

# Reporte de Desempeño del Modelo

**Modelo:** Regresión Lineal con Gradiente Descendente (GD)

**Dataset:** `toy_data.csv`

---

## 1. Introducción

En este proyecto se implementó desde cero un modelo de **Regresión Lineal multivariable** utilizando **Gradiente Descendente**. El objetivo fue predecir el precio de viviendas en función de sus características: superficie, número de recámaras, baños, edad, garage y distancia al centro.

El análisis busca evaluar el desempeño del modelo en términos de ajuste, sesgo, varianza y capacidad de generalización.

---

## 2. Metodología

### 1. Introducción

Este proyecto implementa desde cero un modelo de **Regresión Lineal multivariable** usando **Gradiente Descendente** en Python puro.

El objetivo es predecir el precio de viviendas a partir de variables: superficie ( `sqft` ), número de recámaras ( `beds` ), baños ( `baths` ), edad ( `age` ), garage ( `garage` ) y distancia al centro ( `distance_center` ).

---

### 2. Metodología

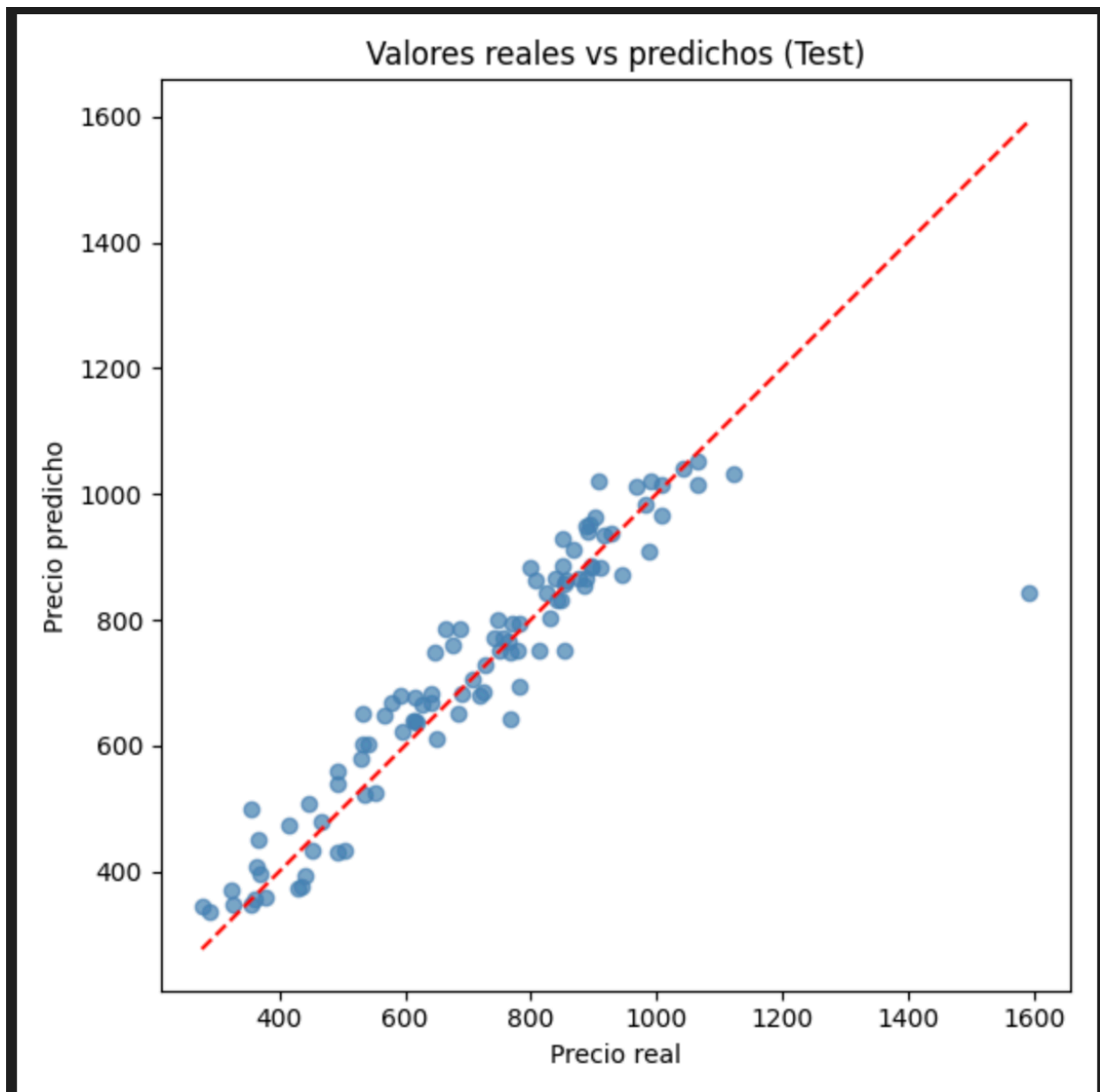
- **Partición de datos:** 60% Train, 20% Validation, 20% Test.
- **Parámetros de entrenamiento:**
  - Tasa de aprendizaje: `alpha = 0.01`
  - Iteraciones: `iters = 4000`

