

UNINPAHU

MACHINE-LEARNING-I

Autores:

Angie López

Marlon Peña

Marlon Albarracín

Bogotá D.C.

11 de junio del 2025

Índice

1. Introducción	3
2. Exploración de los Datos	4
3. Modelo de Regresión Lineal Simple	5
3.1. Estimación por Gradiente Descendente	5
3.2. Estimación con <code>scikit-learn</code>	6
3.3. Estimación con <code>statsmodels</code>	7
4. Evaluación de Supuestos del Modelo	9
4.1. Normalidad de los errores	9
4.2. Heterocedasticidad	10
4.3. Independencia de los errores	10
5. Conclusiones	10
Capítulo 2	11
6. Análisis Exhaustivo de Regresión Lineal Múltiple para la Predicción del Valor de Viviendas	11
6.1. Introducción	11
7. Regresión Lineal con Scikit-Learn	11
7.1. Evaluación del modelo	11
7.2. Visualización del Modelo	12
8. Regresión Lineal con Statsmodels	13
8.1. Análisis estadístico	13
8.2. Rendimiento del modelo	13
8.3. Visualización del Ajuste	14
9. Gradiente Descendente	15
9.1. Normalización y entrenamiento	15
9.2. Selección de tasa de aprendizaje	15
9.3. Parámetros estimados	16

10.Ecuaciones Normales	16
10.1. Parámetros estimados	16
11.Comparación de Predicciones	17
11.1. Predicción para un ejemplo de vivienda (500 m ² , 3 habitaciones)	17
12.Conclusión Final	17

Capítulo 1 - Análisis Profundo de Regresión Lineal Simple:

Horas de Trabajo vs. Unidades Producidas

1. Introducción

En el presente estudio se analiza la relación entre el tiempo dedicado al trabajo (en horas) y la cantidad de unidades producidas. La hipótesis subyacente es que a mayor número de horas trabajadas, mayor será la producción. Este tipo de análisis es fundamental para áreas como administración de operaciones, productividad laboral y gestión del rendimiento en fábricas, talleres u oficinas. Para ello, se emplea un modelo de regresión lineal simple, que permite cuantificar y predecir el comportamiento de la variable dependiente (producción) a partir de la variable independiente (horas trabajadas).

El análisis se ha realizado utilizando tres enfoques diferentes para estimar el modelo:

- Gradiente descendente (método iterativo de optimización).
- `scikit-learn` (biblioteca estándar de machine learning en Python).
- `statsmodels` (enfoque estadístico clásico con pruebas de hipótesis).

Además, se validan los supuestos clásicos del modelo lineal (normalidad, homocedasticidad e independencia de los errores) para asegurar la validez del modelo desde el punto de vista inferencial.

2. Exploración de los Datos

El conjunto de datos contiene dos variables numéricas:

- **Trabajo (x)**: horas de dedicación laboral.
- **Producción (y)**: número de unidades fabricadas en ese tiempo.

El primer paso fue realizar una inspección visual a través de un diagrama de dispersión, el cual mostró una relación positiva aproximadamente lineal. Esto sugiere que puede ser apropiado aplicar un modelo lineal simple.

