

**ANÁLISIS NUMÉRICO II — Práctico N°2 - 2025**  
**Eliminación Gaussiana y Descomposición LU**

1. Demuestre que:
  - a) Si  $A$  y  $B$  son triangulares superiores (inferiores) con elementos diagonales iguales a 1, entonces  $AB$  también es triangular superior (inferior) con elementos diagonales iguales a 1.
  - b) Probar que si  $A$  es triangular superior (inferior) con elementos diagonales iguales a 1, entonces  $A^{-1}$  también es triangular superior (inferior) con elementos diagonales iguales a 1.
2. Si  $B = LA$ , donde  $L$  es triangular inferior con unos en la diagonal, para  $\mathcal{I} = \{1, \dots, k\}$  demuestre que  $B_{\mathcal{I}\mathcal{I}} = L_{\mathcal{I}\mathcal{I}}A_{\mathcal{I}\mathcal{I}}$ .
3. Considere las transformaciones de Gauss  $M_k = I - v^k(e^k)^T$  donde  $e^k \in \mathbb{R}^n$  es el  $k$ -ésimo vector canónico y  $v^k \in \mathbb{R}^n$  con  $v_i^k = 0$  para  $i = 1, \dots, k$ . Demuestre que
  - a)  $M_k^{-1} = I + v^k(e^k)^T$ ,
  - b)  $L = M_1^{-1} \dots M_{n-1}^{-1} = I + \sum_{k=1}^{n-1} v^k(e^k)^T$ .
4. Sea  $A$  una matriz simétrica. Suponga que  $A$  ha sido reducida a la forma

$$\left[ \begin{array}{c|ccc} a_{11} & a_{12} & \dots & a_{1n} \\ \hline 0 & & & \\ \vdots & & A^{(1)} & \\ 0 & & & \end{array} \right]$$

efectuando solo operaciones elementales por filas. Demuestre que  $A^{(1)}$  es simétrica. ¿Qué ventajas tiene esto cuando se efectúa eliminación gaussiana de  $A$ ?

5. **Implemente** una función en **Python** que calcule la eliminación Gaussiana de un sistema  $Ax = b$ . La función debe llamarse **egauss**, tener entradas  $A$ ,  $b$  y salidas  $U$ ,  $y$ . Testear el algoritmo.
6. **Implemente** una función en **Python** que calcule la descomposición LU de una matriz. La función debe llamarse **dlu**, tener entrada  $A$  y salidas  $L$ ,  $U$ . Testear el algoritmo.
7. Escriba un pseudocódigo de un algoritmo de eliminación Gaussiana (sin pivoteo) adaptado a la estructura de una matriz  $A$  cuyos únicos elementos que pueden ser no nulos son  $a_{1n}$ ,  $a_{n1}$  y  $a_{ij}$  con  $|i - j| \leq 1$ . Implementar y testear el algoritmo en **Python**.
8. Cuando aparece un cero en una posición de pivoteo,  $A = LU$  no es posible. Muestre por qué no es posible obtener la descomposición LU directa de estos sistemas:

$$\begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ l & 1 \end{bmatrix} \begin{bmatrix} d & e \\ 0 & f \end{bmatrix}, \quad \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 2 \\ 1 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 1 & & \\ l & 1 & \\ m & n & 1 \end{bmatrix} \begin{bmatrix} d & e & g \\ & f & h \\ & & i \end{bmatrix}.$$

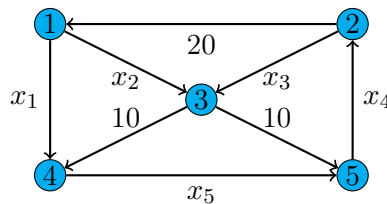
Estas matrices necesitan un intercambio de filas por una matriz de permutación  $P$ .

9. Muestre que si  $P$  es una matriz de permutación, entonces  $P^T P = P P^T = I$
10. **Implemente** dos funciones en **Python**, llamadas **egaussp** y **dlup** que realicen las versiones con permutación de filas de los algoritmos de los ejercicios 5 y 6. Testear los algoritmos.

11. **Implemente** una función en **Python**, llamada `sol_egauss` que utilice eliminación Gaussiana con pivoteo parcial para resolver el sistema lineal  $Ax = b$ . Testear su funcionamiento resolviendo el sistema lineal para :

$$A = \begin{bmatrix} 2 & 10 & 8 & 8 & 6 \\ 1 & 4 & -2 & 4 & -1 \\ 0 & 2 & 3 & 2 & 1 \\ 3 & 8 & 3 & 10 & 9 \\ 1 & 4 & 1 & 2 & 1 \end{bmatrix}, \quad b_1 = \begin{bmatrix} 52 \\ 14 \\ 12 \\ 51 \\ 15 \end{bmatrix} \quad \text{y} \quad b_2 = \begin{bmatrix} 50 \\ 4 \\ 12 \\ 48 \\ 12 \end{bmatrix}.$$

12. **Implemente** una función en **Python**, llamada `inv_lu`, que calcule la inversa de  $A \in \mathbb{R}^{n \times n}$  utilizando descomposición LU con permutaciones y resolviendo  $n$  sistemas lineales. Testear el algoritmo con la matriz  $A$  del ejercicio 11.
13. Calcule el costo computacional (en flops) para obtener la inversa de una matriz (tome en cuenta las entradas nulas de los vectores canónicos).
14. **Implemente** una función en **Python**, llamada `det_lu`, que calcule el determinante de  $A \in \mathbb{R}^{n \times n}$  utilizando su descomposición LU con permutaciones. Calcule el determinante de la matriz  $A$  del ejercicio 11 y compare el tiempo de ejecución con respecto a la función `np.linalg.det`.
15. En una red en equilibrio los flujos entrantes y salientes en cada nodo deben ser iguales. Encuentre valores de flujos positivos para que la red esté en equilibrio.



¿Existe una única solución?

16. Una sociedad tiene tres necesidades básicas: alimento, vivienda y vestimenta. Por lo tanto, existen tres industrias en la sociedad que producen estos bienes, agricultura, vivienda y confección. Cada una de estas industrias consume una determinada proporción de la producción total de cada bien, según la siguiente tabla:

Consumo \ Producción			
	Agricultura	Vivienda	Confección
Agricultura	0.6	0.2	0.3
Vivienda	0.1	0.4	0.2
Confección	0.2	0.5	0.3

Calcular los presupuestos anuales que debe cobrar cada industria, de modo que sus ingresos sean iguales a sus gastos más las necesidades aleatorias de la sociedad.

17. Toda circunferencia en el plano puede definirse implícitamente mediante la ecuación

$$x^2 + y^2 + ax + by + c = 0.$$

Realice una función en **Python** que dado tres puntos no colineales en el plano: (i) resuelva el sistema lineal resultante, (ii) retorne los valores para  $a$ ,  $b$  y  $c$ , (iii) realice el gráfico de la circunferencia junto a los tres puntos dados.