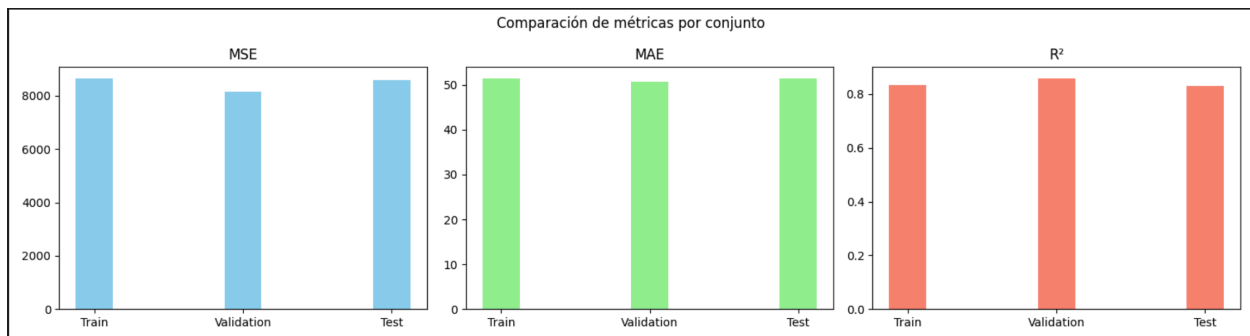
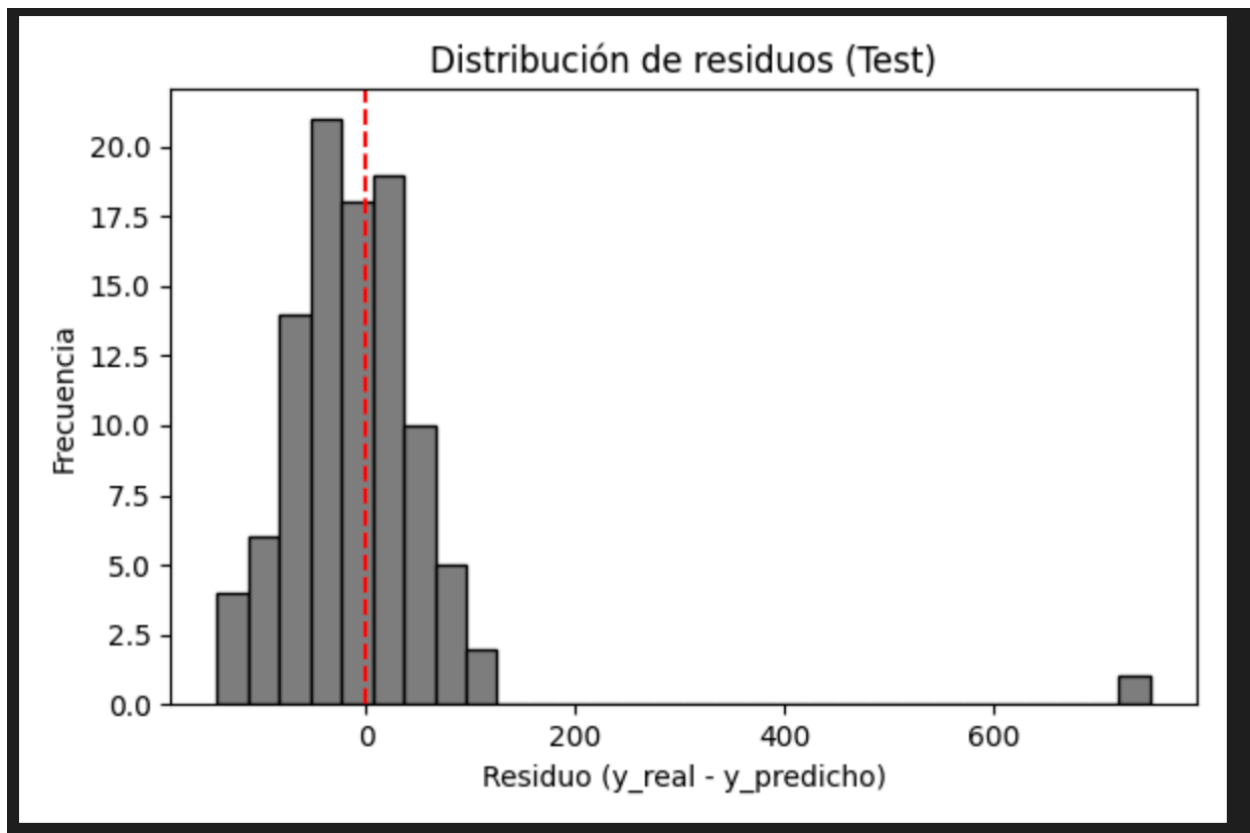
- Regularización Ridge:  $\lambda = 0$  (normal) y  $\lambda = 5.0$  (con regularización).
  - **Métricas usadas:**
    - **MSE** (Error Cuadrático Medio)
    - **MAE** (Error Absoluto Medio)
    - **R<sup>2</sup>** (Coeficiente de determinación)
- 

