|  |  |
| --- | --- |
| **Nombre:** | Luis de la Garza González |
| **Matrícula:** | al03101869 |
| **Nombre del curso:** | Machine Learning. |
| **Trabajo:** | Actividad 01.- Las matemáticas para la resolución de problemas |
| **Nombre del profesor:** | Mtro. |
| **Fecha:** | 14.09.2025 |

**Contenido**

[**Instrucciones** 3](#_Toc206421989)

[**Objetivo** 4](#_Toc206421990)

[**Desarrollo** 4](#_Toc206421991)

[Problema ejemplo 4](#_Toc206421992)

[Habilidades por alcanzar 5](#_Toc206421993)

[Ejercicio 1 5](#_Toc206421994)

[Ejercicio 2 7](#_Toc206421995)

[Ejercicio 3 8](#_Toc206421996)

[**Conclusiones** 11](#_Toc206421997)

[**Liga al código** 12](#_Toc206421998)

# **Instrucciones**

1. **Identifica una problemática que pueda solucionarse mediante algún procedimiento matemático y responde a las siguientes preguntas:**

* ¿Qué tipo de pensamiento matemático está involucrado en la solución del problema?
* ¿Qué procesos matemáticos se deben aplicar para encontrar dicha solución?
* ¿Cuáles son los pasos que se deben realizar para solucionar dicho problema?

Elabora un reporte donde argumentes tus respuestas anteriores.

1. **Elabora una hoja de ruta o plan de estudios donde representes:**

* Las habilidades y conocimientos matemáticos que ya posees.
* Los que consideras que debes alcanzar para dominar las bases necesarias que te permitirán comprender mejor los fundamentos del aprendizaje automático.

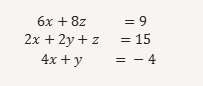
**Actividades de Álgebra Lineal en Python (Jupyter Notebook):**

1. **Utiliza un cuaderno de Jupyter Notebook y el lenguaje de programación Python para realizar las siguientes operaciones:**

* Genera una matriz cuadrada de 4x4.
* Calcula su inversa y su traspuesta.
* Multiplica la matriz por un escalar.
* Realiza el producto interno con una nueva matriz de dimensiones 4x2.

1. **Investiga otras formas computacionales de solucionar sistemas de ecuaciones lineales Elabora una tabla resumen donde presentes tus conclusiones.**
2. **Genera un nuevo cuaderno de Jupyter Notebook y resuelve los siguientes sistemas de ecuaciones lineales, encontrando los valores de las variables utilizando diferentes métodos:**

**a:**

**b**:Texto

El contenido generado por IA puede ser incorrecto.

# **Objetivo**

Identificar un problema de optimización que pueda solucionarse mediante algún procedimiento matemático de álgebra lineal, describir mis habilidades y conocimientos matemáticos actuales y los que debo alcanzar para comprender mejor los fundamentos del aprendizaje automático. Practicar la resolución de sistemas de ecuaciones lineales en una notebook de Jupyter utilizando Python.

# **Desarrollo**

# 1.- Problema ejemplo

Una fábrica produce camisetas de diferentes tallas y colores. Cada tipo de camiseta requiere distintos insumos (tela, tinta, tiempo de máquina) y genera diferentes márgenes de ganancia. La empresa desea maximizar sus ganancias sin exceder los recursos disponibles.

Las variables involucradas son:

* número de camisetas tipo A
* número de camisetas tipo B
* número de camisetas tipo C

Las restricciones son:

* Disponibilidad de tela (en metros cuadrados)
* Horas de máquina disponibles por día
* Cantidad de tinta por color
* Capacidad máxima de producción diaria

Para este tipo de problemas de optimización, utilizaríamos el método simplex por lo que estaríamos haciendo uso principalmente del **algebra lineal** y la **Investigación de operaciones**.

