

Universidad Nacional del Altiplano Puno
Facultad de Ingeniería Estadística e Informática
Escuela profesional de Ingeniería Estadística e
Informática

Optimization Methods

Resolución de Ejercicios de Gauss-Jordan

Estudiante: Eder Elviz Luna Ochochoque

Link GitHub:

<https://acortar.link/xykpH0>

Docente: Dr.Fred Torres Cruz

17 de Enero 2025

Ejercicio 01: Modelo de regresión lineal:

Resolver el siguiente sistema para determinar los valores de los pesos del modelo w_1, w_2, w_3 :

$$2w_1 + 3w_2 - w_3 = 5,$$

$$-w_1 + 2w_2 + 4w_3 = 6,$$

$$3w_1 - w_2 + 2w_3 = 7.$$

Resolución paso a paso:

Hacer 1 el pivote en fila 1 (dividir fila 1 entre 2):

$$\begin{pmatrix} 1 & \frac{3}{2} & -\frac{1}{2} & \frac{5}{2} \\ -1 & 2 & 4 & 6 \\ 3 & -1 & 2 & 7 \end{pmatrix}$$

Reducir fila 2 usando fila 1 (sumar fila 1 a fila 2):

$$\begin{pmatrix} 1 & \frac{3}{2} & -\frac{1}{2} & \frac{5}{2} \\ 0 & \frac{7}{2} & \frac{7}{2} & \frac{17}{2} \\ 3 & -1 & 2 & 7 \end{pmatrix}$$

*Reducir fila 3 usando fila 1 (restar 3 * fila 1 de fila 3):*

$$\begin{pmatrix} 1 & \frac{3}{2} & -\frac{1}{2} & \frac{5}{2} \\ 0 & \frac{7}{2} & \frac{7}{2} & \frac{17}{2} \\ 0 & -\frac{11}{2} & \frac{7}{2} & -\frac{1}{2} \end{pmatrix}$$

Hacer 1 el pivote en fila 2 (dividir fila 2 entre $\frac{7}{2}$):

$$\begin{pmatrix} 1 & \frac{3}{2} & -\frac{1}{2} & \frac{5}{2} \\ 0 & 1 & 1 & \frac{17}{7} \\ 0 & -\frac{11}{2} & \frac{7}{2} & -\frac{1}{2} \end{pmatrix}$$

*Reducir fila 1 usando fila 2 (restar $\frac{3}{2}$ * fila 2 de fila 1):*

$$\begin{pmatrix} 1 & 0 & -2 & -\frac{8}{7} \\ 0 & 1 & 1 & \frac{17}{7} \\ 0 & -\frac{11}{2} & \frac{7}{2} & -\frac{1}{2} \end{pmatrix}$$

*Reducir fila 3 usando fila 2 (sumar $\frac{11}{2}$ * fila 2 de fila 3):*

$$\begin{pmatrix} 1 & 0 & -2 & -\frac{8}{7} \\ 0 & 1 & 1 & \frac{17}{7} \\ 0 & 0 & 9 & \frac{90}{7} \end{pmatrix}$$

Hacer 1 el pivote en fila 3 (dividir fila 3 entre 9):

$$\begin{pmatrix} 1 & 0 & -2 & -\frac{8}{7} \\ 0 & 1 & 1 & \frac{17}{7} \\ 0 & 0 & 1 & \frac{10}{7} \end{pmatrix}$$

Reducir fila 1 usando fila 3 (sumar 2 * fila 3 de fila 1):

$$\begin{pmatrix} 1 & 0 & 0 & \frac{12}{7} \\ 0 & 1 & 1 & \frac{17}{7} \\ 0 & 0 & 1 & \frac{10}{7} \end{pmatrix}$$

Reducir fila 2 usando fila 3 (restar fila 3 de fila 2):

$$\begin{pmatrix} 1 & 0 & 0 & \frac{12}{7} \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & \frac{10}{7} \end{pmatrix}$$

Resultado:

$$w_1 = \frac{12}{7}, \quad w_2 = 1, \quad w_3 = \frac{10}{7}$$

Ejercicio 02: Calibración de hiperparámetros:

Determinar los valores de x , y , z que satisfacen el sistema de ecuaciones:

$$\begin{aligned} 2w_1 + 3w_2 - w_3 &= 5, \\ -w_1 + 2w_2 + 4w_3 &= 6, \\ 3w_1 - w_2 + 2w_3 &= 7. \end{aligned}$$

Resolución paso a paso:

Hacer 1 el pivote en fila 1 (dividir fila 1 entre 2):

