REGRESIÓN LINEAL

INTRODUCCIÓN

La regresión lineal es una herramienta poderosa que nos permite comprender y modelar sistemas reales a partir de datos. Analiza la relación entre variables independientes y una variable dependiente dentro de un conjunto de entrenamiento, con el objetivo de realizar predicciones precisas y confiables.

Gracias a su simplicidad y efectividad, la regresión lineal se utiliza en una gran variedad de aplicaciones positivas y útiles en la vida cotidiana. Por ejemplo, permite estimar la temperatura del día siguiente a partir de registros climáticos históricos, predecir el valor de una vivienda considerando sus características (ubicación, tamaño, servicios), o anticipar el rendimiento académico de un estudiante con base en hábitos de estudio y acompañamiento familiar, lo que puede ayudar a diseñar mejores estrategias educativas.

Debido a su claridad, interpretabilidad y amplio impacto práctico, la regresión lineal es uno de los algoritmos fundamentales y más utilizados en la inteligencia artificial y el aprendizaje automático, siendo frecuentemente el primer paso para comprender modelos más avanzados.

MODELO

Las predicciones pueden obtenerse de manera clara y efectiva utilizando la regresión lineal, una técnica que permite identificar la relación entre dos variables: una independiente y una dependiente.

Un ejemplo positivo de su aplicación es la estimación de la población mundial para el año 2030. En este caso, el tiempo funciona como la variable independiente, mientras que el número de habitantes representa la variable dependiente. Para descubrir la relación entre ambas, se requiere contar con información confiable y suficiente, como los datos históricos anuales de la población mundial. A este conjunto de información se le conoce como conjunto de entrenamiento.

Una vez que el modelo aprende esta relación a partir de los datos históricos, es posible realizar predicciones sobre años futuros. Estas estimaciones suelen ser muy cercanas a los valores reales, ya que la regresión lineal es un algoritmo de aprendizaje supervisado. Esto significa que el modelo aprende de manera iterativa: realiza predicciones iniciales, evalúa el error cometido y utiliza esa información para ajustar y mejorar continuamente sus resultados, logrando así predicciones cada vez más precisas.

En términos simples, la regresión lineal busca encontrar una línea o curva que se ajuste de la mejor manera posible a los datos del conjunto de entrenamiento, capturando la tendencia general del sistema. Este ajuste puede observarse gráficamente, donde la línea resultante representa el comportamiento aprendido a partir de los datos, como se ilustra en la Figura 1.

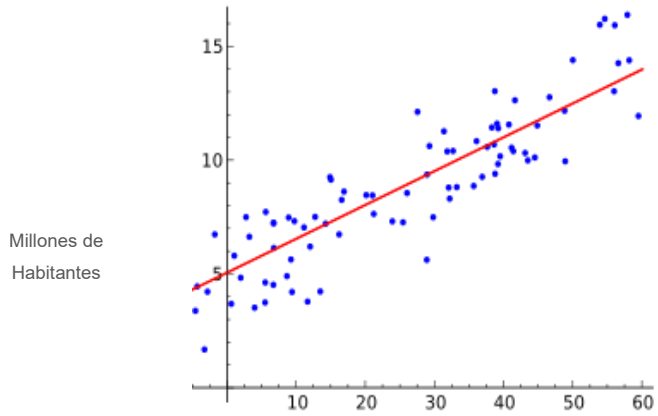

Fig. 1. Ejemplo de regresión lineal

En el eje horizontal tenemos los años (variable independiente) de 1910 a 1960, mientras que en el eje vertical tenemos millones de habitantes (variable dependiente). La regresión lineal trata de encontrar una recta tal que la distancia entre cada punto y ella misma sea lo más pequeña. La ecuación de la recta está definida por:

$$Y = mx + b \tag{1}$$

Donde m es la pendiente y b es la intersección con el eje vertical. En regresión lineal, la nomenclatura para los coeficientes se modifica como:

$$h_\theta(x) = \theta_0 + \theta_1 x \tag{2}$$

Donde $h_\theta(x)$ es predicción que realiza el modelo, θ_0 es la intersección con el eje vertical, mientras que θ_1 es la pendiente de la recta.

El objetivo de la regresión lineal es encontrar los parámetros θ_0 y θ_1 tal que la línea recta se ajuste a los datos de entrenamiento. Visto de otra manera, el objetivo es encontrar qué pendiente y qué intersección se deben utilizar para poder ajustar dicha recta.

Para establecer una notación, usaremos $x^{(i)}$ para denotar las variables de entrada (variable independiente), y $y^{(i)}$ para denotar la salida (variable dependiente) que queremos predecir, como por ejemplo la temperatura para el año 2030. El par de datos $(x^{(i)}, y^{(i)})$ se le conoce como conjunto de entrenamiento. El superíndice (i) indica un índice, y no tiene nada que ver con una ponencia. Llamaremos X a todos los elementos $x^{(i)}$, de la misma manera, llamaremos Y a todos los elementos $y^{(i)}$.

Como ya se mencionó anteriormente, el objetivo de la regresión lineal es, dado un conjunto de entrenamiento, aprender una función h tal que $X \rightarrow Y$, es decir, encontrar una función h la cual sea un buen predictor para el valor Y correspondiente. Esta función h se le conoce como hipótesis

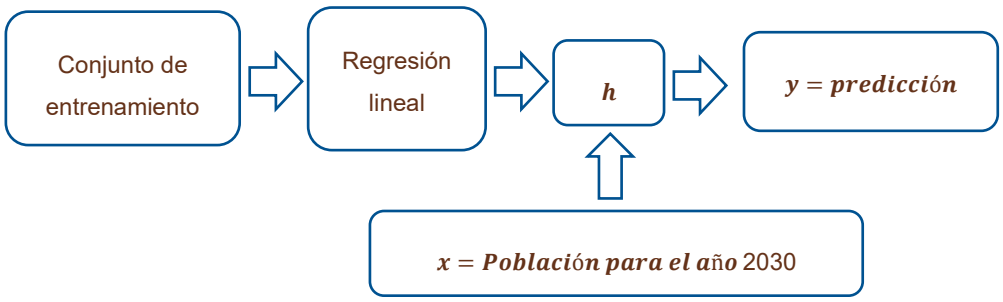

Fig. 2. Diagrama a bloques de un algoritmo de regresión lineal

FUNCIÓN DE COSTO

La función de costo (también conocida como función de pérdida) es un concepto fundamental en *machine learning* que mide qué tan bien (o qué tan mal) las predicciones de un modelo coinciden con los datos reales. En la regresión lineal, esta función cuantifica el error existente entre los valores predichos por el modelo y los valores verdaderos.

La función de costo más común utilizada en la regresión lineal es el Error Cuadrático Medio (Mean Squared Error, MSE). El MSE se define como:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{n=1}^{m} (h(x)_i - y_i)^2 \quad (3)$$

Donde $h(x)$ es el modelo de predicción y m es el número de muestras.

OPTIMIZACIÓN (GRADIENTE DESCENDENTE)

La optimización en machine learning se refiere al proceso de ajustar los parámetros del modelo con el objetivo de minimizar la función de costo. La meta es encontrar aquellos parámetros que hagan que las predicciones del modelo sean lo más precisas posible.

El descenso por gradiente es el algoritmo de optimización más utilizado para este propósito. Funciona como una especie de "brújula", que guía al modelo paso a paso hacia el punto más bajo del paisaje de la función de costo, es decir, hacia el mínimo error.

Analogía Intuitiva

Imagina que estás con los ojos vendados en una montaña (la función de costo) y tu objetivo es llegar al valle (el error mínimo). Para lograrlo, harías lo siguiente:

1. Sientes la pendiente bajo tus pies → calculas el gradiente
2. Das un paso cuesta abajo → actualizas los parámetros en la dirección de mayor descenso
3. Repites el proceso → continúas hasta llegar al punto más bajo (convergencia)

De esta manera, el algoritmo actualiza los parámetros de forma iterativa utilizando la siguiente regla:

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \tag{4}$$

Donde:

θ_j: Parámetro que se optimizará (por ejemplo, θ_0, θ_1)

α: Tasa (ritmo) de aprendizaje (controla el tamaño del paso)

$\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$: Derivada parcial (gradiente) de la función de costo con respecto a los parámetros θ_j

Las derivadas parciales se calculan de la siguiente manera:

$$\frac{\partial}{\partial \theta_j} J(\theta_0) = \frac{1}{m} \sum_i^m (h(x)_i - y_i) \tag{5}$$

$$\frac{\partial}{\partial \theta_j} J(\theta_1) = \frac{1}{m} \sum_i^m (h(x)_i - y_i) x_i \quad (6)$$

Para facilitar el entendimiento de la regresión lineal, vamos a realizar un pequeño ejemplo. Supongamos que hemos colectado información de las notas de estudiantes en una materia y las horas de estudio que invirtieron. El conjunto de datos se ve de la siguiente manera:

Tabla 1. Conjunto de datos Horas Estudio vs Calificación

x (Horas de Estudio)	y (Calificación)
1	**20**
2	**40**
3	**50**
4	**70**

Con base en la colección de datos, implementaremos la regresión lineal para predecir, por ejemplo, la nota de un estudiante si invierte 5 horas de estudio. Si graficamos nuestro conjunto de datos de la Tabla 1, obtenemos la siguiente gráfica:

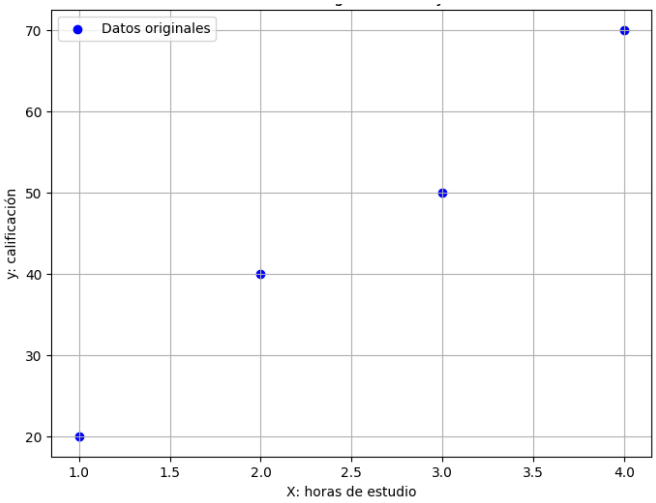

Fig. 3. Visualización del conjunto de datos

El primer paso es inicializar nuestros parámetros de manera aleatoria, supongamos que $\theta_0 = 0$ y $\theta_1 = 10$. Además, inicializaremos el hiper parámetro alfa como $\alpha = 0.1$. Con la inicialización de los parámetros, nuestra hipótesis se convierte en:

$$h_\theta(x) = \theta_0 + \theta_1 x$$

$$h_\theta(x) = 10x$$

El siguiente paso es realizar el cálculo de la predicción con estos parámetros. Para cada valor de x calcularemos nuestra hipótesis $h_\theta(x)$:

Tabla 2. Variables predictoras, objetivo y predicciones

x (Horas de Estudio)	y (Calificación)	$h(x)$ *(Predicción*
1	**20**	**10**
2	**40**	**20**
3	**50**	**30**
4	**70**	**40**

Si graficamos las predicciones se pueden visualizar como:

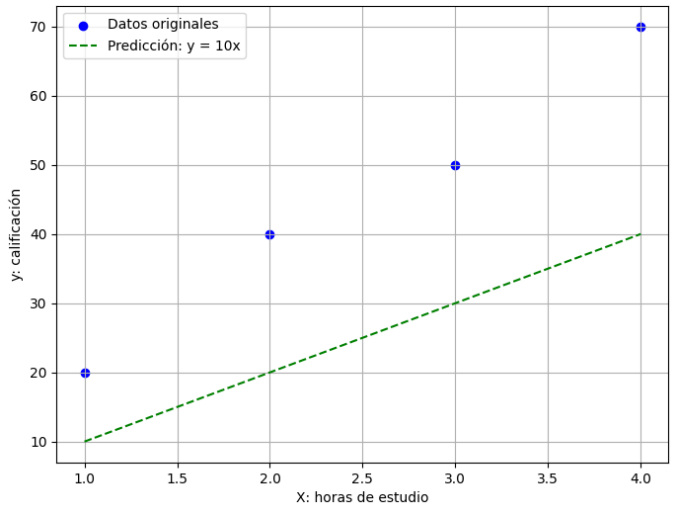

Fig. 4. Predicción (línea verde punteada) de la primera iteración

Como se observa en la Figura 5, las predicciones (línea roja) no se ajustan adecuadamente a los cuatro puntos de datos (en azul). Esta diferencia entre los valores reales y los valores predichos se conoce como error de predicción.

