|  |  |
| --- | --- |
| **Nombre:** | Luis de la Garza González |
| **Matrícula:** | al03101869 |
| **Nombre del curso:** | Machine Learning. |
| **Trabajo:** | Actividad 004.- Introducción al aprendizaje automático supervisado. |
| **Nombre del profesor:** | Mtro. |
| **Fecha:** | 12 de octubre de 2025 |

**Contenido**

[**Objetivo** 3](#_Toc208072148)

[**Instrucciones:** 3](#_Toc208072149)

[**Desarrollo** 4](#_Toc208072150)

[1. Generación de conjuntos de datos experimentales y herramientas de Python. 4](#_Toc208072151)

[a. Conjunto de datos con tres características linealmente relacionadas 5](#_Toc208072152)

[b. Conjunto de datos con cuatro características, relacionadas de forma no lineal 7](#_Toc208072153)

[3. Conjunto de datos que puedan expresar una relación lineal prevista 9](#_Toc208072154)

[2. 11](#_Toc208072155)

[3. análisis de discriminarte lineal Python 12](#_Toc208072156)

[4. métricas del error cuadrático medio y del coeficiente de determinación 12](#_Toc208072157)

[**Conclusiones sobre el aprendizaje automático** 12](#_Toc208072158)

[**Liga al código en Github** 13](#_Toc208072159)

# **Objetivo**

Resolver problemas de aprendizaje automático supervisado utilizando el lenguaje de programación Python.

# **Instrucciones:**

1. Una de las habilidades de un experto en inteligencia artificial y aprendizaje automático es tener la capacidad de generar conjuntos de datos experimentales para poner a prueba los diferentes modelos que va construyendo durante el desarrollo de su carrera. Investiga cuáles de las diversas herramientas de Python incluyen la funcionalidad de realizar esta acción y construye artificialmente los siguientes conjuntos de datos:
2. Conjunto de datos de tres características linealmente relacionados.
3. Conjunto de datos de cuatro características, relacionados de forma completa, no lineal.
4. Conjunto de datos que, de manera prevista, puedan expresar una relación lineal.
5. Investiga la manera en que se realiza el análisis de discriminarte lineal Python y utiliza este conocimiento para llevar a cabo la reducción de la dimensión y la representación gráfica del primer conjunto de datos construido en la pregunta anterior.
6. Utilizando la librería Scikit-learn, investiga cómo se implementan las métricas del error cuadrático medio y del coeficiente de determinación para evaluar la calidad de un modelo de aprendizaje supervisado.

# **Desarrollo**

# Generación de conjuntos de datos experimentales y herramientas de Python.

Una de las habilidades de un experto en inteligencia artificial y aprendizaje automático es tener la capacidad de generar conjuntos de datos experimentales para poner a prueba los diferentes modelos que va construyendo durante el desarrollo de su carrera. Investiga cuáles de las diversas herramientas de Python incluyen la funcionalidad de realizar esta acción y construye artificialmente los siguientes conjuntos de datos:

* + Conjunto de datos de tres características linealmente relacionados.
  + Conjunto de datos de cuatro características, relacionados de forma completa, no lineal.
  + Conjunto de datos que, de manera prevista, puedan expresar una relación lineal.

¡Perfecto, Luis! Justo en tu línea de dominio técnico y estructuración científica, aquí tienes una guía clara y funcional para generar conjuntos de datos sintéticos en Python, usando herramientas especializadas que permiten experimentar con modelos de aprendizaje automático.

**Herramientas de Python para generar datos sintéticos**

Las siguientes bibliotecas son ampliamente utilizadas por expertos en IA para crear conjuntos de datos artificiales:

|  |  |
| --- | --- |
| **Herramienta** | **Descripción** |
| scikit-learn | Incluye funciones como `make\_regression`, `make\_classification`, `make\_blobs`. |
| Faker | Genera datos realistas (nombres, direcciones, etc.) para simulaciones. |
| numpy + matplotlib | Útiles para crear relaciones matemáticas personalizadas y visualizarlas. |
| CTGAN / SDV | Modelos generativos para datos tabulares complejos. |

En nuestro caso, utilizaremos principalmente scikit-learn y numpy.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.**a. Conjunto de datos con tres características linealmente relacionadas**

El primer conjunto de datos se genera con **make\_regression** de **scikit-learn**, siguiendo los pasos

La función `make\_regression` crea datos sintéticos donde las características (variables independientes) están relacionadas linealmente con una variable objetivo (target). Es útil para validar que un modelo de regresión pueda aprender correctamente una relación lineal.