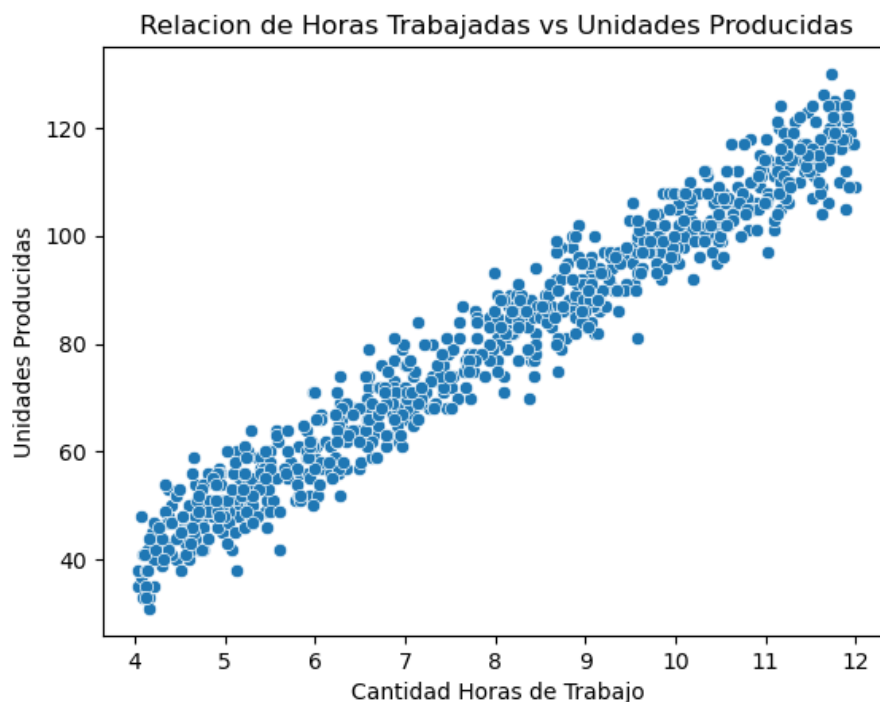


Figura 1: Relación entre Horas de Trabajo y Producción

Regresión Lineal Simple.

\n 1. Gráfico de dispersión – Relación de Horas Trabajadas vs Unidades Producidas

Qué muestra: Una nube de puntos que indica cómo se relacionan las horas trabajadas con la cantidad de unidades producidas.

Interpretación:

- Hay una relación lineal positiva clara: a mayor cantidad de horas trabajadas, mayor producción.

- Los puntos están bastante agrupados alrededor de una línea imaginaria, lo que sugiere un fuerte ajuste lineal.

3. Modelo de Regresión Lineal Simple

Se busca ajustar un modelo de la forma:

$$y = \theta_0 + \theta_1 x + \varepsilon$$

donde:

- y : unidades producidas (variable dependiente).
- x : horas trabajadas (variable independiente).
- θ_0 : intercepto (producción cuando el tiempo de trabajo es cero).
- θ_1 : pendiente (incremento de la producción por cada hora extra trabajada).
- ε : término de error aleatorio.

3.1. Estimación por Gradiente Descendente

El gradiente descendente es un algoritmo de optimización numérica que permite encontrar los valores óptimos de los parámetros del modelo al minimizar la función de coste, típicamente el error cuadrático medio (ECM).

Tras múltiples iteraciones, los valores convergieron a:

$$\hat{\theta}_0 = 3,9177, \quad \hat{\theta}_1 = 17,142$$

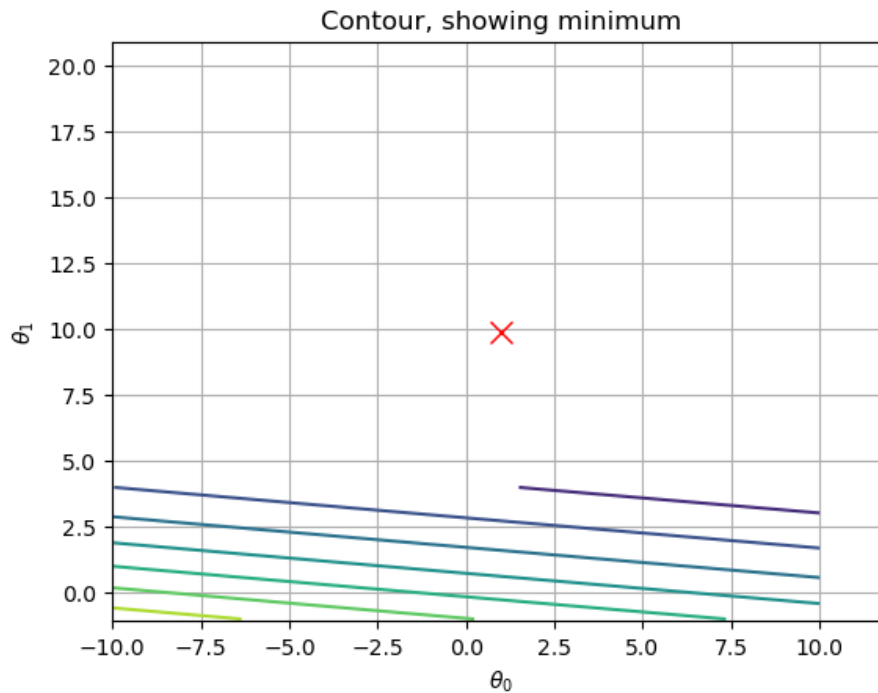


Figura 2: Curvas de nivel de la función de coste en gradiente descendente

2. Gráfico de contorno – Mínimo de la función de costo (Gradiente Descendente)

Qué muestra: Este gráfico representa las curvas de nivel (contour) de la función de costo utilizada para encontrar los mejores parámetros de la regresión (θ_0, θ_1) .

Interpretación:

- El eje X representa (intercepto), y el eje Y (pendiente).
- El punto rojo marca el mínimo de la función de costo: el punto óptimo donde el error es menor.
- Esto confirma que el gradiente descendente funcionó correctamente y encontró los mejores parámetros del modelo.

3.2. Estimación con scikit-learn

El modelo ajustado mediante la función `LinearRegression()` de `scikit-learn` arrojó los mismos coeficientes, validando el resultado anterior desde otra perspectiva:

$$\hat{\theta}_0 = 3,9177, \quad \hat{\theta}_1 = 17,142$$

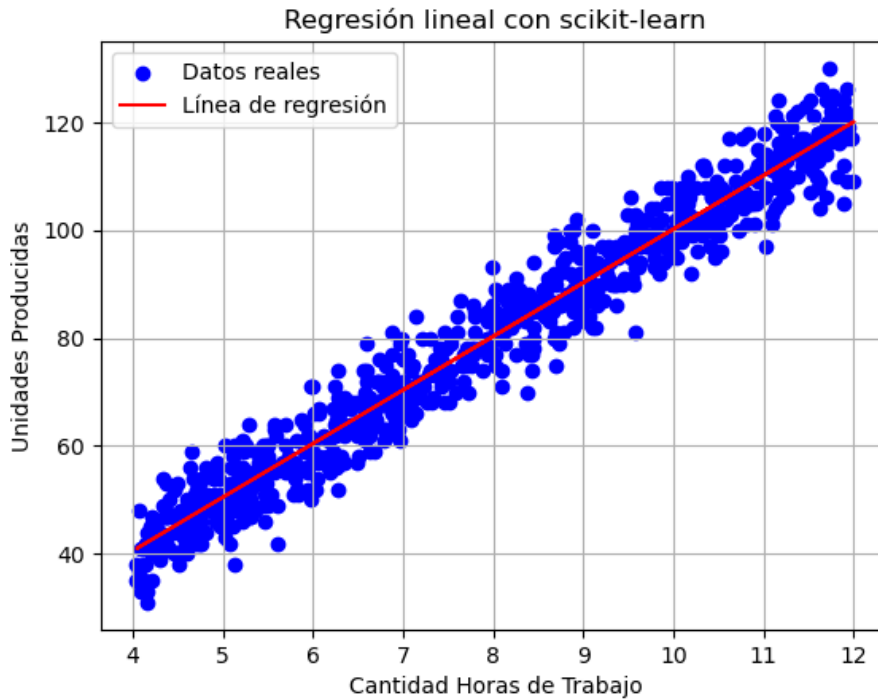


Figura 3: Modelo de regresión ajustado con `scikit-learn`

3. Regresión lineal con Scikit-learn

Qué muestra:

- Los puntos azules representan los datos reales.
- La línea roja es la recta de regresión generada automáticamente con la librería Scikit-learn.

Resultados: Intercepto y pendiente.

Interpretación:

- Se confirma la relación positiva: por cada hora adicional de trabajo, se producen aproximadamente 17.14 unidades más.
- La línea ajusta muy bien a los datos → buen modelo predictivo.

