PUCE Virtual

CLASE

4

Métodos Numéricos

Sistemas de Ecuaciones Lineales I

QS WORLD UNIVERSITY RANKINGS

Educación de calidad

INTRODUCCIÓN DE LA CLASE

GRADO / POSGRADO / TECNOLOGÍAS

Estudia a tu tiempo
son interrupciones

EDUCACIÓN EN LÍNEA DE CALIDAD

En esta clase se aborda el estudio de los sistemas de ecuaciones lineales como uno de los problemas fundamentales del análisis numérico y del álgebra lineal aplicada, debido a su presencia en modelos de ingeniería, física, economía y ciencias computacionales, comenzando con su formulación general tanto en forma algebraica como en su representación matricial (Ax = b), lo que permite interpretar el problema desde una perspectiva vectorial y entender cómo muchos modelos discretizados, como los de interpolación o ecuaciones diferenciales, conducen naturalmente a sistemas lineales cuya solución es clave en el proceso computacional, analizando además la existencia y unicidad de solución a partir de propiedades de la matriz como el rango, el determinante y su carácter singular o no singular, para luego introducir los métodos directos como estrategias que permiten obtener la solución en un número finito de pasos, destacando el método de eliminación de Gauss y la factorización LU, junto con consideraciones de estabilidad numérica y errores de redondeo.

Además, se estudian los métodos iterativos como una alternativa eficiente para sistemas de gran tamaño o con matrices dispersas, partiendo de la idea de reescribir el sistema ($\mathrm{Ax} = \mathrm{b}$) para generar aproximaciones sucesivas desde un valor inicial, analizando métodos como Jacobi y Gauss–Seidel y sus diferencias en la forma de actualizar las soluciones, junto con criterios de convergencia como la dominancia diagonal y el radio espectral que permiten determinar si el método converge, cerrando con una comparación entre métodos directos e iterativos en términos de eficiencia, uso de memoria y estabilidad, con el objetivo de que el estudiante pueda aplicar estos métodos y elegir el más adecuado según el problema.

Resultado o resultados de aprendizaje que será abordado con el contenido de la clase.

1) Analizar problemas matemáticos y computacionales utilizando métodos numéricos adecuados para su solución aproximada.
2) Aplicar técnicas numéricas en la resolución de problemas propios de la ingeniería y las ciencias computacionales.

Reto # (2)

Sistemas de Ecuaciones Lineales I

En las clases anteriores nos centramos en ecuaciones no lineales de una variable. Ahora pasamos a sistemas de ecuaciones lineales, que constituyen uno de los pilares fundamentales de los métodos numéricos y tienen aplicaciones críticas en computación y ciberseguridad.

Presentamos primero la formulación matemática formal, propiedades y teoremas relevantes, y posteriormente las implicaciones directas en ciberseguridad.

Los sistemas de ecuaciones lineales son conjuntos de ecuaciones donde las incógnitas aparecen solo con exponente uno y sin productos entre ellas. Estos sistemas se pueden expresar de manera compacta utilizando matrices.

Formulación de sistemas de ecuaciones lineales

Un sistema de m ecuaciones lineales con n incógnitas se escribe en forma matricial como:

$$\mathbf{Ax} = \mathbf{b}$$

Donde:

- **A** $\in \mathbb{R}^{m \times n}$ es la matriz de coeficientes,
- **x** $\in \mathbb{R}^{n}$ **e**s el vector de incógnitas,
- **b** $\in \mathbb{R}^{m}$ es el vector de términos independientes.

En el caso cuadrado (m = n), el sistema es determinado si det(A) ≠ 0 (matriz invertible), sobredeterminado si m > n, o subdeterminado si m < n. En la práctica numérica nos enfocamos principalmente en sistemas cuadrados no singulares.

Propiedades clave:

- **Matriz invertible** ⇔ det(A) ≠ 0 ⇔ rango(A) = n ⇔ columnas linealmente independientes ⇔ solución única $x = A^{-1}$ b.
- **Número de condición** $\kappa(A) = ||A|| \cdot ||A^{-1}||$ (en alguna norma inducida, usualmente 2-norma o ∞-norma): mide sensibilidad a perturbaciones. Si κ(A) grande (mal condicionado), pequeños errores en A o b producen grandes errores en x.

Teorema de existencia y unicidad (sistemas cuadrados): El sistema $Ax = b$ tiene solución única si y solo si A es invertible. En ese caso, $x = A^{-1} b$.

En ciberseguridad: estos sistemas aparecen en criptoanálisis basado en lattices (e.g., resolución de sistemas lineales modulares o aproximados en ataques BKZ/LLL), criptografía lineal (ataques lineales a cifrados de bloques como AES), reconstrucción de claves en side-channel (regresión lineal sobre múltiples trazas), y en machine learning para seguridad (resolución de mínimos cuadrados en entrenamiento o detección de anomalías).

