frac{a_{31}}{a_{11}} a_{13}$.

We can compose elmats that eliminate elements in the same column into a Gaussian matrix G:

$$\begin{bmatrix} 1 & 0 & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\frac{a_{31}}{a_{11}} & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & 0 \\ -\frac{a_{31}}{a_{11}} & 0 & 1 \end{bmatrix} \triangleq G$$

If we proceed like that, we assign matrix G_k with the task of eliminating an entire column, the k column, in A. Let us complete the GE process that we started in $A_{3\times 3}$:

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\frac{a_{32}^{(1)}}{a_{22}^{(1)}} & 1 \end{bmatrix}}_{G_2} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} \\ 0 & 0 & a_{33}^{(2)} \end{bmatrix}, \quad a_{33}^{(2)} = a_{33}^{(1)} - \frac{a_{32}^{(1)}}{a_{22}^{(1)}} a_{23}^{(1)} \quad //$$

We can compose G and G_2 into:

$$G_2 G A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33}^{(2)} \end{bmatrix} \triangleq U.$$

$$\boxed{E = I - \alpha e_k e_l^T}$$

$$A = \begin{bmatrix} r_1 \\ r_2 \\ r_l \\ r_k \\ r_N \end{bmatrix}$$

If $r_l = l^{th}$ row of A

$$EA = \begin{bmatrix} r_1 \\ r_2 \\ r_l \\ r_k \\ r_N \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\alpha r_l \\ r_N \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_l \\ r_k - \alpha r_l \\ r_N \end{bmatrix}$$

For $\boxed{E^T = I - \alpha e_l e_k^T}$, with $A = [c_1 \; c_2 \; c_l \; c_k \; c_N]$

$c_l = l^{th}$ col of A

$$AE^T = [c_1 \; c_2 \; c_l \; c_k \; c_N] + [0 \;\; 0 \;\; 0 \;\; -\alpha c_l \;\; c_N]$$

$$AE^T = [c_1 \;\; c_2 \;\; c_l \;\; c_k - \alpha c_l \;\; c_N]$$

$$E^T A = \begin{bmatrix} r_1 \\ r_2 \\ r_l - \alpha r_k \\ r_k \\ r_N \end{bmatrix}, \quad AE = \boxed{c_1 \,|\, c_2 \,|\, (c_l - \alpha c_k) \,|\, c_N}$$

writes αe_l^T in k^{th} row

$(\alpha e_k e_l^T)$

selects ~~c_k~~ c_k in AE

~~case~~ $\underbrace{\alpha e_k}_{\text{writes } \alpha e_l^T \text{ in } k^{th} \text{ row}} \underbrace{e_l^T}_{\text{selects } r_l \text{ in } EA}$

In order to operate over the rows of some given matrix A, we perform

$$\bar{A} = E A,$$

likewise, for the cols, we proceed as

$$\bar{A} = A E.$$

The consequence is that GE and GJ may be encoded into a sequence of elmats properly chosen to triangularize A, or make it identity (in GJ's case).

<u>Thm 2.8.1</u>: the product of nonsingular matrices is nonsingular.

<u>Proof</u>: let $A, B \mid \exists A^{-1}, B^{-1}$; then build a matrix $X = B^{-1}A^{-1}$. Thereby,

$$ABX = AB(B^{-1}A^{-1}) = A(BB^{-1})A^{-1} = AIA^{-1} = I.$$

In the same vein,

$$XAB = (B^{-1}A^{-1})AB = B^{-1}(A^{-1}A)B = B^{-1}IB = I.$$

Corolary : A matrix A is invertible if, and only if (iff) it is a product of elmnts.

Proof: If $\exists A^{-1}$, then

$$E_k \cdots E_2 E_1 A = I.$$

So,

$$(E_1^{-1} E_2^{-1} \cdots E_k^{-1}) E_k \cdots E_2 E_1 A = (E_1^{-1} E_2^{-1} \cdots E_k^{-1}) I$$

or $\quad A = E_1^{-1} E_2^{-1} \cdots E_k^{-1}$. $\therefore$ A = product of elmnts.

If $A = E_1^{-1} E_2^{-1} \cdots E_k^{-1}$, then

$$(E_k \cdots E_2 E_1) A = (E_k \cdots E_2 E_1)(E_1^{-1} E_2^{-1} \cdots E_k^{-1})$$

$$E_k \cdots E_2 E_1 A = I.$$ $\therefore$ A is nonsingular.

REMARK: this implies that elementary rows/cols ops are reversible.

2.9. Equivalence Relations & Matrix Factorizations

The concept of an equivalence relation, from Algebra, is useful to classify matrices that share some given property; for instance, the set of matrices that can be triangularized; the set of diagonalizable matrices, etc.

Equivalence Relations

Let us start with the definition of a binary relation R, i.e., a relation between two input arguments.

Given any ordered pair (a,b) of elements in a set S, a relation R on S is a law that classifies elements in S that are related, or not, via the formation law that defines R. If a and b are related via R, we write

$$a R b \qquad \text{or} \qquad (a,b) \in R.$$

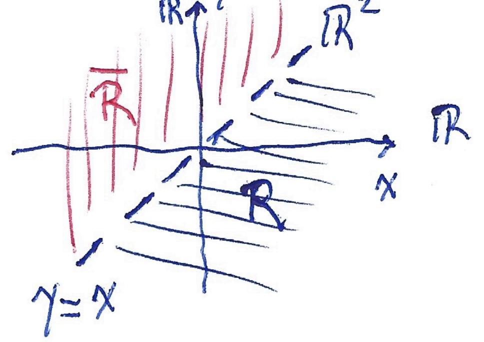
Formally, a relation R on S is any subset of $S \times S$ (cartesian product).