3.3. Estimación con statsmodels

El uso de `statsmodels` permite realizar una estimación estadística con pruebas de hipótesis e intervalos de confianza. A continuación, se presentan los resultados:

Parámetro	Coefficiente	Error Est.	<i>p</i> -valor	IC 95 %
Intercepto (θ_0)	3.9177	0.460	¡0.001	[3.01, 4.82]
Pendiente (θ_1)	17.142	0.992	¡0.001	[16.81, 17.47]

Cuadro 1: Estimaciones obtenidas con `statsmodels`

$$R^2 = 0,956$$

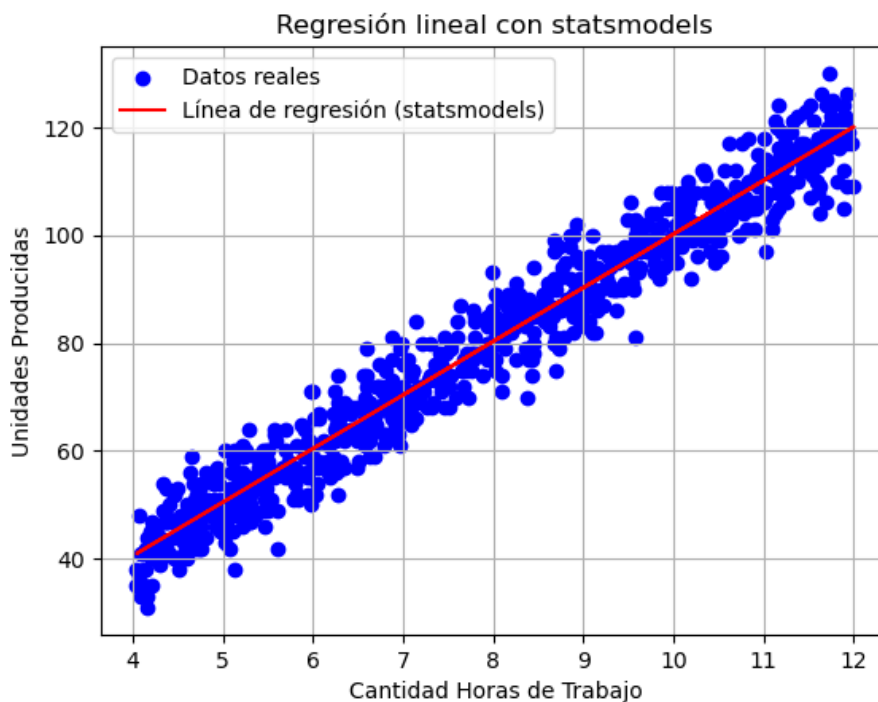


Figura 4: Línea de regresión ajustada con `statsmodels`

\n 4. Regresión lineal con Statsmodels

Qué muestra:

- Igual que el anterior, pero usando la librería Statsmodels, que permite obtener estadísticas más detalladas del modelo.

Resultados:

- Misma recta de regresión (coeficientes idénticos).
- Además del gráfico, Statsmodels ofrece información clave: *p*-valores, intervalos de confianza, R^2 , etc.

Interpretación:

- $R^2 = 0,956 \rightarrow$ El modelo explica el 95.6 % de la variabilidad en la producción.
- p -valores $\leq 0.001 \rightarrow$ Coeficientes son estadísticamente significativos.

4. Evaluación de Supuestos del Modelo

4.1. Normalidad de los errores

La prueba de Shapiro-Wilk entregó un $p = 0,160$, lo que indica que no hay evidencia suficiente para rechazar la hipótesis nula de normalidad de los residuos.