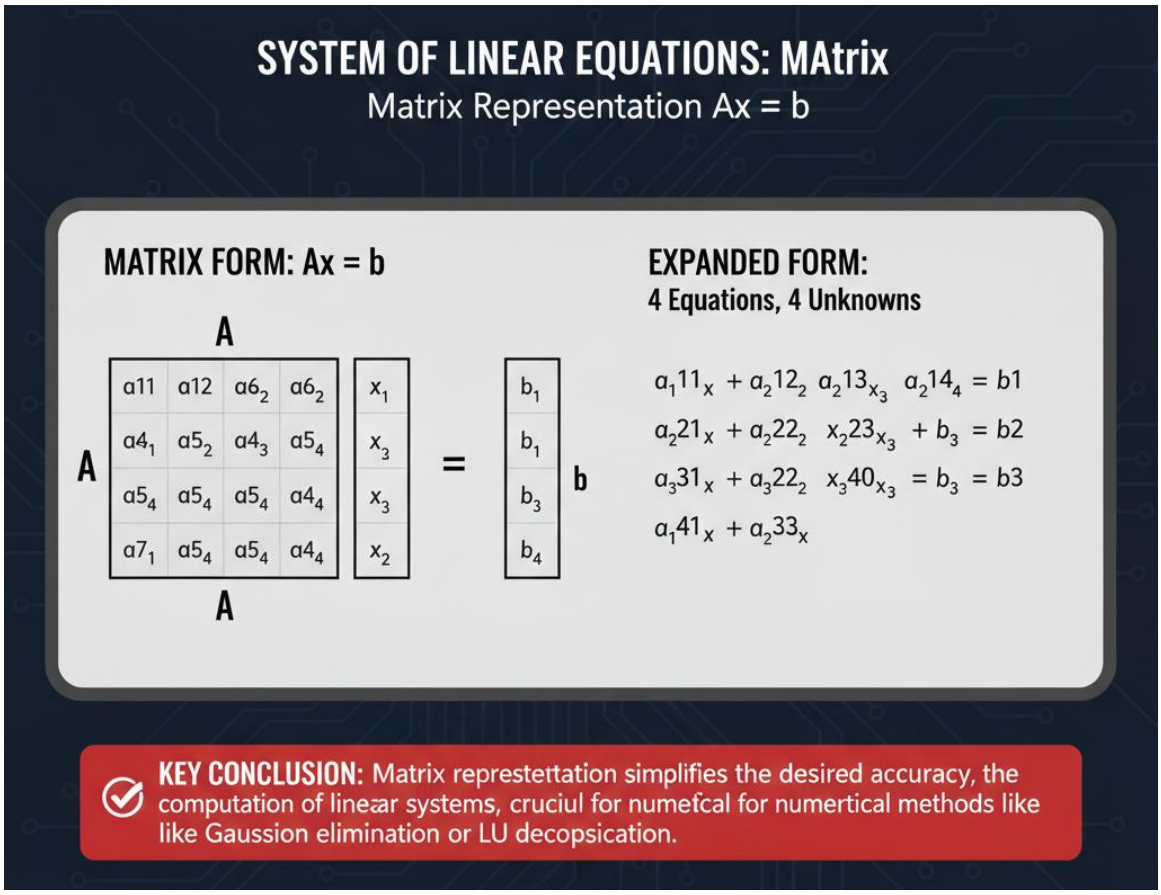

Figura 1: Representación matricial Ax = b para un sistema de 4 ecuaciones y 4 incógnitas.

Creación propia.

Este sistema puede resolverse usando diversos métodos de álgebra lineal, como:

- Método de eliminación de Gauss (o Gauss-Jordan),
- Método de matrices inversas,
- Método de factorización LU,
- O utilizando métodos iterativos si se trata de un sistema grande.

Este formato de Ax = b facilita la aplicación de algoritmos computacionales para resolver sistemas de ecuaciones de manera eficiente.

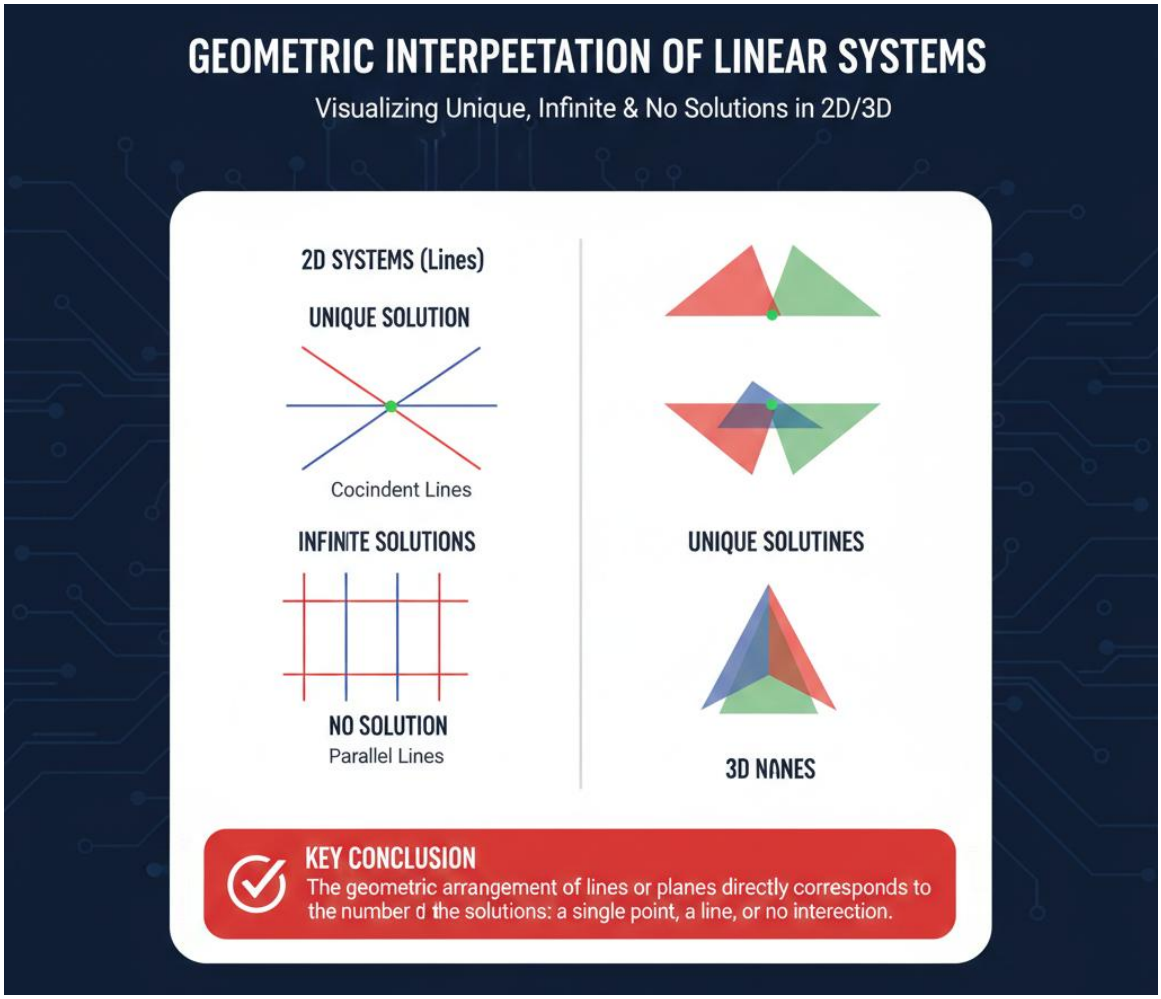

Figura 2: Ilustración geométrica de solución única, infinitas o ninguna en sistemas 2D y 3D.

Creación propia.