<u>Example</u>: R is the ordering relation "$<$" on $\mathbb{R}$.

$$3 < 7 \quad \text{or} \quad (3,7) \in R$$

$$-\sqrt{2} < 0 \quad \text{or} \quad (-\sqrt{2}, 0) \in R$$

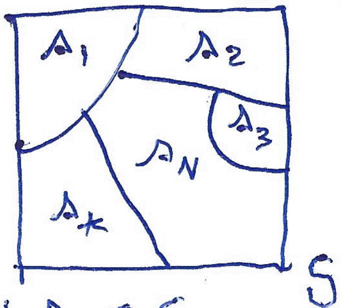
$$\pi \text{ and } 3.14 \notin R$$



An equivalence relation E in S is a relation R with some special properties. For any a, b, c in S:

1. aEa (reflexive);
2. $aEb \Rightarrow bEa$ (symmetry);
3. aEb and $bEc \Rightarrow aEc$ (transitivity).

An equivalence relation provides a partition of S.



partition in S

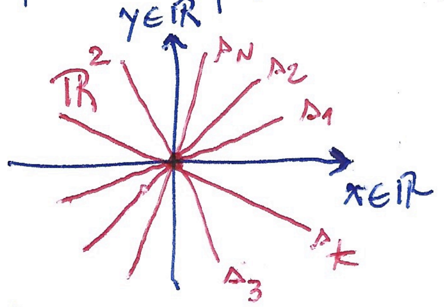
$A_k \subseteq S$

$\bigcup A_k = S$

$\bigcap A_k = \emptyset$

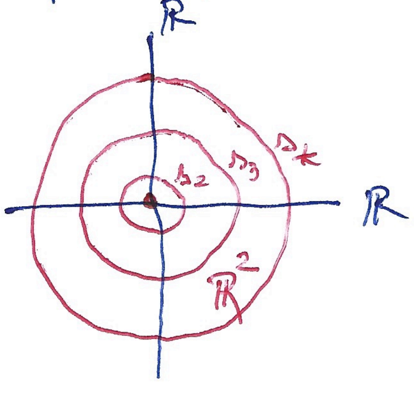
Examples: $S = \mathbb{R}$

$E_1: \{xEy \mid y = hx\}, h \in \mathbb{R}$



E_1 partitions S into subsets that are lines crossing the origin in $\mathbb{R}^2$

$E_2: \{xEy \mid x^2 + y^2 = r^2\}, r \in \mathbb{R}$



E_2 partitions S into subsets that are concentric circumferences of radius r.

Note: E_k here is not related to elmat E_k.

Refs: 1) "A book of Abstract Algebra", Charles Pinter

2) "Basic Algebra I", Nathan Jacobson

Matrix Equivalence

The fundamental operation in matrix algebra is the matrix product. Using the matrix product together with equivalence relations sets the foundations for a central topic in matrix analysis: matrix factorizations.

We consider the set S to be the set of $M \times N$ matrices

Row equivalence: $\boxed{A \overset{R}{\sim} B}$. Two matrices A and B, not necessarily invertible, are row-equivalent via the matrix product if $\exists P^{-1} \mid PA = B$. Operate over the rows of A to obtain B.

Col equivalence: $\boxed{A \overset{C}{\sim} B}$. Matrices A and B, not necessarily invertible, are col-equivalent via the matrix product if $\exists Q^{-1} \mid AQ = B$. Operate over cols of A to obtain B.

Equivalence: $\boxed{A \sim B}$. Likewise, matrices A and B are equivalent if $\exists P^{-1}, Q^{-1} \mid PAQ = B$. Operate over rows AND cols of A to obtain B.

Remarks: 1) Note that $A \overset{R}{\sim} B$ and $A \overset{C}{\sim} B$ are particular cases of $A \sim B$

$$PAQ = B \xrightarrow{Q=I} PA = B$$
$$PAQ = B \xrightarrow{P=I} AQ = B$$

The idea here is that if we operate over both the rows and cols, we may find more pairs (A, B) that are equivalent than just $A \overset{R}{\sim} B$ or $A \overset{C}{\sim} B$

2) Can we find singular matrices P and/or Q so that $PA = B$ / $AQ = B$ / $PAQ = B$?

YES! But they do not establish equivalence relations.

eventually, for some pairs (A, B). Maybe not for all A, B

3) That is why P and Q must be invertible: we must respect the 3 properties, or: ops must be reversible. (Back)

<u>thm 2.9.1</u>: Every matrix A is equivalent to a matrix of the form

$$\bar{I}_r \triangleq \begin{bmatrix} I_r & 0 \\ 0 & 0 \end{bmatrix} \quad \text{for some } r = \operatorname{rank}(A)$$

$\bar{I}_r$ is the RANK NORMAL FORM.

<u>Sketch of Proof</u>:

$$A \xrightarrow[GJ]{\text{rows}} \underset{\text{reduced row echelon}}{E_A} \xrightarrow[GJ]{\text{cols}} \bar{E}_A = \text{bunch of zeroes and ones} \xrightarrow[\text{permutations}]{\text{rows cols}} \begin{bmatrix} I_r & 0 \\ 0 & 0 \end{bmatrix}$$

$$PA = E_A \qquad PAQ = \bar{E}_A \qquad A \sim \bar{I}_r$$

<u>Example</u>:

$$\begin{matrix} 1 & 2 & 2 & 1 \\ 1 & 2 & 3 & 2 \\ 0 & 0 & 1 & 2 \\ 2 & 4 & 4 & 2 \end{matrix} \xrightarrow{\text{rows}} \begin{matrix} 1 & 2 & 2 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 \end{matrix} \xrightarrow{\text{rows}} \begin{matrix} 1 & 2 & 0 & -1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{matrix} \xrightarrow{\text{rows}} \underbrace{\begin{matrix} 1 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{matrix}}_{E_A}$$

We can only go further if operating over cols now

$$\xrightarrow{\text{cols}} \underbrace{\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{matrix}}_{\bar{E}_A} \xrightarrow[\text{Permutations}]{\text{row/col}} \left[\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 0 \end{array}\right] = \bar{I}_r \quad \text{RANK NORMAL FORM.}$$

the rank normal form provides a way to find, within a certain precision the rank of a Matrix.

All that two matrices A and B must have to be equivalent ($A \sim B$) is the same rank.

rank = N° LI cols / N° LI rows

Matrix Decompositions / Factorizations

A matrix decomposition is the expression of an original matrix A ~~into~~ in terms of the product of other matrices, usually two or three. An unstructured matrix A is written as the product of structured matrices ~~so that~~ that are more convenient in some applications, or that unveils explicitly parameters of A, such as eigenvalues, rank, determinant, inertia, etc. In general,

$$A = BC \qquad \text{or} \qquad A = BCD$$

in which equivalence relations apply. There is some ambiguity involved, as

$$A = BC\,, \quad A = (-B)(-C) \quad \text{or} \quad A = B\,\overbrace{FG}^{I_M}_{M\times M}\,C_{M\times N}$$

in which $FG = I_M$ for some F, G, represent equivalent decompositions, unless B and OR C are required to have specific formats / properties.

We will explore many different decompositions in this course, such as LU, QR, eigendecomposition, SVD, etc.

We start with the LU factorization.

2.10. The LU factorization

Tadeusz Banachiewicz 1938

Alan TURING 1948

The encoding ~~...~~ of GE as a product of properly chosen elmats is usually called Gaussian Transformation, which posed in a decomposition form returns a widely used factorization, the LU decomposition.

Definitions and the algebra of triangular matrices

1) Lower / upper (Left / Right) triang: all els above / below main diagonal are zero.
   $\triangle$ ∇

2) Lower / upper (Left / Right) UNIT triang: a lower / upper Δ-matrix with 1's across main diagonal

$\boxed{\Delta = \text{either } \nabla \text{ or } \triangle}$

3) $\triangle^{-1} = \triangle$, $\nabla^{-1} = \nabla$

4) $\triangle\triangle = \triangle$, $\nabla\nabla = \nabla$

5) $(\text{UNIT } \Delta)^{-1} = \text{UNIT } \Delta$
   SAME KIND

6) $(\text{UNIT } \Delta)(\text{UNIT } \Delta) = \text{UNIT } \Delta$
   same kind   same kind

7) $\det \Delta = \prod_i [\Delta]_{ii}$. thus, a Δ-matrix with a zero in the main diagonal is SINGULAR!

Gaussian Transform

It is the matrix representation of GE via elmats. Recall that row ops can be performed by pre-multiplying a given matrix A by ~~that~~ some elmat, i.e., EA. In GE, each column represents a stage in the method. For an $M \times N$ matrix $A_{M\times N}$, there will be N sequences of elmats as follows. For simplicity, let us consider $A_{N\times N}$.

$$\underbrace{E_{N-1}\, E_{N-2} \cdots E_2\, E_1}_{G_1} A = \left[\begin{array}{c|c} a_{11} & a_{12} \cdots a_{1N} \\ \hline 0 & \\ \vdots & A^{(1)}_{N-1\times N-1} \\ 0 & \end{array}\right]$$

First stage in GE (i.e., elimination of col #1)

G_k annihilates the entire k^{th} col below the a_{kk} pivot, which is of size $k-1 \times 1$, so that, after $N-1$ stages, we get an upper triangular matrix

$$G_{N-1}\, G_{N-2} \cdots G_2 G_1 A_{NN} = U_{NN} \quad \left(\text{upper triang.}\right)$$

Or $A \overset{R}{\sim} U$. We may group the sequence of G_k's into an entire "GAUSSIAN" matrix G:

$$GA = U \quad , \quad G \triangleq G_{N-1} \cdots G_2 G_1$$

where G is the matrix form of GE. G transforms A into an upper triangular matrix U. The general form of G_k is

$$\boxed{G_k = I - c_k e_k^T} \;, \quad c_k^T = [\overbrace{0 \cdots 0}^{k \text{ zeros}} \; \overbrace{c(k+1)\;\; c(k+2) \cdots c(N)}^{\text{multipliers}}] \;, \quad c(i) = \frac{a_{ik}}{a_{kk}} \quad i = k+1, \dots, N$$

canonical vec, selects k^{th} row in A

Back

As it is done in GE, we do not really need to compute all the G_k's, multiply them together into G, then perform $GA \to U$. In matrix Analysis the way the decompositions are represented theoretically is <u>not</u> usually how they are implemented numerically in practice!

<u>Example</u>: By inspection. Find G and U from A as in

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

$$G_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ -2 & 0 & 0 & 1 \end{bmatrix} \qquad G_1A = \begin{bmatrix} 2 & 1 & 0 & -1 \\ 0 & 3 & 2 & 1 \\ 0 & 3 & 1 & -1 \\ 0 & -6 & 2 & 4 \end{bmatrix} \qquad A^{(1)}_{3\times 3}$$

$$G_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{bmatrix} \qquad G_2G_1A = \begin{bmatrix} 2 & 1 & 0 & -1 \\ 0 & 3 & 2 & 1 \\ 0 & 0 & -1 & -2 \\ 0 & 0 & 6 & 6 \end{bmatrix} \qquad A^{(2)}_{2\times 2}$$

$$G_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 6 & 1 \end{bmatrix} \qquad \underbrace{\begin{bmatrix} 2 & 1 & 0 & -1 \\ 0 & 3 & 2 & 1 \\ 0 & 0 & -1 & -2 \\ 0 & 0 & 0 & -6 \end{bmatrix}}_{U} \qquad A^{(3)}_{1\times 1}$$

$$\underbrace{G_3G_2G_1}_{G}A = U$$

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 2 & -1 & 1 & 0 \\ 8 & -4 & 6 & 1 \end{bmatrix}$$

$$\boxed{GA = U}$$

Constructing LU

Since $GA = U$, then we have

$$A = G^{-1} U = G_1^{-1} G_2^{-1} G_3^{-1} \cdots G_{N-1}^{-1} U$$

$$= L_1 L_2 L_3 \cdots L_{N-1} U$$

$$= \underbrace{L}_{\text{Unit } \Delta} \cdot \underbrace{U}_{\nabla}, \quad L \triangleq L_1 L_2 \cdots L_{N-1},$$

or $\boxed{A = LU}$ LU factorization of A.

Matrix L has a striking property that allows for a much easier calculation by inspection than G.T. Consider the previous example, then calculate $L = G^{-1}$:

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ -1 & 1 & 1 & 0 \\ 2 & -2 & -6 & 1 \end{bmatrix} \qquad U = \begin{bmatrix} 2 & 1 & 0 & -1 \\ 1 & 3 & 2 & 1 \\ -1 & 1 & -1 & -2 \\ 2 & -2 & -6 & -6 \end{bmatrix}$$

Compare the k^{th} col in L with the k^{th} col in G_k

L (relevant numbers)

It turns out that the numbers in k^{th} col below main diagonal are the negative of the multipliers in the Gaussian transformation matrix at stage k, not in G G_k. Moreover, the final Gaussian matrix does not have this property!

Because of this remarkable property, it is common to store the $[l_{ij}] = L$ numbers from L directly in U, below the main diagonal and replacing zeros. By doing so, we save space in memory (numbers in green in U above)

FINDING L

A natural question is why L has the nice property of directly storing the (negative of) multipliers, while G doesn't. From the definition

$$L_k = G_k^{-1} = \left(I - c_k e_k^T\right)^{-1}.$$

the first clue comes from the MIL:

$$\left(I - c_k e_k^T\right)^{-1} = I + c_k e_k^T$$

$c_k^T e_k = 0 \ \forall k$, so L_k is always defined ($c_k^T e_k$ is an inner product, $c_k e_k^T$ is an outer prod)!

That is, we multiply by -1 the multipliers c_k that come from G_k to obtain the k^{th} col in L_k.

the second reason comes from the rule to invert a product of compatible matrices

$$\left(G_{N-1} G_{N-2} \cdots G_k \cdots G_2 G_1\right)^{-1} = G_1^{-1} G_2^{-1} \cdots G_k^{-1} \cdots G_{N-1}^{-1}$$

$$= L_1 L_2 \cdots L_k \cdots L_{N-1},$$

that is, the L_k's operate in reverse order over U, compared to the action of the G_k's over A. We may go further:

$$L = L_1 L_2 \cdots L_{N-1} = \left(I + c_1 e_1^T\right)\left(I + c_2 e_2^T\right) \cdots \left(I + c_{N-1} e_{N-1}^T\right).$$

Let's expand the first two terms in the product above:

$$\left(I + c_1 e_1^T\right)\left(I + c_2 e_2^T\right) = I + c_1 e_1^T + c_2 e_2^T + c_1 \underbrace{e_1^T c_2}_{0} e_2^T$$

$$= I + c_1 e_1^T + c_2 e_2^T.$$

0, $e_\ell^T c_k = 0$, $\ell \leq k$

~~Now the first two terms wrgt.~~

$$\cancel{\left(I - c_2 e_2^T\right)\left(I - c_1 e_1^T\right) = I - c_1 e_1^T - c_2 e_2^T + c_2 \underbrace{e_2^T c_1}_{\neq 0} e_1^T} \rightarrow$$

Next page

therefore, the general L is of the form

$$L = I + \underbrace{c_1 e_1^T}_{\text{in col 1}} + \underbrace{c_2 e_2^T}_{\text{in col 2}} + c_3 e_3^T + \dots + c_{N-1} e_{N-1}^T$$

$$\boxed{L = I + \sum_{k=1}^{N-1} c_k e_k^T}$$

that is why we may write L directly by inspection, filling in with the negative of the multipliers used during the triangularization of A

Conclusion: No need to form the G_k's, invert them and perform the product of the L_k's to find L

For comparison, take the first two products of G_k's

$$(I - c_2 e_2^T)(I - c_1 e_1^T) = I - c_2 e_2^T - c_1 e_1^T + c_2 e_2^T c_1 e_1^T$$

$$= I - c_2 e_2^T - \underbrace{c_1 e_1^T} + \boxed{\underbrace{(e_2^T c_1)}_{\text{scalar}} c_2 e_1^T}$$

this term is added to c_1 in col 1

Example: A and B are lower triang. with the same structure as the G_k's and L_k's. Form AB and BA and compare.

$$A = \begin{bmatrix} 1 & 0 & 0 \\ \alpha_1 & 1 & 0 \\ \alpha_2 & 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \beta_1 & 1 \end{bmatrix}$$

$$BA = \begin{bmatrix} 1 & 0 & 0 \\ \alpha_1 & 1 & 0 \\ \alpha_2 + \beta_1 \alpha_1 & \beta_1 & 1 \end{bmatrix} \qquad AB = \begin{bmatrix} 1 & 0 & 0 \\ \alpha_1 & 1 & 0 \\ \alpha_2 & \beta_1 & 1 \end{bmatrix}$$

That is, the order of the L_k's product (AB) preserves the multipliers in the right cols.

Existence and Uniqueness

LU factorization does not always exist:

Example 1

$$A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Example 2

$$\underset{G_1}{\begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -3 & 0 & 1 \end{bmatrix}} \underset{A}{\begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 7 \\ 3 & 5 & 3 \end{bmatrix}} = \underset{G_1 A}{\begin{bmatrix} 1 & 2 & 3 \\ 0 & 0 & 1 \\ 0 & -1 & -6 \end{bmatrix}}$$

In the examples above, there is no way we can finish the triangularization over A in its original form. Without altering A somehow, there is no LU decomposition.

Example 3: We may check on the existence of LU via a linear systems approach

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 7 \\ 3 & 5 & 3 \end{bmatrix} = \underset{L}{\begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix}} \underset{U}{\begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}} = B$$

If we multiply L by U to form B, we have a system of 9 equations, from A = B, which is <u>inconsistent</u>.
Note that A is <u>nonsingular</u> ($\exists A^{-1}$), but $\nexists$ LU.
Nonsingularity and existence are not necessarily bonded.

NOTATION: $\nexists A^{-1} = \overline{A^{-1}}$, $\nexists LU = \overline{LU}$

$\overline{A^{-1}}$, LU

$$\begin{matrix} 1 & 2 & 1 & 3 \\ 1 & 1 & 1 & 1 \\ 2 & 3 & 2 & 4 \\ 0 & 1 & 0 & 2 \end{matrix} \rightarrow \begin{matrix} 1 & 2 & 1 & 3 \\ 0 & -1 & 0 & -2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{matrix}$$

LU $\neq$ Nonsingularity

$\overline{A^{-1}}$, $\overline{LU}$

$$\begin{matrix} 1 & 2 & 1 & 3 \\ 1 & 1 & 1 & 1 \\ 2 & 3 & 2 & 4 \\ 0 & 1 & 4 & 4 \end{matrix} \rightarrow \begin{matrix} 1 & 2 & 1 & 3 \\ 0 & -1 & 0 & -2 \\ 0 & 0 & \boxed{0} & 0 \\ 0 & 0 & 4 & 2 \end{matrix}$$

$\overline{LU}$ $\neq$ singularity

Thm 2.10.1: Existence of LU (sufficient condition)

Let A be an $M \times N$ matrix and for $k = 1, \dots, \min(M,N) - 1$, let $A(k) = [A]_{1:k,1:k}$ be the $k \times k$ submatrix of A. If $A(k)$ is nonsingular for all k, then there exists an LU decomposition for A. Proof for square matrices: Meyer. The proof is also an algorithm for triangularization.

Remarks

1) It is only a sufficient condition, i.e., it is possible to find matrices that fail the $A(k)$ nonsingularity tests, but that have LU decompositions.
2) the submatrices $A(k)$ may be tested on-the-fly, during the triangularization process: if a zero shows up as a pivot (i.e., in the main diagonal) then A does not have an LU factorization. This is equivalent to $\det A(k) \neq 0$, $k = 1, \dots, \min(M,N) - 1$.
3) the theorem considers A in its original form, that is: <u>no pivoting</u>!
4) Matrix A does not have to be invertible. Note that a matrix whose last col is all zeros may pass the $A(k)$'s tests, but A itself will be <u>singular</u>.
5) In the generic rectangular case, we will end up with trapezoidal L or U factors, assuming $A = LU$ exists. Depending whether $M < N$ or $M > N$, L will be either $M \times M$ or $N \times N$. Matrix U will be $M \times N$.

Examples

1)
$$A = \begin{bmatrix} 1 & 2 & 1 & 3 \\ 1 & 1 & 1 & 1 \\ 2 & 3 & 2 & 4 \\ 0 & 1 & 0 & 2 \end{bmatrix}$$

$A(1) = [1]$, $A(2) = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix}$,

$$A(3) = \begin{bmatrix} 1 & 2 & 1 \\ 1 & 1 & 1 \\ 2 & 3 & 2 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 2 & 1 & 1 & 0 \\ 0 & -1 & \boxed{\alpha} & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 & 3 \\ 0 & -1 & 0 & -2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \forall \alpha$$

Recall the thm is only a sufficient, not necessary, condition

$A(1)$ and $A(2)$ are nonsingular. But $A(3)$ is singular. It fails the $A(k)$ tests, but it has an LU factorization. It is also non-unique: take $\alpha_1 \neq \alpha_2$ and test!

