Inteligencia Artificial

Regresión

Lineal y Logística

Contenido

Regresión

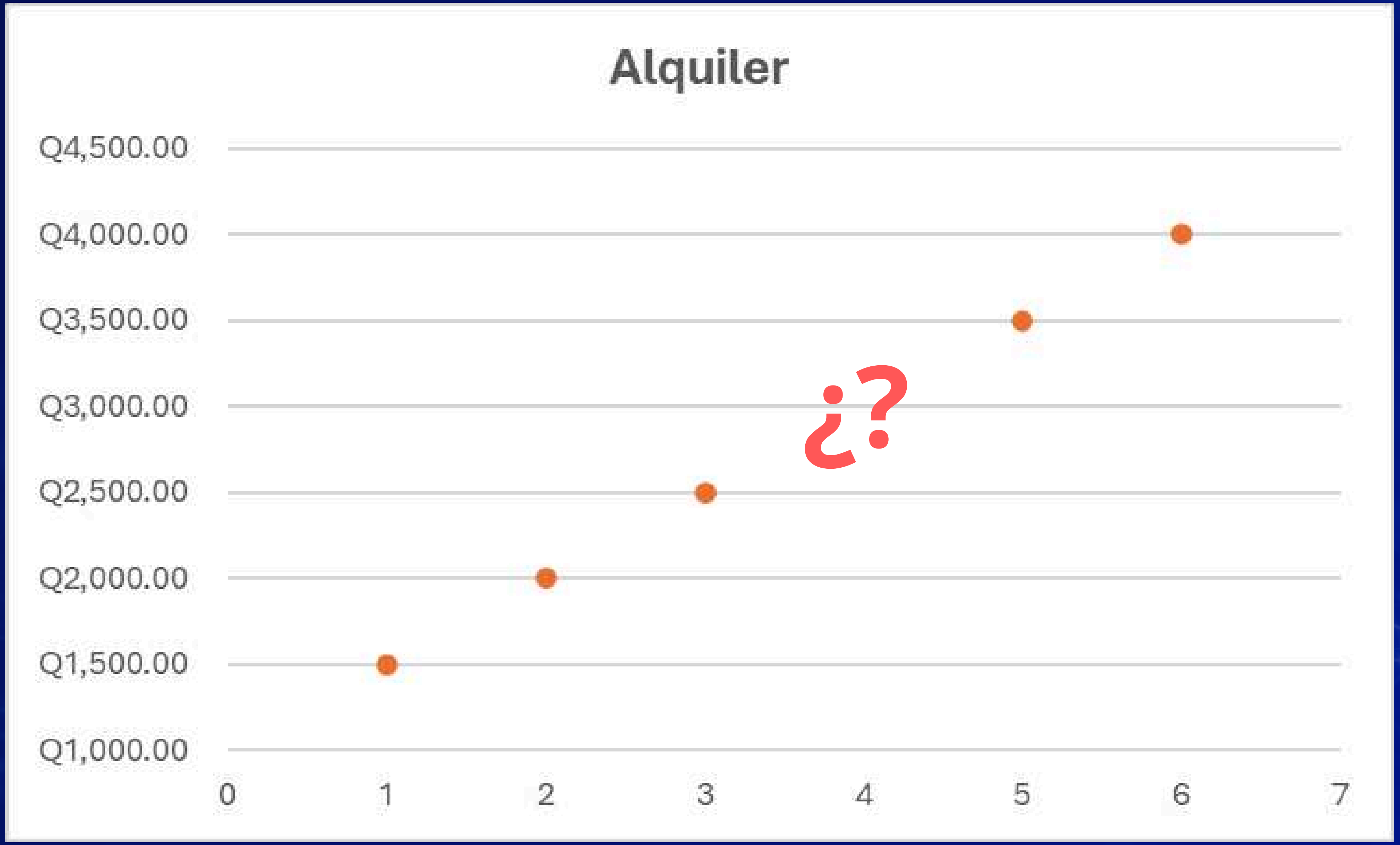
- Introducción Regresión
- Regresión Lineal
  - Fundamentos Matemáticos
  - Evaluación de modelos de regresión
- Regresión Logística
  - Introducción a la Clasificación y Regresión Logística
  - Función Sigmoide
- Optimización
  - Gradiente descendente
  - Regularización

Regresión

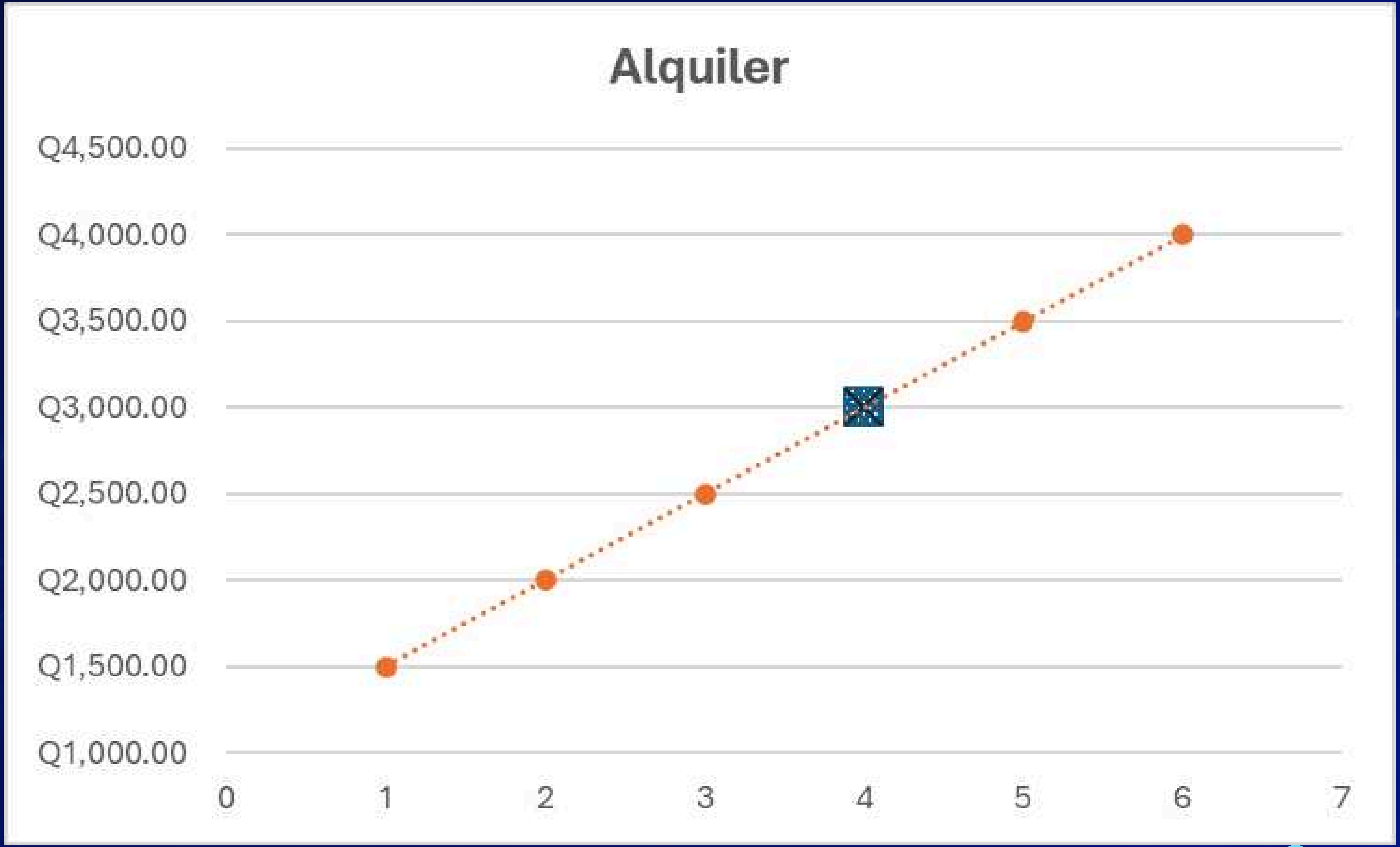
Predecir valores continuos con base en datos previos.

Ejemplo: Valor de alquiler de casas en función del número de habitaciones

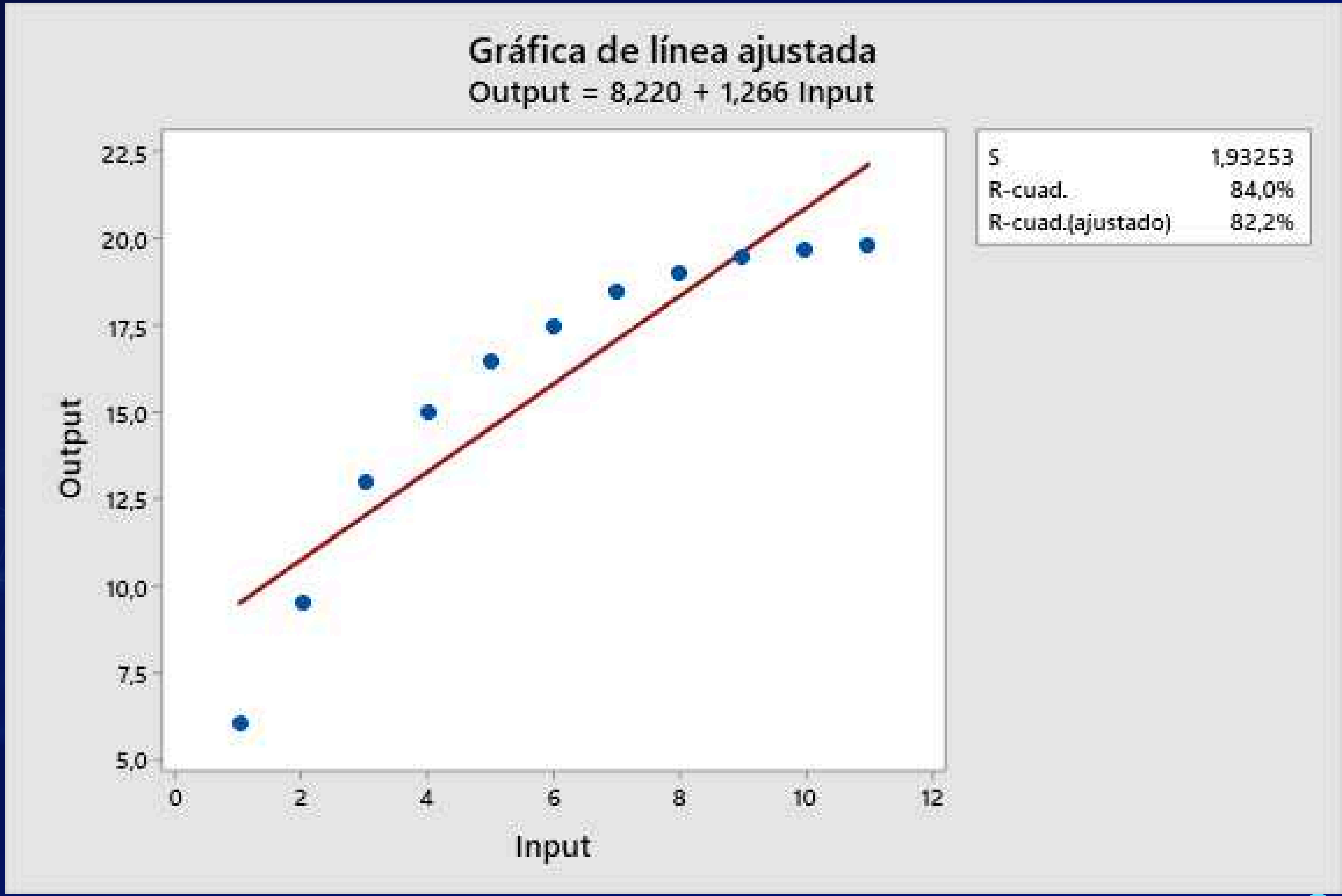
Habitaciones (*x*)	Alquiler (*y*)
1	1500 Q
2	2000 Q
3	2500 Q
4	¿?
5	3500 Q
6	4000 Q