Sistemas en 2D (Dos Ecuaciones, Dos Incógnitas)

Para un sistema de 2 ecuaciones lineales con 2 incógnitas (por ejemplo, $ax + by = cy\ dx + ey = f$), la representación geométrica se puede visualizar en el plano cartesiano como dos **líneas**. Dependiendo de la relación entre estas dos líneas, se puede tener uno de los siguientes casos:

a) **Solución Única (Intersección de dos líneas)**:

- En este caso, las dos líneas se intersectan en un único punto.
- La solución del sistema de ecuaciones es el punto de intersección, que representa los valores de las incógnitas xy $y$que satisfacen ambas ecuaciones.
- **Geometría**: Dos líneas que se cortan en un solo punto.
- **Condición**: Las dos ecuaciones tienen una pendiente diferente, es decir, $\frac{a}{b} \neq \frac{d}{e}$.

b) **Infinitas Soluciones (Líneas coincidentes)**:

- En este caso, ambas ecuaciones representan la misma línea en el plano, es decir, las dos ecuaciones son dependientes y no proporcionan nueva información.
- La solución del sistema es cualquier punto sobre la línea común, ya que todos esos puntos satisfacen ambas ecuaciones.
- **Geometría**: Dos líneas coincidentes.
- **Condición**: Las ecuaciones son proporcionales entre sí, es decir, $\frac{a}{b} = \frac{d}{e}$ y $\frac{c}{b} = \frac{f}{e}$.

c) **Ninguna Solución (Líneas paralelas)**:

- En este caso, las dos ecuaciones representan líneas paralelas que nunca se intersectan, lo que significa que no hay ningún punto que satisfaga ambas ecuaciones.
- **Geometría**: Dos líneas paralelas.
- **Condición**: Las ecuaciones son inconsistentes, es decir, las pendientes son iguales pero los términos independientes no lo son (por ejemplo, $\frac{a}{b} = \frac{d}{e}$pero $\frac{c}{b} \neq \frac{f}{e}$).

Sistemas en 3D (Tres Ecuaciones, Tres Incógnitas)

Para un sistema de 3 ecuaciones lineales con 3 incógnitas (por ejemplo, $ax + by + cz = d$, $ex + fy + gz = h$, y $ix + jy + kz = l$), la representación geométrica se extiende al espacio tridimensional. En este caso, las soluciones se representan como planos en 3D, y el comportamiento del sistema dependerá de cómo se relacionen estos planos entre sí.

a) **Solución Única (Intersección de tres planos en un punto)**:

- En este caso, los tres planos se intersectan en un único punto en el espacio 3D.
- La solución del sistema es el punto de intersección de los tres planos, que representa los valores de x, y, y $z$que satisfacen todas las ecuaciones.
- **Geometría**: Tres planos que se intersectan en un solo punto.
- **Condición**: Las tres ecuaciones son independientes, y los planos no son paralelos ni coincidentes.

b) **Infinitas Soluciones (Planos coincidentes o paralelos)**:

- Si dos o más planos son coincidentes o paralelos, el sistema tendrá infinitas soluciones, ya que cualquier punto sobre la intersección de los planos será una solución.

- Dependiendo del caso, las soluciones pueden estar restringidas a una línea o una superficie en el espacio.
- **Geometría**: Planos coincidentes o paralelos.
- **Condición**: Las ecuaciones son dependientes, y los planos no tienen una intersección única, sino que se solapan en una línea o se paralelizan.

c) **Ninguna Solución (Planos paralelos o no coincidentes)**:

- Si los tres planos son paralelos pero no coincidentes, no existe ninguna solución, ya que los planos no se intersectan en ningún punto.
- **Geometría**: Planos paralelos y no coincidentes.
- **Condición**: Las ecuaciones son inconsistentes; los planos son paralelos pero tienen diferentes términos independientes.

Métodos directos

Los métodos directos encuentran la solución exacta (en aritmética exacta) en un número finito de operaciones aritméticas, sin iteraciones ni aproximaciones sucesivas. Incluyen eliminación de Gauss, factorización LU, Cholesky (para matrices simétricas definidas positivas), QR, etc.

Ventajas matemáticas:

- Precisión teórica alta si no hay errores de redondeo.
- Costo computacional predecible: $O(n^3)$ para Gauss/LU en sistemas densos n×n.

Desventajas: sensibles a errores de redondeo en matrices mal condicionadas; no escalan bien para matrices muy grandes o dispersas (donde se prefieren métodos iterativos).

Eliminación de Gauss

La eliminación de Gauss transforma A en una matriz triangular superior U mediante operaciones elementales de fila (intercambio, multiplicación por escalar ≠0, suma de múltiplo de una fila a otra).

Proceso:

1. Para k = 1 a n-1:
   - Pivote: a_{kk} (si =0, intercambiar filas para pivote ≠0 → pivoteo parcial).
   - Eliminadores: $l_{ik} = a_{ik}/a_{kk}$ para $i = k+1$ a n.
   - Actualizar: $\text{fila } i \leftarrow \text{fila} i - l_{ik} \cdot \text{fila } k$ (para i > k, columnas j ≥ k).

Resultado: Ax = b → Ux = c, donde U es triangular superior.

Luego, sustitución hacia atrás (back-substitution): $x_n = c_n/u_{nn} x_k = \left(c_k - \sum_{j=k+1}^{n} u_{kj} x_j\right)/$ $u_{kk} parak = n - 1$ downto 1.

Complejidad: O(n³/3) operaciones flotantes.

Estabilidad: con pivoteo parcial, numéricamente estable en la mayoría de los casos (aunque no siempre óptimo).