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 12 \\ 2 & -1 & 1 & 4 \\ -1 & 2 & -2 & 0 \end{array} \right)$$

Reducir fila 2 usando fila 1 (restar 2 * fila 1 de fila 2):

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 12 \\ 0 & -5 & -5 & -20 \\ -1 & 2 & -2 & 0 \end{array} \right)$$

Reducir fila 3 usando fila 1 (sumar fila 1 a fila 3):

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 12 \\ 0 & -5 & -5 & -20 \\ 0 & 4 & 1 & 12 \end{array} \right)$$

Hacer 1 el pivote en fila 2 (dividir fila 2 entre -5):

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 12 \\ 0 & 1 & 1 & 4 \\ 0 & 4 & 1 & 12 \end{array}\right)$$

Reducir fila 1 usando fila 2 (restar 2 * fila 2 de fila 1):

$$\left(\begin{array}{ccc|c} 1 & 0 & 1 & 4 \\ 0 & 1 & 1 & 4 \\ 0 & 4 & 1 & 12 \end{array}\right)$$

Reducir fila 3 usando fila 2 (restar 4 * fila 2 de fila 3):

$$\left(\begin{array}{ccc|c} 1 & 0 & 1 & 4 \\ 0 & 1 & 1 & 4 \\ 0 & 0 & -3 & -4 \end{array}\right)$$

Hacer 1 el pivote en fila 3 (dividir fila 3 entre -3):

$$\left(\begin{array}{ccc|c} 1 & 0 & 1 & 4 \\ 0 & 1 & 1 & 4 \\ 0 & 0 & 1 & \frac{4}{3} \end{array}\right)$$

Reducir fila 1 usando fila 3 (restar 1 * fila 3 de fila 1):

$$\left(\begin{array}{ccc|c} 1 & 0 & 0 & \frac{8}{3} \\ 0 & 1 & 1 & 4 \\ 0 & 0 & 1 & \frac{4}{3} \end{array}\right)$$

Reducir fila 2 usando fila 3 (restar 1 * fila 3 de fila 2):

$$\left(\begin{array}{ccc|c} 1 & 0 & 0 & \frac{8}{3} \\ 0 & 1 & 0 & \frac{8}{3} \\ 0 & 0 & 1 & \frac{4}{3} \end{array}\right)$$

Resultado:

$$x = \frac{8}{3}, \quad y = \frac{8}{3}, \quad z = \frac{4}{3}.$$

Ejercicio 03: Asignacion optima de recursos:

Resolver para a, b, c, que representan las proporciones de recursos asignados a cada m'odulo, el sistema:

$$\begin{aligned} a + b + c &= 6, \\ 2a - b + 3c &= 13, \\ -a + 2b - c &= 2. \end{aligned}$$

Resolución paso a paso:

Hacer 1 el pivote en fila 1 (dividir fila 1 entre 1):

$$\begin{pmatrix} 1 & 1 & 1 & 6 \\ 2 & -1 & 3 & 13 \\ -1 & 2 & -1 & 2 \end{pmatrix}$$

*Reducir fila 2 usando fila 1 (restar 2 * fila 1 de fila 2):*

$$\begin{pmatrix} 1 & 1 & 1 & 6 \\ 0 & -3 & 1 & 1 \\ -1 & 2 & -1 & 2 \end{pmatrix}$$

Reducir fila 3 usando fila 1 (sumar fila 1 a fila 3):

$$\begin{pmatrix} 1 & 1 & 1 & 6 \\ 0 & -3 & 1 & 1 \\ 0 & 3 & 0 & 8 \end{pmatrix}$$

Hacer 1 el pivote en fila 2 (dividir fila 2 entre -3):

$$\begin{pmatrix} 1 & 1 & 1 & 6 \\ 0 & 1 & -\frac{1}{3} & -\frac{1}{3} \\ 0 & 3 & 0 & 8 \end{pmatrix}$$

Reducir fila 1 usando fila 2 (restar fila 2 de fila 1):

$$\begin{pmatrix} 1 & 0 & \frac{4}{3} & \frac{19}{3} \\ 0 & 1 & -\frac{1}{3} & -\frac{1}{3} \\ 0 & 3 & 0 & 8 \end{pmatrix}$$

*Reducir fila 3 usando fila 2 (restar 3 * fila 2 de fila 3):*

$$\begin{pmatrix} 1 & 0 & \frac{4}{3} & \frac{19}{3} \\ 0 & 1 & -\frac{1}{3} & -\frac{1}{3} \\ 0 & 0 & 1 & 9 \end{pmatrix}$$

Hacer 1 el pivote en fila 3 (dividir fila 3 entre 1):

$$\begin{pmatrix} 1 & 0 & \frac{4}{3} & \frac{19}{3} \\ 0 & 1 & -\frac{1}{3} & -\frac{1}{3} \\ 0 & 0 & 1 & 9 \end{pmatrix}$$

*Reducir fila 1 usando fila 3 (restar $\frac{4}{3}$ * fila 3 de fila 1):*

$$\begin{pmatrix} 1 & 0 & 0 & -\frac{17}{3} \\ 0 & 1 & -\frac{1}{3} & -\frac{1}{3} \\ 0 & 0 & 1 & 9 \end{pmatrix}$$

Reducir fila 2 usando fila 3 (sumar $\frac{1}{3}$ * fila 3 a fila 2):

$$\begin{pmatrix} 1 & 0 & 0 & -\frac{17}{3} \\ 0 & 1 & 0 & \frac{8}{3} \\ 0 & 0 & 1 & 9 \end{pmatrix}$$

Resultado:

$$a = -\frac{17}{3}, \quad b = \frac{8}{3}, \quad c = 9$$

Ejercicio 04: Optimizacion de parametros de un Bosque Aleatorio:

Resolver el siguiente sistema para los hiperparametros p, q, r:

$$p + 2q + 3r = 10,$$

$$2p - q + 4r = 12,$$

$$3p + 3q - r = 6.$$

Resolución paso a paso:

Hacer 1 el pivote en fila 1 (dividir fila 1 entre 1):

$$\begin{pmatrix} 1 & 2 & 3 & 10 \\ 2 & -1 & 4 & 12 \\ 3 & 3 & -1 & 6 \end{pmatrix}$$

Reducir fila 2 usando fila 1 (restar $2 \cdot$ fila 1 de fila 2):

$$\begin{pmatrix} 1 & 2 & 3 & 10 \\ 0 & -5 & -2 & -8 \\ 3 & 3 & -1 & 6 \end{pmatrix}$$

Reducir fila 3 usando fila 1 (restar $3 \cdot$ fila 1 de fila 3):

$$\begin{pmatrix} 1 & 2 & 3 & 10 \\ 0 & -5 & -2 & -8 \\ 0 & -3 & -10 & -24 \end{pmatrix}$$

Hacer 1 el pivote en fila 2 (dividir fila 2 entre -5):

$$\begin{pmatrix} 1 & 2 & 3 & 10 \\ 0 & 1 & \frac{2}{5} & \frac{8}{5} \\ 0 & -3 & -10 & -24 \end{pmatrix}$$

Reducir fila 1 usando fila 2 (restar $2 \cdot$ fila 2 de fila 1):

$$\begin{pmatrix} 1 & 0 & \frac{11}{5} & \frac{34}{5} \\ 0 & 1 & \frac{2}{5} & \frac{8}{5} \\ 0 & -3 & -10 & -24 \end{pmatrix}$$

Reducir fila 3 usando fila 2 (sumar $3 \cdot$ fila 2 a fila 3):

$$\begin{pmatrix} 1 & 0 & \frac{11}{5} & \frac{34}{5} \\ 0 & 1 & \frac{2}{5} & \frac{8}{5} \\ 0 & 0 & -\frac{44}{5} & -\frac{96}{5} \end{pmatrix}$$

Hacer 1 el pivote en fila 3 (dividir fila 3 entre $-\frac{44}{5}$):

$$\begin{pmatrix} 1 & 0 & \frac{11}{5} & \frac{34}{5} \\ 0 & 1 & \frac{2}{5} & \frac{8}{5} \\ 0 & 0 & 1 & \frac{24}{11} \end{pmatrix}$$

Reducir fila 1 usando fila 3 (restar $\frac{11}{5} \cdot$ fila 3 de fila 1):

$$\begin{pmatrix} 1 & 0 & 0 & 2 \\ 0 & 1 & \frac{2}{5} & \frac{8}{5} \\ 0 & 0 & 1 & \frac{24}{11} \end{pmatrix}$$

Reducir fila 2 usando fila 3 (restar $\frac{2}{5} \cdot$ fila 3 de fila 2):

$$\begin{pmatrix} 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & \frac{8}{11} \\ 0 & 0 & 1 & \frac{24}{11} \end{pmatrix}$$

Resultado final:

$$p = 2, \quad q = \frac{8}{11}, \quad r = \frac{24}{11}.$$

Ejercicio 05: Estimaci3n de demanda de inventario:

Resolver para u , v , w , que representan los coeficientes del modelo de series de tiempo:

$$\begin{aligned} u + v + 2w &= 9, \\ 2u - 3v + 4w &= 5, \\ u - 2v + w &= 1. \end{aligned}$$