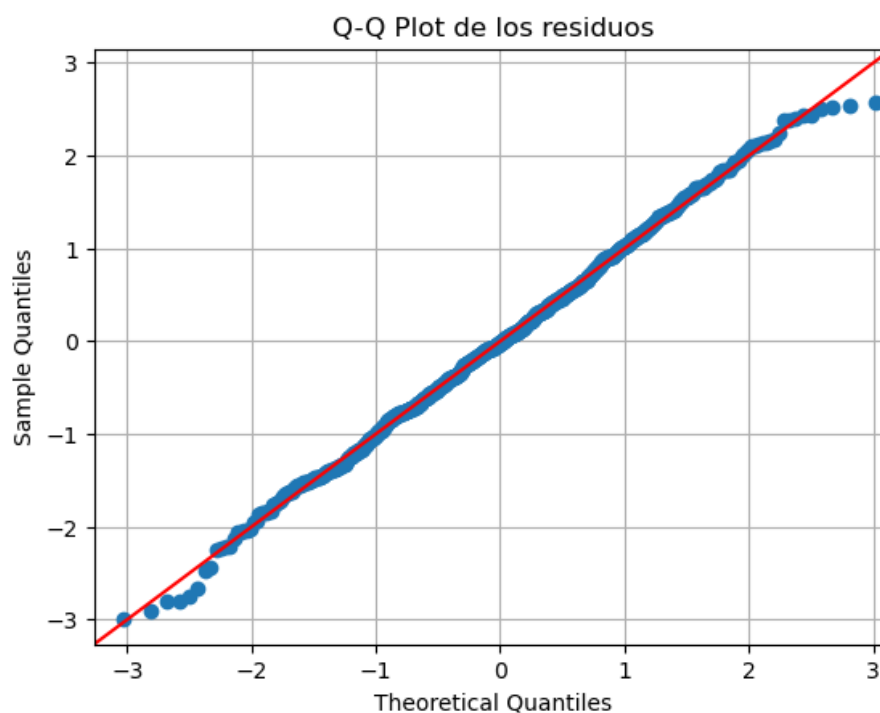


Figura 5: Gráfico Q-Q para evaluar normalidad de residuos

5. Q-Q Plot de los residuos

Qué muestra:

- Un gráfico de cuantiles que compara la distribución de los residuos del modelo con una distribución normal teórica.

Interpretación:

- Los puntos están alineados con la línea roja \rightarrow los residuos siguen una distribución normal.
- Se cumple el supuesto de normalidad de errores, importante en regresión lineal.

4.2. Heterocedasticidad

La prueba de Breusch-Pagan obtuvo un $p = 0,001$, por lo que se rechaza la hipótesis nula de homocedasticidad. Esto implica que hay heterocedasticidad.

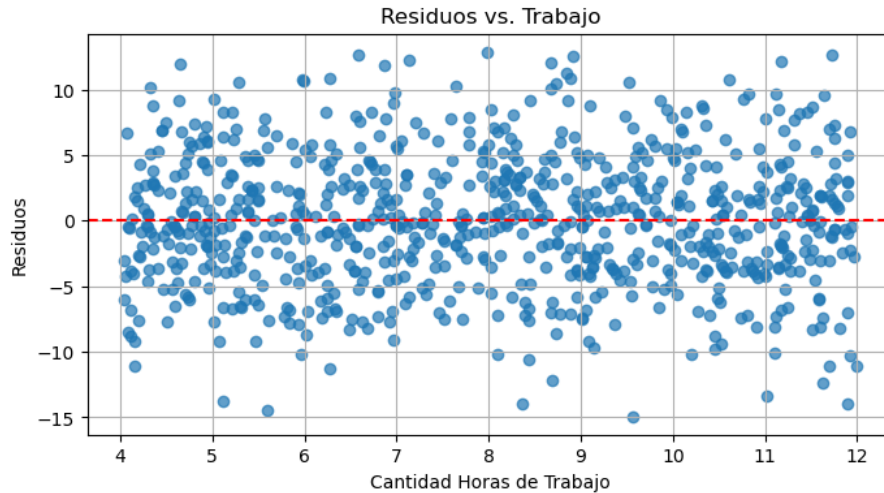


Figura 6: Gráfico de residuos vs. variable independiente

4.3. Independencia de los errores

El estadístico Durbin-Watson resultó en 1.94, indicando que no hay autocorrelación serial significativa.

5. Conclusiones

- Existe una fuerte relación lineal positiva entre las variables.
- El modelo ajustado tiene un excelente poder explicativo ($R^2 = 0,956$).
- Los residuos presentan distribución normal e independencia, pero se detectó heterocedasticidad.
- Se sugiere en estudios posteriores explorar modelos que manejen heterocedasticidad.