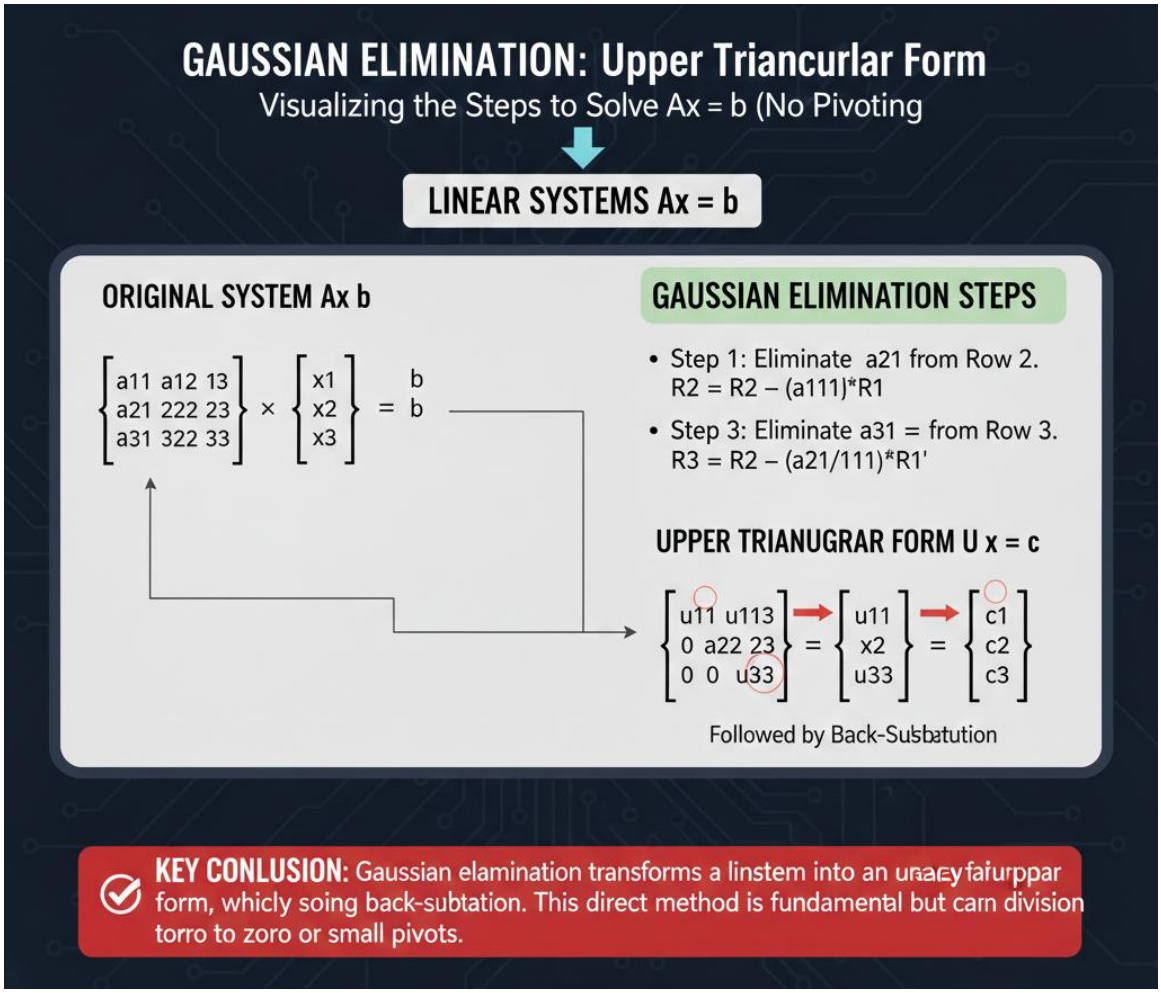

Figura 3: Pasos de eliminación de Gauss sin pivoteo: transformación a triangular superior.

Creación propia.

Factorización LU

Formulación matemática formal

La factorización LU descompone A = L U, donde:

- L es triangular inferior con 1's en la diagonal (unitaria),
- U es triangular superior.

Si no se requiere pivoteo: $A = LU \Rightarrow Ax = b \Leftrightarrow L(Ux) = b \Leftrightarrow Ly = b$(sustitución hacia adelante), luego $Ux = y$ (hacia atrás).

Con pivoteo parcial: $A = P\,L\,U$, donde P es matriz de permutación. Entonces $PAx = Pb \Leftrightarrow L\,U\,x = P\,b$.

Algoritmo de factorización LU con pivoteo parcial (Doolittle o Crout):

- Inicializar L con 1's en diagonal, U vacía.
- Para cada columna k:
    - Buscar pivote máximo en fila $\geq k$ → intercambiar filas (registrar en P).
    - Calcular multiplicadores $l_{ik} = a_{ik}^{(k)} / a_{kk}^{(k)}$ (para $i > k$).
    - Actualizar submatriz restante.

Complejidad: O(n³/3), mismo que Gauss-Jordan, pero más eficiente en almacenamiento y reutilización (factorizar una vez, resolver múltiples b).

Teorema de existencia: Toda matriz cuadrada no singular admite factorización LU con pivoteo parcial (sin crecimiento excesivo de elementos si se usa pivoteo).

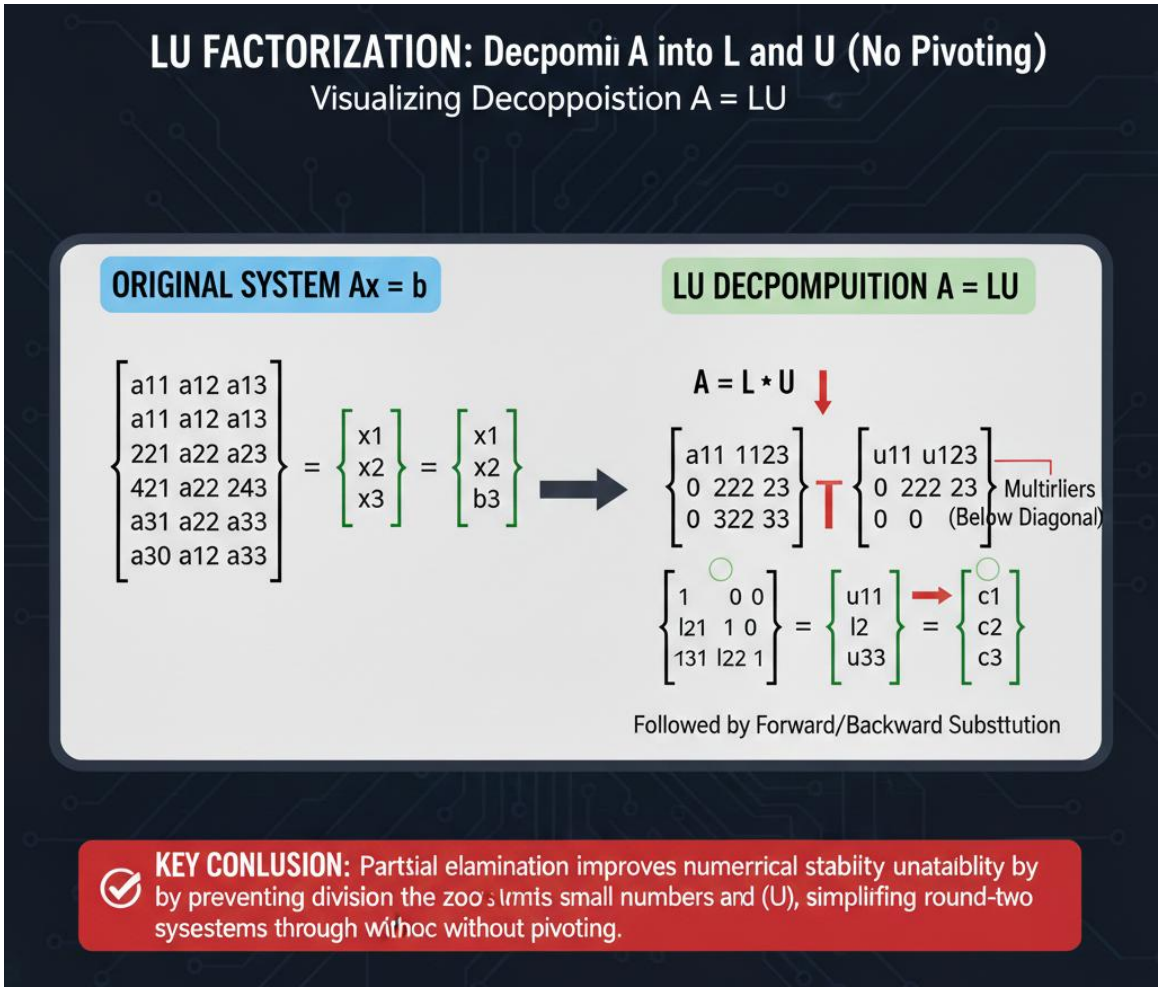

Figura 4: Factorización LU sin pivoteo: L (multiplicadores debajo diagonal) y U (triangular superior).