Ejemplo: Valor de alquiler de casas en función del número de habitaciones

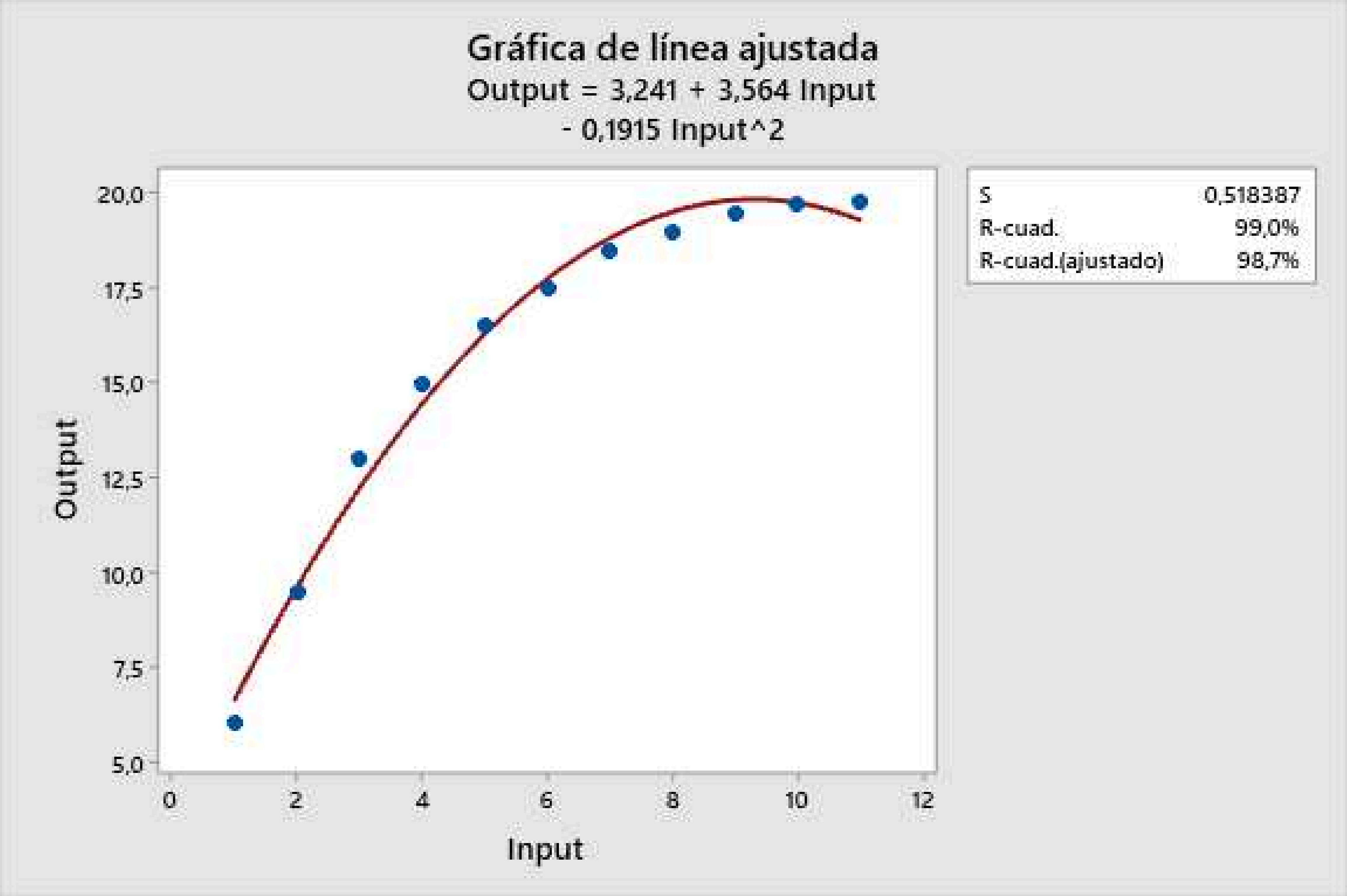


Ejemplo



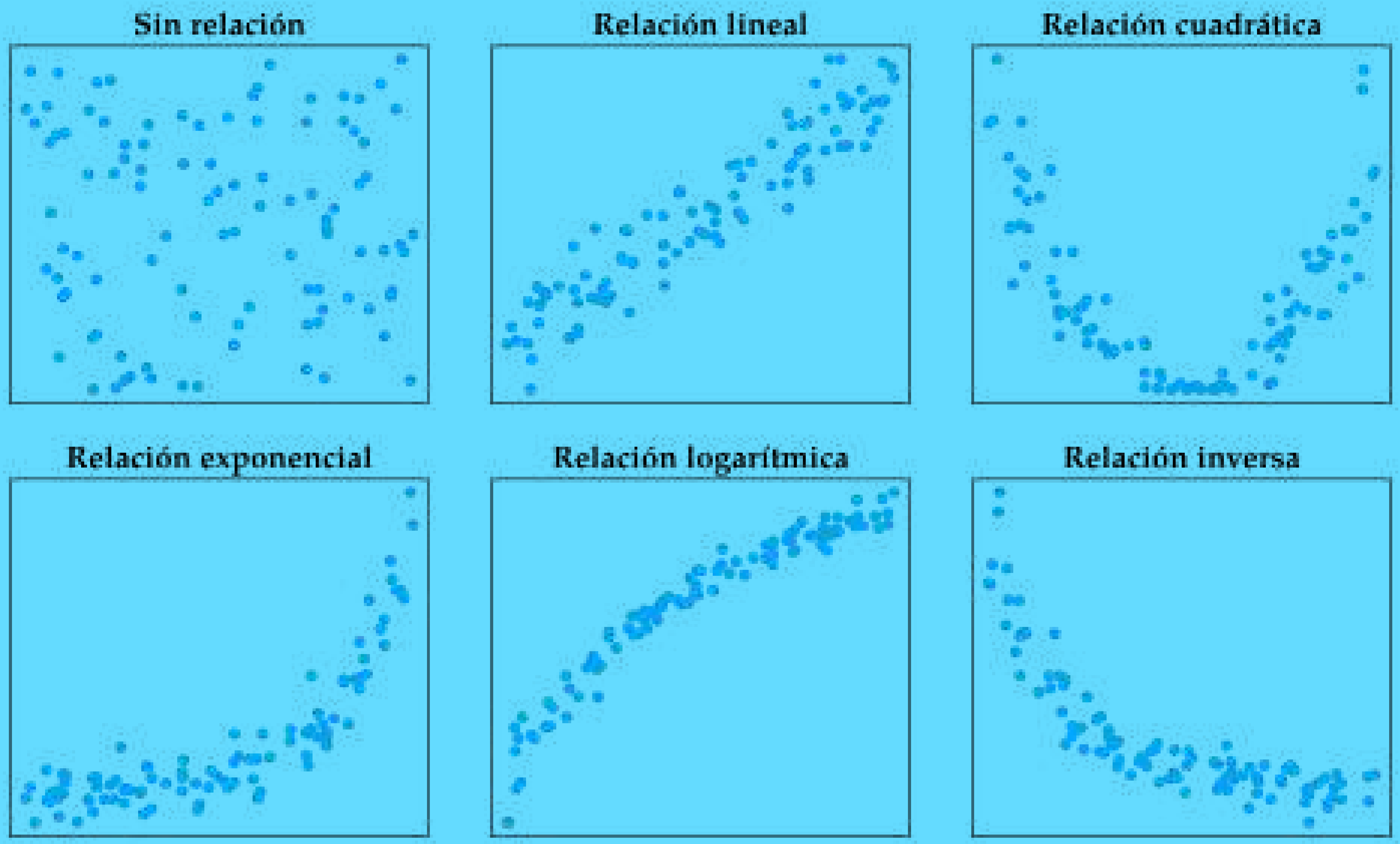
Gráfica de línea ajustada
Output = 8,220 + 1,266 Input
S 1,93253
R-cuad. 84,0%
R-cuad.(ajustado) 82,2%
Output
22,5
20,0
17,5
15,0
12,5
10,0
7,5
5,0
0
2
4
6
8
10
12
Input

Ejemplo



Gráfica de línea ajustada
Output = 3,241 + 3,564 Input
- 0,1915 Input^2
S 0,518387
R-cuad. 99,0%
R-cuad.(ajustado) 98,7%
Output
20,0
17,5
15,0
12,5
10,0
7,5
5,0
0
2
4
6
8
10
12
Input

Otros Modelos

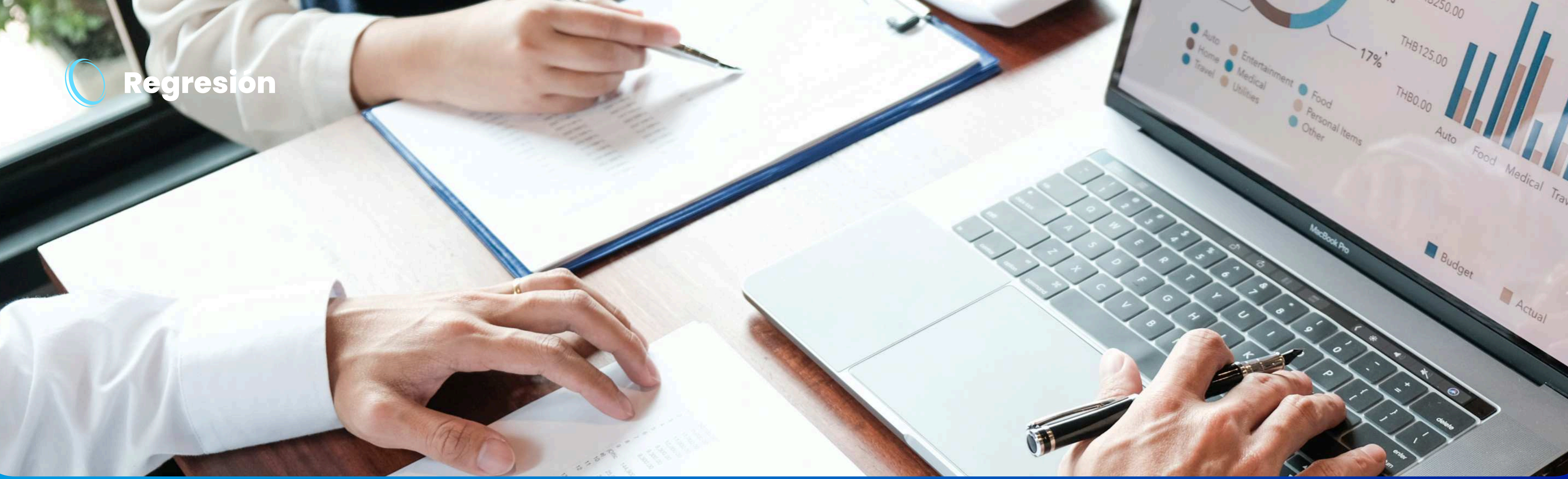


Modelos Lineales vs Modelos No Lineales

Característica	Modelos Lineales	Modelos No Lineales
Relación entrada-salida	Lineal	No lineal
Fórmula general	$y = w_1x_1 + w_2x_2 + \ldots + b$	$y = f(x)$, donde f es no lineal
Ejemplo simple	Regresión lineal	Regresión polinómica
Interpretabilidad	Fácil	Difícil
Computación	Rápida	Lenta
Flexibilidad	Baja	Alta
Tendencia al Overfitting	Menor	Mayor

Diferencia entre Clasificación y Regresión

- **Clasificación:** La salida es booleana (0 o 1) o pertenece a una clase discreta.
  - *Ejemplo*: "¿Es spam o no?"
- **Regresión:** La salida es numérica y continua.
  - *Ejemplo*: "¿Cuál será el precio de una casa?"



Problema de Regresión

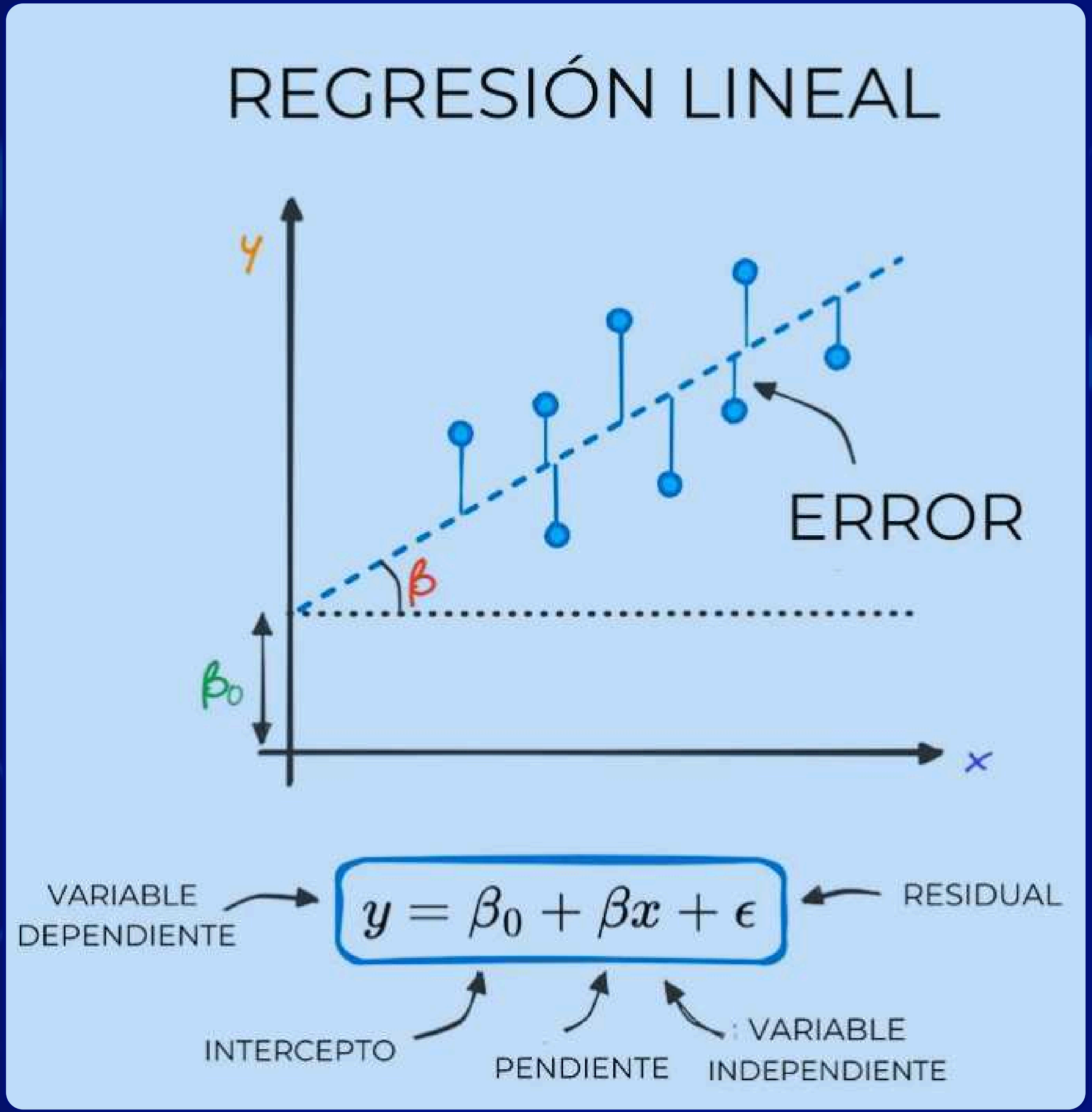
En regresión, queremos encontrar una función *f(x)* que relacione los datos de entrada *x* con las salidas reales *r*.

- Si no hay ruido en los datos, es simplemente un problema de interpolación (ajustar la función exacta).
- Si hay ruido, se debe a variables ocultas que no observamos. Entonces, modelamos la relación como:

$$r_t = f(x_t) + \epsilon$$

Donde ϵ representa el ruido o factores externos que no podemos medir.

Regresión Lineal Simple



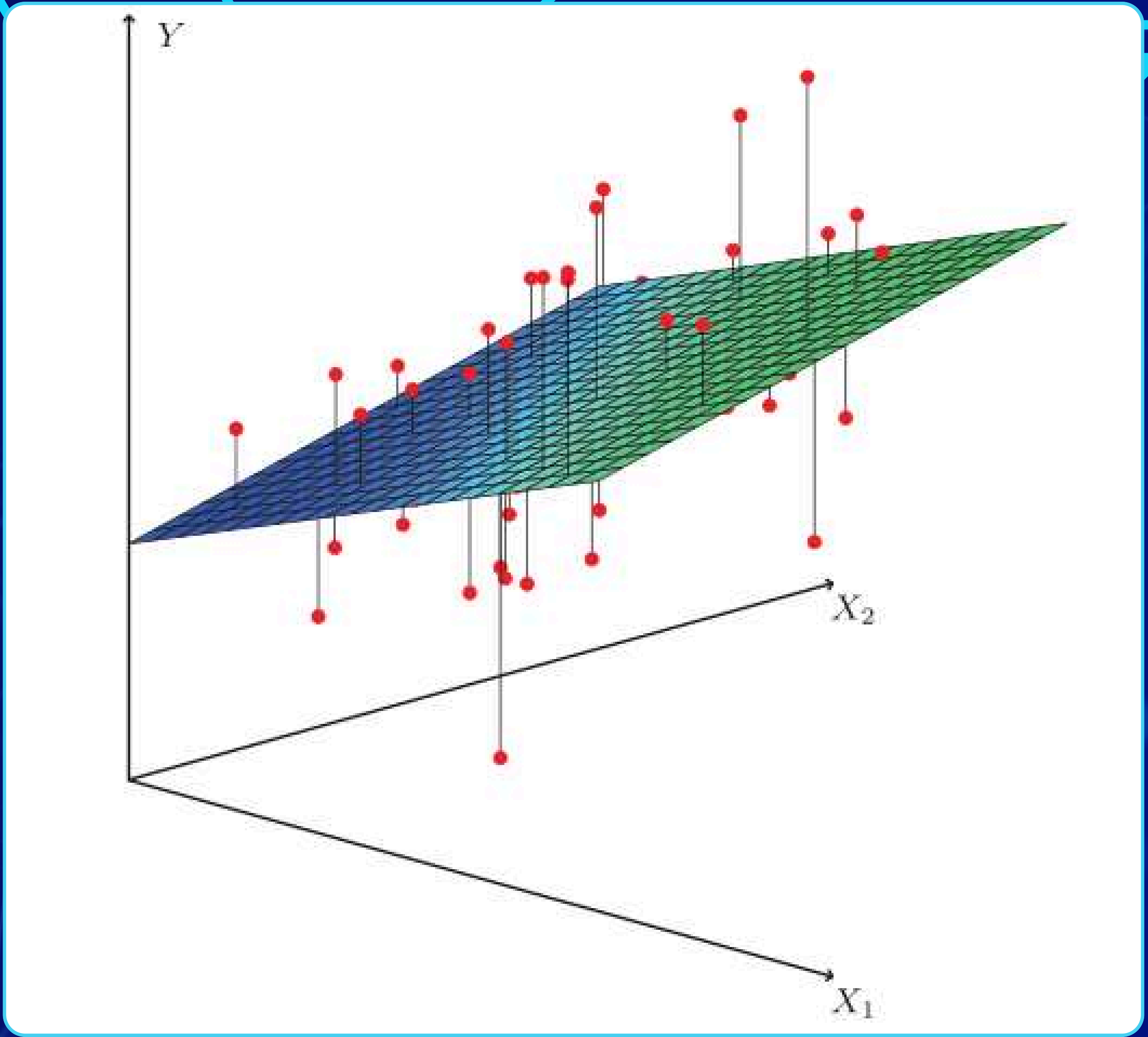
Objetivo: Encontrar β0 y β1 que minimicen el error

Ecuación:

$$y = \beta_0 + \beta_1 x + \epsilon$$

Parámetros (β0 y β1): Intercepto y pendiente.