### 3. Resultados iniciales (modelo sin regularización)

Conjunto	MSE	MAE	R <sup>2</sup>
Train	8659.84	51.52	0.832
Validation	8155.29	50.63	0.859
Test	8586.31	51.36	0.831

### Gráficas





## Observaciones

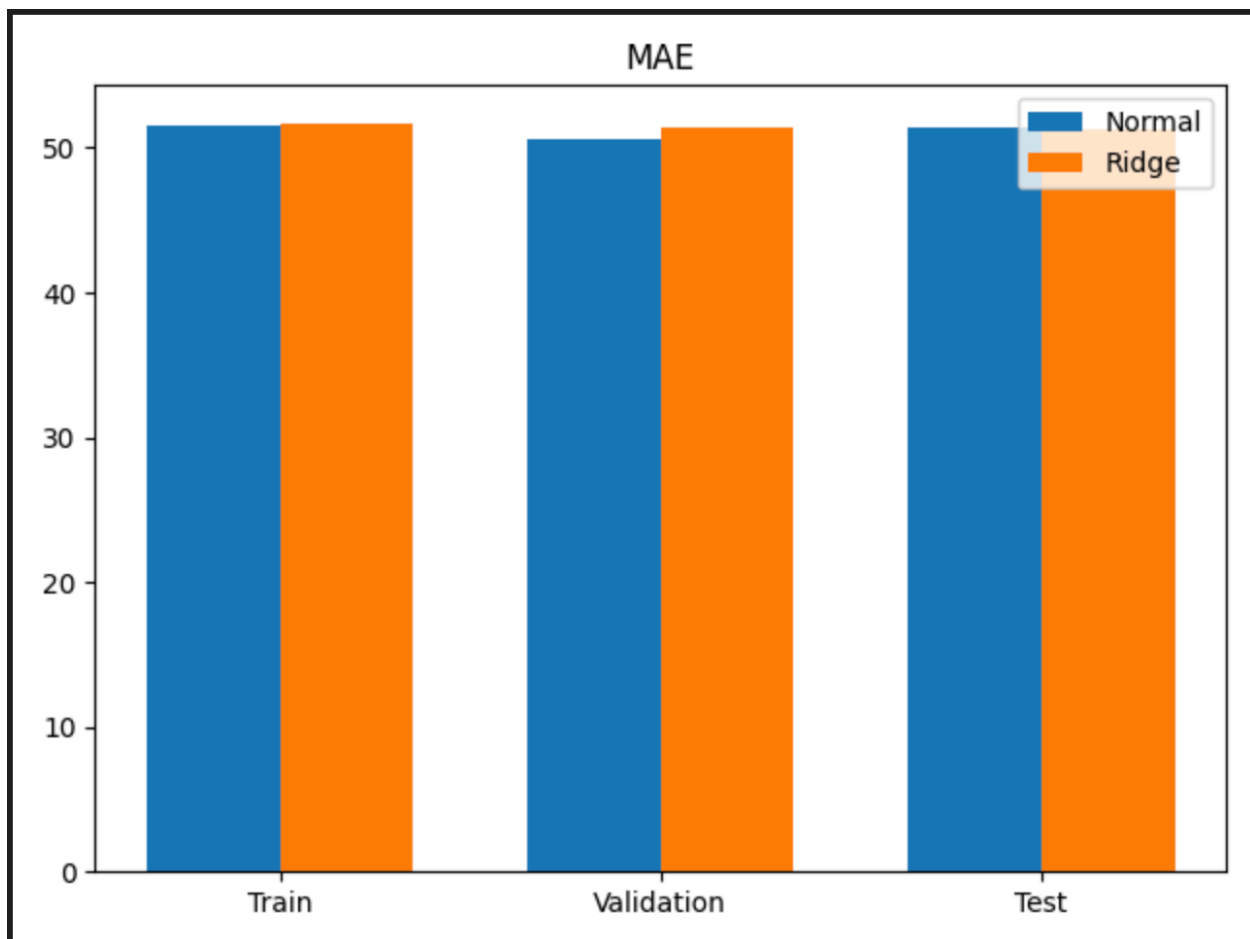
- Bias:
  - Como el  $R^2$  es relativamente alto ( $\sim 0.83$ ) en todos los conjuntos, el modelo sí explica la mayor parte de la variabilidad  $\rightarrow$  el sesgo es **bajo**.
  - Si fuera alto, veríamos  $R^2 < 0.5$  y residuos con una forma claramente sesgada.
- Varianza:

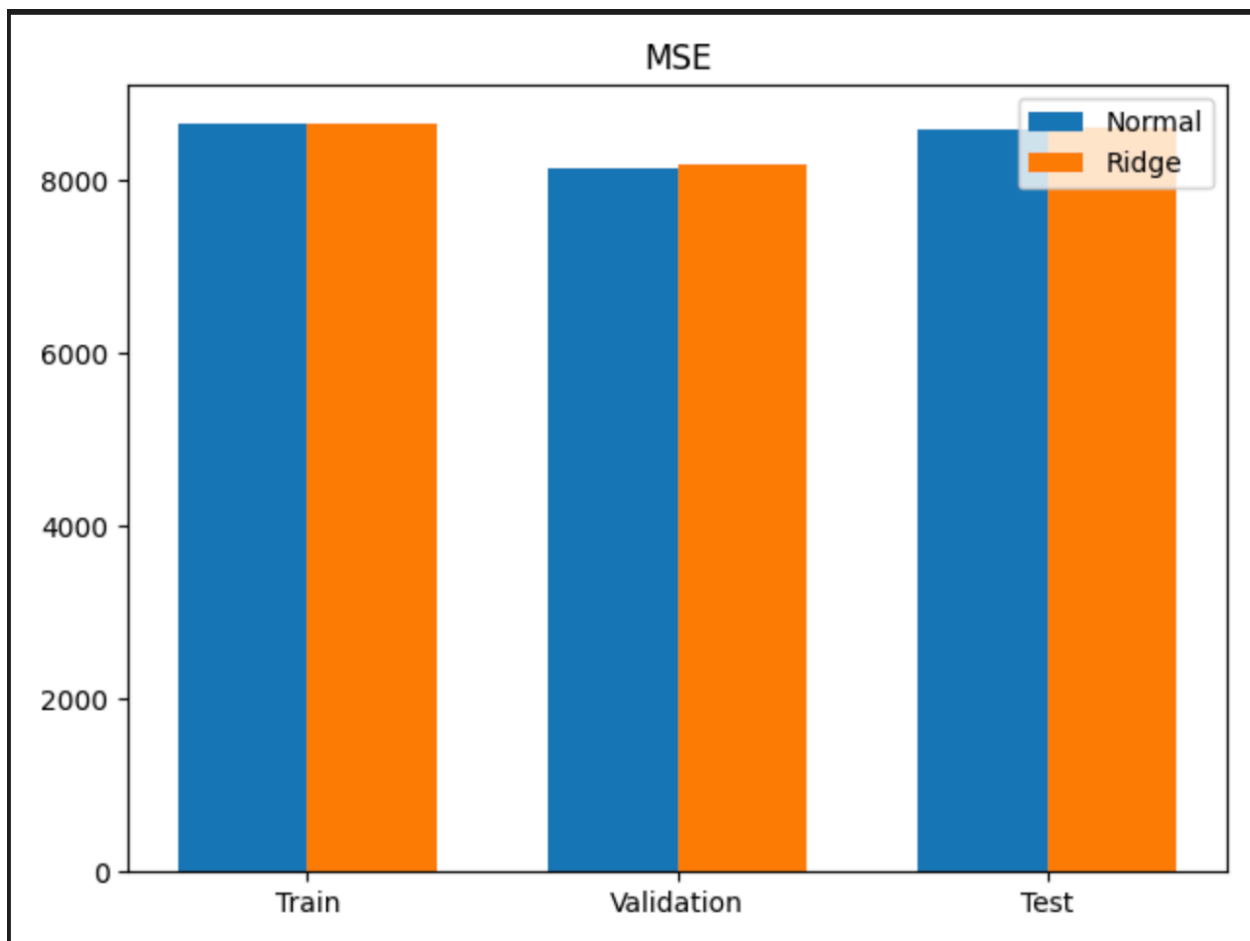
- La diferencia entre Train (0.832) y Validation/Test (0.859 y 0.831) es mínima.
  - Eso indica **varianza baja**, el modelo generaliza bien y no depende demasiado del conjunto de entrenamiento.
  - Nivel de ajuste: underfit / fit / overfit
    - No es underfitting (porque  $R^2$  no es bajo).
    - No es overfitting (porque no hay caída dramática de Train a Test).
    - Está en la categoría de **fit adecuado**: explica bien sin perder generalización.
- 