# 2.- Habilidades por alcanzar

Aunque tengo el antecedente de mis estudios en la facultad de ingeniería, tengo que volver a desarrollar la habilidad en las siguientes áreas de las matemáticas.

|  |  |
| --- | --- |
| Área | Objetivo |
| Álgebra lineal | Comprender operaciones con matrices, vectores, y espacios vectoriales |
| Cálculo | Entender derivadas parciales, gradientes y optimización |
| Probabilidad | Manejar distribuciones, inferencia y funciones de densidad |
| Estadística | Aplicar métricas de evaluación y análisis de errores |
| Lógica matemática | Formalizar problemas y estructuras de decisión |

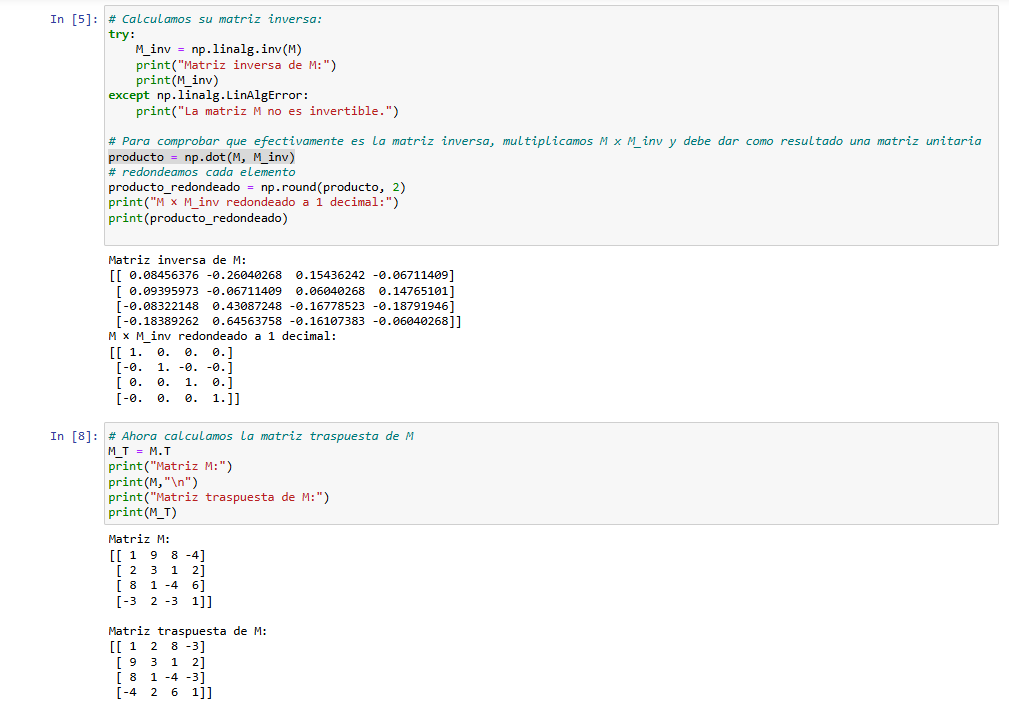
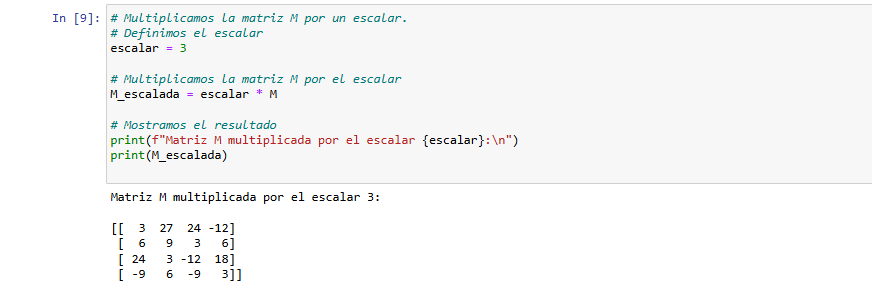
# 3.- Ejercicio 1

**Utiliza un cuaderno de Jupyter Notebook y el lenguaje de programación Python para realizar las siguientes operaciones:**

* Genera una matriz cuadrada de 4x4.
* Calcula su inversa y su traspuesta.
* Multiplica la matriz por un escalar.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.