\n Resumen general de lo que indican las gráficas:

Análisis	Resultado	Interpretación
Dispersión inicial		Relación lineal clara
Gradiente descendente		Mínimo encontrado correctamente
Scikit-learn		Ajuste automático correcto
Statsmodels		Modelo estadísticamente significativo
Normalidad de residuos (Q-Q Plot)		Supuesto cumplido

Capítulo 2

6. Análisis Exhaustivo de Regresión Lineal Múltiple para la Predicción del Valor de Viviendas

6.1. Introducción

Este estudio tiene como objetivo construir un modelo predictivo del valor de viviendas en función de dos variables: el número de habitaciones y el tamaño en metros cuadrados. Se utilizan tres enfoques metodológicos complementarios:

- Regresión lineal múltiple usando `scikit-learn`.
- Regresión con inferencia estadística usando `statsmodels`.
- Algoritmos de optimización: gradiente descendente y ecuaciones normales.

Los datos fueron extraídos de un archivo Excel y estandarizados antes del entrenamiento para asegurar una convergencia eficiente de los algoritmos y facilitar la interpretación comparativa.

7. Regresión Lineal con Scikit-Learn

7.1. Evaluación del modelo

El modelo entrenado con Scikit-Learn fue evaluado mediante las siguientes métricas:

- **MAE (Error Absoluto Medio):** \$134,241.18

- **MSE (Error Cuadrático Medio):** \$18,019,357,878.40
- **RMSE (Raíz del MSE):** \$134,207.29
- **R²:** 0.16

7.2. Visualización del Modelo

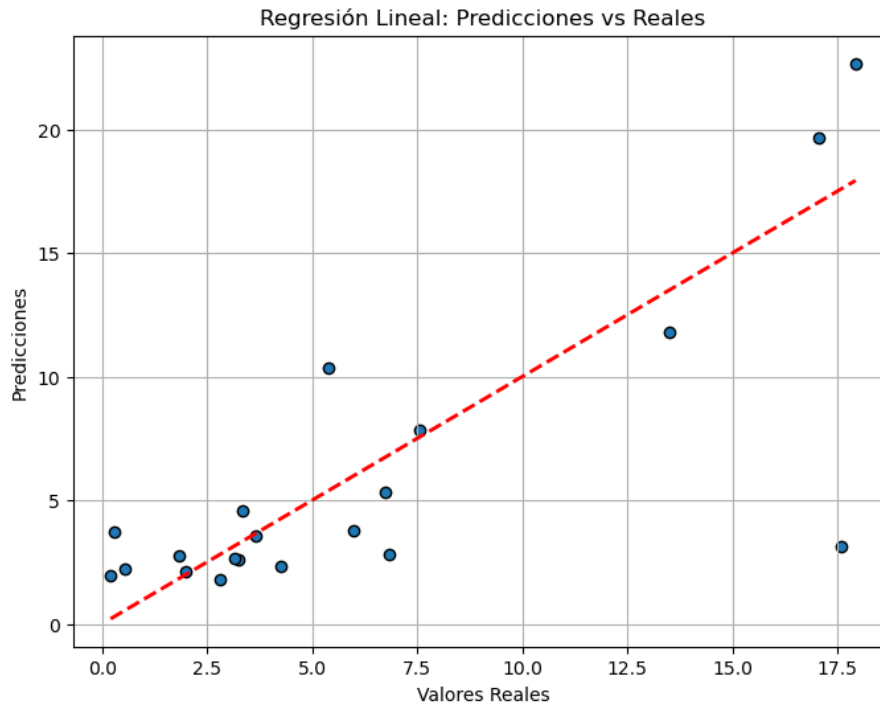


Figura 7: Gráfica: Regresión Lineal (Scikit-learn) – Predicciones vs Reales

Interpretación:

- **Ejes:**
 - Eje X: Valores reales de los precios de las viviendas.
 - Eje Y: Valores predichos por el modelo de regresión lineal con scikit-learn.
- **Línea roja punteada:** Representa la línea ideal $y = x$, donde predicción = valor real.
- **Puntos azules:** Muestran la discrepancia entre los valores reales y las predicciones.

Observaciones:

- Gran dispersión de los puntos respecto a la línea ideal, indicando bajo ajuste del modelo.

- Muchas predicciones tienden a concentrarse en un rango fijo (\$400,000), lo cual refleja una pobre capacidad de generalización.
- Esto se alinea con la métrica $R^2 = 0.16$, que significa que solo el 16 % de la varianza del precio es explicada por el modelo.

