



Momento de Retroalimentación:

Módulo 2

Análisis y Reporte sobre el desempeño del modelo.

Portafolio Análisis

Inteligencia artificial avanzada para
la ciencia de datos

Monica Escobedo Cota
A01642647

14/08/2025

Código de regresión lineal:

```
def mean(values):
    return sum(values) / len(values)

def covariance(x, y, x_mean, y_mean):
    result = 0
    for i in range(len(x)):
        result += (x[i] - x_mean) * (y[i] - y_mean)
    return result

def variance(values, mean_value):
    result = 0
    for i in range(len(values)):
        result += (values[i] - mean_value) ** 2
    return result

# Entrenamiento del modelo
def linear_regression(x, y):
    x_mean = mean(x)
    y_mean = mean(y)

    b1 = covariance(x, y, x_mean, y_mean) / variance(x, x_mean) #
    pendiente  $\beta$ 
    b0 = y_mean - b1 * x_mean #
    intercepto  $\alpha$ 
    return b0, b1

# Predicción
def predict(x, b0, b1):
    return b0 + b1 * x

# Programa principal
if __name__ == "__main__":
    # Datos de ejemplo
    x = [1, 2, 3, 4, 5] # horas de estudio
    y = [2, 4, 5, 4, 5] # calificación obtenida

    # Entrenamos el modelo
    b0, b1 = linear_regression(x, y)

    print("Modelo entrenado:")
    print(f"y = {b0:.2f} + {b1:.2f}x")
```

```

# Predicciones de prueba
test_x = [6, 7, 8]
print("\nPredicciones:")
for val in test_x:
    pred = predict(val, b0, b1)
    print(f"x={val} -> y={pred:.2f}")

# Predicción manual con input del usuario
while True:
    entrada = input("\nIngresa un valor de x (o escribe 'salir'): ")

    if entrada.lower() == "salir":
        break
    try:
        val = float(entrada)
        pred = predict(val, b0, b1)
        print(f"Predicción: y ≈ {pred:.2f}")
    except ValueError:
        print("Por favor ingresa un número válido.")

```

Análisis:

1. Separación de datos:

Se introducen solo los datos necesarios, lo que evita el sobreajuste o la falta de ajuste. La mayoría de los conjuntos de datos dividen sus datos en tres partes según las tareas correspondientes: conjuntos de entrenamiento, validación y prueba.

El conjunto de entrenamiento se utiliza para el entrenamiento del modelo y el ajuste de parámetros; el conjunto de validación, para la selección del modelo y la determinación de los parámetros finales; y el conjunto de prueba, para evaluar el rendimiento del modelo final.

2. Métricas de desempeño:

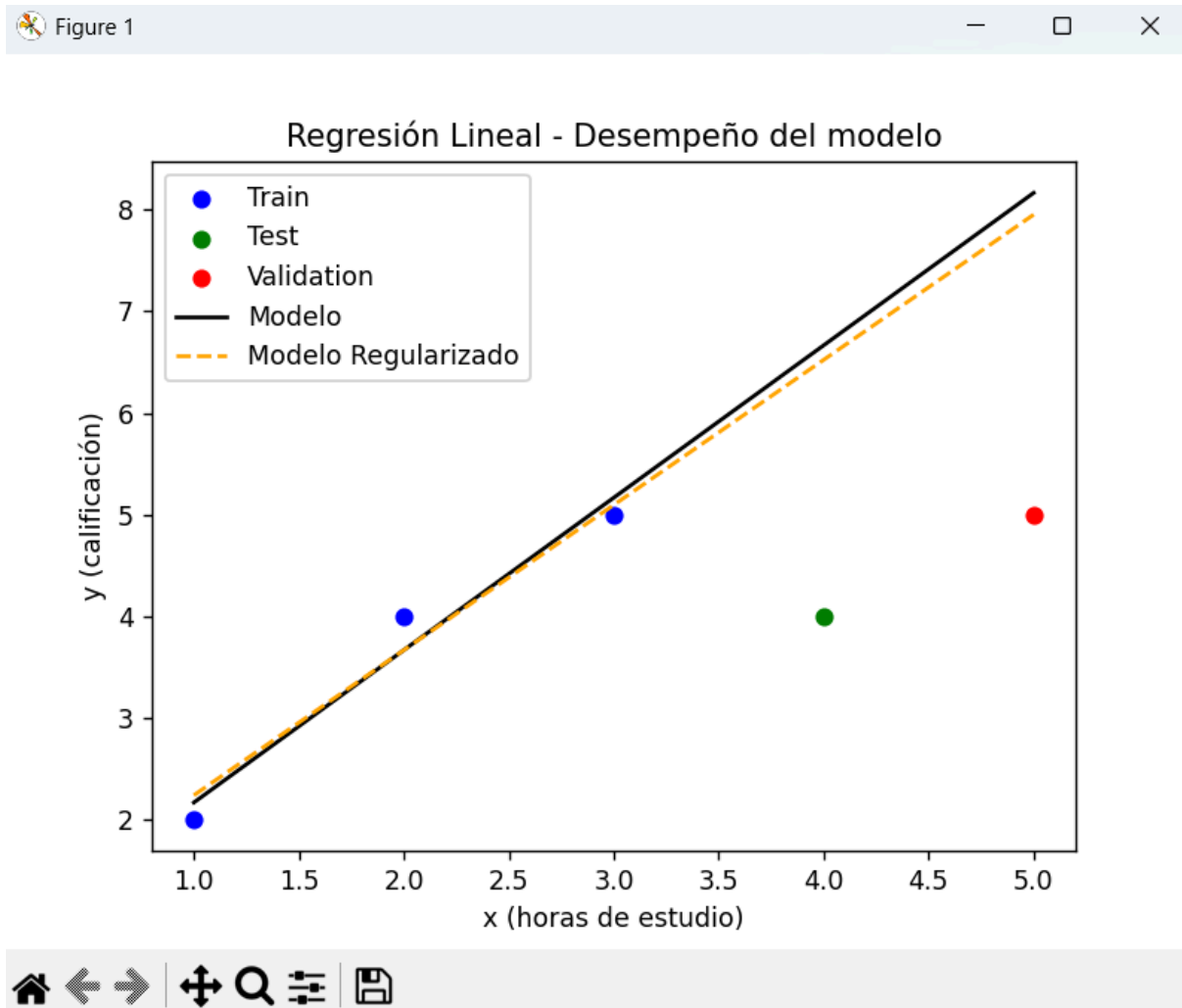
$$MSE = \frac{1}{n} \sum (y_{real} - y_{pred})^2$$

Error Cuadrático Medio:

- Entrenar: indica qué tan bien se ajusta el modelo a los datos con los que fue entrenado.

- Testing: evalúa si el modelo logra generalizar cuando se enfrenta a datos que no vio en el entrenamiento.
- Validación: permite verificar el desempeño final y confirmar si el modelo mantiene un rendimiento consistente.

3. Diagnóstico del modelo:



4. Mejora:

Los parámetros de la muestra estén demasiado inflados, dificulta su control de inestabilidad característica.

Una manera de mejora es añadir un parámetro adicional en el cálculo de la pendiente. Pero no puede ser una figura real.

Teniendo en cuenta los resultados de las ecuaciones aleatorias como "verdades" significa que es un resultado accidental.

Bibliografía:

Wikipedia contributors. (2025, September 14). *Simple linear regression*. In *Wikipedia, The Free Encyclopedia*. Retrieved from https://en.wikipedia.org/wiki/Simple_linear_regression