# 4.- Ejercicio 2

**Investiga otras formas computacionales de solucionar sistemas de ecuaciones lineales Elabora una tabla resumen donde presentes tus conclusiones.**

**Tabla resumen:** **Métodos computacionales para sistemas de ecuaciones lineales**

| Método | Tipo | Características principales | Ventajas | Desventajas |
| --- | --- | --- | --- | --- |
| Eliminación Gaussiana | Directo | Reduce la matriz a forma escalonada para resolver por sustitución regresiva | Preciso, aplicable a cualquier sistema consistente | Sensible a errores numéricos en matrices mal condicionadas |
| Eliminación Gauss-Jordan | Directo | Lleva la matriz a forma escalonada reducida (identidad) para obtener solución directa | Da solución única directamente | Más costoso computacionalmente que Gauss |
| Regla de Cramer | Directo | Usa determinantes para resolver sistemas con solución única | Simple para sistemas pequeños (n ≤ 3) | Ineficiente para sistemas grandes; requiere determinantes |
| Descomposición LU | Directo | Factoriza la matriz en producto de matrices triangular inferior y superior | Eficiente para resolver múltiples sistemas con misma matriz | Requiere que la matriz sea factorizable |
| Método de Jacobi | Iterativo | Aproxima soluciones sucesivas usando valores anteriores | Fácil de implementar, útil en sistemas dispersos | Convergencia lenta; requiere condiciones como diagonal dominante |
| Método de Gauss-Seidel | Iterativo | Similar a Jacobi pero usa valores actualizados en cada paso | Más rápido que Jacobi; útil en grandes sistemas | No siempre converge; depende de propiedades de la matriz |
| Gradiente Conjugado | Iterativo | Optimiza la solución para sistemas simétricos y definidos positivos | Muy eficiente en sistemas grandes y dispersos | Solo aplicable a matrices simétricas y positivas definidas |

**Conclusiones sobre métodos computacionales para sistemas de ecuaciones lineales:**

Los métodos directos, como: Eliminación Gaussiana, Eliminación Gauss-Jordan, Regla de Cramer y

Descomposición LU; son ideales para sistemas pequeños o cuando se requiere precisión exacta.

Los métodos iterativos, como: Método de Jacobi, Método de Gauss-Seidel y Gradiente Conjugado; son preferibles en sistemas grandes, dispersos o cuando se busca eficiencia computacional.

La elección depende de la estructura de la matriz, el tamaño del sistema y los recursos computacionales disponibles.

# 5.- Ejercicio 3

**Genera un nuevo cuaderno de Jupyter Notebook y resuelve los siguientes sistemas de ecuaciones lineales, encontrando los valores de las variables utilizando diferentes métodos:**

**a:**

Texto

El contenido generado por IA puede ser incorrecto.**b**:Texto

El contenido generado por IA puede ser incorrecto.