Regresión Lineal Múltiple



Ecuación:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \epsilon$$

- *y*: Variable dependiente (objetivo).
- *β0*: Intercepto.
- *β1,β2,...,βn*: Coeficientes de las variables independientes *x1,x2,...,xn*.
- *ϵ*: Término de error (residuo).

Regresión Lineal Múltiple

Un modelo lineal se simplifica como:

$$y = X\beta + \epsilon$$

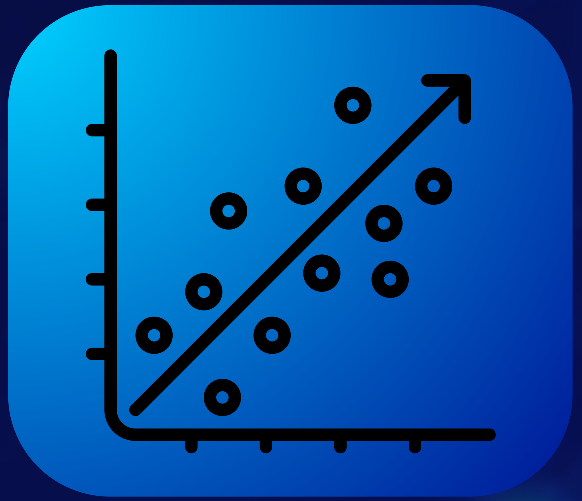
Donde $y = (y_1, \ldots, y_n)^T$ es un vector de outputs, ***X*** es una n × k matriz, $\beta = (\beta_1, \ldots, \beta_n)^T$ es un vector de parámetros

$$X = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{i,k-1} \\ 1 & x_{21} & x_{22} & \cdots & x_{2,k-1} \\ \vdots & \vdots & & & \\ 1 & x_{n1} & x_{n2} & \cdots & x_{n,k-1} \end{pmatrix}$$

$$\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_{k-1})^T$$

ϵ es un vector de errores aleatorios.

Supuestos de la Regresión Lineal Simple



Linealidad

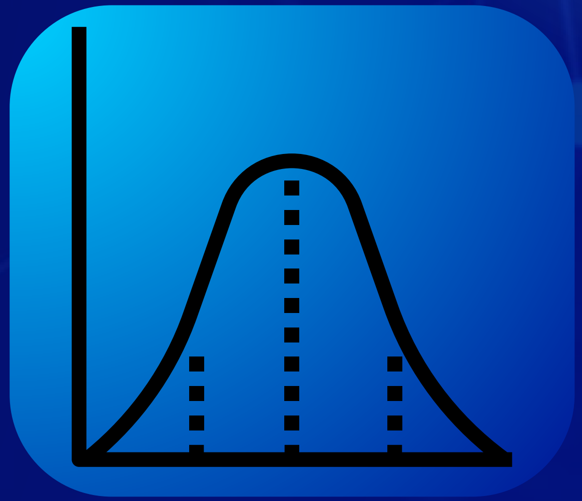
Esto significa que debe existir una línea recta que pueda trazarse a través de los puntos de datos.

Independencia

Cada observación en el conjunto de datos debe ser independiente de las demás. Esto significa que el valor de una observación no debe influir en el valor de otra.

Homoscedasticidad

La variabilidad de los errores debe ser constante en todos los niveles de las variables independientes. Esto significa que la cantidad de la variable independiente no debe afectar cuánto varían los errores. Si la variabilidad de los errores no es constante, la regresión lineal no será precisa.



Normalidad

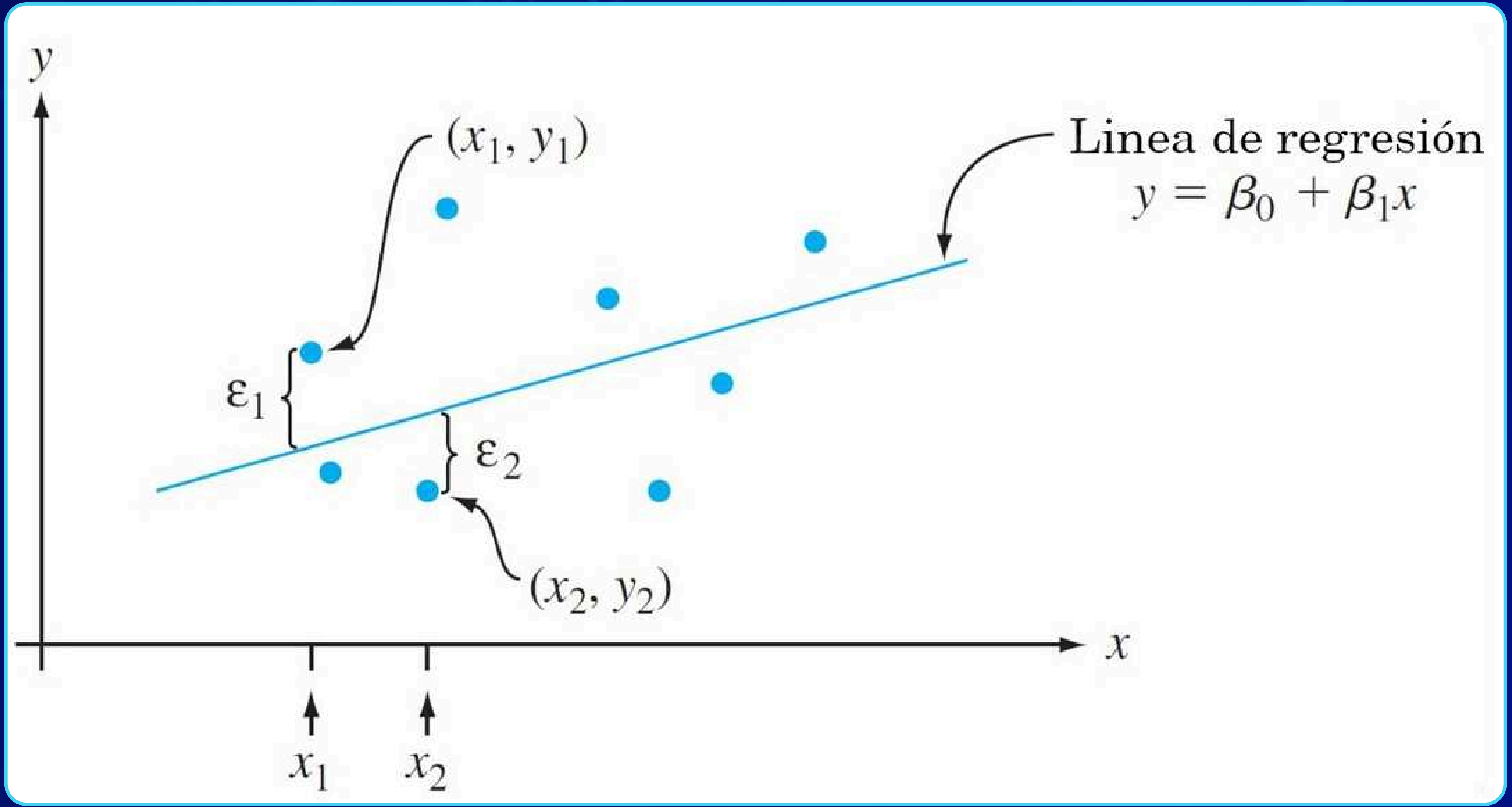
Los residuos (errores) deben seguir una distribución normal, que es una curva en forma de campana.

Si no se dan estos supuestos, la regresión lineal no será precisa.

Función de Costo

La función de costo (cost function) es una función matemática que el modelo minimiza durante el entrenamiento.

Su propósito es medir el error entre las predicciones del modelo y los valores reales de los datos de entrenamiento.

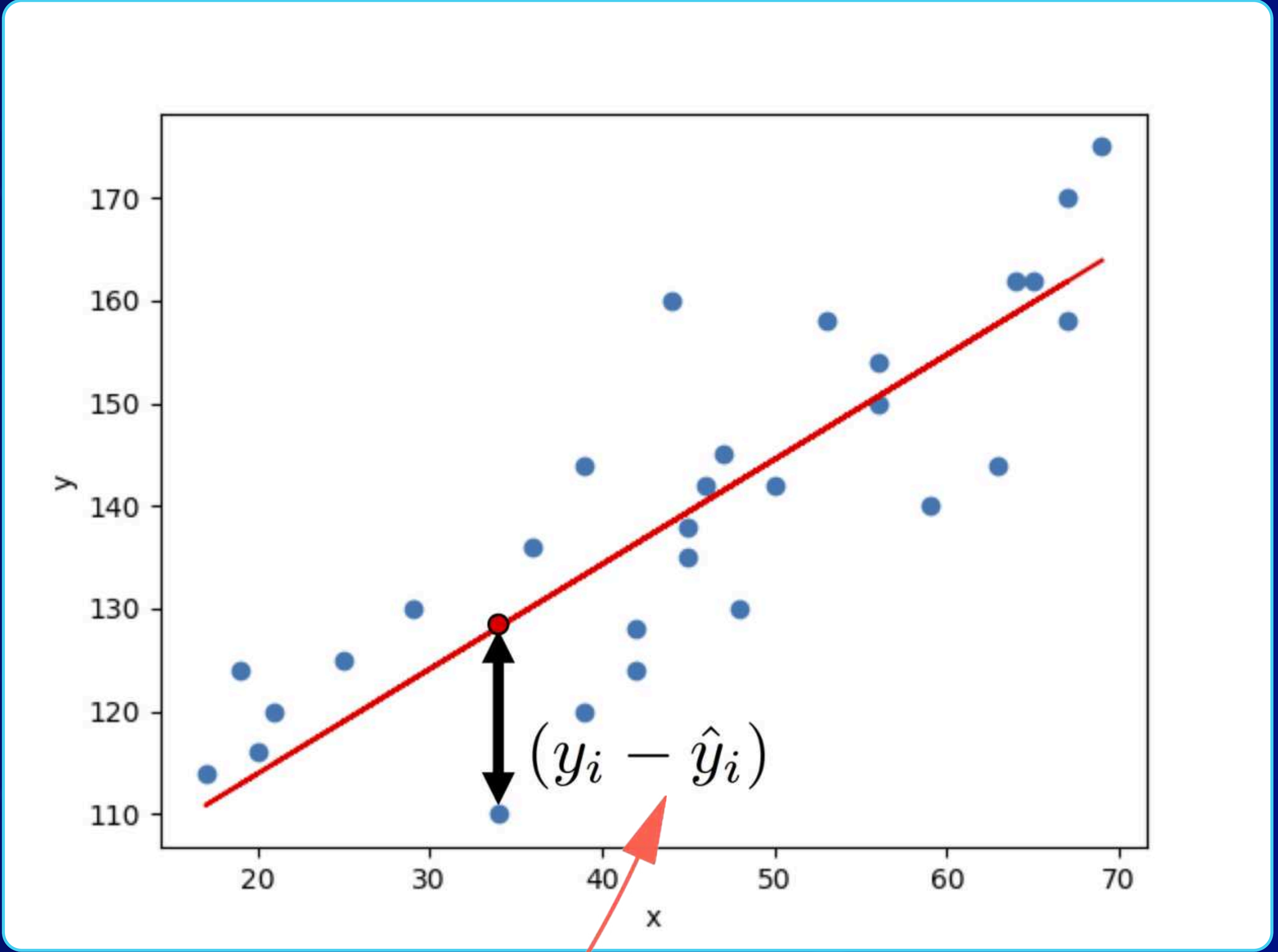


Función de Costo

$$J(w) = \frac{1}{N} \sum_{i=1}^{N} L(y_i, \hat{y}_i)$$

Donde:

- $J(w)$ es la función de costo.
- N es el número total de ejemplos de entrenamiento.
- $L(y_i, \hat{y}_i)$ es la función de pérdida (cómo se mide el error para un solo dato).
- y_i es el valor real.
- $\hat{y}_i$ es el valor predicho.



¡Se vuelve un problema de Optimización!

Funciones de Pérdida

Error Cuadrático (Squared Error Loss)

$$L(y_i, \hat{y}_i) = (y_i - \hat{y}_i)^2$$

- Penaliza los errores grandes de forma cuadrática.
- Base del Error Cuadrático Medio (MSE).

Error Absoluto (Absolute Error Loss)

$$L(y_i, \hat{y}_i) = |y_i - \hat{y}_i|$$

- Penaliza errores de forma lineal.
- Más robusto a valores atípicos que el error cuadrático.

Funciones de Pérdida

Error Huber (Huber Loss)

$$L(y_i, \hat{y}_i) = \begin{cases} \frac{1}{2}(y_i - \hat{y}_i)^2 & \text{for } |y_i - \hat{y}_i| \le \delta, \\ \delta|y_i - \hat{y}_i| - \frac{1}{2}\delta^2 & \text{for } |y_i - \hat{y}_i| > \delta. \end{cases}$$

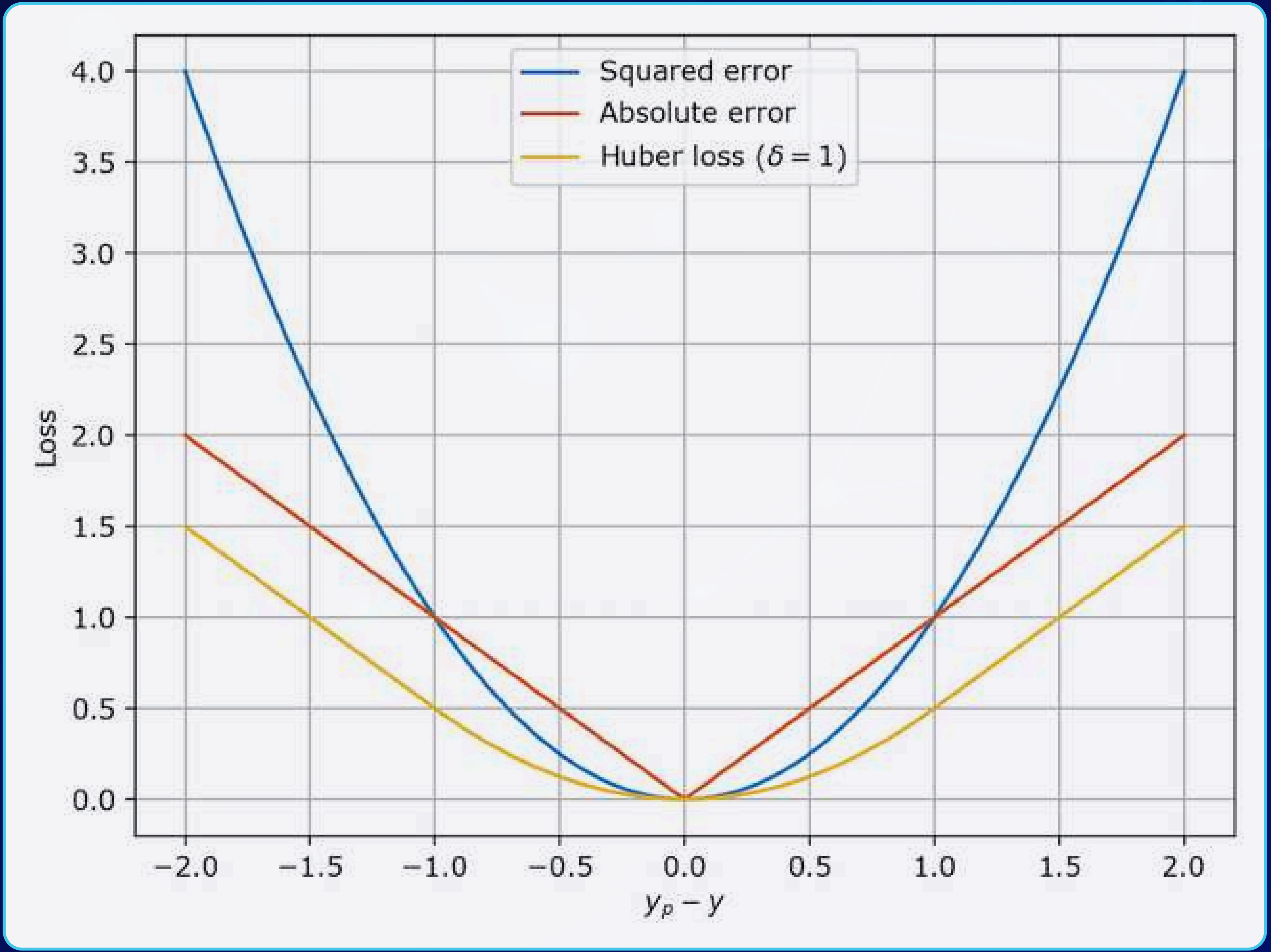
- Combina el MSE y MAE para ser menos sensible a valores atípicos.
- Para pequeños errores, actúa como MSE. Para errores grandes, como MAE.

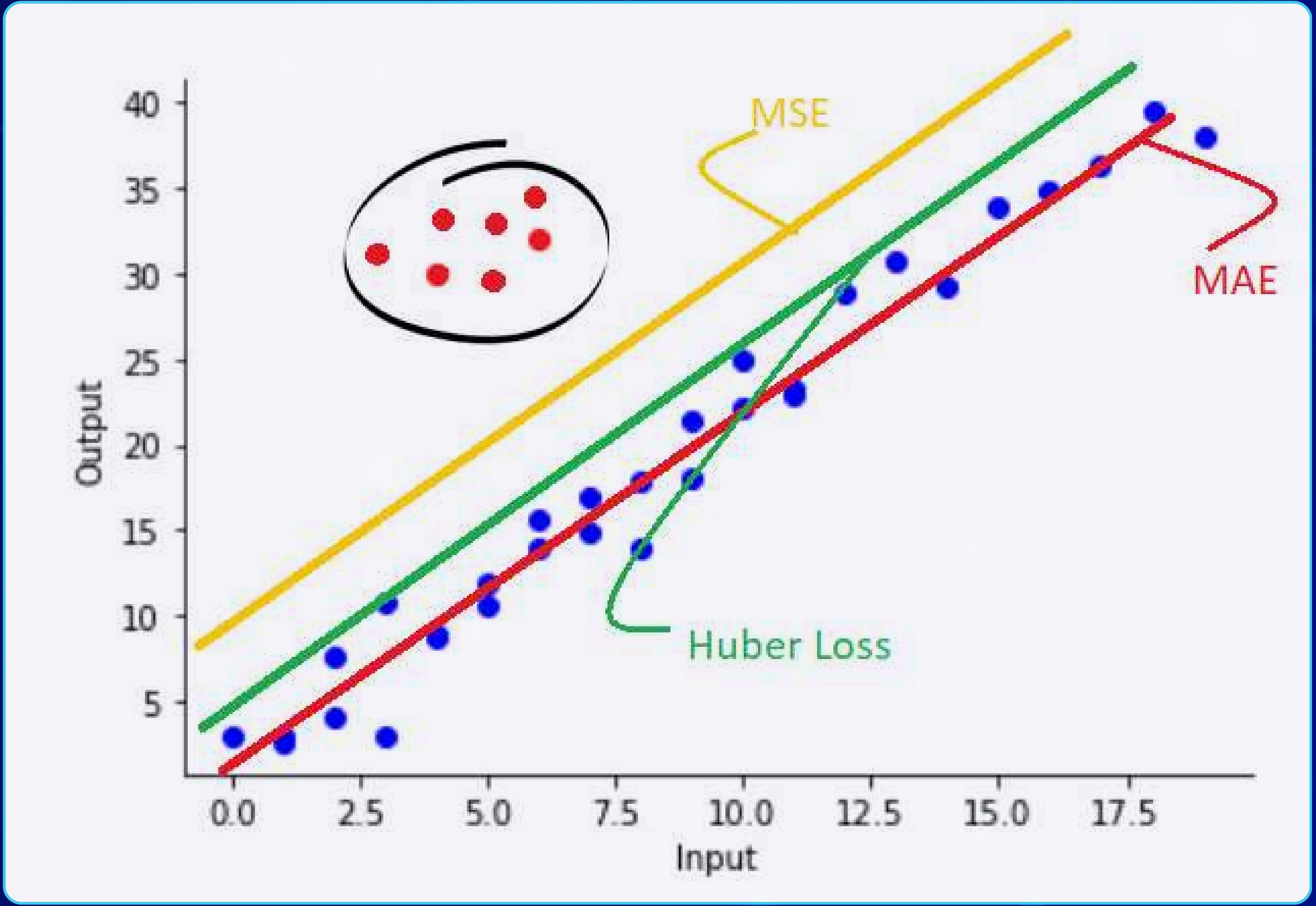
Error Log-Cosh (Log-Cosh Loss)

$$L(y_i, \hat{y}_i) = \log\left(\cosh(y_i - \hat{y}_i)\right)$$

- Similar al Error Huber, pero más suave y diferenciable en todo su dominio.
- Se comporta como MSE para errores pequeños y MAE para errores grandes.

Funciones de Pérdida





Mínimos Cuadrados Ordinarios

Es un método de estimación utilizado para encontrar los parámetros de un modelo de regresión lineal.

La Suma de los Errores al Cuadrado es una función de costo que se utiliza para evaluar qué tan bien se ajusta el modelo a los datos. Cuanto menor sea la SSE, mejor será el ajuste del modelo.

$$J(w) = \sum_{i=1}^{N} \left(y_i - \hat{y}_i\right)^2$$

Mínimos Cuadrados Ordinarios

El objetivo es encontrar los valores de β0 y β1 que minimicen la suma de los errores al cuadrado (Mínimos Cuadrados Ordinarios, OLS):

$$S = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

Reemplazamos *yi* por la ecuación de la recta:

$$S = \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_i))^2$$

Mínimos Cuadrados Ordinarios

La Suma de los Errores al Cuadrado es una función de costo que se utiliza para evaluar qué tan bien se ajusta el modelo a los datos. Cuanto menor sea la SSE, mejor será el ajuste del modelo.

$$S = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_i))^2$$

Seguidamente debemos derivar la ecuación con respecto a β0 y β1

Mínimos Cuadrados Ordinarios

Ahora tenemos las ecuaciones finales:

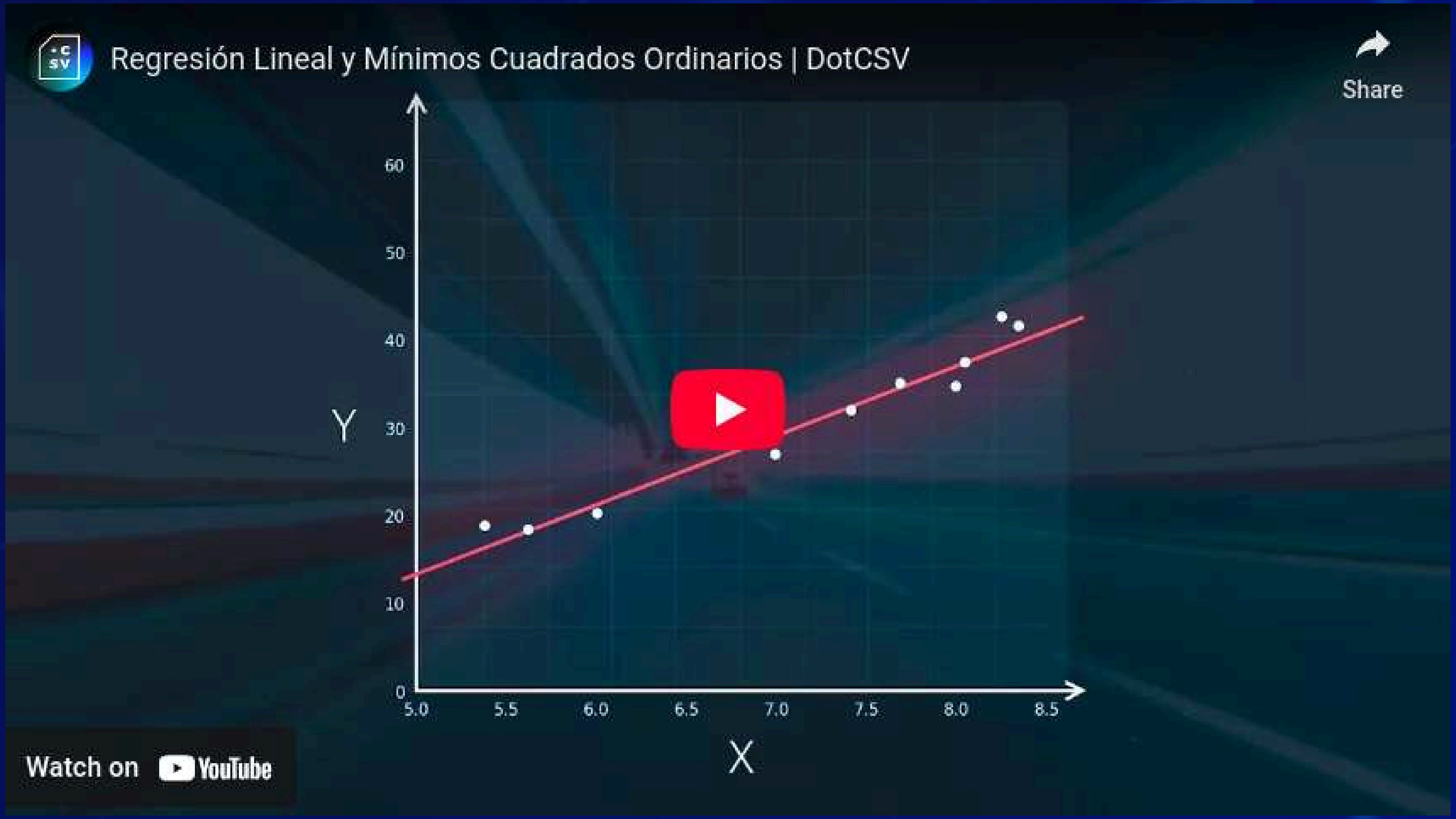
$$\beta_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$\beta_0 = \bar{y} - \beta_1 \bar{x}$$

Donde x¯ y y¯ son los promedios de x e y, respectivamente.

- β1 representa la pendiente de la recta: mide cómo cambia *y* por cada unidad de cambio en *x*.
- β0 es el intercepto: el valor esperado de *y* cuando *x=0*

Resumen



Regresión Lineal y Mínimos Cuadrados Ordinarios | DotCSV
Share
Y
X
60
50
40
30
20
10
0
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5
Watch on YouTube

Ejemplo Regresión Lineal Simple

Habitaciones (*x*)	Alquiler (*y*)
1	1500 Q
2	2000 Q
3	2500 Q
4	3000 Q
5	3500 Q
6	4000 Q

Paso 1: Relación entre las variables

Queremos encontrar una ecuación lineal que se ajuste a estos datos:

$$y = \beta_0 + \beta_1 x + \epsilon$$

Donde:

- *Y* es el alquiler (valor dependiente).
- X es el número de habitaciones (valor independiente).
- β0 es la intersección (cuando X=0).
- β1 es la pendiente (cambio en Y por cada unidad de X).
- ϵ representa el término de error.

Ejemplo

Paso 2: Cálculo de promedios

Primero calculamos los promedios de X y Y:

$$\bar{X} = \frac{1 + 2 + 3 + 4 + 5 + 6}{6} = 3.5$$

$$\bar{Y} = \frac{1500 + 2000 + 2500 + 3000 + 3500 + 4000}{6} = 2750$$

Ejemplo

Paso 3: Cálculo de la pendiente (β1)

Para calcular la pendiente, usamos la fórmula:

$$\beta_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

a) Calculamos el numerador

$$\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

b) Calculamos el denominador

$$\sum_{i=1}^{n}(x_i - \bar{x})^2$$

c) Calculamos β1

Ejemplo

Paso 3a:

X_i	Y_i	$(X_i - \bar{X})$	$(Y_i - \bar{Y})$	$(X_i - \bar{X})(Y_i - \bar{Y})$
1	1500	$1 - 3.5 = -2.5$	$1500 - 2750 = -1250$	$(-2.5)(-1250) = 3125$
2	2000	$2 - 3.5 = -1.5$	$2000 - 2750 = -750$	$(-1.5)(-750) = 1125$
3	2500	$3 - 3.5 = -0.5$	$2500 - 2750 = -250$	$(-0.5)(-250) = 125$
4	3000	$4 - 3.5 = 0.5$	$3000 - 2750 = 250$	$(0.5)(250) = 125$
5	3500	$5 - 3.5 = 1.5$	$3500 - 2750 = 750$	$(1.5)(750) = 1125$
6	4000	$6 - 3.5 = 2.5$	$4000 - 2750 = 1250$	$(2.5)(1250) = 3125$

$$\sum_{i=1}^{6}(x_i - \bar{x})(y_i - \bar{y}) = 3125 + 1125 + 125 + 125 + 1125 + 3125 = 8750$$

Ejemplo

Paso 3b:

X_i	$(X_i - \bar{X})^2$
1	$(-2.5)^2 = 6.25$
2	$(-1.5)^2 = 2.25$
3	$(-0.5)^2 = 0.25$
4	$(0.5)^2 = 0.25$
5	$(1.5)^2 = 2.25$
6	$(2.5)^2 = 6.25$

$$\sum_{i=1}^{6}(x_i - \bar{x})^2 = 6.25 + 2.25 + 0.25 + 0.25 + 2.25 + 6.25 = 17.5$$

Ejemplo

Paso 3c:

$$\beta_1 = \frac{8750}{17.5}$$

$$\beta_1 = 500$$

Ejemplo

Paso 4: Cálculo de la intersección (β0)

Identificando $\beta_1 = 500$, resolvemos:

$$\beta_0 = \bar{y} - \beta_1 \bar{x}$$

$$\beta_0 = 2750 - 500 \times 3.5$$

$$2750 - 1750$$

$$\beta_0 = 1000$$

Ejemplo

Paso 5: Modelo final

Usamos la fórmula:

$$y = \beta_0 + \beta_1 x + \epsilon$$

El modelo ajustado es:

$$y = 1000 + 500x$$

El ajuste de la línea busca minimizar ϵ, así que al calcular β0 y β1, estamos optimizando el modelo para que ϵ sea lo más pequeño posible.
En este sentido, ϵ no desaparece, sino que ya está considerado como parte del proceso de ajuste.

Con este modelo, podemos predecir el alquiler para un número dado de habitaciones.

Ejemplo Regresión Lineal Múltiple

El objetivo es encontrar una relación lineal entre la calificación final del curso de IA (Y) y las variables independientes: la calificación del examen parcial (X1) y el número de clases perdidas (X2).

Estudiante	***Y (Calificación de IA)***	***X1 (Calificación del examen)***	***X2 (Clases perdidas)***
1	85	65	1
2	74	50	7
3	76	55	5
4	90	65	2
5	85	55	6
6	87	70	3
7	94	65	2
8	98	70	1
9	81	55	5
10	91	70	3
11	76	50	1
12	74	55	4

Ejemplo

Paso 1: Relación entre las variables

Queremos encontrar una ecuación lineal que se ajuste a estos datos:

$$y = \beta_0 + \beta_1 x + \beta_2 x + \epsilon$$

Donde:

- β0: Intersección (término constante).
- β1: Coeficiente asociado a la calificación del parcial
- β2: Coeficiente asociado al número de clases perdidas
- ϵ representa el término de error.

$$X = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{i,k-1} \\ 1 & x_{21} & x_{22} & \cdots & x_{2,k-1} \\ \vdots & \vdots & & & \\ 1 & x_{n1} & x_{n2} & \cdots & x_{n,k-1} \end{pmatrix}$$

$$\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_{k-1})^T$$

ϵ es un vector de errores aleatorios.

Ejemplo

Matriz de variables independientes (X)

$$X = \begin{bmatrix} 1 & 65 & 1 \\ 1 & 50 & 7 \\ 1 & 55 & 5 \\ 1 & 65 & 2 \\ 1 & 55 & 6 \\ 1 & 70 & 3 \\ 1 & 65 & 2 \\ 1 & 70 & 1 \\ 1 & 55 & 5 \\ 1 & 70 & 3 \\ 1 & 50 & 1 \\ 1 & 55 & 4 \end{bmatrix}$$

Vector de Valores observados (Y)

$$Y = \begin{bmatrix} 85 \\ 74 \\ 76 \\ 90 \\ 85 \\ 87 \\ 94 \\ 98 \\ 81 \\ 91 \\ 76 \\ 74 \end{bmatrix}$$

Ejemplo

Paso 2: Fórmula para calcular el vector β

Usamos la fórmula de mínimos cuadrados en forma matricial:

$$\beta = (X^T X)^{-1} X^T Y$$

Donde:

- X^T: Transpuesta de la matriz X.
- (X^T X)^-1: Inversa de la matriz X^T X.
- β: Vector que contiene los coeficientes β0,β1,β2

$$X = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{i,k-1} \\ 1 & x_{21} & x_{22} & \cdots & x_{2,k-1} \\ \vdots & \vdots & & & \\ 1 & x_{n1} & x_{n2} & \cdots & x_{n,k-1} \end{pmatrix} \qquad \boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_{k-1})^T$$

Ejemplo

Paso 2: Fórmula para calcular β

Usamos la fórmula de mínimos cuadrados en forma matricial:

$$\beta = (X^T X)^{-1} X^T Y$$

$$X^T = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 65 & 50 & 55 & 65 & 55 & 70 & 65 & 70 & 55 & 70 & 50 & 55 \\ 1 & 7 & 5 & 2 & 6 & 3 & 2 & 5 & 4 & 3 & 1 & 4 \end{pmatrix}$$

Ejemplo

Paso 3a: Calcular $X^T X$

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 65 & 50 & 55 & 65 & 55 & 70 & 65 & 70 & 55 & 70 & 50 & 55 \\ 1 & 7 & 5 & 2 & 6 & 3 & 2 & 5 & 4 & 3 & 1 & 4 \end{pmatrix} \cdot \begin{pmatrix} 1 & 65 & 1 \\ 1 & 50 & 7 \\ 1 & 55 & 5 \\ 1 & 65 & 2 \\ 1 & 55 & 6 \\ 1 & 70 & 3 \\ 1 & 65 & 2 \\ 1 & 70 & 5 \\ 1 & 55 & 4 \\ 1 & 70 & 3 \\ 1 & 50 & 1 \\ 1 & 55 & 4 \end{pmatrix} = \begin{pmatrix} 12 & 725 & 43 \\ 725 & 44475 & 2540 \\ 43 & 2540 & 195 \end{pmatrix}$$

Ejemplo

Paso 3b: Calcular $\left(X^T X\right)^{-1}$

$$\begin{pmatrix} 12 & 725 & 43 \\ 725 & 44475 & 2540 \\ 43 & 2540 & 195 \end{pmatrix}^{(-1)} = \begin{pmatrix} 7{,}654748 & -0{,}110822 & -0{,}244443 \\ -0{,}110822 & 0{,}001692 & 0{,}002395 \\ -0{,}244443 & 0{,}002395 & 0{,}027830 \end{pmatrix}$$

Ejemplo

Paso 3c: Calcular X^TY

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 65 & 50 & 55 & 65 & 55 & 70 & 65 & 70 & 55 & 70 & 50 & 55 \\ 1 & 7 & 5 & 2 & 6 & 3 & 2 & 5 & 4 & 3 & 1 & 4 \end{pmatrix} \cdot \begin{pmatrix} 85 \\ 74 \\ 76 \\ 90 \\ 85 \\ 87 \\ 94 \\ 98 \\ 81 \\ 91 \\ 76 \\ 74 \end{pmatrix} = \begin{pmatrix} 1011 \\ 61685 \\ 3581 \end{pmatrix}$$

Ejemplo

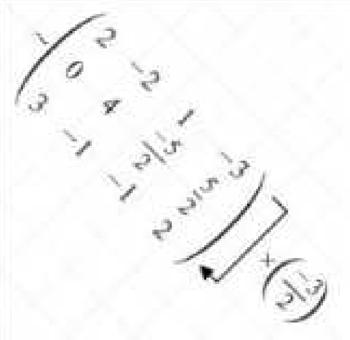
Paso 3d: Calcular β

$$\begin{pmatrix} \frac{88841}{11606} & \frac{-6431}{58030} & \frac{-2837}{11606} \\ \frac{-6431}{58030} & \frac{491}{290150} & \frac{139}{58030} \\ \frac{-2837}{11606} & \frac{139}{58030} & \frac{323}{11606} \end{pmatrix} \cdot \begin{pmatrix} 1011 \\ 61685 \\ 3581 \end{pmatrix} = \begin{pmatrix} 27{,}546700 \\ 0{,}921678 \\ 0{,}284250 \end{pmatrix}$$

La solución nos dará los coeficientes β0, β1, y β2, que representan:

- β0: El término constante.
- β1: El impacto de la calificación del parcial en la nota final del curso de IA.
- β2: El impacto de las clases perdidas en la nota final del curso de IA.

Ejemplo



Calculadora de Matrices

Cálculo de suma de matrices, de diferencia de matrices, de producto de matrices, matriz inversa, de determinante, de matriz transpuesta; Reducir...

matrixcalc

Paso 4: Modelo final

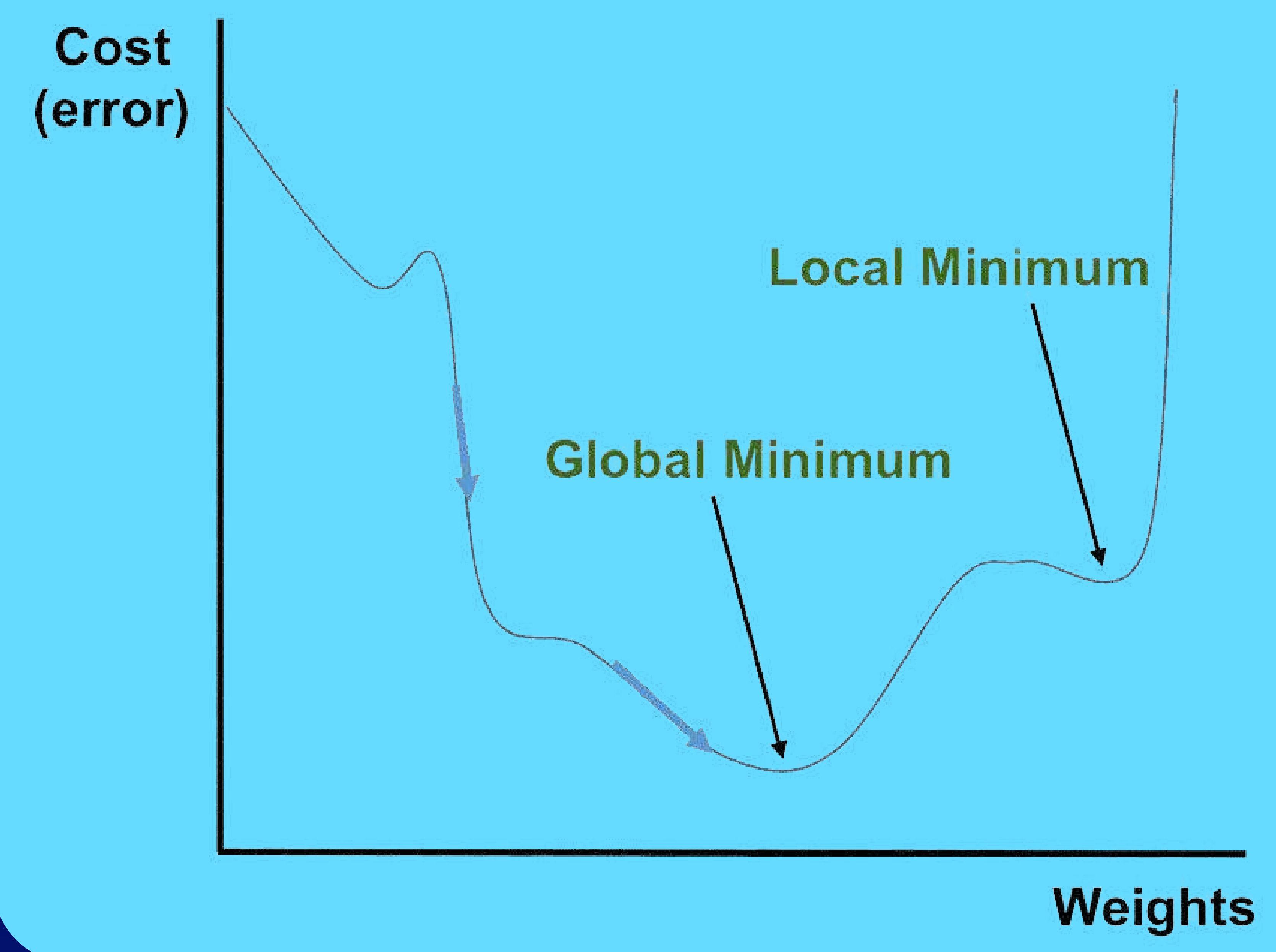
=== MODELO DE REGRESIÓN LINEAL MÚLTIPLE ===
Intercepto ($\beta0$): 27.5467
Coeficiente para Examen ($\beta1$): 0.9217
Coeficiente para Clases_Perdidas ($\beta2$): 0.2842

Ecuación del modelo:

$$Estadistica = 27.55 + 0.92*Examen + 0.28*Clases\_Perdidas$$

Gradiente Descendente

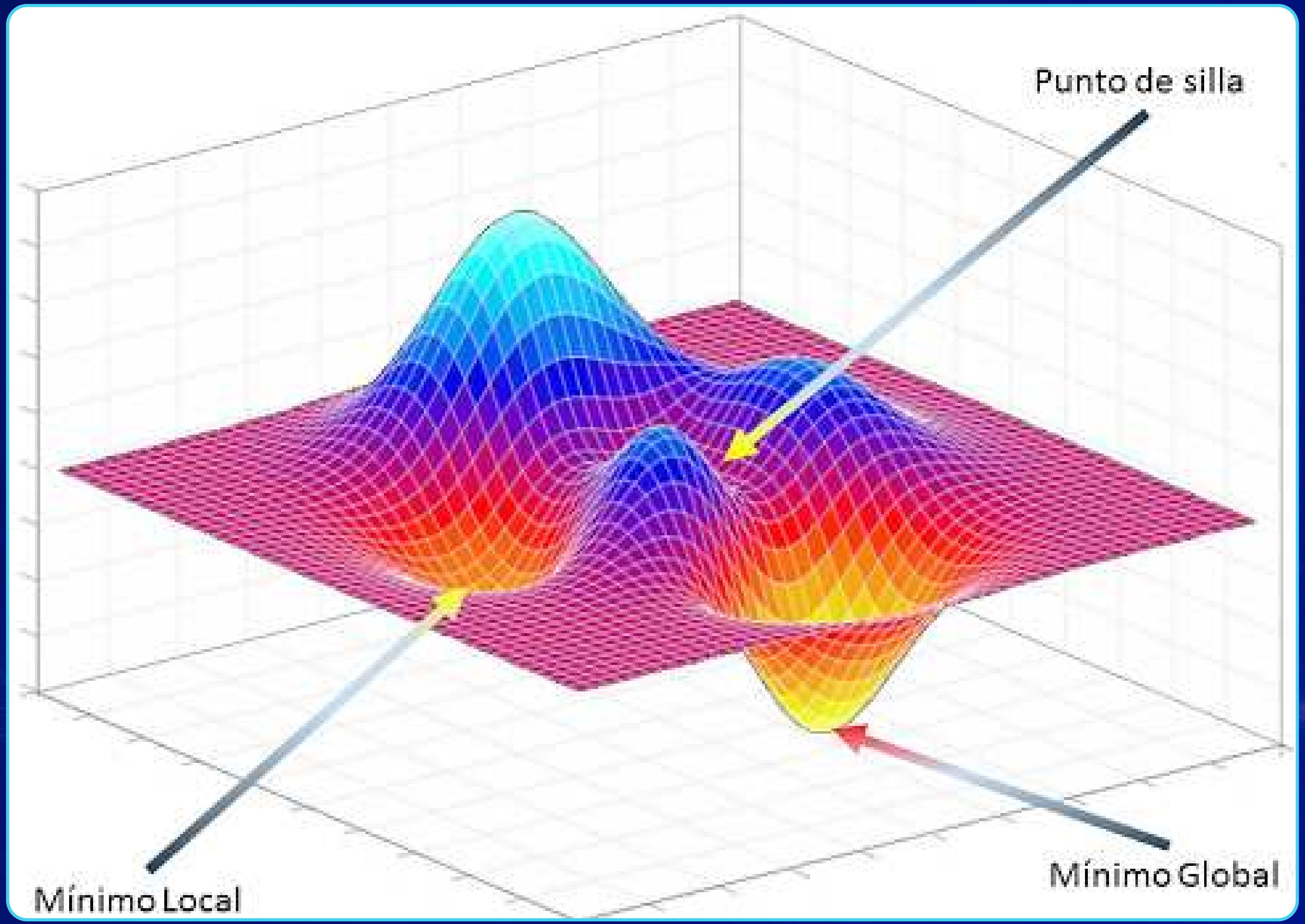
Este es un enfoque numérico utilizado para minimizar funciones de costo, especialmente en modelos complejos o con grandes volúmenes de datos.



Gradiente Descendente

Minimización iterativa de la función de costo.

El gradiente indica la dirección de mayor aumento de la función de costo. En el gradiente descendente, nos movemos en la dirección opuesta (por eso "descendente") para encontrar el mínimo de J.



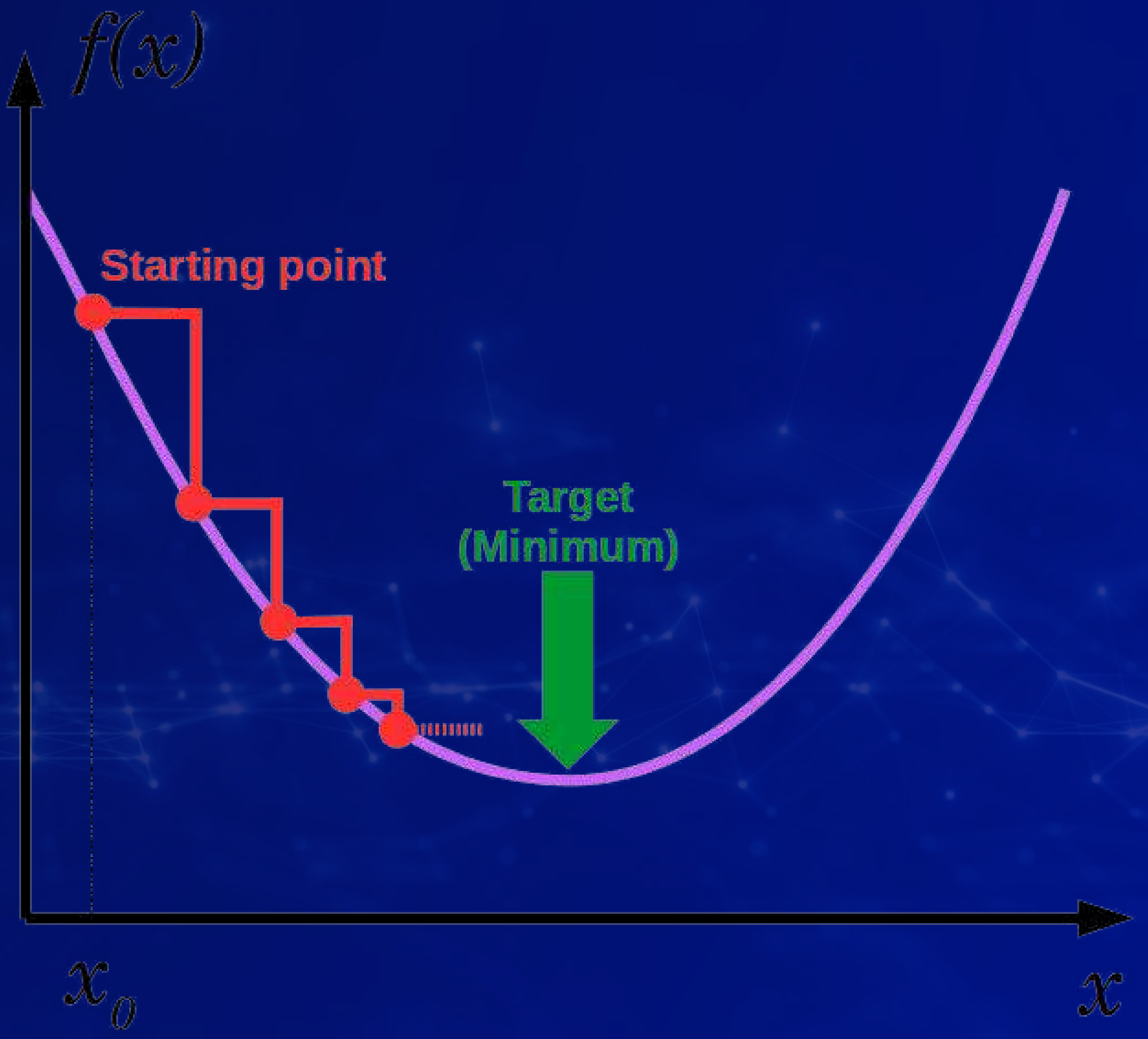
Gradiente Descendente

Para β0 (intersección):

$$\beta_0 := \beta_0 - \alpha \cdot \frac{\partial J}{\partial \beta_0}$$

Para β1 (pendiente):

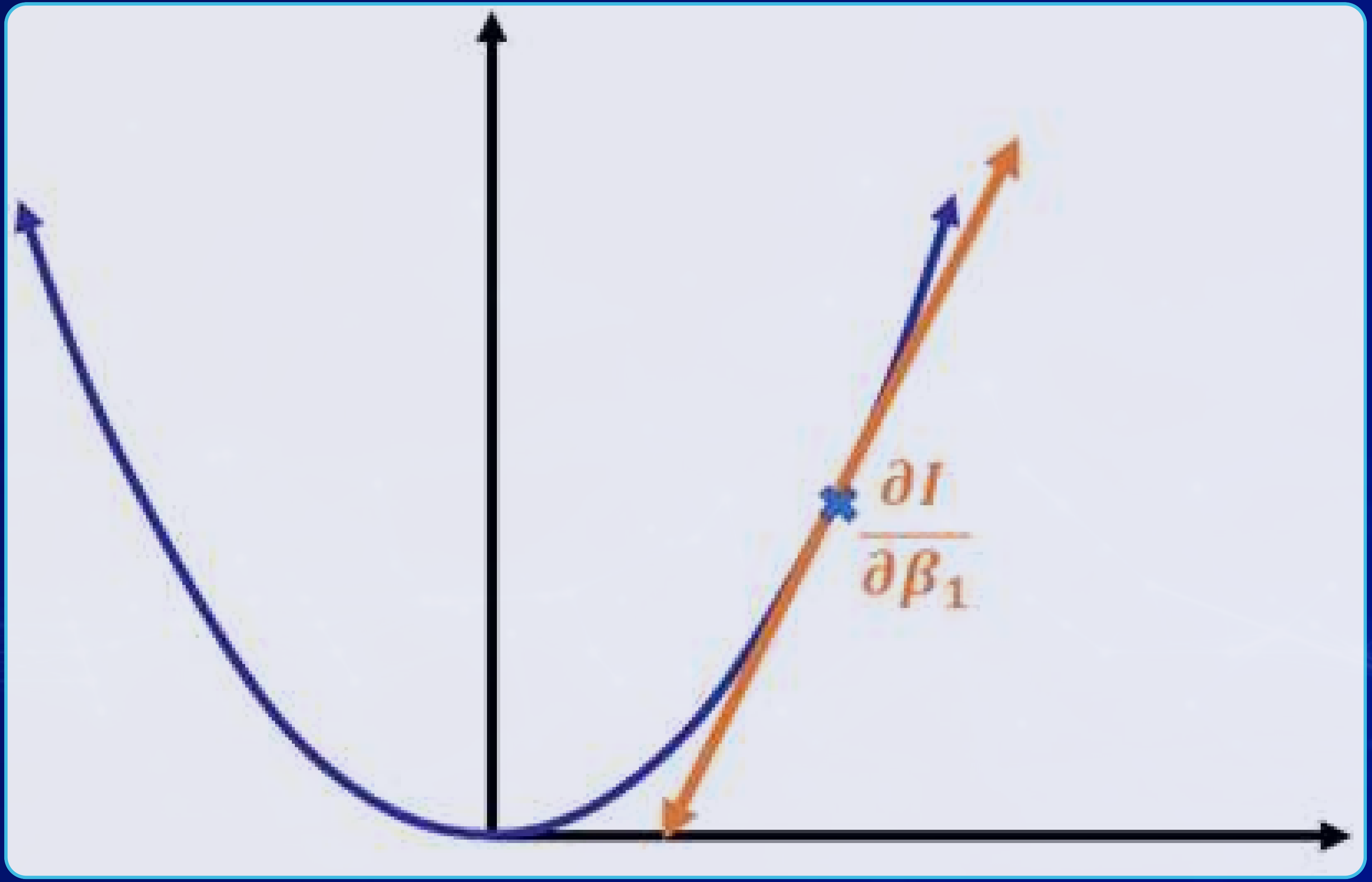
$$\beta_1 := \beta_1 - \alpha \cdot \frac{\partial J}{\partial \beta_1}$$



Gradiente Descendente

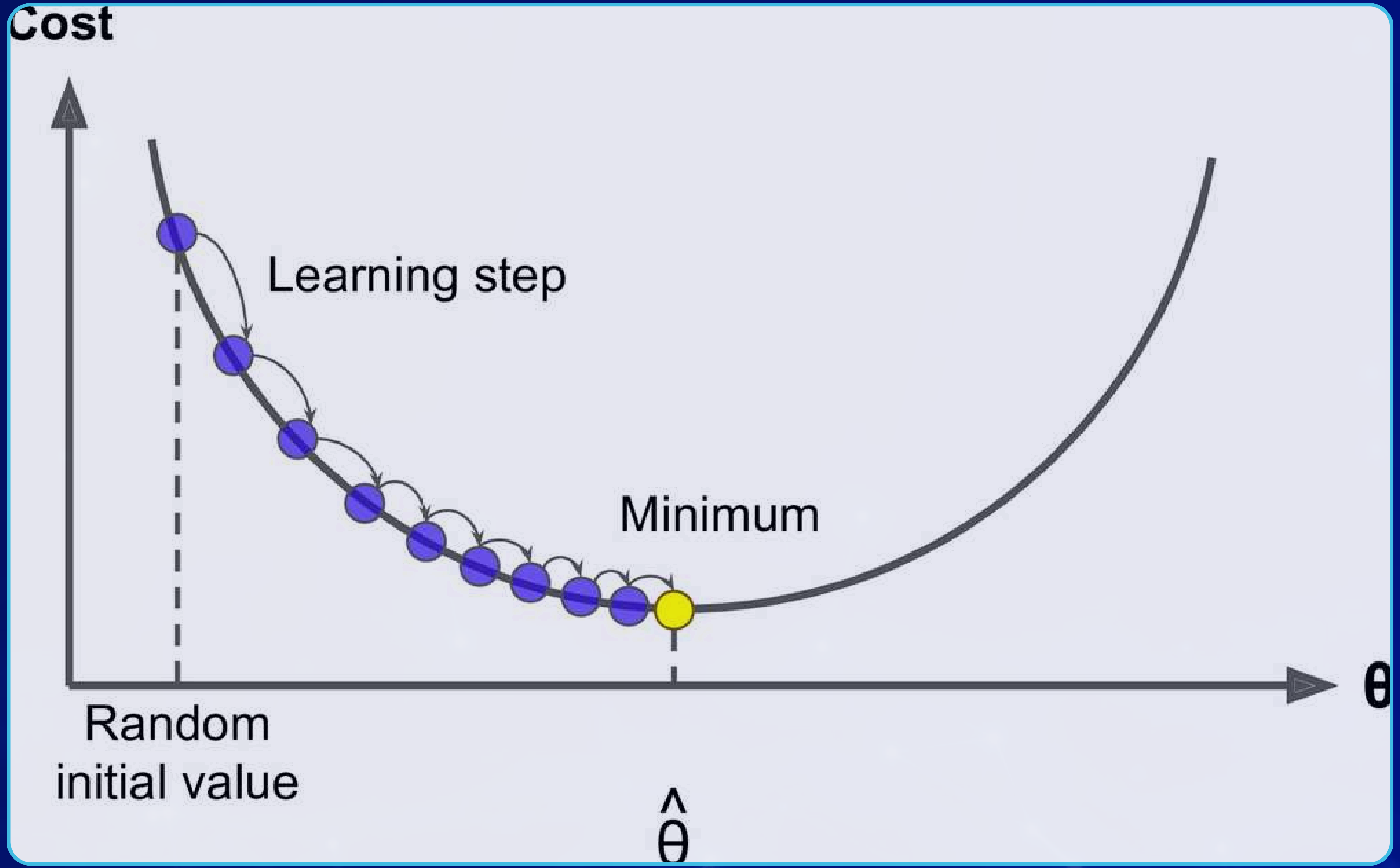
La actualización de los parámetros se hace utilizando las derivadas parciales de J:

$$\frac{\partial J}{\partial \beta_0} \quad \text{y} \quad \frac{\partial J}{\partial \beta_1}$$



Estas derivadas nos indican cómo cambian los errores (el costo) cuando se ajustan β_0 y β_1.

Gradiente Descendente



Donde:

- α: Tasa de aprendizaje, un número pequeño (como 0.01) que controla qué tan rápido avanzamos.