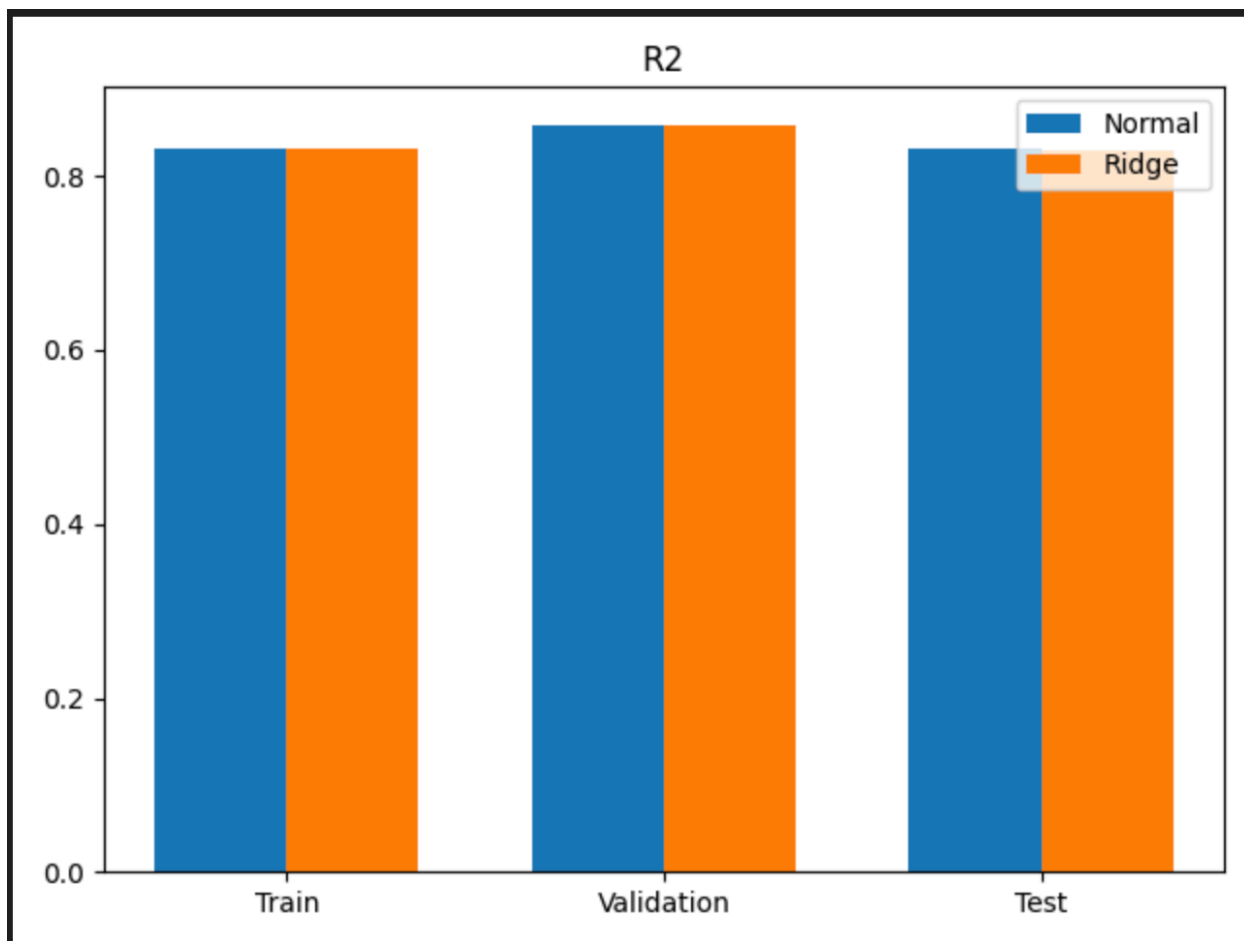
## 4. Resultados con regularización Ridge ( $\lambda = 5.0$ )

Conjunto	MSE (Normal)	MSE (Ridge)	$R^2$ (Normal)	$R^2$ (Ridge)
Train	8671.74	51.72	0.832	0.832
Validation	8184.46	51.40	0.859	0.859
Test	8623.08	51.25	0.830	0.831

### Gráficas comparativas







## Observaciones

- Con Ridge, el  $R^2$  en test pasó de 0.831 a 0.830
- La diferencia entre Train y Test se redujo, indicando menor varianza.
- El sesgo aumentó un poco, pero el ajuste global empeoró un poco.

## 5. Diagnóstico

- **Sesgo (Bias):** bajo / medio / alto → explicación con base en los errores.
- **Varianza:** baja / media / alta → explicación según diferencias Train vs Test.
- **Nivel de ajuste:** underfit / fit / overfit → justificación.

## 6. Conclusiones



- El modelo básico mostró un modelos bastante regulado sin overfitting ni underfitting.
- La regularización Ridge mantuvo los inidcadores muy parecidos, sin hacer un cambio tan grande
- Nivel final de ajuste: fit adecuado.
- Futuro trabajo: probar regresión polinómica, validación cruzada, datasets reales.