UNIVERSIDAD NACIONAL DEL ALTIPLANO PUNO

FACULTAD DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA

ESCUELA PROFESIONAL DE INGENIERÍA ESTADÍSTICA E INFORMÁTICA

Proyecto Web Aplicativo Método Sustitucion, Gauss-Jordan y Cramer

CURSO: MÉTODOS DE OPTIMIZACIÓN

DOCENTE:

ING. Fred Torres Cruz

PRESENTADO POR:

Edilfonso Muñoz Anccori

SEMESTRE: V NIV

PUNO-PERÚ
2025

1. Introducción

En la actualidad, los métodos numéricos desempeñan un papel crucial en la resolución de problemas matemáticos complejos que no pueden resolverse de manera analítica. Los sistemas de ecuaciones lineales son una clase de problemas que surgen con frecuencia en diversas áreas de la ingeniería, la física, la economía y muchas otras disciplinas. Estos sistemas de ecuaciones pueden representarse mediante matrices y vectores, y existen varios métodos numéricos que permiten encontrar sus soluciones.

En este proyecto, se ha desarrollado una aplicación utilizando la biblioteca Streamlit de Python, que permite resolver sistemas de ecuaciones lineales a través de tres métodos numéricos: Sustitución, Gauss-Jordan y Cramer. La aplicación es de fácil uso e interactiva, lo que permite al usuario ingresar las matrices de coeficientes y el vector de términos constantes, y luego seleccionar el método para obtener la solución del sistema.

El objetivo de este proyecto es proporcionar una herramienta que facilite la comprensión y la resolución de sistemas de ecuaciones lineales, tanto para estudiantes como para profesionales en el campo de las ciencias e ingenierías. La interfaz gráfica de la aplicación está diseñada de manera sencilla e intuitiva, de modo que cualquier usuario, sin importar su nivel de conocimiento previo, pueda interactuar con ella de manera eficiente.

2. Metodología

3. Metodología

La resolución de sistemas de ecuaciones lineales es uno de los problemas fundamentales en álgebra lineal, y se pueden utilizar diversos métodos para obtener soluciones de manera eficiente. En este trabajo, se han implementado tres métodos clásicos: **Método de Sustitución**, **Método de Gauss-Jordan** y **Método de Cramer**. A continuación se detallan los procedimientos y fórmulas involucradas en cada uno de estos métodos.

3.1. Método de Sustitución

El **Método de Sustitución** es un enfoque paso a paso que se utiliza principalmente para resolver sistemas de ecuaciones lineales en los cuales una de las incógnitas puede ser despejada de una de las ecuaciones. El procedimiento es el siguiente:

1. Seleccionar una de las ecuaciones y despejar una de las incógnitas en términos de las otras.
2. Sustituir esta expresión en las demás ecuaciones.
3. Repetir el proceso de sustitución hasta obtener el valor de todas las incógnitas.

Ejemplo de procedimiento:
Considere el siguiente sistema de ecuaciones:

$$\begin{aligned} 2x + 3y &= 5, \\ x - y &= 1. \end{aligned}$$

En este caso, se puede despejar x de la segunda ecuación:

$$x = y + 1.$$

Luego, sustituimos esta expresión en la primera ecuación:

$$2(y + 1) + 3y = 5.$$

Resolviendo esta ecuación:

$$2y + 2 + 3y = 5 \quad \Rightarrow \quad 5y = 3 \quad \Rightarrow \quad y = \frac{3}{5}.$$

Sustituyendo el valor de y en la ecuación $x = y + 1$:

$$x = \frac{3}{5} + 1 = \frac{8}{5}.$$

Por lo tanto, la solución es $x = \frac{8}{5}$ y $y = \frac{3}{5}$.

3.2. Método de Gauss-Jordan

El **Método de Gauss-Jordan** es una extensión del método de eliminación de Gauss. Consiste en transformar la matriz aumentada del sistema a una forma escalonada reducida, de manera que la solución del sistema sea evidente.

El procedimiento es el siguiente:

1. Escriba el sistema en forma de matriz aumentada.
2. Realice operaciones elementales sobre las filas (intercambio, multiplicación por un escalar, suma de filas) para obtener ceros debajo y encima de cada pivote.
3. Cuando la matriz esté en forma escalonada reducida, las soluciones serán los elementos de la última columna de la matriz.