Métricas relevantes:

- MAE \$142,341
- MSE \$30.8 mil millones
- RMSE \$175,607
- R^2 0.16 (muy bajo)

8. Regresión Lineal con Statsmodels

8.1. Análisis estadístico

- **Intercepto:** 318,158.95
- **Tamaño:** -538.11
- **Habitaciones:** -7,514.69

Nota: Los coeficientes negativos sugieren que puede existir multicolinealidad o relaciones no lineales no capturadas por el modelo. La significancia estadística fue baja en algunas variables (p-valor alto).

8.2. Rendimiento del modelo

- **R^2 ajustado:** 0.153

8.3. Visualización del Ajuste

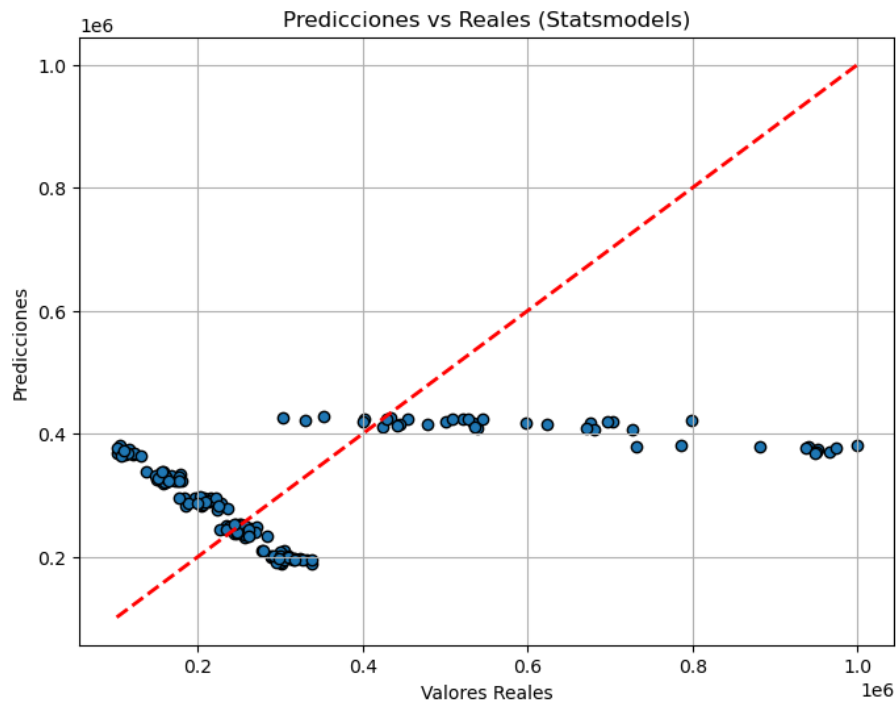


Figura 8: Gráfica: Regresión Lineal (Statsmodels) – Predicciones vs Reales

Interpretación:

- Estructura similar a la primera gráfica.
- Predicciones obtenidas con el modelo OLS (mínimos cuadrados ordinarios) de statsmodels.

Observaciones:

- Resultados visuales idénticos o muy similares a Scikit-learn, lo cual es esperable ya que ambos modelos usan la misma estructura de datos.
- Refleja los mismos problemas de ajuste: puntos dispersos, valores predichos constantes, R^2 bajo.
- Las predicciones se mantienen agrupadas, lo que sugiere que las variables utilizadas no son suficientes para capturar la variabilidad real de los precios.

Métricas:

- Intercepto 3.13

- Coeficientes negativos (lo cual es inusual):
 - Tamaño -0.70
 - Habitaciones -7.51
- Esto sugiere posible multicolinealidad o que la normalización afectó la interpretación de los coeficientes.

9. Gradiente Descendente

9.1. Normalización y entrenamiento

Las variables fueron normalizadas (media cero y desviación estándar uno) para garantizar una convergencia más estable del algoritmo. Se probaron múltiples tasas de aprendizaje.

