

Curso base

# INTELIGENCIA ARTIFICIAL DESDE CERO

## LECCIÓN 15 Regresión: predicción numérica

Cómo entrenar un modelo sencillo para predecir un valor con datos básicos y una interpretación comprensible

<b>Lección</b>	15	<b>Tema</b>	Regresión: predicción numérica
<b>Nombre del participante</b>	_____	<b>Fecha</b>	_____

### Actividad central

**Actividad principal:** construcción de un primer modelo de regresión lineal simple con dos ejemplos guiados e interpretación de predicciones numéricas.

**Producto esperado:** primer modelo de regresión con dos ejemplos guiados, al menos una predicción nueva en cada caso y una interpretación breve escrita con lenguaje comprensible.

**Meta de la lección.** Comprender de manera intuitiva qué es una regresión, para qué sirve y cómo entrenar un modelo sencillo que permita predecir un valor numérico a partir de datos básicos. Al finalizar esta guía, el participante podrá ejecutar un modelo elemental, interpretar con palabras sencillas qué significa una predicción y apoyarse en la IA generativa para entender mejor el proceso.

### Idea central

La regresión es una forma de aprendizaje automático que se utiliza cuando lo que deseamos predecir es un número. No se usa para responder sí o no, ni para clasificar casos en categorías como alto, medio o bajo. Se usa cuando queremos estimar valores como precio, temperatura, ventas, consumo, tiempo o nota final.

En esta lección no buscamos profundizar en teoría compleja. Lo que buscamos es construir una intuición clara. Si contamos con algunos datos conocidos, podemos entrenar un modelo sencillo para que encuentre una relación aproximada entre una variable de entrada y una variable de salida. Después, ese modelo puede utilizarse para generar una predicción nueva.

### ¿Qué vamos a hacer en esta guía?

- Comprender qué significa una regresión y cuándo conviene usarla.
- Reconocer las partes básicas de un modelo sencillo: entrada, salida, modelo y predicción.
- Desarrollar un primer ejemplo guiado para predecir el precio de una vivienda según su área.
- Trabajar un segundo ejemplo guiado para predecir una nota final según horas de estudio.
- Usar prompts de IA generativa para comprender, interpretar y corregir el proceso.

### ¿Cuándo conviene usar regresión?

La regresión conviene cuando el problema pide responder con un valor numérico. Por ejemplo, puede servir para estimar el precio de una vivienda según su tamaño, calcular ventas aproximadas a partir de una campaña, predecir el consumo de energía según las horas de uso o aproximar una nota final a partir de ciertas variables de trabajo académico.

Si el resultado esperado fuera una categoría, entonces no hablaríamos de regresión sino de clasificación. Esta diferencia es muy importante. En regresión la salida puede ser, por ejemplo, 25,4; 70,1; 182 o 950, mientras que en clasificación la salida suele ser una etiqueta o clase.

<b>Situación</b>	¿Qué valor se quiere predecir?
<b>Vivienda</b>	Precio estimado de una casa o apartamento según área, ubicación o número de habitaciones.
<b>Educación</b>	Nota o puntaje final según asistencia, talleres, tareas o exámenes.
<b>Ventas</b>	Cantidad vendida en un período a partir de promociones, campañas o fechas especiales.

### Vocabulario mínimo para comenzar

<b>Elemento</b>	Explicación sencilla
<b>x</b>	VARIABLES DE ENTRADA. Por ejemplo: horas de estudio, área en metros cuadrados o inversión publicitaria.
<b>y</b>	Valor numérico que queremos predecir. Por ejemplo: nota, precio, ventas o consumo.
<b>Modelo</b>	Regla aprendida por el algoritmo para relacionar X con y.
<b>Predicción</b>	Valor estimado que el modelo entrega cuando recibe un caso nuevo.

#### Recomendación pedagógica antes de empezar

En esta primera experiencia no interesa la perfección técnica. Lo importante es que el estudiante pueda responder preguntas como estas: ¿qué datos le entrego al modelo?, ¿qué valor quiero que aprenda a predecir?, ¿qué significa el número que devuelve?, ¿esa predicción parece coherente con los datos observados?

## Ejemplo guiado 1. Predecir el precio de una vivienda según su área

Vamos a empezar con un caso sencillo y fácil de visualizar. Imaginemos una pequeña tabla donde la entrada es el área de una vivienda en metros cuadrados y la salida es el precio en millones de pesos.

Área (m <sup>2</sup> )	Precio (millones)
40	120
55	155
70	190
85	230
100	265

### ¿Qué observamos a simple vista?

Antes de correr cualquier código, conviene mirar la tabla y pensar. Cuando el área aumenta, el precio también aumenta. Eso sugiere que puede existir una relación aproximadamente creciente entre ambas variables. El modelo intentará capturar esa tendencia.

### Secuencia sugerida

1. Crear o cargar la tabla de datos.
2. Identificar la variable de entrada y la variable objetivo.
3. Preparar el notebook e importar las herramientas necesarias.
4. Construir la tabla, separar X e y, entrenar el modelo y hacer una predicción nueva.
5. Interpretar la pendiente, el intercepto y el valor predicho con palabras sencillas.

### Paso 1. Crear o cargar la tabla de datos

En este caso usaremos una tabla pequeña creada manualmente dentro del notebook.

### Paso 2. Identificar la variable de entrada y la variable objetivo

La variable de entrada será `area_m2`. La variable objetivo será `precio_millones`. Es decir, queremos que el modelo aprenda a predecir el precio a partir del área.

### Pasos 3 a 9. Preparar y entrenar el modelo

```
import pandas as pd
from sklearn.linear_model import LinearRegression

datos = pd.DataFrame({
    "area_m2": [40, 55, 70, 85, 100],
    "precio_millones": [120, 155, 190, 230, 265]
})
datos

X = datos[["area_m2"]]
```

```
y = datos["precio_millones"]

modelo = LinearRegression()
modelo.fit(X, y)

nueva_area = pd.DataFrame({"area_m2": [75]})
prediccion = modelo.predict(nueva_area)
print("Precio estimado:", round(prediccion[0], 2), "millones")

print("Pendiente aproximada:", round(modelo.coef_[0], 2))
print("Intercepto:", round(modelo.intercept_, 2))
```

### Paso 10. Explicar el resultado con palabras sencillas

Si la predicción da un valor cercano a 203 millones, eso no significa que ese sea el precio exacto de toda vivienda de 75 m<sup>2</sup>. Significa que, con base en los datos que le entregamos, el modelo estima un valor razonable alrededor de esa cantidad. La predicción debe verse como una aproximación, no como una verdad absoluta.

### Prompts de IA generativa para acompañar el ejemplo 1

#### Prompt 1. Comprender la idea del ejercicio

*Explícame con palabras muy sencillas qué hace un modelo de regresión lineal cuando recibe una tabla con área de viviendas y precios. Usa lenguaje para principiantes.*

#### Prompt 2. Interpretar la predicción

*Si un modelo de regresión predice 203 millones para una vivienda de 75 m<sup>2</sup>, ayúdame a redactar una explicación corta, clara y prudente para un estudiante principiante.*

#### Prompt 3. Interpretar la pendiente

*Explícame qué significa la pendiente en una regresión lineal simple usando el ejemplo de área de vivienda y precio.*

#### Prompt 4. Revisar el código

*Revisa este código de regresión lineal simple y dime si está correcto. Si hay errores, explícamelos paso a paso y propón una versión corregida.*

### Comentario pedagógico

El estudiante no debe usar la IA generativa para reemplazar su razonamiento, sino para mejorar su comprensión. Lo ideal es que primero observe la tabla, haga una hipótesis sencilla y luego use la IA para confirmar o mejorar su explicación.

## Ejemplo guiado 2. Predecir la nota final según horas de estudio

En esta versión de la guía, la antigua actividad principal se reemplaza por un segundo ejemplo guiado para que el estudiante tenga un acompañamiento más fuerte antes de trabajar con mayor autonomía.

Horas de estudio	Nota final
2	2.8
3	3.1
4	3.5
5	3.9
6	4.2
7	4.5

### ¿Qué observamos a simple vista?

Parece que cuando aumentan las horas de estudio, también aumenta la nota final. Otra vez aparece una tendencia creciente. El modelo intentará representarla.

### Pasos 1 a 5. Crear y entrenar el modelo

```
import pandas as pd
from sklearn.linear_model import LinearRegression

datos_estudio = pd.DataFrame({
    "horas_estudio": [2, 3, 4, 5, 6, 7],
    "nota_final": [2.8, 3.1, 3.5, 3.9, 4.2, 4.5]
})
datos_estudio

X2 = datos_estudio[["horas_estudio"]]
y2 = datos_estudio["nota_final"]

modelo2 = LinearRegression()
modelo2.fit(X2, y2)

nuevas_horas = pd.DataFrame({"horas_estudio": [5.5]})
prediccion2 = modelo2.predict(nuevas_horas)
print("Nota estimada:", round(prediccion2[0], 2))
```

### Paso 6. Interpretar el resultado

Si el modelo estima una nota cercana a 4.0, debemos leer ese resultado como una aproximación basada en los datos de ejemplo. No significa que todas las personas que estudien 5.5 horas obtendrán exactamente esa nota, porque en la realidad influyen muchos otros factores.

## Paso 7. Comparar los dos ejemplos

En el primer ejemplo predecimos precio a partir de área. En el segundo ejemplo predecimos nota a partir de horas de estudio. Aunque los contextos son diferentes, la lógica es la misma: el modelo recibe datos numéricos de entrada, aprende una relación aproximada y luego produce una predicción nueva.

## Prompts de IA generativa para acompañar el ejemplo 2

### Prompt 1. Comparar ejemplos

*Compara en lenguaje sencillo estos dos ejemplos de regresión: predecir precio según área y predecir nota según horas de estudio. Explica qué tienen en común.*

### Prompt 2. Ayuda para redactar interpretación

*Redáctame un comentario de tres líneas, claro y sencillo, para explicar la predicción de la nota final en un ejercicio de regresión lineal.*

### Prompt 3. Detección de errores comunes

*Voy a entrenar una regresión lineal simple en Google Colab. Dime cuáles son los errores más comunes que puede cometer un principiante al separar  $X$  e  $y$  y al hacer una predicción.*

### Prompt 4. Reflexión crítica

*Explícame por qué una predicción numérica no debe tomarse como una verdad exacta, sino como una aproximación basada en datos.*

## Buenas prácticas para esta primera experiencia

<b>Recomendación</b>	¿Por qué importa?
<b>Usar datos limpios</b>	Errores de formato o valores extraños pueden afectar la predicción.
<b>Empezar con un caso simple</b>	Una sola variable ayuda a entender mejor el proceso.
<b>Interpretar, no solo ejecutar</b>	No basta con correr el código; hay que comprender qué significa.
<b>No prometer exactitud total</b>	Toda predicción tiene margen de error.
<b>Comparar la predicción con la tabla</b>	Ayuda a decidir si el resultado parece razonable.
<b>Usar IA generativa con criterio</b>	La IA debe apoyar la comprensión, no reemplazarla.

## Mini ruta de trabajo sugerida para el participante

### Recorrido recomendado

- Observe la tabla y describa la tendencia con palabras.
- Identifique con claridad cuál es la entrada y cuál es la salida.
- Ejecute el modelo paso a paso.
- Haga una predicción para un caso nuevo.
- Escriba una interpretación breve y prudente del resultado.
- Si encuentra una dificultad, use uno de los prompts sugeridos para pedir ayuda puntual.

## Producto esperado

El producto esperado en esta versión de la guía es un notebook corto y ordenado que contenga dos ejemplos básicos de regresión lineal simple, al menos una predicción nueva en cada caso y una interpretación breve redactada con palabras comprensibles.

## Lista de verificación

Revisión	Listo
¿Se identificó correctamente la variable de entrada en cada ejemplo?	<input type="checkbox"/>
¿Se identificó correctamente la variable objetivo en cada ejemplo?	<input type="checkbox"/>
¿Se entrenó el modelo sin errores?	<input type="checkbox"/>
¿Se realizó al menos una predicción nueva en cada ejemplo?	<input type="checkbox"/>
¿Se explicó con palabras sencillas qué representa la predicción?	<input type="checkbox"/>
¿Se usó al menos un prompt de IA generativa para comprender o mejorar el trabajo?	<input type="checkbox"/>
¿El notebook quedó organizado, limpio y comentado?	<input type="checkbox"/>

## Cierre

La regresión es una excelente puerta de entrada al aprendizaje automático porque permite comprender una idea poderosa con herramientas sencillas: a partir de datos conocidos, un modelo puede estimar valores nuevos. En esta etapa importa mucho más entender el flujo completo que alcanzar una gran sofisticación técnica. Si el estudiante comprende qué entra al modelo, qué sale del modelo y cómo interpretar esa salida, ya ha dado un paso muy importante. Más adelante será mucho más fácil estudiar error, evaluación y generalización.