Creación propia.

Dado un sistema de ecuaciones lineales $Ax = b$, el objetivo es descomponer la matriz Aen el producto de una matriz triangular inferior Ly una matriz triangular superior U, es decir:

$$\boldsymbol{A = LU}$$

Sin embargo, para mejorar la estabilidad numérica, se realiza un proceso de pivoteo parcial, lo que implica intercambiar filas de la matriz Adurante el proceso de eliminación para garantizar que los pivotes sean lo más grandes posible, lo que minimiza el error numérico.

Enlace 1: https://en.wikipedia.org/wiki/LU_decomposition Ver en Wikipedia: LU decomposition Recurso con demostración de existencia, algoritmo Doolittle/Crout y pivoteo parcial.

Sistemas de Ecuaciones Lineales II

En la Clase 7 estudiamos métodos directos para resolver sistemas lineales $Ax = b$ (eliminación de Gauss, factorización LU). Ahora exploramos los métodos iterativos, ideales para matrices grandes, dispersas o mal condicionadas donde los métodos directos son costosos o inestables numéricamente.

Presentamos primero la formulación matemática formal, algoritmos, teoremas de convergencia y criterios, y posteriormente las aplicaciones en ciberseguridad.

Los métodos iterativos son herramientas esenciales para resolver sistemas de ecuaciones lineales, particularmente en situaciones donde se enfrentan sistemas grandes que no son prácticos de resolver mediante métodos directos como la eliminación de Gauss o la descomposición LU. Estos métodos no proporcionan una solución exacta de inmediato, sino que se aproximan a la solución mediante sucesivas iteraciones. El Método de Jacobi, por ejemplo, se basa en descomponer cada ecuación del sistema en términos de las incógnitas que aún no han sido actualizadas. Esto implica que en cada iteración, se usan los valores de las incógnitas obtenidas en la iteración anterior para calcular los nuevos valores. Este proceso se repite de forma sucesiva hasta que las diferencias entre los valores de las incógnitas en dos iteraciones consecutivas son mínimas, indicando que se ha alcanzado una solución aproximada suficientemente precisa.

El Método de Gauss-Seidel es una variante que mejora el proceso iterativo de Jacobi. A diferencia de este último, en Gauss-Seidel se utilizan los valores más recientes de las incógnitas dentro de la misma iteración, lo que puede resultar en una convergencia más rápida, ya que se aprovecha la información actualizada de inmediato. Este método es generalmente más eficiente, pero aún depende de las características del sistema de ecuaciones para su efectividad. Ambos métodos, aunque muy útiles, requieren un análisis cuidadoso de la convergencia para garantizar que las soluciones obtenidas sean válidas. Existen ciertos

criterios matemáticos que permiten determinar si un sistema de ecuaciones lineales se comportará de manera convergente bajo estos métodos iterativos. La convergencia se refiere a la capacidad del proceso iterativo para acercarse cada vez más a la solución exacta, y si estos criterios no se cumplen, los métodos podrían no converger o hacerlo muy lentamente, lo que puede llevar a resultados inexactos.

Métodos iterativos

Los métodos iterativos generan una secuencia $\{x^{(k)}\}$ que converge a la solución $x^* = A^{-1}b$ (asumiendo A invertible). Se basan en una **división** (splitting) de $A = M - N$, donde M es fácil de invertir (e.g., diagonal o triangular), y la iteración es:

$\boldsymbol{Mx^{(k+1)} = Nx^{(k)} + b}$ o equivalentemente $\boldsymbol{x^{(k+1)} = M^{-1}Nx^{(k)} + M^{-1}b = Gx^{(k)} + c}$

donde $G = M^{-1}$ N es la **matriz de iteración** y $c = M^{-1}b$.

Teorema fundamental de convergencia (para cualquier método iterativo $x\text{^}\{(k+1)\} = G\ x\text{^}\{(k)\} + c$): La iteración converge a la solución únicax^* para cualquier inicial $x^{(0)}$ si y solo si el radio espectral ρ(G) < 1, es decir, todas las autofunciones de G tienen módulo <1.

Además, si ρ(G) < 1, el error $e^{(k)} = x^{(k)} - x^*$ satisface $\|e^{(k)}\| \leq C\rho(G)^k\|e^{(0)}\|$ para alguna constante C > 0 y norma subordinada. La tasa asintótica de convergencia es -$\log_{10}$(ρ(G)) dígitos por iteración (cuanto menor ρ(G), más rápido).

Método de Jacobi

División: A = D + (L + U), donde D diagonal, L estrictamente triangular inferior, U estrictamente superior. Entonces M = D, N = -(L + U).

Iteración: $\boldsymbol{Dx^{(k+1)} = -(L+U)x^{(k)} + b}$ o componente a componente:

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j\neq i} a_{ij}x_j^{(k)}\right), \quad i = 1, \dots, n$$

Matriz de iteración: $G_J = -D^{-1}(L+U) = I - D^{-1}A$.

Teorema de convergencia

El método de Jacobi converge para cualquier $x^{(0)}$ si $\rho(G_J) < 1$. Condición suficiente (no necesaria): A estrictamente diagonalmente dominante por filas: $\forall \boldsymbol{i}: |\boldsymbol{a_{ii}}| > \sum_{\mathbf{j\neq i}}|\mathbf{a_{ij}}|$ En este caso, ‖G_J‖∞ < 1 (usando norma ∞), lo que implica $\boldsymbol{\rho(G_J) \leq \|G_J\|\infty < 1}$.

Orden de convergencia: lineal (orden 1), tasa ρ(G_J).

Enlace 2: https://en.wikipedia.org/wiki/Jacobi_method Ver en Wikipedia: Jacobi method Este artículo detalla la formulación, matriz de iteración y condición de convergencia (ρ(G_J) < 1, diagonal dominance).