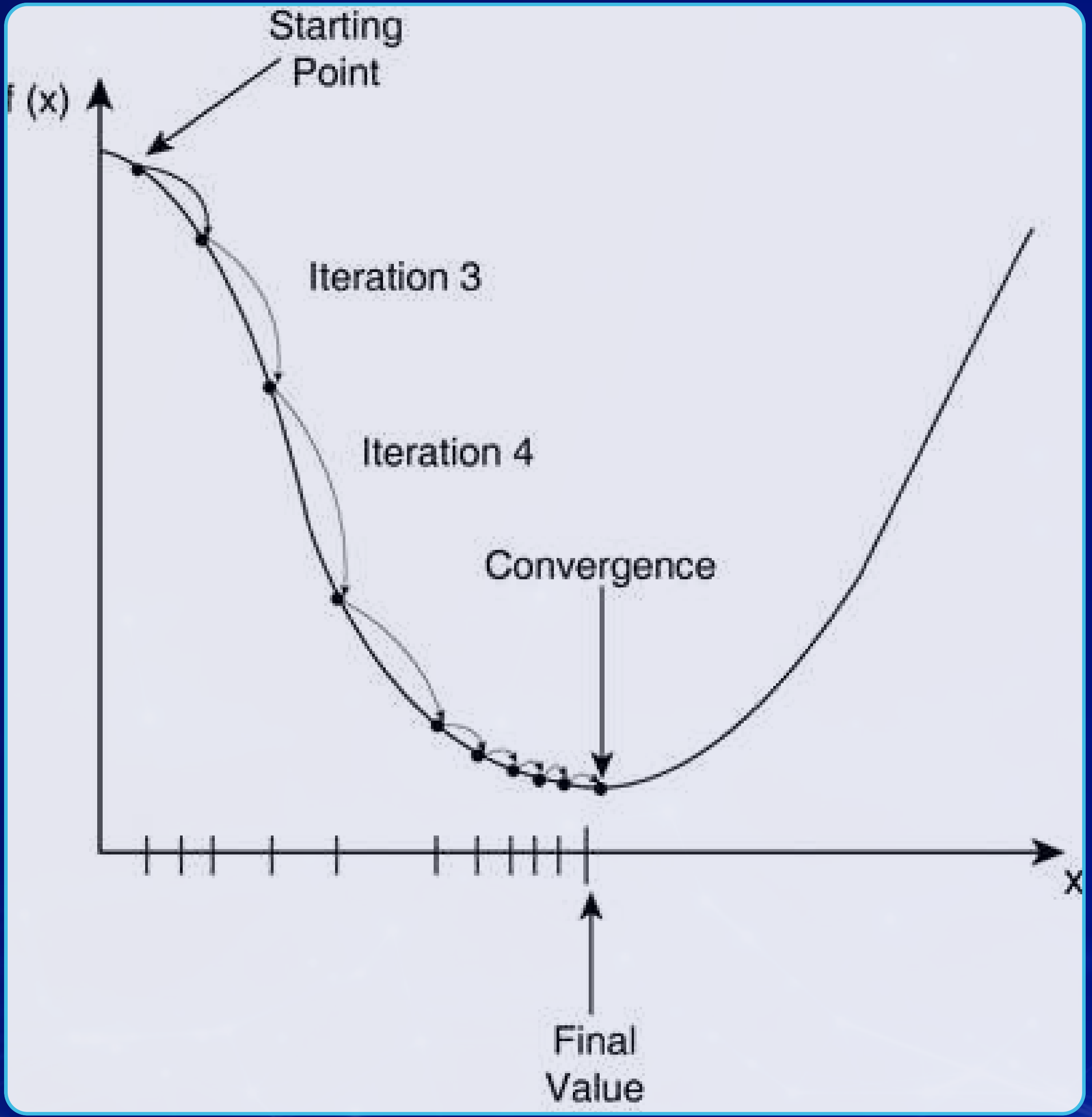
$$\frac{\partial J}{\partial \beta_0} = -\frac{1}{m}\Sigma_{i=1}^{m}\left(Y_i - (\beta_0 + \beta_1 \cdot X_i)\right)$$

$$\frac{\partial J}{\partial \beta_1} = -\frac{1}{m}\Sigma_{i=1}^{m}\left(Y_i - (\beta_0 + \beta_1 \cdot X_i)\right) \cdot X_i$$

Optimización

Proceso Iterativo

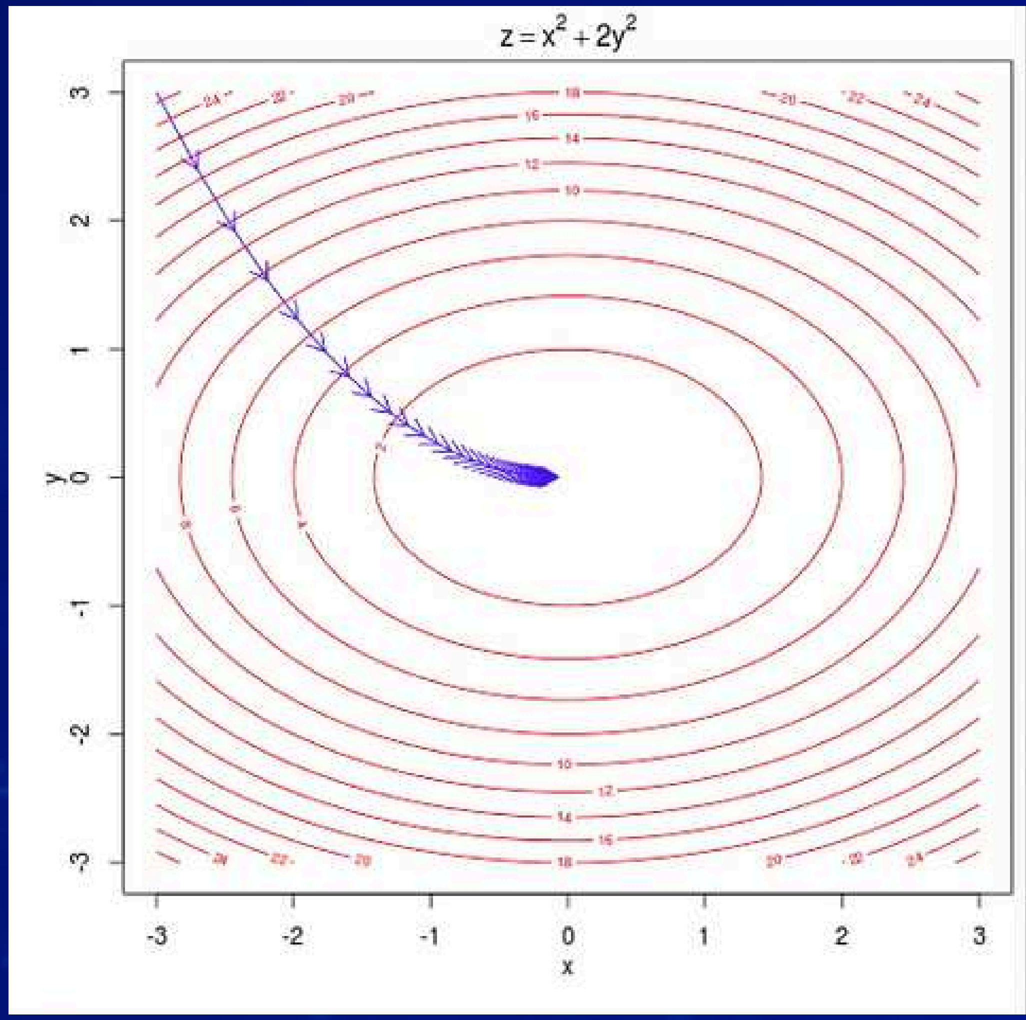
1. **Inicialización:** Asigna valores iniciales a β0 y β1 (pueden ser 0 o valores aleatorios).
2. **Calcular el costo inicial:** Usar los valores iniciales para calcular J.
3. **Actualizar los parámetros:** Usar las fórmulas de actualización para β0 y β1.
4. **Repetir:** Volver a calcular el costo J y repetir hasta que:
   - J sea lo suficientemente pequeño.
   - Ó el número de iteraciones alcance un límite predefinido.



Ejemplo intuitivo

Imagina que tienes un modelo inicial:

$$Y_{predicho} = \beta_0 + \beta_1 x$$



Si tu predicción inicial está lejos de los valores reales (*Yreal*), el gradiente descendente ajustará *β0* y *β1* paso a paso para alinear mejor las predicciones con los datos.

Después de suficientes iteraciones, los valores de β0 y β1 convergerán a los mismos que encontrarías usando un método más exacto como los mínimos cuadrados.

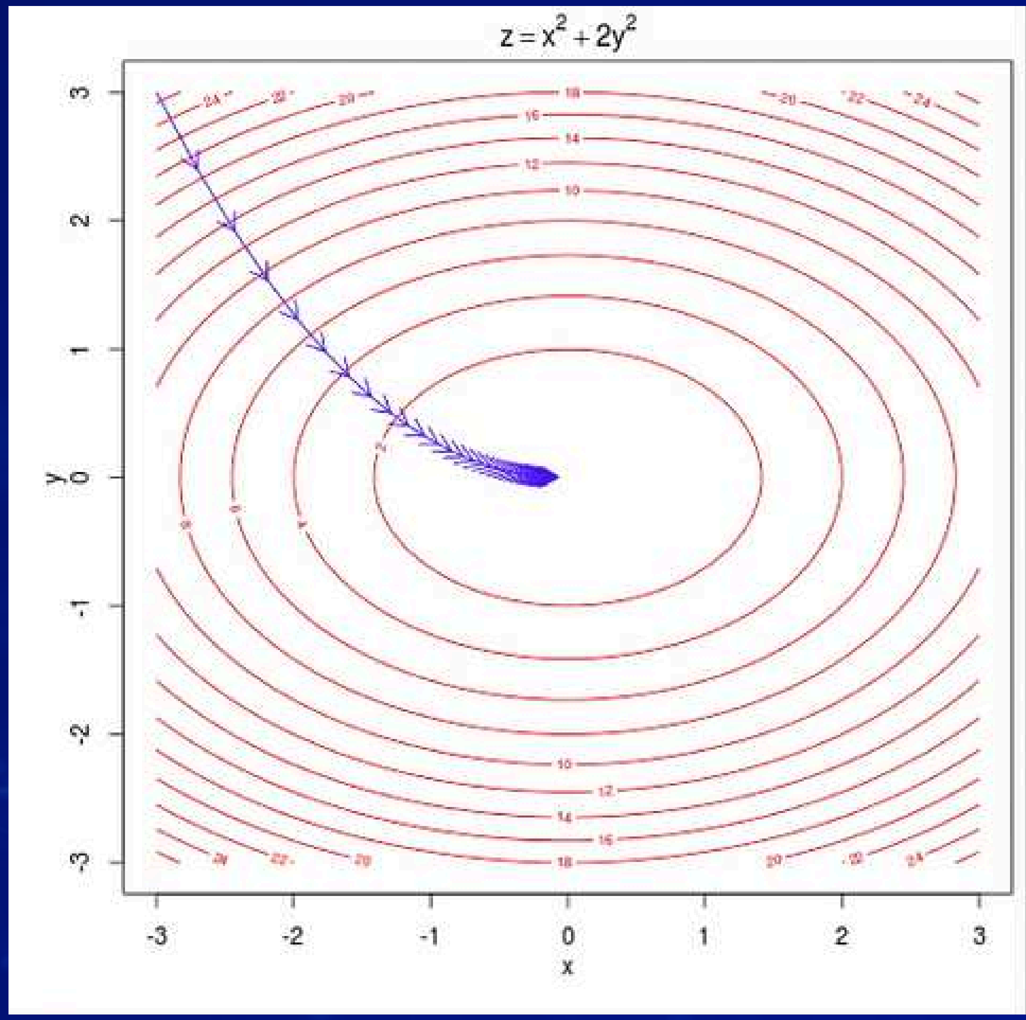
Optimización

Aplicado a la regresión lineal

Cuando usar...

mínimos cuadrados ordinarios:

- Cuando el número de datos es mayor que el número de variables y la matriz $X^T X$ es invertible
- Cuando queremos una solución exacta y rápida (cuando X es de tamaño manejable)
- Cuando no hay demasiadas características (variables), porque la inversión de la matriz $(X^T X)^{-1}$ es computacionalmente costosa si el numero de variables es grande.

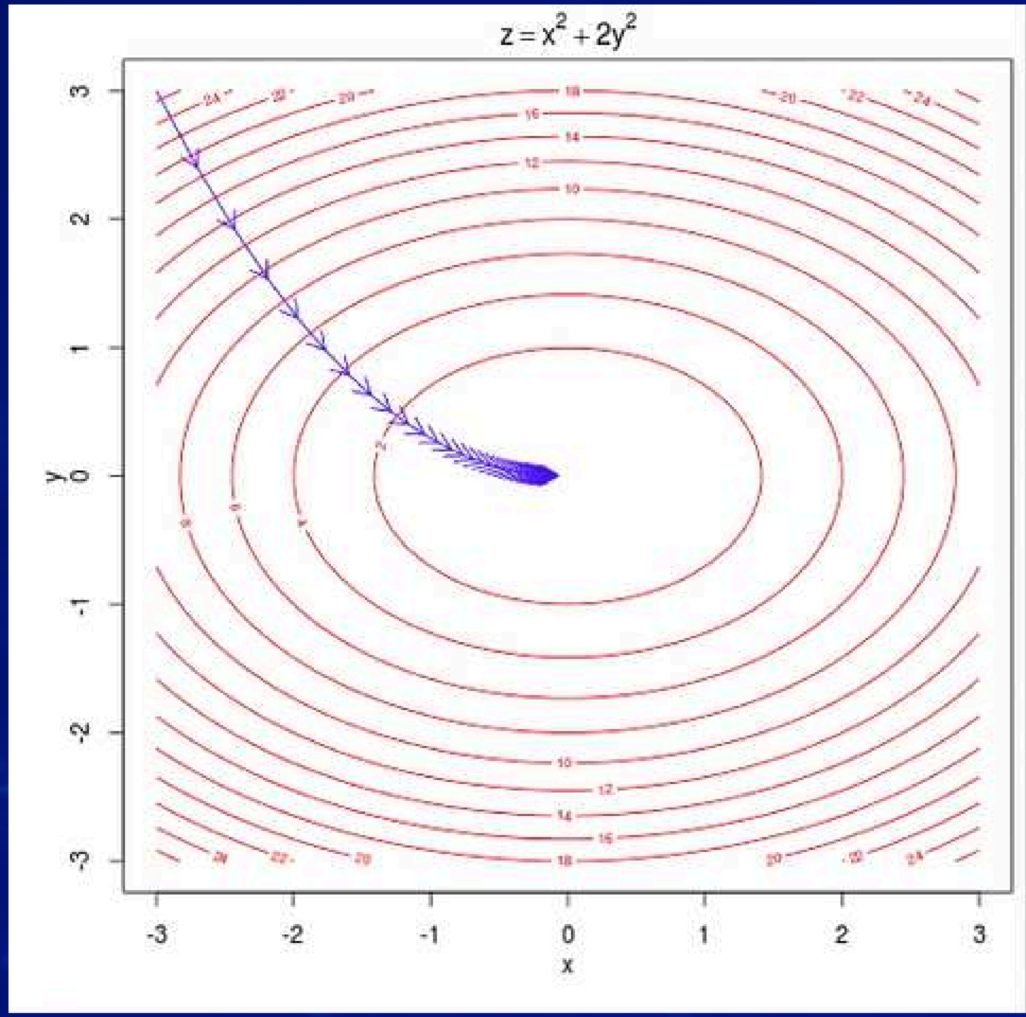


se usa cuando podemos calcular la solución exacta y tenemos una cantidad razonable de datos y características.

Cuando usar...

descenso del gradiente:

- Cuando el conjunto de datos es muy grande (alta dimensionalidad o muchas muestras), porque evitar calcular $(X^TX)^{-1}$ ahorra memoria y tiempo.
- Cuando no podemos invertir X^TX porque es singular o mal condicionada.
- Cuando trabajamos con modelos de aprendizaje profundo, donde OLS no es viable debido a la no linealidad y la gran cantidad de parámetros.



es más flexible y escalable, pero requiere elegir una tasa de aprendizaje y más iteraciones para converger a la solución óptima

Evaluación del modelo

Para medir qué tan buena es nuestra aproximación, usamos el error sobre los datos de entrenamiento

Métricas principales:

- Error Cuadrático Medio (MSE): Penaliza errores grandes.
- Coeficiente de Determinación: Explica qué porcentaje de la varianza de *y* es explicado por *x*.