Para mejorar la calidad de la predicción, el primer paso es medir ese error de manera cuantitativa. Para ello, implementaremos la función de costo de Error Cuadrático Medio (MSE), la cual nos permite evaluar qué tan lejos están las predicciones del modelo respecto a los valores reales y, con base en ello, ajustar el modelo para obtener resultados más precisos.

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{n=1}^{m} (h(x)_i - y_i)^2$$

$$J(\theta_0, \theta_1) = \frac{1}{2 \times 4} \sum_{n=1}^{4} ([10, 20, 30, 40] - [20, 40, 50, 70])^2$$

$$J(\theta_0, \theta_1) = \frac{1}{8} \sum_{n=1}^{4} ([-10, -20, -20, -30])^2$$

$$J(\theta_0, \theta_1) = \frac{1}{8} \sum_{n=1}^{4} ([100, 400, 400, 900])^2$$

$$J(\theta_0, \theta_1) = \frac{1}{8} ([1800])$$

$$J(\theta_0, \theta_1) = 225$$

El valor del error es 225, lo cual confirma lo que se observa en la Figura 5: el modelo de regresión lineal está realizando predicciones poco precisas.

El siguiente paso es la optimización, en donde se calculan nuevos parámetros para que, en la siguiente iteración, el modelo produzca una predicción mejor y el error sea menor que 225. Para minimizar el error, implementaremos el algoritmo de descenso por gradiente. Como primer paso, es necesario calcular las derivadas de la función de costo con respecto a los parámetros del modelo.

En el caso del parámetro θ_0, se tiene lo siguiente:

$$\frac{\partial}{\partial\theta_j}J(\theta_0) = \frac{1}{m}\sum_{i}^{m}(h(x)_i - y_i)$$

$$\frac{\partial}{\partial\theta_j}J(\theta_0) = \frac{1}{4}\sum_{i}^{4}([10,20,30,40] - [20,40,50,70])$$

$$\frac{\partial}{\partial\theta_j}J(\theta_0) = \frac{1}{4}\sum_{i}^{4}([-10,-20,-20,-30])$$

$$\frac{\partial}{\partial\theta_j}J(\theta_0) = \frac{1}{4}([-80])$$

$$\frac{\partial}{\partial\theta_j}J(\theta_0) = -20$$

Ahora para θ_1:

$$\frac{\partial}{\partial\theta_j}J(\theta_1) = \frac{1}{m}\sum_{i}^{m}(h(x)_i - y_i)x_i$$

$$\frac{\partial}{\partial\theta_j}J(\theta_1) = \frac{1}{4}\sum_{i}^{4}([10,20,30,40] - [20,40,50,70])[1,2,3,4]$$

$$\frac{\partial}{\partial\theta_j}J(\theta_1) = \frac{1}{4}\sum_{i}^{4}([-10,-20,-20,-30])[1,2,3,4]$$

$$\frac{\partial}{\partial\theta_j}J(\theta_1) = \frac{1}{4}\sum_{i}^{4}([-10,-40,-60,-120])$$

$$\frac{\partial}{\partial\theta_j}J(\theta_1) = \frac{1}{4}([-230])$$

$$\frac{\partial}{\partial\theta_j}J(\theta_1) = -57.5$$

Ahora que tenemos las derivadas parciales de la función de costo, podemos calcular los nuevos valores para θ_0 y θ_1. Para θ_0 tenemos que:

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$

$$\theta_0 := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0)$$

$$\theta_0 := 0 - (0.1) \times [-20]$$

$$\theta_0 := 2$$

Ahora necesitamos encontrar el nuevo valor de θ_1:

$$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_1)$$

$$\theta_1 := 10 - (0.1) \times (-57.5)$$

$$\theta_1 := 10 + 5.75$$

$$\theta_1 = 15.75$$

Ahora que ya tenemos los nuevos valores de θ_j volvemos a iterar hasta obtener un valor de error muy cercano a cero. En la figura 5 se muestran los valores del error, θ_0 y θ_1 en cada iteración.

```
Iteración 1: Error = 225.00, theta_0 = 2.00, theta_1 = 15.75
Iteración 2: Error = 9.11, theta_0 = 2.36, theta_1 = 16.69
Iteración 3: Error = 3.22, theta_0 = 2.45, theta_1 = 16.83
Iteración 4: Error = 3.04, theta_0 = 2.50, theta_1 = 16.84
Iteración 5: Error = 3.02, theta_0 = 2.54, theta_1 = 16.84
Iteración 6: Error = 3.01, theta_0 = 2.58, theta_1 = 16.82
Iteración 7: Error = 2.99, theta_0 = 2.61, theta_1 = 16.81
Iteración 8: Error = 2.98, theta_0 = 2.65, theta_1 = 16.80
Iteración 9: Error = 2.96, theta_0 = 2.68, theta_1 = 16.79
Iteración 10: Error = 2.95, theta_0 = 2.72, theta_1 = 16.78
Iteración 11: Error = 2.93, theta_0 = 2.75, theta_1 = 16.76
Iteración 12: Error = 2.92, theta_0 = 2.79, theta_1 = 16.75
Iteración 13: Error = 2.91, theta_0 = 2.82, theta_1 = 16.74
Iteración 14: Error = 2.90, theta_0 = 2.85, theta_1 = 16.73
Iteración 15: Error = 2.89, theta_0 = 2.88, theta_1 = 16.72
Iteración 16: Error = 2.87, theta_0 = 2.92, theta_1 = 16.71
Iteración 17: Error = 2.86, theta_0 = 2.95, theta_1 = 16.70
Iteración 18: Error = 2.85, theta 0 = 2.98, theta 1 = 16.69
```

Fig. 5. Valores de error y θ_j por iteración

Después de 345 iteraciones se obtienen los valores $\theta_0 = 5$ y $\theta_1 = 16$ los cuales producen un error de 2.5. La ecuación de la recta obtenida por estos valores es $h(x) = 5 + 16x$. La predicción se muestra en la Figura 6.

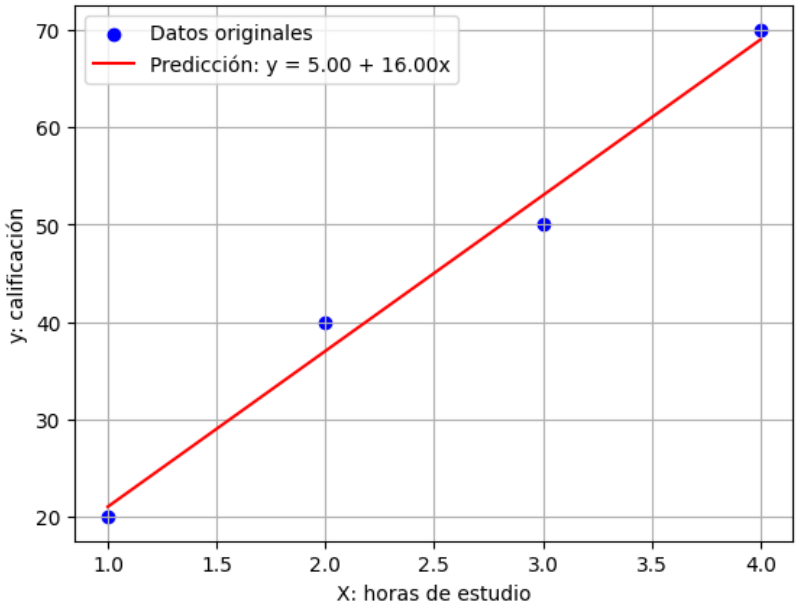

Fig. 6. Predicción final con un error cercano a cero