2)
$$A = \left[\begin{array}{ccc:cc} 2 & 2 & -4 & 8 & 2 \\ 1 & 2 & 2 & 2 & -2 \\ 0 & 2 & 4 & 0 & 2 \end{array}\right] = \begin{bmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix} \left[\begin{array}{ccc:cc} 2 & 2 & -4 & 8 & 2 \\ 0 & 1 & 4 & -2 & -3 \\ 0 & 0 & -4 & 4 & 8 \end{array}\right]$$

LU exists and it is unique. Note that the $A(k)$'s tests and L are exactly the same as compared to when $A = \begin{bmatrix} 2 & 2 & -4 \\ 1 & 2 & 2 \\ 0 & 2 & 4 \end{bmatrix}$, that is, the "square" part to the left.
U, in this case, will also be the "square" part of the rectangular U. This is the idea behind "extending" the proof on square matrices (usually given in books) to the rectangular case: The existence proof also applies to the rectangular case, although a formal derivation is required.

BACK

Example

3) $$B = A^T = \begin{bmatrix} 2 & 1 & 0 \\ 2 & 2 & 2 \\ -4 & 2 & 4 \\ 8 & 2 & 0 \\ 2 & -2 & 2 \end{bmatrix}$$

$$B = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ -2 & 4 & 1 & 0 & 0 \\ 4 & -2 & 1 & 1 & 0 \\ 1 & -3 & -2 & \boxed{\alpha} & 1 \end{bmatrix}}_{L} \cdot \underbrace{\begin{bmatrix} 2 & 1 & 0 \\ 0 & 1 & 2 \\ 0 & 0 & -4 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{U} \quad \forall \alpha$$

thm 2.10.2 : Uniqueness of LU

Any matrix A that has an LU factorization and is also nonsingular has a unique LU factorization, i.e., the L and U factors are unique.

Proof: Assume $L_1 U_1 = A = L_2 U_2$ are two different decompositions of A. then

$$L_1 U_1 = L_2 U_2$$

since $\exists A^{-1}$, then it holds that the L's and U's are invertible:

$$L_1^{-1} L_1 U_1 = L_1^{-1} L_2 U_2$$

$$L_1^{-1} L_1 U_1 U_2^{-1} = L_1^{-1} L_2 U_2 U_2^{-1}$$

$$U_1 U_2^{-1} = L_1^{-1} L_2 \text{ . Recall } \left\{ \begin{array}{l} \nabla \nabla = \nabla \\ \Delta \Delta = \Delta \\ \nabla^{-1} = \nabla , \; \Delta^{-1} = \Delta \end{array} \right.$$

We have a situation in which

$$\nabla = \Delta$$

so that **all** elements off the main diagonal are zero.

We also have that the L_1 and L_2 are unit lower triang, so L_1^{-1} is also unit-Δ and $L_1^{-1} L_2$ is unit-Δ as well. Since its main diagonal is ones and all terms off diagonal are zero, $L_1^{-1} L_2 = I$. Then follows:

$L_1^{-1} L_2 = I$	$U_1 U_2^{-1} = L_1^{-1} L_2 = I$
$L_2 = L_1$ //	$U_1 = U_2$ //

$\blacksquare$

LU Decomposition with row pivoting

Consider the first elimination step over matrix A:

$$G_1 A = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -3 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 7 \\ 3 & 5 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & \boxed{0} & 1 \\ 0 & -1 & -6 \end{bmatrix}.$$

Matrix A does not have an LU factorization. However, if we permute rows 2 and 3, we get

$$P_2 G_1 A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 0 & 0 & 1 \\ 0 & -1 & -6 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & -1 & -6 \\ 0 & 0 & 1 \end{bmatrix} = U.$$

Where is L then? Note that $(P_2 G_1)^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 0 & 1 \\ 3 & 1 & 0 \end{bmatrix} \neq \Delta.$

We can use the fact that $P_2^2 = I$ (that is permuting rows 2 and 3 twice returns the same original rows).

$$P_2 G_1 A = U \Rightarrow P_2 G_1 (P_2 P_2) A = U$$

$$\underbrace{P_2 G_1 P_2}_{\bar{G}_1} \underbrace{P_2 A}_{\substack{\text{pivoted} \\ A}} = U \Rightarrow \bar{G}_1 P_2 A = U.$$ And we get

L as usual: $(\bar{G}_1)^{-1} = (P_2 G_1 P_2)^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ -3 & 1 & 0 \\ -2 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 2 & 0 & 1 \end{bmatrix}$

That is $L = (P_2 G_1 P_2)^{-1}$. Moreover:

$P_2 A = LU$ while $A \neq LU$. Which is a way to say that under proper pivoting, it is always possible to find LU factors for PA. In many books A is required to be invertible, but it is ~~a~~ only a ~~necessary~~ sufficient condition. This strategy can be generalized to when many permutations may be needed. First a note on permutation matrices.