**Parámetros usados**

X, y = make\_regression(n\_samples=100, n\_features=3, noise=5.0, random\_state=42)

- n\_samples=100: Se generan 100 observaciones (filas).

- n\_features=3: Cada observación tiene 3 características (columnas).

- noise=5.0: Se añade ruido gaussiano para simular variabilidad real.

- random\_state=42: Fija la semilla para reproducibilidad.

**Esto genera:**

X: Matriz de características (100 filas × 3 columnas).

y: Vector de valores objetivo (100 valores).

Convertimos X en un DataFrame con nombres de columnas descriptivos:

df = pd.DataFrame(X, columns=["Feature 1", "Feature 2", "Feature 3"])

Y se añade la columna “Target” con los valores de y:

df["Target"] = y

Mostramos las primeras 5 filas del DataFrame para una inspección rápida:

print(df.head())

**Estructura del conjunto**

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature\_1** | **Feature\_2** | **Feature\_3** | **Target** |
| -0.792521 | 0.504987 | -0.114736 | 13.149332 |
| 0.280992 | -0.208122 | -0.622700 | -20.311291 |
| 0.791032 | 1.402794 | -0.909387 | 101.284137 |

* Las columnas Feature\_1, Feature\_2, Feature\_3 son variables independientes.
* La columna Target es la variable dependiente, calculada como una combinación lineal de las tres características más el ruido.

Matemáticamente, se puede expresar como:

Donde son coeficientes generados aleatoriamente, y es el ruido.

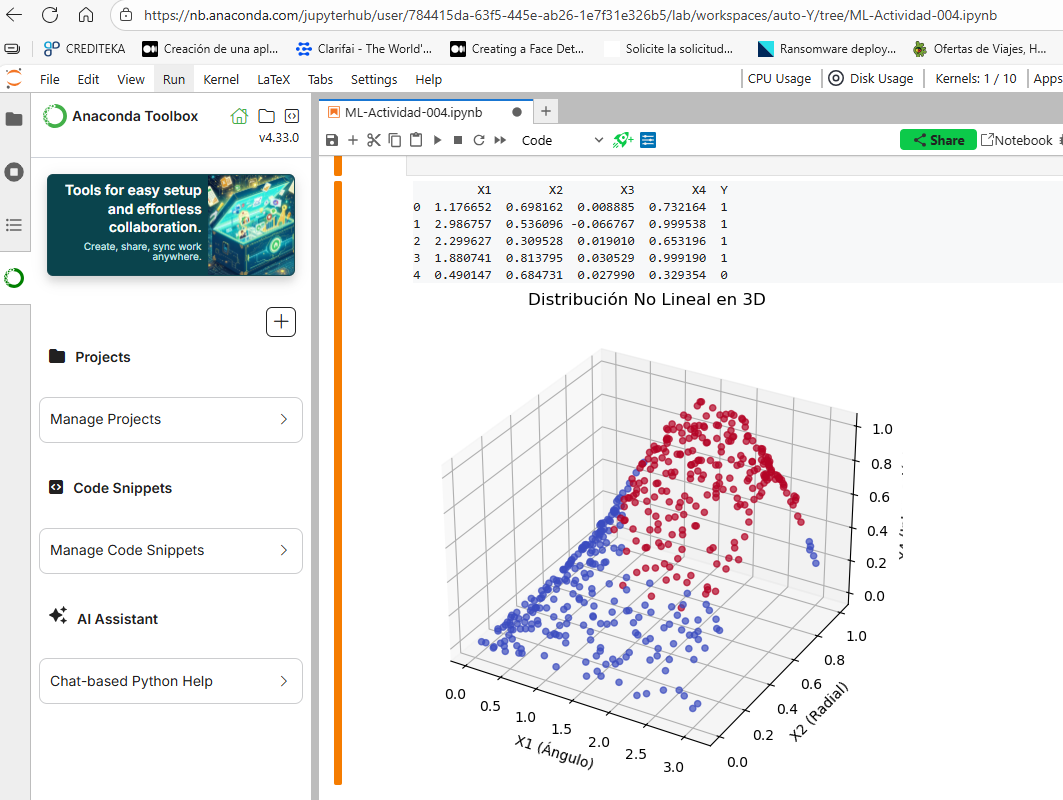
**Este tipo de conjunto es ideal para:**

* Verificar que un modelo de regresión lineal (como `LinearRegression`) puede recuperar los coeficientes.
* Evaluar el impacto del ruido en la predicción.
* Probar visualizaciones de correlación y ajuste.