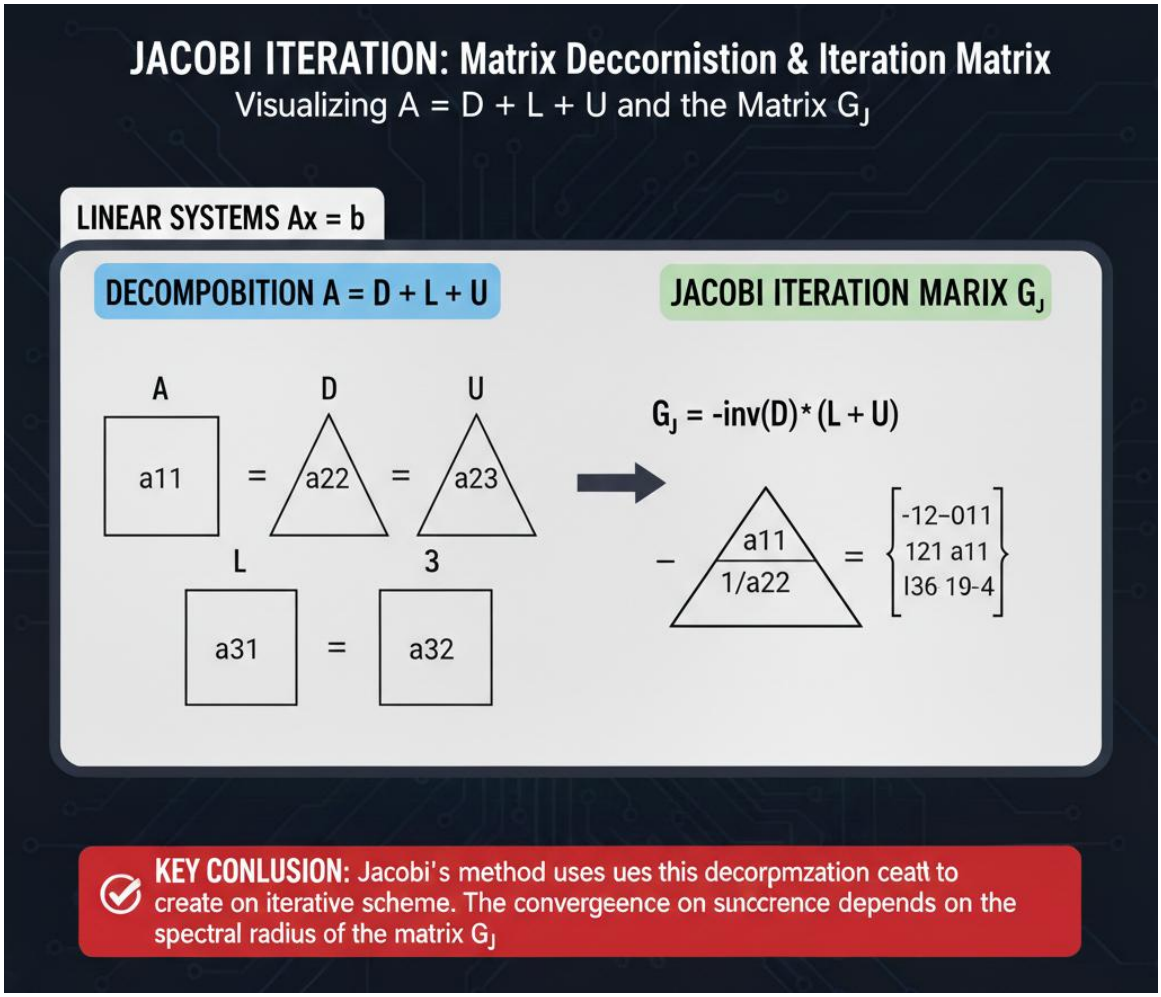

Figura 5: Descomposición A = D + L + U para Jacobi y matriz de iteración G_J.

Creación propia.

Método de Gauss–Seidel

División: A = (D + L) + U, donde D + L triangular inferior. Entonces M = D + L, N = -U.

Iteración: $(\mathbf{D}+\mathbf{L})\mathbf{x}^{(\mathbf{k}+\mathbf{1})} = -\mathbf{U}\mathbf{x}^{(\mathbf{k})} + \mathbf{b}$ o componente a componente (actualización inmediata): $\mathbf{x}_\mathbf{i}^{(\mathbf{k}+\mathbf{1})} = \frac{\mathbf{1}}{\mathbf{a}_{\mathbf{ii}}}\left(\mathbf{b}_\mathbf{i} - \sum_{\mathbf{j}=\mathbf{1}}^{\mathbf{i}-\mathbf{1}} \mathbf{a}_{\mathbf{ij}}\mathbf{x}_\mathbf{j}^{(\mathbf{k}+\mathbf{1})} - \sum_{\mathbf{j}=\mathbf{i}+\mathbf{1}}^{\mathbf{n}} \mathbf{a}_{\mathbf{ij}}\mathbf{x}_\mathbf{j}^{(\mathbf{k})}\right), \quad \mathbf{i} = \mathbf{1}, \dots, \mathbf{n}$

Matriz de iteración: $\mathrm{G_{GS}} = -(\mathrm{D}+\mathrm{L})^{-1}\mathrm{U}$.

Teorema de convergencia

El método converge para cualquier $\mathrm{x}^{(0)}$ si $\rho(\mathrm{G_{GS}}) < 1$. Condiciones suficientes:

a) A estrictamente diagonalmente dominante (por filas o columnas).

b) A simétrica y definida positiva (entonces converge, y Gauss-Seidel es más rápido que Jacobi).

Gauss-Seidel converge más rápido que Jacobi cuando converge ($\rho(G_{GS}) \leq \rho(G_J)^2$ en muchos casos, e.g., matrices simétricas positivas definidas).