BACK

matrix $A = \begin{bmatrix} 1 & 2 & 1 & 3 \\ 1 & 2 & 4 & 4 \\ 1 & 1 & 1 & 1 \\ 2 & 6 & 2 & 10 \end{bmatrix}$ is singular, however

$G_2 P_2 G_1 A = U$, $G_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -2 & 0 & 0 & 1 \end{bmatrix}$, $P_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

$G_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{bmatrix}$, $U = \begin{bmatrix} 1 & 2 & 1 & 3 \\ 0 & -1 & 0 & 2 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

Where is L ?

$$G_2 P_2 G_1 \underbrace{P_2 P_2}_{I} A = U$$

~~Some books include another description~~

$$G_2 \underbrace{P_2 G_1 P_2}_{\overline{G}_1} P_2 A = U$$

$$G_2 \overline{G}_1 P_2 A = U$$

$$P_2 A = (G_2 \overline{G}_1)^{-1} U$$

$$\boxed{P_2 A = L U}$$

Permutation Matrices P

A permutation matrix is a matrix that has only a 1 in each row and a 1 in each col, the rest are zeros. It is basically a switch of rows of the identity matrix: $P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$

PA exchange rows 2 and 3. AP exchange cols 2 and 3. Matrices P may perform more than one row/col exchange at once, for instance $P = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$. This matrix is a <u>composite permutation matrix</u>:

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \underbrace{\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}}_{p} = \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix},$$

which switches two times the test vector p. It can be decomposed into two single permutations:

$$\begin{matrix} 1 \\ 2 \\ 3 \end{matrix} \xrightarrow{P_{1\leftrightarrow 2}} \begin{matrix} 2 \\ 1 \\ 3 \end{matrix} \xrightarrow{P_{2\leftrightarrow 3}} \begin{matrix} 2 \\ 3 \\ 1 \end{matrix} \quad \text{or} \quad \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} \xrightarrow{P_{2\leftrightarrow 3}} \begin{matrix} 1 \\ 3 \\ 2 \end{matrix} \xrightarrow{P_{1\leftrightarrow 3}} \begin{matrix} 2 \\ 3 \\ 1 \end{matrix}.$$

This is the kind of matrix we need to find in PA=LU. In the general case is composed of several <u>elementary permutation matrices</u> P_k. Such matrices perform <u>ONLY one</u> row/col switch each time.

$\boxed{P_k^2 = I}$. A composite P also has this property, but for some integer $K > 2$: $P^K = I$.

For the composite case above we have $P^3 = I$.

<u>thm 2.10.3</u>: A nonsingular matrix A has a unique LU factorization upon proper pivoting, i.e., $PA = LU$, for some permutation matrix P.

<u>Proof</u>: Consider the general case, with elementary permutations P_k at every stage. That is at every stage a zero shows up at the pivot position.

We show the 4x4 case as the method naturally generalizes.

<u>CASE 4x4</u>: $G_3 P_3 G_2 P_2 G_1 P_1 A = U$

$$G_3 P_3 G_2 P_2 G_1 \underbrace{P_2 P_2}_{I} P_1 A = U$$

$$G_3 P_3 G_2 \underbrace{P_3 P_3} P_2 G_1 \underbrace{P_2 P_2} P_1 A = U$$

$$G_3 P_3 G_2 \underbrace{P_3 P_3} P_2 G_1 \overbrace{P_2 \underbrace{P_3 P_3} P_2} P_1 A = U$$

$$\underbrace{G_3}_{\bar{G}_3 = G_3} \underbrace{P_3 G_2 P_3}_{\bar{G}_2} \underbrace{P_3 P_2 G_1 P_2 P_3}_{\bar{G}_1} \underbrace{P_3 P_2 P_1}_{P} A = U$$

$$\bar{G}_3 \bar{G}_2 \bar{G}_1 P A = U$$

$$\bar{G} P A = U$$

$$\boxed{PA = LU}, \quad L = (\bar{G})^{-1}.$$

1) Some of the P_k might be identity.

2) P_k permutes zeros (small numbers) at the a_{kk} pivotal positions with numbers below a_{kk}, i.e., $a_{kk} \to a_{\ell k}$, $\ell > k$

here we assumed a 0 showed up, o.w. $P_2 = I$

$$\underbrace{\begin{bmatrix} 0 & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}}_{A^{(0)} = A} \xrightarrow{G_1 P_1} \underbrace{\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} \\ 0 & 0 & a_{23}^{(1)} & a_{24}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & a_{34}^{(1)} \\ 0 & a_{42}^{(1)} & a_{43}^{(1)} & a_{44}^{(1)} \end{bmatrix}}_{A^{(1)} = G_1 P_1 A} \xrightarrow{G_2 P_2} \underbrace{\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & a_{24}^{(2)} \\ 0 & 0 & 0 & a_{34}^{(2)} \\ 0 & 0 & a_{43}^{(2)} & a_{44}^{(2)} \end{bmatrix}}_{A^{(2)} = G_2 P_2 A^{(1)}} \xrightarrow{G_3 P_3} \underbrace{\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & a_{24}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & a_{34}^{(3)} \\ 0 & 0 & 0 & a_{44}^{(3)} \end{bmatrix}}_{U = A^{(3)} = G_3 P_3 A^{(2)}}$$

In practice, zero means a small number, then pivoting is <u>always</u> used to improve numerical stability, just like in GE. Actually, LU is the decomposition form of GE. For a given A and a fixed P, $PA = LU$ is unique if A is invertible; the proof is the same as before: $\exists L^{-1}, \exists U^{-1}$.