Resoluci3n paso a paso:

Hacer 1 el pivote en fila 1 (dividir fila 1 entre 1):

$$\begin{pmatrix} 1 & 1 & 2 & 9 \\ 2 & -3 & 4 & 5 \\ 1 & -2 & 1 & 1 \end{pmatrix}$$

Reducir fila 2 usando fila 1 (restar $2 \cdot$ fila 1 de fila 2):

$$\begin{pmatrix} 1 & 1 & 2 & 9 \\ 0 & -5 & 0 & -13 \\ 1 & -2 & 1 & 1 \end{pmatrix}$$

Reducir fila 3 usando fila 1 (restar $1 \cdot$ fila 1 de fila 3):

$$\begin{pmatrix} 1 & 1 & 2 & 9 \\ 0 & -5 & 0 & -13 \\ 0 & -3 & -1 & -8 \end{pmatrix}$$

Hacer 1 el pivote en fila 2 (dividir fila 2 entre -5):

$$\begin{pmatrix} 1 & 1 & 2 & 9 \\ 0 & 1 & 0 & \frac{13}{5} \\ 0 & -3 & -1 & -8 \end{pmatrix}$$

Reducir fila 1 usando fila 2 (restar $1 \cdot$ fila 2 de fila 1):

$$\begin{pmatrix} 1 & 0 & 2 & \frac{32}{5} \\ 0 & 1 & 0 & \frac{13}{5} \\ 0 & -3 & -1 & -8 \end{pmatrix}$$

Reducir fila 3 usando fila 2 (sumar $3 \cdot$ fila 2 a fila 3):

$$\begin{pmatrix} 1 & 0 & 2 & \frac{32}{5} \\ 0 & 1 & 0 & \frac{13}{5} \\ 0 & 0 & -1 & -\frac{1}{5} \end{pmatrix}$$

Hacer 1 el pivote en fila 3 (dividir fila 3 entre -1):

$$\begin{pmatrix} 1 & 0 & 2 & \frac{32}{5} \\ 0 & 1 & 0 & \frac{13}{5} \\ 0 & 0 & 1 & \frac{1}{5} \end{pmatrix}$$

Reducir fila 1 usando fila 3 (restar $2 \cdot$ fila 3 de fila 1):

$$\begin{pmatrix} 1 & 0 & 0 & 6 \\ 0 & 1 & 0 & \frac{13}{5} \\ 0 & 0 & 1 & \frac{1}{5} \end{pmatrix}$$

Reducir fila 2 usando fila 3 (restar $0 \cdot$ fila 3 de fila 2):

$$\begin{pmatrix} 1 & 0 & 0 & 6 \\ 0 & 1 & 0 & \frac{13}{5} \\ 0 & 0 & 1 & \frac{1}{5} \end{pmatrix}$$

Resultado final:

$$u = 6, \quad v = \frac{13}{5}, \quad w = \frac{1}{5}.$$

Codigo

```
import customtkinter as ctk
import numpy as np
from fractions import Fraction

def gauss_jordan_with_steps(matrix):
    """Resuelve un sistema de ecuaciones lineales utilizando el
        m todo de Gauss-Jordan con pasos en fracciones."""
    # Convertir la matriz a fracciones
    matrix = np.array([[Fraction(element).limit_denominator() for
        element in row] for row in matrix])
    rows, cols = matrix.shape
    steps = [] # Lista para almacenar los pasos intermedios

    def format_matrix(mat):
        """Formatea una matriz para mostrarla en pasos intermedios.
            """
        return "\n".join(["\t".join([str(element) for element in
            row]) for row in mat])

    # M todo de Gauss-Jordan
    for i in range(rows):
        # Hacer el pivote igual a 1
        pivot = matrix[i, i]
        matrix[i] = [element / pivot for element in matrix[i]]
        steps.append(f"Hacer el pivote en fila {i+1} (dividir
            fila {i+1} entre {pivot}): \n{format_matrix(matrix)}")

        for j in range(rows):
            if i != j:
                factor = matrix[j, i]
                # Resto de la fila j por el factor y la fila i
                matrix[j] = [matrix[j][k] - factor * matrix[i][k]
                    for k in range(cols)]
                steps.append(f"Reducir fila {j+1} usando fila {i+1}
                    (restar {factor} * fila {i+1} de fila {j+1}): \n
                    {format_matrix(matrix)}")

    solutions = [matrix[i, -1] for i in range(rows)] # ltima
        columna contiene las soluciones
    return solutions, steps

def update_matrix_inputs():
    """Actualiza la interfaz de entrada de la matriz seg n las
        dimensiones ingresadas."""
    try:
        rows = int(rows_entry.get())
        cols = int(cols_entry.get())
        if rows <= 0 or cols <= 0:
            raise ValueError("Las dimensiones deben ser mayores a
                0.")
    except ValueError as e:
        result_label.configure(text=f"Error: {e}")
        return
```

```

# Limpiar los cuadros de entrada existentes
for widget in matrix_frame.winfo_children():
    widget.destroy()

global matrix_entries
matrix_entries = []

# Crear cuadros de entrada dinamicos
for i in range(rows):
    row_entries = []
    for j in range(cols): # Exactamente 'cols' columnas
        entry = ctk.CTkEntry(matrix_frame, width=50, justify="center")
        entry.grid(row=i, column=j, padx=2, pady=2)
        row_entries.append(entry)
    matrix_entries.append(row_entries)

def solve_system():
    """Obtiene los valores de la matriz, resuelve el sistema y muestra el resultado."""
    try:
        rows = len(matrix_entries)
        cols = len(matrix_entries[0])

        # Construir la matriz aumentada desde los cuadros de entrada
        augmented_matrix = np.array(
            [[float(matrix_entries[i][j].get()) for j in range(cols)] for i in range(rows)]
        )

        # Resolver el sistema con pasos
        solutions, steps = gauss_jordan_with_steps(augmented_matrix)

        # Formatear las soluciones como fracciones
        formatted_solutions = [f"x{i+1}={solutions[i]}" for i in range(len(solutions))]
        solution_text = "Soluciones:\n" + "\n".join(formatted_solutions)

        # Formatear los pasos intermedios
        steps_text = "\n\nResolución paso a paso:\n"
        for step in steps:
            steps_text += f"{step}\n\n"

        # Actualizar el resultado
        result_label.configure(text=solution_text)
        steps_textbox.delete(1.0, "end")
        steps_textbox.insert("1.0", steps_text)
    except ValueError:
        result_label.configure(text="Error: Asegurate de ingresar valores válidos.")
    except Exception as e:
        result_label.configure(text=f"Error: {str(e)}")

```

```

# Configuraci3n de la ventana principal
app = ctk.CTk()
app.title("Calculadora Gauss-Jordan")
app.geometry("1000x700")

# T tulo
title_label = ctk.CTkLabel(app, text="Calculadora M todo Gauss-
Jordan", font=("Arial", 20, "bold"))
title_label.pack(pady=10)

# Dimensiones de la matriz
dims_frame = ctk.CTkFrame(app)
dims_frame.pack(pady=10)

rows_label = ctk.CTkLabel(dims_frame, text="Filas:")
rows_label.grid(row=0, column=0, padx=5)
rows_entry = ctk.CTkEntry(dims_frame, width=50, justify="center")
rows_entry.insert(0, "2") # Valor por defecto
rows_entry.grid(row=0, column=1, padx=5)

cols_label = ctk.CTkLabel(dims_frame, text="Columnas:")
cols_label.grid(row=0, column=2, padx=5)
cols_entry = ctk.CTkEntry(dims_frame, width=50, justify="center")
cols_entry.insert(0, "3") # Valor por defecto
cols_entry.grid(row=0, column=3, padx=5)

update_button = ctk.CTkButton(dims_frame, text="Actualizar matriz",
command=update_matrix_inputs)
update_button.grid(row=0, column=4, padx=10)

# Entrada de la matriz
matrix_frame = ctk.CTkFrame(app)
matrix_frame.pack(pady=10)

# Bot3n para resolver
solve_button = ctk.CTkButton(app, text="Resolver", command=
solve_system)
solve_button.pack(pady=20)

# Etiqueta para mostrar el resultado
result_label = ctk.CTkLabel(app, text="", font=("Arial", 14),
justify="left", wraplength=900)
result_label.pack(pady=10)

# Cuadro de texto con deslizador para los pasos intermedios
steps_textbox_frame = ctk.CTkFrame(app)
steps_textbox_frame.pack(pady=10)

steps_textbox = ctk.CTkTextbox(steps_textbox_frame, width=900,
height=300, wrap="word")
steps_textbox.grid(row=0, column=0, padx=5, pady=5)

steps_scrollbar = ctk.CTkScrollbar(steps_textbox_frame, command=
steps_textbox.yview)
steps_scrollbar.grid(row=0, column=1, sticky="ns", pady=5)

```

```
steps_textbox.configure(yscrollcommand=steps_scrollbar.set)

# Generar entradas iniciales
update_matrix_inputs()

# Iniciar la aplicaci n
app.mainloop()
```