La matriz aumentada es de la forma:

$$\left[\begin{array}{ccc|c} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{array}\right]$$

Donde las filas representan las ecuaciones del sistema, y la última columna contiene los términos constantes.

El objetivo es convertir la matriz en la forma:

$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & x_1 \\ 0 & 1 & 0 & x_2 \\ 0 & 0 & 1 & x_3 \end{array}\right]$$

Las soluciones del sistema serán los valores de $x_1, x_2, \ldots$.

3.3. Método de Cramer

El **Método de Cramer** es un método directo basado en determinantes, que se aplica cuando el sistema tiene un número igual de ecuaciones e incógnitas, y la matriz de coeficientes es invertible (es decir, su determinante no es cero).

El procedimiento es el siguiente:

1. Escriba el sistema en forma matricial:
$$A\mathbf{x} = \mathbf{b},$$
donde A es la matriz de coeficientes, $\mathbf{x}$ es el vector de incógnitas, y $\mathbf{b}$ es el vector de términos constantes.

2. Calcule el determinante de la matriz A, denotado por $\det(A)$.

3. Si $\det(A) \neq 0$, calcule los determinantes de las matrices obtenidas sustituyendo las columnas de A por el vector $\mathbf{b}$. Cada incógnita x_i se obtiene mediante la fórmula:
$$x_i = \frac{\det(A_i)}{\det(A)},$$
donde A_i es la matriz obtenida reemplazando la i-ésima columna de A por el vector $\mathbf{b}$.

Por ejemplo, para el sistema:
$$\begin{aligned} 2x + y &= 5, \\ 3x + 4y &= 6, \end{aligned}$$
la matriz A es:
$$A = \begin{bmatrix} 2 & 1 \\ 3 & 4 \end{bmatrix},$$
y el vector $\mathbf{b}$ es:
$$\mathbf{b} = \begin{bmatrix} 5 \\ 6 \end{bmatrix}.$$
Se calculan los determinantes:
$$\det(A) = (2)(4) - (1)(3) = 5.$$
Luego, se calculan los determinantes de las matrices A_1 y A_2 obtenidas al reemplazar la primera y segunda columna de A por $\mathbf{b}$, respectivamente:
$$A_1 = \begin{bmatrix} 5 & 1 \\ 6 & 4 \end{bmatrix}, \quad \det(A_1) = (5)(4) - (1)(6) = 14,$$
$$A_2 = \begin{bmatrix} 2 & 5 \\ 3 & 6 \end{bmatrix}, \quad \det(A_2) = (2)(6) - (5)(3) = -7.$$
Finalmente, las soluciones son:
$$x = \frac{\det(A_1)}{\det(A)} = \frac{14}{5}, \quad y = \frac{\det(A_2)}{\det(A)} = \frac{-7}{5}.$$

4. Implementación

Para la implementación de la aplicación, se utilizó el marco `Streamlit`, que permite crear aplicaciones web interactivas de manera sencilla con Python. A continuación, se presenta el código utilizado para crear la interfaz de la aplicación y resolver los sistemas de ecuaciones.

Listing 1: Código de la aplicación en Streamlit

```
import streamlit as st
import numpy as np

# Funciones para resolver sistemas de ecuaciones
def resolver_sustitucion(A, b):
    st.write("### Paso 1: Resoluci n por el m todo de sustituci n")
    # F rmulas: x_i = (b_i - sum(A_ij * x_j)) / A_ii para cada i
    st.write("La ecuaci n general es: x_i = (b_i -    A_ij * x_j) / A_ii para cada i")

    # Soluci n paso a paso
    st.write("### Pasos para resolver cada inc gnita:")
    st.write("Para cada \( x_i \):")

    solucion = np.linalg.solve(A, b)

    for i in range(len(b)):
        st.write(f"**Ecuaci n {i+1}:** x_{i+1} = {solucion[i]:.4f}")
        # Mostrar la ecuaci n que se us   para obtener el valor
        st.write(f"Ecuaci n para \( x_{i+1} \): x_{i+1} = ({b[i]} - sum(A_{i,:} * x)) / A_{i,i}")

    return solucion

def resolver_gauss_jordan(A, b):
    st.write("### Paso 1: Resoluci n por el m todo de Gauss-Jordan")
    augmented_matrix = np.hstack([A, b.reshape(-1, 1)])
    n = A.shape[0]

    st.write("Matriz aumentada inicial:")
    st.write(augmented_matrix)
    st.write("La forma general de Gauss-Jordan es:")
    st.write("Aumentada: [ A | b ]")

    for i in range(n):
        augmented_matrix[i] = augmented_matrix[i] / augmented_matrix[i, i]  # Normalizar la fila
        for j in range(n):
            if i != j:
                augmented_matrix[j] -= augmented_matrix[i] * augmented_matrix[j, i]

        # Mostrar los pasos de la matriz aumentada
        st.write(f"Paso {i+1}: Eliminar elementos debajo y arriba de A[{i+1},{i+1}]")
        st.write(augmented_matrix)

    st.write("### Resultado:")
    st.write(f"Soluciones: {augmented_matrix[:, -1]}")
```

```
    return augmented_matrix[:, -1]

def resolver_cramer(A, b):
    st.write("### Paso 1: Resoluci n por el m todo de Cramer")
    det_A = np.linalg.det(A)
    if det_A == 0:
        st.error("El sistema no tiene soluci n  nica .")
        return
    st.write(f"Determinante de A: |A| = {det_A}")
    st.write("La f rmula de Cramer es: x_i = |A_i| / |A|, donde A_i es la matriz A reemplazada por la columna b.")

    soluciones = []
    for i in range(A.shape[1]):
        Ai = A.copy()
        Ai[:, i] = b
        det_Ai = np.linalg.det(Ai)
        st.write(f"Determinante de A_{i+1} (reemplazando columna {i+1} con b): |A_{i+1}| = {det_Ai}")
        soluciones.append(det_Ai / det_A)
        st.write(f"x_{i+1} = {det_Ai / det_A:.4f}")

    return soluciones

# Interfaz de usuario
st.set_page_config(page_title="Resoluci n de Sistemas de Ecuaciones Lineales", layout="wide")
st.title("       Resoluci n de Sistemas de Ecuaciones Lineales")

# Barra lateral para ingresar datos
st.sidebar.header("Configuraci n del Sistema")
n = st.sidebar.number_input("N mero de ecuaciones (y variables):", min_value=2, max_value=5, value=3)

# Selecci n de m todo
metodo = st.sidebar.selectbox("Selecciona el m todo:", ["Sustituci n", "Gauss-Jordan", "Cramer"])

# Crear las matrices
st.sidebar.header("Entrada de la Matriz y Vector de T rminos Independientes")

# Entradas din micas para las matrices
A = np.zeros((n, n))
b = np.zeros(n)

# Crear campos para la matriz A
for i in range(n):
    for j in range(n):
        A[i, j] = st.sidebar.number_input(f"a[{i+1},{j+1}]", value=0.0)

# Crear campo para el vector b
for i in range(n):
    b[i] = st.sidebar.number_input(f"b[{i+1}]", value=0.0)

# Mostrar la matriz A y el vector b
st.subheader("Matriz de Coeficientes (A) y Vector de T rminos Independientes (B)")
```

```
st.write("Matriz A:")
st.write(A)
st.write("Vector B:")
st.write(b)

# Resoluci n del sistema
if st.sidebar.button("Resolver"):
    if metodo == "Sustituci n":
        resultado = resolver_sustitucion(A, b)
    elif metodo == "Gauss-Jordan":
        resultado = resolver_gauss_jordan(A, b)
    elif metodo == "Cramer":
        resultado = resolver_cramer(A, b)

    # Mostrar resultados
    st.subheader("Resultados Finales:")
    if isinstance(resultado, str):  # Si el resultado es un mensaje de error
        st.error(resultado)
    else:
        for i, res in enumerate(resultado):
            st.write(f"x{i+1} = {res:.4f}")
```

5. Resultados

El código proporcionado genera una interfaz interactiva en la cual el usuario puede ingresar las matrices de coeficientes y el vector de términos constantes de un sistema de ecuaciones. Dependiendo del método seleccionado (Sustitución, Gauss-Jordan, o Cramer), la aplicación resolverá el sistema y mostrará el resultado correspondiente.

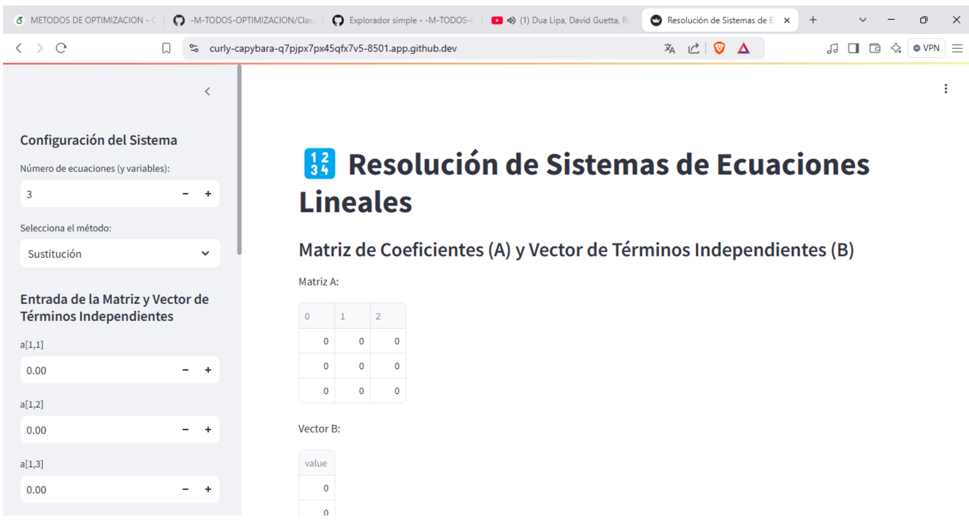

Figura 1: Visualizacion del aplicacion de los tres metodos

Figura 2: Codigo QR del web

6. Conclusión

El desarrollo de la aplicación para la resolución de sistemas de ecuaciones lineales utilizando métodos numéricos como Sustitución, Gauss-Jordan y Cramer ha permitido crear una herramienta interactiva y accesible para estudiantes, ingenieros y profesionales en diversas disciplinas. A través de esta aplicación, los usuarios pueden resolver sistemas de ecuaciones lineales de forma rápida y eficiente, sin necesidad de realizar cálculos manuales complicados, lo cual facilita la comprensión de los conceptos matemáticos detrás de estos métodos.

Cada uno de los métodos implementados tiene características y ventajas que los hacen útiles en diferentes contextos. El Método de Sustitución es adecuado para sistemas pequeños y cuando es posible despejar fácilmente una incógnita en términos de las demás. El Método de Gauss-Jordan, por otro lado, es un método directo que transforma la matriz del sistema en una forma escalonada reducida, permitiendo resolver sistemas de cualquier tamaño. Finalmente, el Método de Cramer es útil cuando el determinante de la matriz de coeficientes es distinto de cero, y es adecuado para sistemas pequeños o medianos.

La aplicación fue diseñada con una interfaz intuitiva utilizando Streamlit, lo que facilita la entrada de datos y la visualización de los resultados. La posibilidad de seleccionar el método de resolución y ver los resultados paso a paso contribuye a una mayor comprensión de los procesos involucrados en la resolución de sistemas lineales.

Además de su aplicabilidad práctica, este proyecto también ha servido para profundizar en los fundamentos teóricos de los métodos numéricos y su implementación en programación, lo que resulta valioso para quienes buscan aprender o reforzar sus conocimientos en esta área. Sin embargo, es importante tener en cuenta que cada método tiene limitaciones, como la necesidad de que el determinante sea distinto de cero en el caso de Cramer o el costo computacional en sistemas más grandes para el método de Gauss-Jordan.

Referencias

[1] Leslie Lamport, *LaTeX: A Document Preparation System*, 2nd Edition, Addison-Wesley, 1994.

[2] Gilbert Strang, *Introduction to Linear Algebra*, 4th Edition, Wellesley-Cambridge Press, 2002.

[3] M.S. Grewal, *Higher Engineering Mathematics*, 40th Edition, Khanna Publishers, 2010.

[4] Lloyd N. Trefethen, *Numerical Linear Algebra*, SIAM, 1997.

[5] Ronald M. Brach, *Introduction to Dynamics and Control*, Springer, 2008.

[6] Wikipedia contributors, *Gauss-Jordan Elimination*, https://en.wikipedia.org/wiki/Gauss%E2%80%93Jordan_elimination, Accessed: January 2025.

[7] F. B. Hildebrand, *Introduction to Numerical Analysis*, 2nd Edition, Dover Publications, 2004.