Solving Linear Systems via Pivoted LU

In practical implementations it is not necessary to form all the P_k's in order to calculate P. We use an auxiliary vector $p^T = [1\ 2\ 3 \cdots M]$ for $A_{M\times N}$.

$$A \xrightarrow{G_1 P_1} \overset{(1)}{A} \xrightarrow{G_2 P_2} \overset{(2)}{A} \xrightarrow{G_3 P_3} \overset{(3)}{A} = U$$

$$p = \begin{bmatrix}1\\2\\3\\4\end{bmatrix} \xrightarrow{P_1} \begin{bmatrix}3\\2\\1\\4\end{bmatrix} \xrightarrow[\text{(here assumes no perm. required)}]{P_2 = I} \begin{bmatrix}3\\2\\1\\4\end{bmatrix} \xrightarrow{P_3} \begin{bmatrix}3\\2\\4\\1\end{bmatrix} = p_f.$$

The final p_f stores how the rows of $\overset{(t)}{A}$ were permuted during factorization, thus we may use p_f to switch the respective rows of an identity matrix to obtain P, if P is required. Usually p_f suffices.

the solution for $Ax = b$

$$Ax = b$$

$$PAx = Pb \triangleq \bar{b}$$

but $PA = LU$, then

$LUx = \bar{b}$. Define

$Ux \triangleq y$ so that

$$Ly = \bar{b}.$$

Solve for y: $Ly = \bar{b}$

Solve for x: $Ux = y$.

The LU factorization is the universal choice to solve dense linear systems in which A is not structured. If A has some special structure, then other more efficient methods apply.

2.11. Structured Matrices and Linear Systems

There are matrices with special structures that appear in many applications. For such matrices there exists efficient solutions when they describe linear systems. The solutions explore the structure in order to devise methods that are less complex in terms of computations. Structure also has impact on matrix parameters, such as eigenvalues.

Circulant Matrices are matrices in which every row is a 1-shift of the previous row, as follows:

$$A = \begin{bmatrix} c(1) & c(2) & c(3) & \cdots & c(N) \\ c(N) & c(1) & c(2) & \cdots & c(N-1) \\ c(N-1) & c(N) & c(1) & \cdots & c(N-2) \\ & & \vdots & & \\ c(2) & c(3) & \cdots & & c(1) \end{bmatrix}$$

c

A circulant matrix has exactly N different elements $c = [c(1)\ c(2) \cdots c(N)]$.

If we define the basic circulant permutation matrix P,

$$P = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix},$$

then A can be written as follows:

$$A = \sum_{k=0}^{N-1} c(k+1) P^{k}, \quad \text{or} \quad A = \sum_{k=0}^{N-1} e_{k+1} c \, P^{k}.$$

1) the product of circulant matrices is also circulant;
2) Circulant matrices commute under multiplication;
3) A circulant matrix A is diagonalizable by the FFT unitary matrix:

$$D = FAF^{-1}$$

where $D = \mathrm{Diag}(\lambda_1, \lambda_2, \cdots, \lambda_N)$ contains the eigenvalues of A.

or

$$A = F^{-1} D F$$

$$[F]_{kl} = \frac{1}{\sqrt{N}} e^{-j\frac{2\pi k l}{N}},$$

N-point unitary discrete Fourier matrix.

unitary: $F^{-1} = F^{*}$

A linear system whose coefficients matrix is circular can be solved very effiently via FFT:

$$Ax = b$$
$$A(F^{-1}F)x = b$$
$$FAF^{-1}Fx = Fb$$
$$Dy = \bar{b}$$

$D = FAF^{-1}$, $y \triangleq Fx$.

Solve for y: $\boxed{y = D^{-1}\bar{b}}$ N scalar divisions

Solve for x: $\boxed{x = F^{-1}y}$ Inverse FFT

Where $[F]_{kl} = \frac{1}{\sqrt{N}} e^{-j\frac{2\pi k l}{N}}$, N-point Discrete Fourier unitary matrix: $F^{-1} = F^{*}$.

Matrix F here is merely notational: it means we take the FFT (inverse or direct) over the rows/cols. Of course we can form F and carry out the products, but $\mathcal{F}\{\cdot\}$ via Butterfly Algorithm is more efficient.

<u>Toeplitz Matrix</u> is a matrix that has identical elements across the diagonals. For a set of elements $t_{-N}, t_{-N+1}, t_{-N+2}, \dots, t_{-1}, t_0, \dots, t_N$, the general term is $\boxed{a_{ij} = t_{j-i}}$, $i,j = 1,N$.

$$A = \begin{bmatrix} t_0 & t_1 & t_2 & \cdots & t_N \\ t_{-1} & t_0 & t_1 & & \vdots \\ t_{-2} & t_{-1} & t_0 & & \\ \vdots & & & \ddots & t_1 \\ t_{-N} & & & & t_0 \end{bmatrix}$$

Matrix A has $2N+1$ different elements. Defining forward and backward shift matrices

$$F = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

We can write any Toeplitz matrix as

$$A = \sum_{k=1}^{N} t_{-k} F^{k} + \sum_{k=0}^{N} t_{k} B^{k}.$$

<u>Hankel Matrix</u> is a matrix in which the elements across the counter diagonals are identical, in terms of a set of elements $h_0, h_1, h_2, \dots, h_{2N}$.

$$A = \begin{bmatrix} h_0 & h_1 & h_2 & h_3 & \cdots & h_N \\ h_1 & h_2 & h_3 & & & h_{N+1} \\ h_2 & h_3 & h_4 & & & \\ h_3 & & & & \ddots & \\ \vdots & & & & & \\ h_N & h_{N+1} & & & & h_{2N} \end{bmatrix}$$

A Hankel matrix has exactly $2N+1$ different elements. The general term is

$$a_{ij} = h_{i+j-2} \quad i,j = 1, N.$$

A trivial Hankel matrix is the backward identity

$$P = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

Matrix P relates Toeplitz and Hankel matrices as

$$T = HP \quad \text{and} \quad H = PT.$$

Both Toeplitz and Hankel linear systems may be solved efficiently via the <u>Levinson-Durbin algorithm</u> with complexity $O(N^2)$.