**b. Conjunto de datos con cuatro características, relacionadas de forma no lineal**

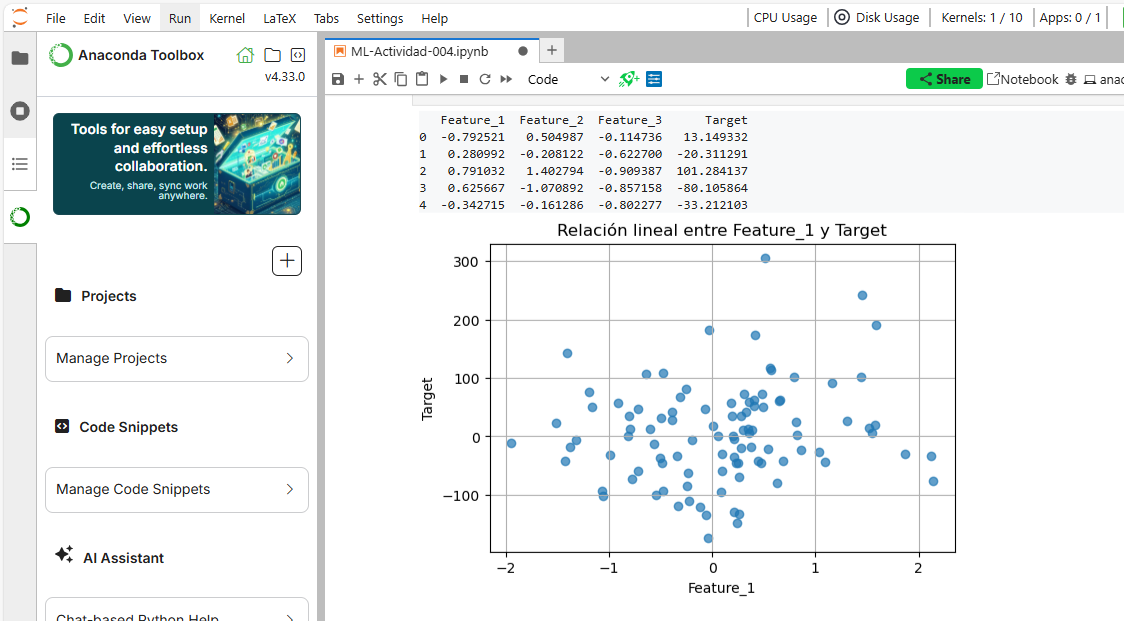
Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.Generación de un conjunto de datos sintético con cuatro características relacionadas de forma no lineal, ideal para pruebas con modelos de clasificación:

Ejecutamos el código y obtenemos la siguiente salida:

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.**3. Conjunto de datos que puedan expresar una relación lineal prevista**

Salida:

1. Análisis de discriminante lineal Python y reducción de la dimensión.

Investiga la manera en que se realiza el análisis de discriminante lineal Python y utiliza este conocimiento para llevar a cabo la reducción de la dimensión y la representación gráfica del primer conjunto de datos construido en la pregunta anterior.

# Métricas del error cuadrático medio y del coeficiente de determinación

Utilizando la librería Scikit-learn, investiga cómo se implementan las métricas del error cuadrático medio y del coeficiente de determinación para evaluar la calidad de un modelo de aprendizaje supervisado.

# **Conclusiones sobre el aprendizaje automático**

El aprendizaje automático incluye varias estrategias, puede ser: supervisado, no supervisado, de refuerzo y aprendizaje profundo; de acuerdo con los tipos particulares de datos (estructurados y no estructurados) así como los objetivos de análisis.

Es importante enfatizar la importancia del tiempo en la selección de modelos, especialmente en problemas donde los datos evolucionan con el tiempo y esa evolución afecta la predicción o clasificación.

Importancia de la preparación de los datos a utilizar en el aprendizaje automático: El uso de todo tipo de herramientas durante la preparación de eventos, adaptadas a las condiciones locales, permite lograr una mayor calidad de entrenamiento. Esto puede implicar limpieza de datos, transformación de datos y la selección de características apropiadas.

Por otro lado, el entendimiento de la naturaleza de los datos (lineal o no lineal) y su dimensionalidad, nos permite elegir los modelos más apropiados y prevenir errores de interpretación.

La elección del modelo a utilizar para la predicción debe basarse en el análisis de la naturaleza de los datos y del resultado esperado, así como en las restricciones con las que está vinculado el problema, lo que significa que requiere comprensión y juicio técnico del problema.

# **Liga al código en Github**

Archivo: