# Resumen

En este laboratorio, se lleva a cabo un análisis exhaustivo de los algoritmos de eliminación de Gauss y Gauss-Jordán[1]**,** con el objetivo de entender su funcionamiento, evaluar su eficiencia y precisión, y proponer mejoras para casos específico.

**Fecha límite de entrega**: miércoles, **27-marzo-2024 12:00**. Se puede trabajar colaborativamente, pero el reporte es individual.

Cada alumno remitirá **cinco (5) archivos** en el siguiente formato:

**TP0X-T-S3pY-Nombre-Apellido.m** Archivo de Matlab/Octave conteniendo las pruebas realizadas. La **Y** representa la tarea según corresponda a la sección 3.**1**,…,3.**4**. (son 4 archivos).

**TP0X-I-Nombre-Apellido.pdf** Informe conforme el formato de de la sección Redacción de Informe.

La X representa el número de tarea asignada.

# Objetivos

* Realizar pruebas exhaustivas de la función de eliminación de Gauss utilizando matrices y vectores proporcionados.
* Verificar la corrección de la función para diversos escenarios, incluyendo matrices singulares, diagonales y con ceros en posiciones estratégicas.
* Analizar la eficiencia y precisión de la función en cada caso de prueba y discutir posibles mejoras o consideraciones adicionales.
* Proponer e implementar una mejora para manejar el pivoteo parcial (cuando el pivote es cero) para una estabilidad numérica mejorada.
* Comparar la función de eliminación de Gauss implementada con las proporcionadas por el Profesor Strang, incluyendo los archivos **alim.m** y **plu.m** de **Tcodes.zip** [2][3], que consideran la permutación de ecuaciones cuando un pivote es cero.
* Realizar pruebas de escritorio (cálculos manuales paso a paso) para matrices de hasta 3x3 para garantizar la corrección del algoritmo.

# Actividades

## Pruebas al algoritmo de eliminación propuesto gauss\_elimination.m:

**Nombre del script: TP0X-T-S3p1-Nombre-Apellido.m**

Utilizar matrices y vectores de prueba proporcionados para pruebas exhaustivas.

Los escenarios de prueba incluyen:

Matrices cuadradas no singulares (ejemplo: A = [1 2; 3 4], b = [5; 6]).

Matrices no cuadradas (ejemplo: A\_no\_cuadrada = [1 2; 3 4; 5 6], b\_no\_cuadrada = [5; 6; 7]).

Matrices mal condicionadas (ejemplo: A\_mal\_formada = [0.400 99.6; 75.3 100], b\_mal\_formado = [100; 30.0]).

Verificar los resultados y discutir cualquier discrepancia o problema encontrado.

Realizar pruebas de escritorio y pruebas en MATLAB/Octave para validar la implementación.

## Comparación con las implementaciones del profesor Strang (Tcodes.zip)

**Nombre del script: TP0X-T-S3p2-Nombre-Apellido.m**

Comparar la función de eliminación de Gauss implementada con las implementaciones proporcionadas (alim.m y plu.m) por el Profesor Strang.

Crear un diagrama de flujo para el algoritmo.

Discutir similitudes, diferencias y posibles áreas de mejora

## Mejora de implementación para pivoteo parcial

**Nombre del script: TP0X-T-S3p3-Nombre-Apellido.m**

Explicar la importancia del pivoteo en la eliminación de Gauss basado en el artículo de Sudrastawa (2022) [1]**.**

Crear un diagrama de flujo para el algoritmo.

Proponer e implementar una mejora de pivoteo parcial para manejar pivotes cero para una estabilidad numérica mejorada.

Probar la función mejorada con escenarios relevantes y comparar los resultados con la implementación original.

## Implementación del algoritmo de Gauss-Jordan

**Nombre del script: TP0X-T-S3p4-Nombre-Apellido.m**

Implementar el algoritmo de Gauss-Jordan basado en el artículo de Sudrastawa (2022)[1]**.**

Crear un diagrama de flujo para el algoritmo.

Realizar pruebas de escritorio y pruebas en MATLAB/Octave para validar la implementación.

## Redacción de informe

**Nombre del script: TP0X-I-Nombre-Apellido.pdf**

Escribir un informe en LaTeX utilizando [Overleaf](https://www.overleaf.com/latex/templates/template-for-journal-of-computational-design-and-engineering-jcde-v-dot-2-0-updated-march-2024/zzdzmdyggczf) [4]**.**

Incluir secciones como Introducción, Metodología, Resultados, Discusión y Conclusiones.

Seguir el formato IEEE [5] para el artículo.

Incluir un análisis detallado de los algoritmos implementados, mejoras y comparaciones con métodos existentes.