**3.a**

**El sistema de ecuaciones lo podemos representar como MxV =I**

**Entonces M = V-1x I**

**En donde M = [**

**[1,2,5,-4],**

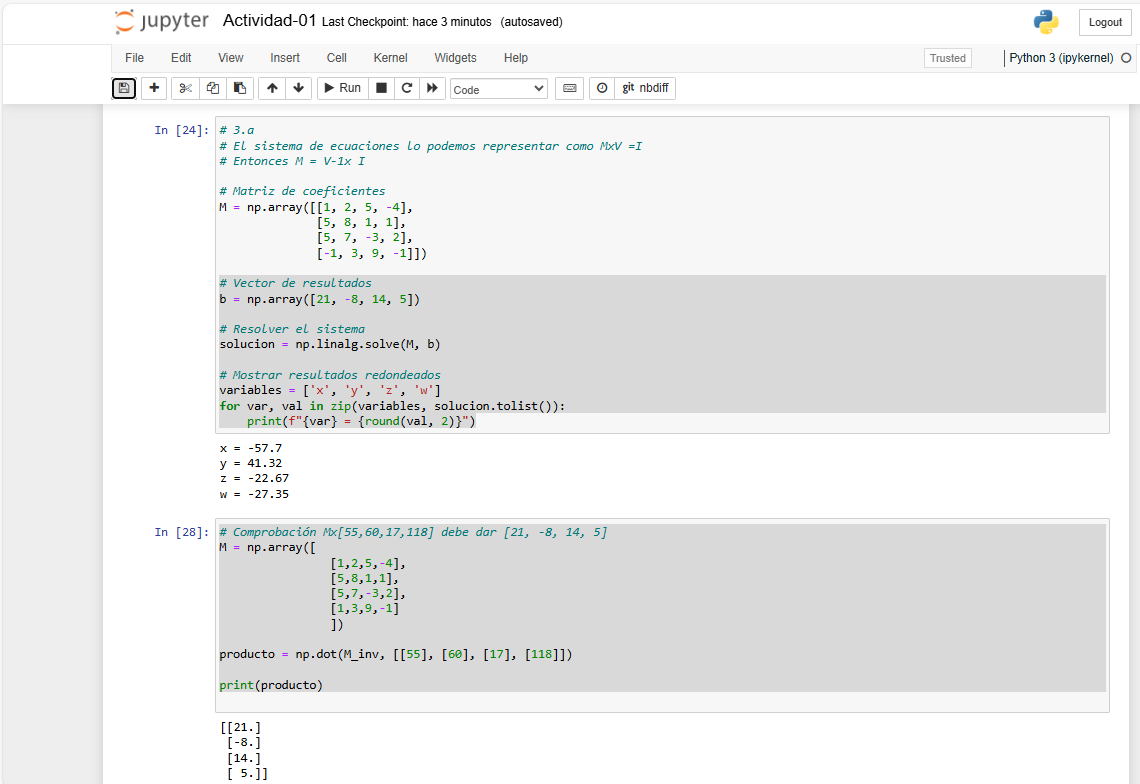
**[5,8,1,1],**

**[5,7,-3,2],**

**[-1,3,9,-1]**

**]**

**La siguiente imagen muestra la codificación y los resultados:**



**3.b**

**Aplicamos la misma metodología para encontrar los valores de x, y, z.**

**Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.**

# **Conclusiones**

Las matemáticas son un metalenguaje con un profundo trasfondo filosófico que mantiene la indispensabilidad de las matemáticas como una herramienta esencial para el progreso científico. Aunque este vínculo es menos obvio con el campo más técnico de la lógica matemática, también resulta estar arraigado en la lógica general y, por lo tanto, en la estructura del lenguaje. El pensamiento matemático nos ayuda a modelar y resolver problemas del mundo real utilizando modelos numéricos, geométricos y probabilísticos.

Resolver problemas matemáticos implica un proceso estructurado: comprensión, planificación, cálculo e interpretación.

Las matemáticas computacionales han cambiado drásticamente desde el final de la Segunda Guerra Mundial, y este cambio también está haciendo posible el desarrollo de la inteligencia artificial. El entendimiento del álgebra lineal, el cálculo y la probabilidad es necesario para entender las bases del aprendizaje automático o Machine Learning.

Lo que aprendimos en la segunda mitad de esta lección:

Álgebra lineal y sus fundamentos: Para desarrollar modelos matemáticos, así como aplicaciones de inteligencia artificial, la comprensión del álgebra lineal es un requisito básico. Esa misma definición es la que nos permite usar vectores en diferentes contextos.

Operaciones de matrices en Python: Por ejemplo, la inversa o la transpuesta, lo que facilita la resolución de problemas computacionales complejos. Es un trasfondo teórico básico para estudiar las siguientes estructuras algebraicas más avanzadas: grupos, espacios, subespacios vectoriales.

Software matemático: Las bibliotecas de software matemático permiten concentrarse en la lógica y el razonamiento sobre un cálculo, y no en su instanciación mecánica.

Conexión teoría-práctica: La conexión de la teoría con la práctica computacional acelera el aprendizaje y resuelve mejor los problemas reales.

Los ejercicios prácticos de Jupyter Notebook en Python fueron esenciales para comprender y aplicar conceptos matemáticos a la resolución de problemas. En este trabajo, generamos la matriz cuadrada 4x4 mediante este método y encontramos su inversa y la transposición de la matriz, lo que ayuda en la visualización y manipulación de estructuras algebraicas. Multiplicar la matriz por un escalar y realizar un producto interno con otra matriz 4×2 ha ayudado a proporcionar una visión de las operaciones básicas de matrices.

En general, el trabajo práctico ha proporcionado esencialmente un medio para aplicar y reforzar la comprensión teórica para ayudar a resolver problemas matemáticos, así como seguir desarrollando el razonamiento analítico y lógico.

# **Liga al código**

La codificación (actividad-01.ipynb) la podemos encontrar en siguiente liga a Github:

<https://github.com/luisgg121/ML-Actividad-01.git>