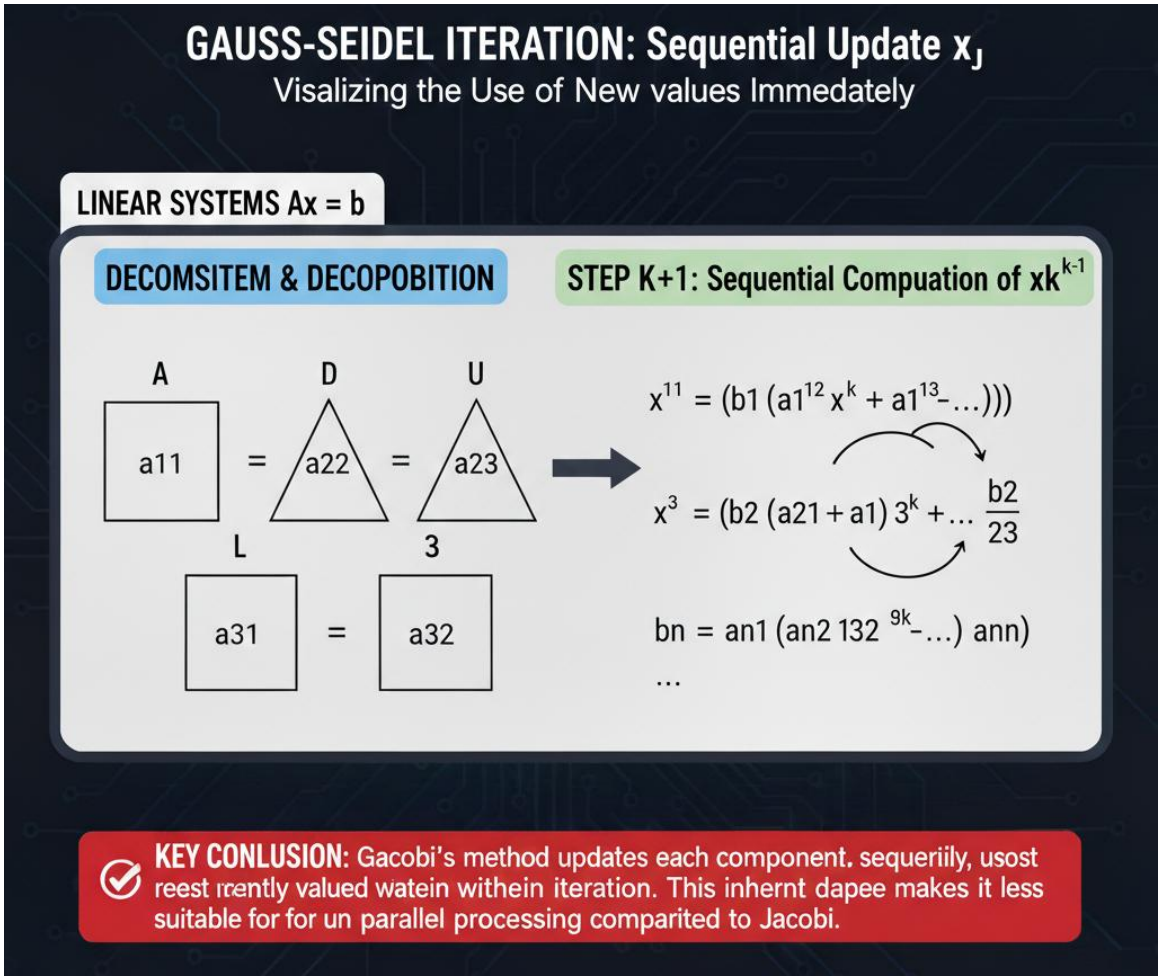

Figura 6: Actualización secuencial en Gauss-Seidel: usa valores nuevos inmediatamente.

Creación propia.

La figura muestra el proceso de actualización secuencial en el método de Gauss-Seidel, en el cual los nuevos valores de la solución se utilizan inmediatamente para calcular las siguientes iteraciones. A diferencia del método de Jacobi, donde todos los valores de la iteración anterior se emplean para calcular los nuevos valores en paralelo, en Gauss-Seidel se actualiza el valor de cada componente de la solución de manera secuencial, es decir, tan pronto como se calcula un nuevo valor ($x_i^{(k+1)}$), este se utiliza de inmediato para actualizar el siguiente valor ($x_j^{(k+1)}$). Esto significa que, durante una misma iteración, los componentes de la solución se van actualizando uno a uno, lo que puede acelerar la convergencia del método en comparación con el método de Jacobi, ya que cada actualización puede aprovechar los nuevos valores que ya se han calculado.

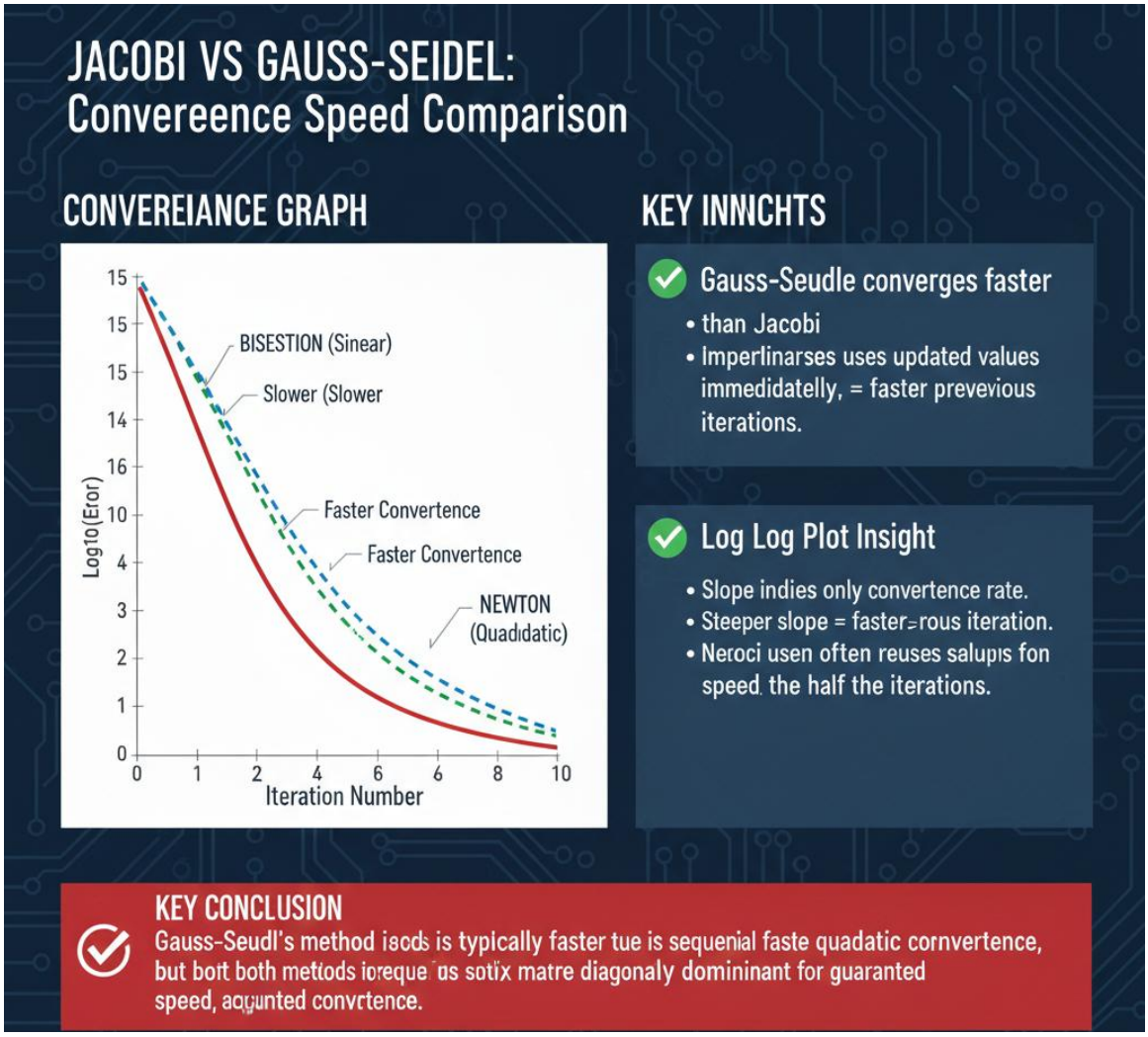

Figura 7: Comparación de error vs iteraciones: Gauss-Seidel más rápido que Jacobi.