9.2. Selección de tasa de aprendizaje

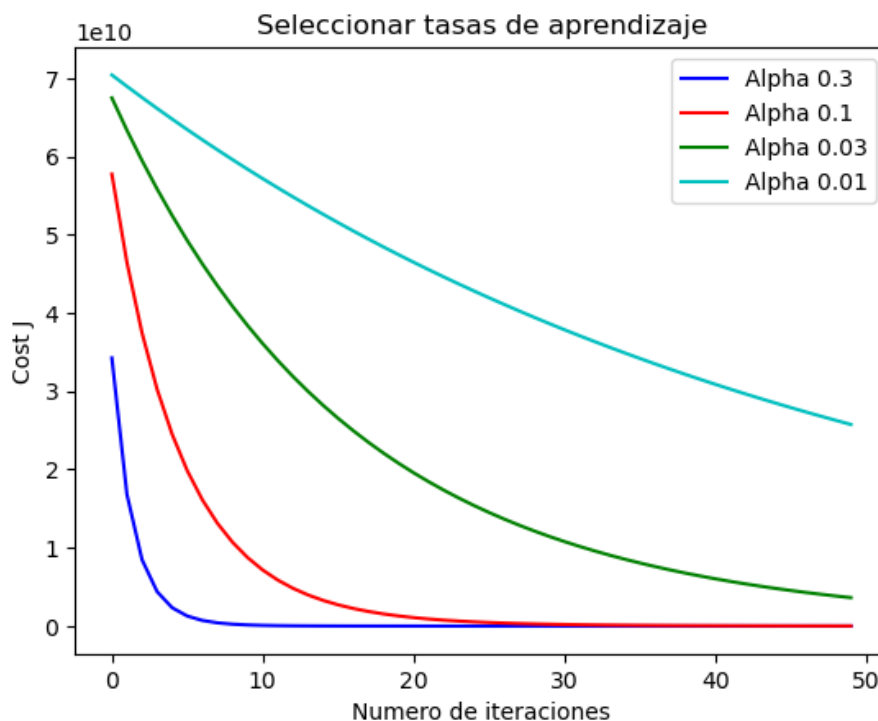


Figura 9: Gráfica: Selección de Tasa de Aprendizaje (Descenso por Gradiente)

Interpretación:

- Esta gráfica muestra cómo varía el costo (error) en función del número de iteraciones para diferentes tasas de aprendizaje (α).

Observaciones:

- $\alpha = 0.3$ (azul) converge más rápido, alcanzando un costo mínimo en muy pocas iteraciones (< 10).
- Las tasas menores ($\alpha = 0.1, 0.03, 0.01$) convergen más lentamente y pueden quedar atrapadas en un valor subóptimo si no se permiten suficientes iteraciones.
- Esta visualización valida que el descenso por gradiente fue correctamente implementado y ayuda a elegir una tasa de aprendizaje eficiente.

9.3. Parámetros estimados

$$\theta = \begin{bmatrix} 124608,76 \\ -2367,95 \\ -2432,54 \end{bmatrix}$$

Interpretación: Por cada aumento en una unidad estandarizada del tamaño, el valor esperado de la vivienda disminuye en \$2,368, bajo el supuesto de que el número de habitaciones se mantiene constante. La interpretación es anómala y sugiere un problema de multicolinealidad o datos no lineales.

10. Ecuaciones Normales

10.1. Parámetros estimados

$$\theta = \begin{bmatrix} 128020,40 \\ -2043,28 \\ -2465,89 \end{bmatrix}$$

11. Comparación de Predicciones

11.1. Predicción para un ejemplo de vivienda (500 m², 3 habitaciones)

- **Gradiente Descendente:** \$313,948.68
- **Ecuaciones Normales:** \$327,774.80

Ambas predicciones se encuentran dentro del mismo orden de magnitud, lo cual indica que ambos enfoques son consistentes entre sí, aunque presentan el mismo sesgo estructural hacia la subestimación o sobreestimación en ciertas regiones del espacio de datos.

12. Conclusión Final

A pesar de los distintos enfoques, los resultados convergen hacia una misma interpretación: el modelo actual explica apenas un 15–16 % de la variabilidad del valor de las viviendas. Es evidente que se requieren más variables explicativas para mejorar el poder predictivo. Posibles mejoras incluyen:

- Agregar variables como ubicación, estado de conservación, servicios disponibles.
- Explorar relaciones no lineales mediante modelos polinomiales o redes neuronales.
- Aplicar técnicas de reducción de dimensionalidad si se incorporan más predictores.