# Bibliografía

|  |  |
| --- | --- |
| [1] | I. P. A. Sudrastawa, A. Parwata, M. W. Adhi Saputra, I. G. A. Made Wirautama y I. W. S. Malan Vergantana, «Conceptual and Practical Review of Gaussian Elimination and Gauss-Jordan Reduction,» de *Jurnal Ilmu Komputer Indonesia*, Indonesia, 2022. |
| [2] | G. Strang, Introduction to linear algebra, Cambridge Press, 2009. |
| [3] | M. OCW, «Matlab Teaching Codes,» [En línea]. Disponible en: https://web.mit.edu/18.06/www/Course-Info/Tcodes.html. [Último acceso: 20 marzo 2024]. |
| [4] | Overleaf, «Template for Journal of Computational Design and Engineering (JCDE) v2.0,» [En línea]. Disponible en: https://www.overleaf.com/latex/templates/template-for-journal-of-computational-design-and-engineering-jcde-v-dot-2-0-updated-march%202024/zzdzmdyggczf. [Último acceso: 20 marzo 2024]. |
| [5] | IEEE, «IEEE Editorial Style Manual (Online),» [En línea]. Disponble en: https://www.ieee.org/content/dam/ieee-org/ieee/web/org/conferences/style\_references\_manual.pdf. [Último acceso: 20 marzo 2024]. |

Anexo 01 – Diagrama de flujo del algoritmo **gauss\_elimination.m**

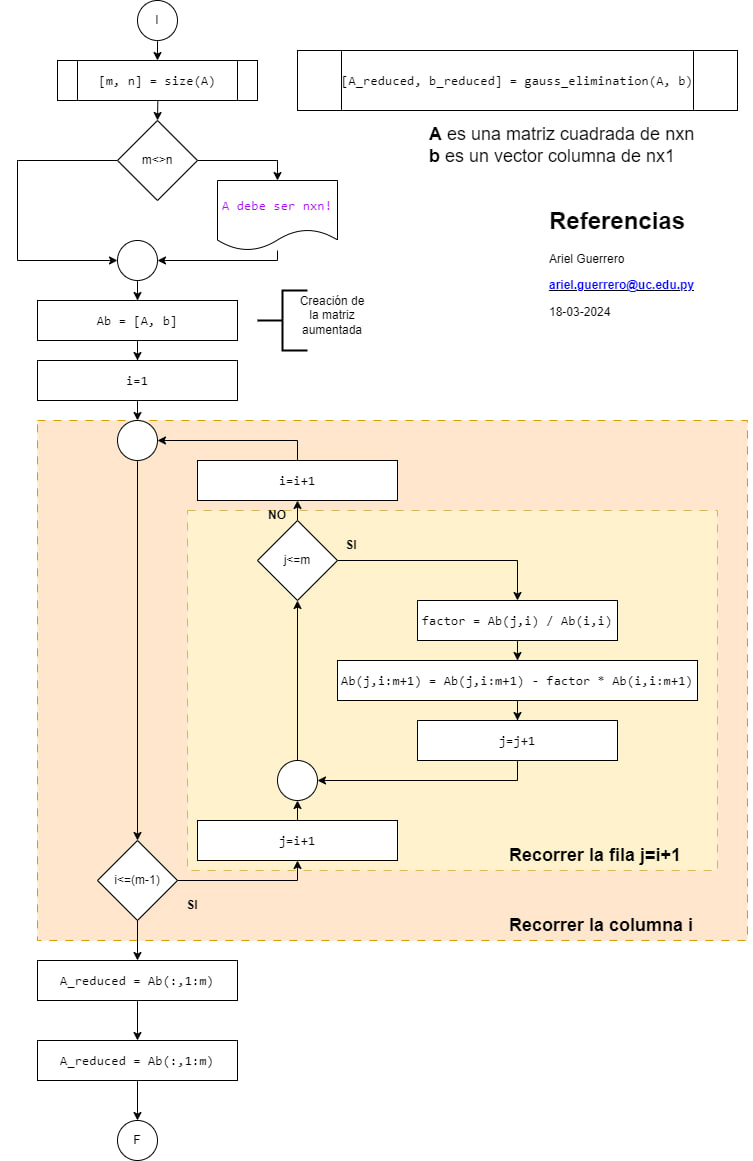


Figura - Algoritmo de eliminación de gauss sin pivoteo parcial

Anexo 02 – Listado del código fuente de **gauss\_elimination.m**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | function [A\_reduced, b\_reduced] = gauss\_elimination(A, b)  % Verificar si la matriz es cuadrada  [m, n] = size(A);  if m ~= n  error('La matriz A debe ser cuadrada para aplicar el método de eliminación de Gauss.');  end    % Combinar la matriz A y el vector b en una sola matriz aumentada  Ab = [A, b];    %% Aplicar la eliminación de Gauss sin pivoteo parcial  for i = 1:m-1 % Se recorren las columnas de la matriz aumentada  for j = i+1:m % se recorren las filas  factor = Ab(j,i) / Ab(i,i);  Ab(j,i:m+1) = Ab(j,i:m+1) - factor \* Ab(i,i:m+1);  end  end    % Obtener la matriz A reducida y el vector b reducido  A\_reduced = Ab(:,1:m);  b\_reduced = Ab(:,m+1);  end |

Anexo 03 – Prueba de escitorio **gauss\_elimination.m**

Realizar primero en el anotador (lápiz y papel) la prueba, luego puede automatizar con un script en Matlab.

Por ejemplo, sea A = [1 2 3; 2 5 6; 3 6 9]. La prueba de escritorio ejecutando paso a paso sería:

|  |
| --- |
| -------------------------------------------------------------  Paso | i | j | factor | Ab  -------------------------------------------------------------  1 | 1 | 2 | 2.0000 | [1 2 3 ; 0 1 0 ; 3]  1 | 1 | 3 | 3.0000 | [1 2 3 ; 0 1 0 ; 0]  2 | 2 | 3 | 0.0000 | [1 2 3 ; 0 1 0 ; 0]  ------------------------------------------------------------- |