Criterios de convergencia en sistemas

Criterios prácticos de parada:

- Residuo pequeño: $\|r^{(k)}\| = \|Ax^{(k)} - b\| < \varepsilon$ (norma ∞ o 2 usual).
- Cambio relativo: $\|x^{(k+1)} - x^{(k)}\| / \|x^{(k+1)}\| < \varepsilon$.
- Máximo de iteraciones (para evitar bucles infinitos si no converge).

Criterios teóricos para convergencia garantizada:

1. **$\rho(G) < 1$** (necesario y suficiente).
2. **$\|G\| < 1$** en alguna norma subordinada (suficiente).
    - Para Jacobi: si A estrictamente diagonalmente dominante → ‖G_J‖_∞ < 1.
    - Para Gauss-Seidel: lo mismo, más simétrica positiva definida.
3. **Diagonal dominance estricta o irreducible** (suficiente para ambos métodos).

Teorema general: Si ρ(G) < 1, converge para cualquier inicial; si ρ(G) ≥ 1, diverge o converge condicionalmente.

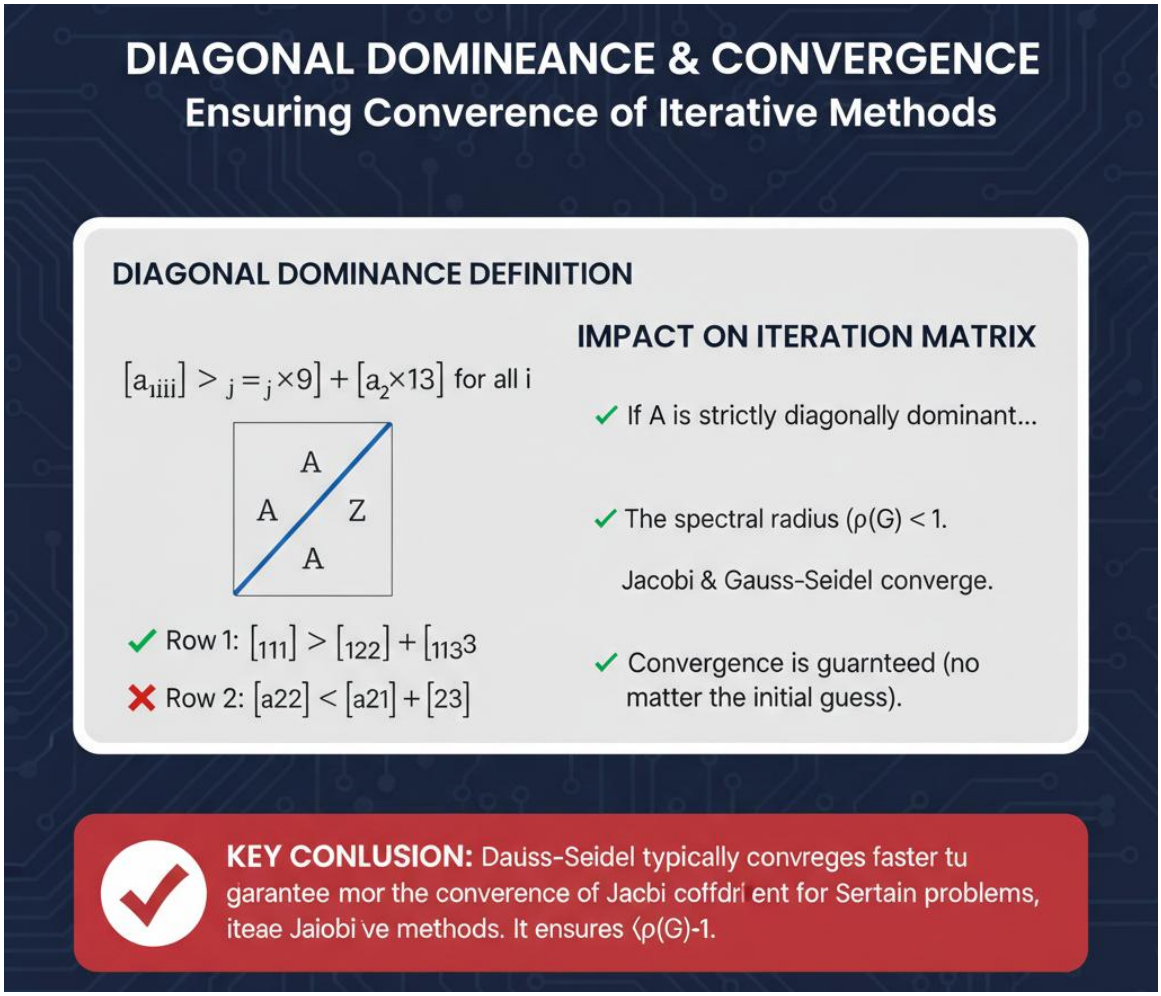

Figura 8: Matriz estrictamente diagonalmente dominante y su impacto en ρ(G).

Creación propia.

Una matriz (A) es estrictamente diagonalmente dominante si, para cada fila (i), el valor absoluto del elemento en la diagonal es mayor que la suma de los valores absolutos de los elementos fuera de la diagonal en esa misma fila, es decir,

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}|$$

Glosario:

- Golub, G. H., & Van Loan, C. F. (2013). Matrix computations (4ª ed.). Johns Hopkins University Press.
- Higham, N. J. (2002). Accuracy and stability of numerical algorithms (2ª ed.). SIAM.
- Burden, R. L., Faires, J. D., & Burden, A. M. (2015). Numerical analysis (10ª ed.). Cengage Learning.
- Paar, C., & Pelzl, J. (2010). Understanding cryptography: A textbook for students and practitioners. Springer.
- Boneh, D., & Shoup, V. (2020). A graduate course in applied cryptography. https://toc.cryptobook.us

PUCE Virtual

síguenos en:

www.pucevirtual.puce.edu.ec