Métricas

Estas métricas nos ayudan a entender qué tan bien el modelo es capaz de aproximarse a los valores reales de la variable que estamos intentando predecir.

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}|$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y})^2$$

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y})^2}$$

$$R^2 = 1 - \frac{\sum (y_i - \hat{y})^2}{\sum (y_i - \bar{y})^2}$$

Where,

$\hat{y}$ – *predicted value of y*

$\bar{y}$ – *mean value of y*

Evaluación del Modelo de Regresión

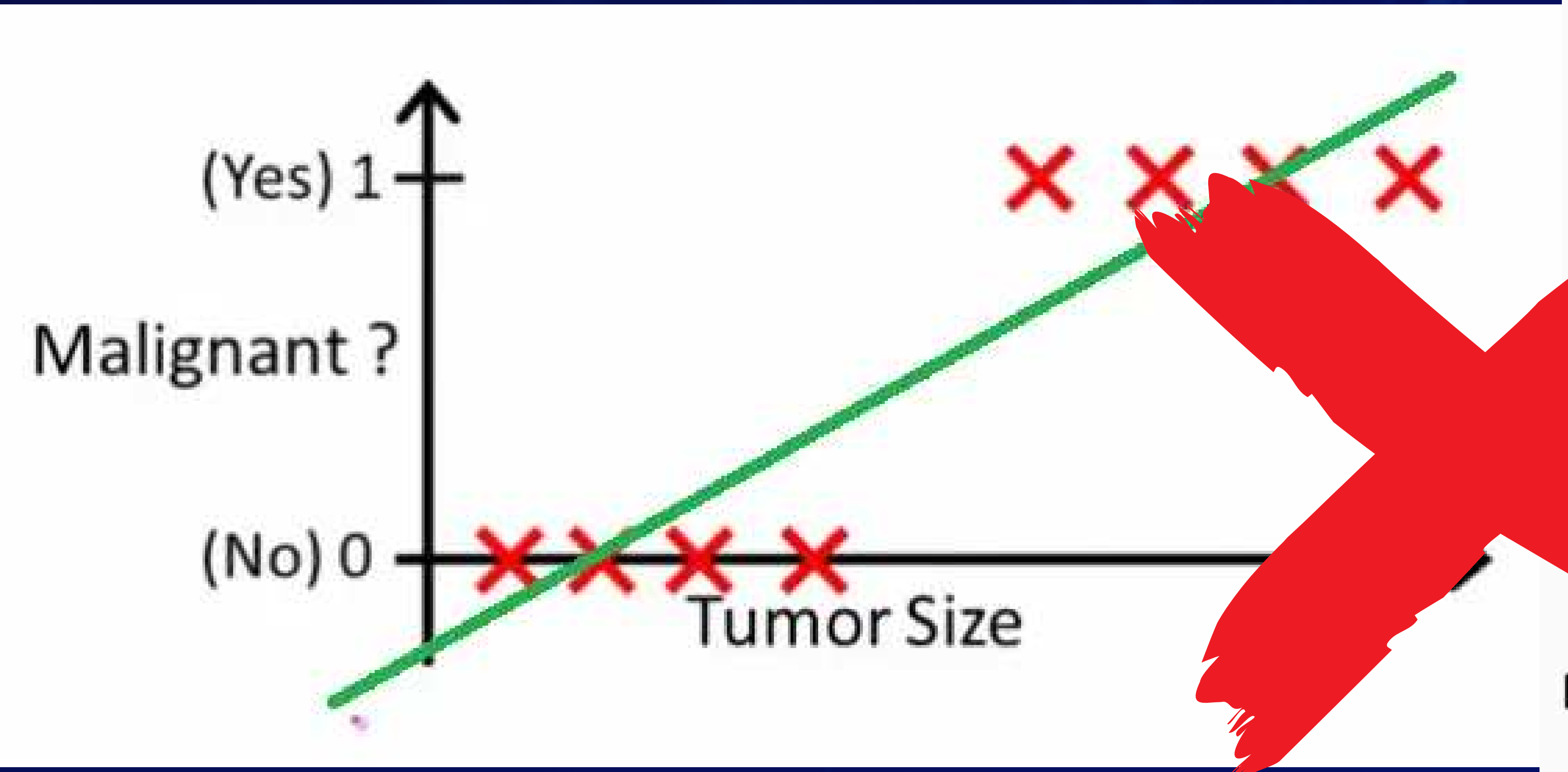
Métrica	Descripción	Ventajas	Desventajas	Cuándo usar
Error Absoluto Medio (MAE)	Promedio de las diferencias absolutas entre los valores predichos y los valores reales.	Robusto a valores atípicos. Fácil de interpretar en las unidades originales de la variable objetivo.	No penaliza los errores grandes tan fuertemente como el MSE.	Cuando los valores atípicos son comunes y se desea una métrica fácil de interpretar.
Error Cuadrático Medio (MSE)	Promedio de los cuadrados de las diferencias entre los valores predichos y los valores reales.	Penaliza más los errores grandes. Ampliamente utilizada en optimización.	Sensible a valores atípicos. No es tan fácil de interpretar en las unidades originales de la variable objetivo.	Cuando los errores grandes son especialmente indeseables y se desea una métrica matemáticamente conveniente.
Raíz del Error Cuadrático Medio (RMSE)	Raíz cuadrada del MSE.	Penaliza los errores grandes, pero es más interpretable que el MSE porque está en las unidades originales de la variable objetivo.	Sensible a valores atípicos.	Cuando los errores grandes son indeseables y se desea una métrica en las unidades originales de la variable objetivo.
Coeficiente de Determinación (R^2)	Proporción de la varianza de la variable dependiente que es predecible a partir de las variables independientes.	Indica 1 qué tan bien se ajusta el modelo a los datos. Varía entre 0 y 1, donde 1 indica un ajuste perfecto.	No indica si el modelo es sesgado. No indica si las variables independientes son estadísticamente significativas.	Para evaluar el ajuste general del modelo y comparar diferentes modelos.

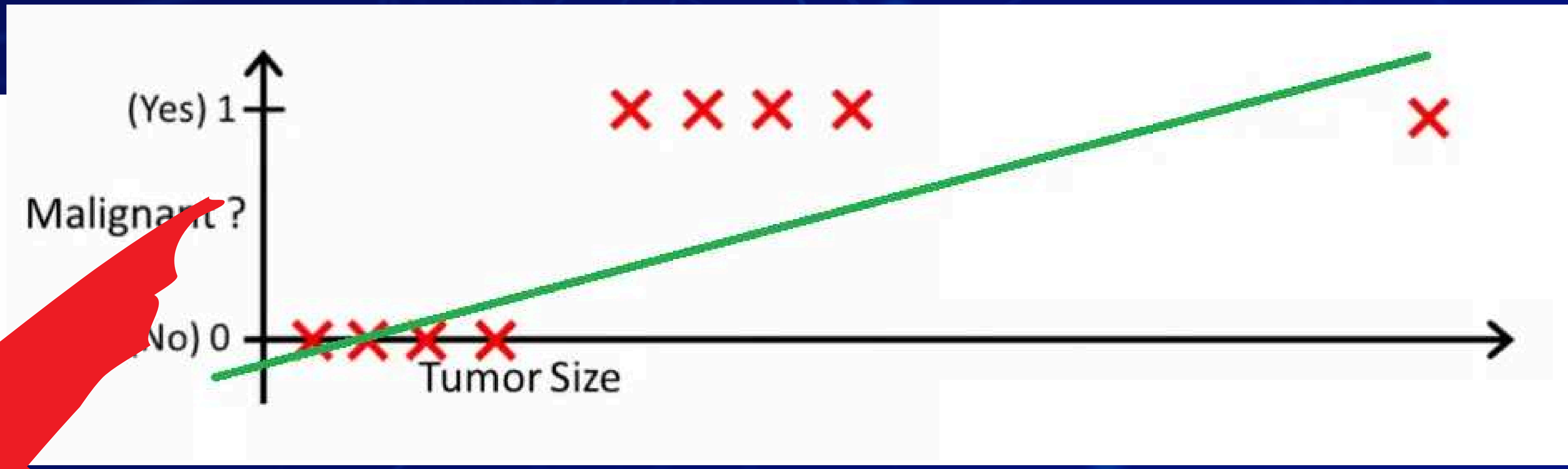
Diferencias Claves entre Función de Coste y Métricas de Evaluación

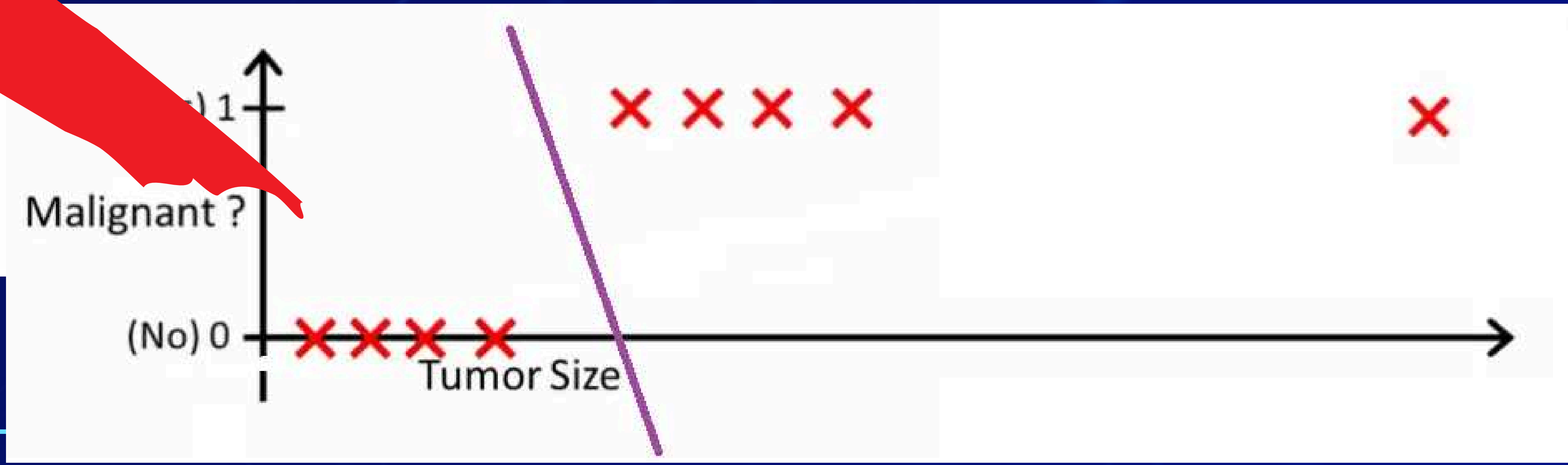
Característica	Función de Coste	Métrica de Evaluación
Propósito	Optimizar el modelo minimizando errores.	Evaluar la calidad del modelo.
Cuándo se usa	Durante el entrenamiento.	Después del entrenamiento.
¿Afecta el modelo?	Sí, el modelo ajusta los pesos para minimizarla.	No, solo mide qué tan bueno es el modelo.
Ejemplos	MSE, MAE, Log-Cosh, Regularización.	RMSE, R^2, MAPE.

Clasificación

Regresión Logística

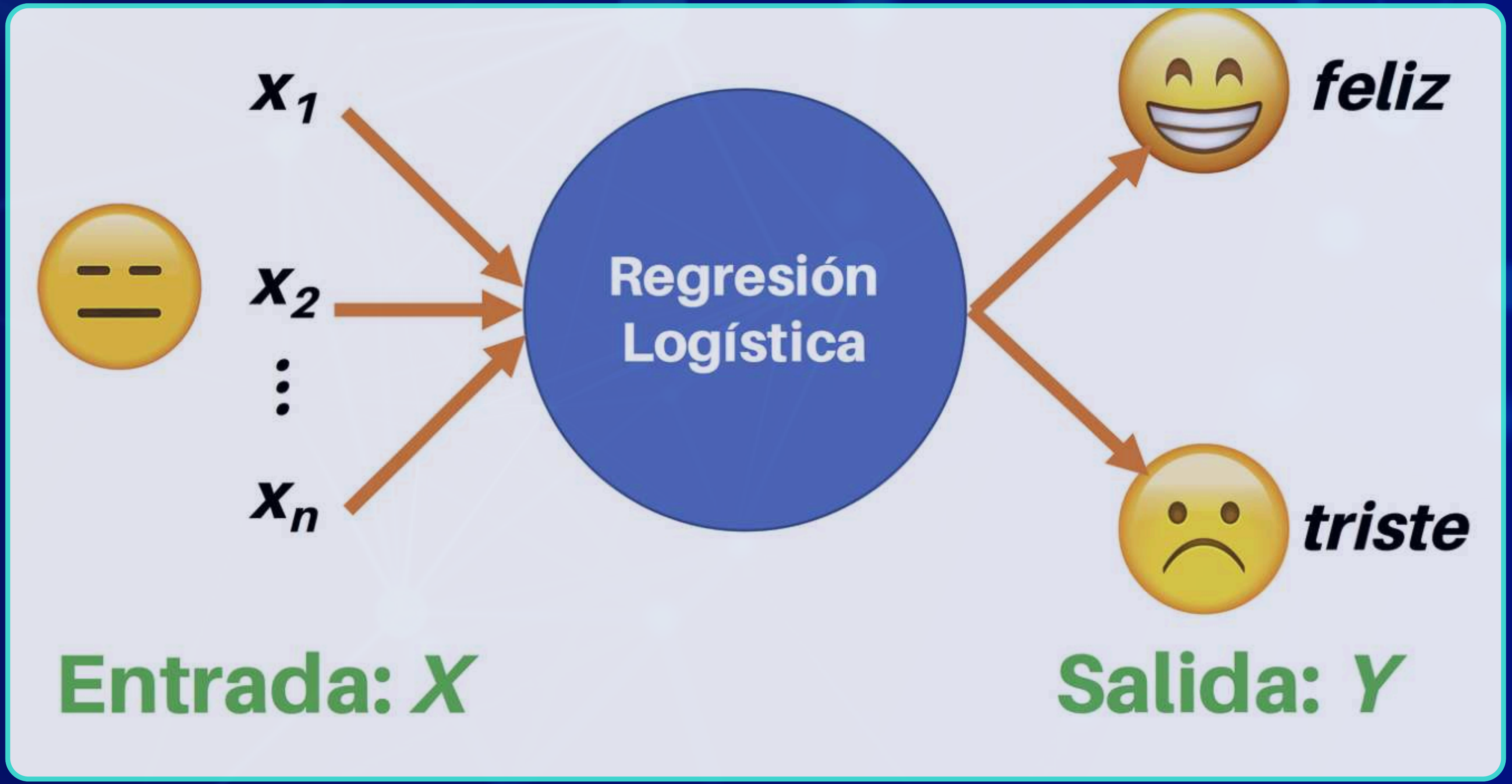




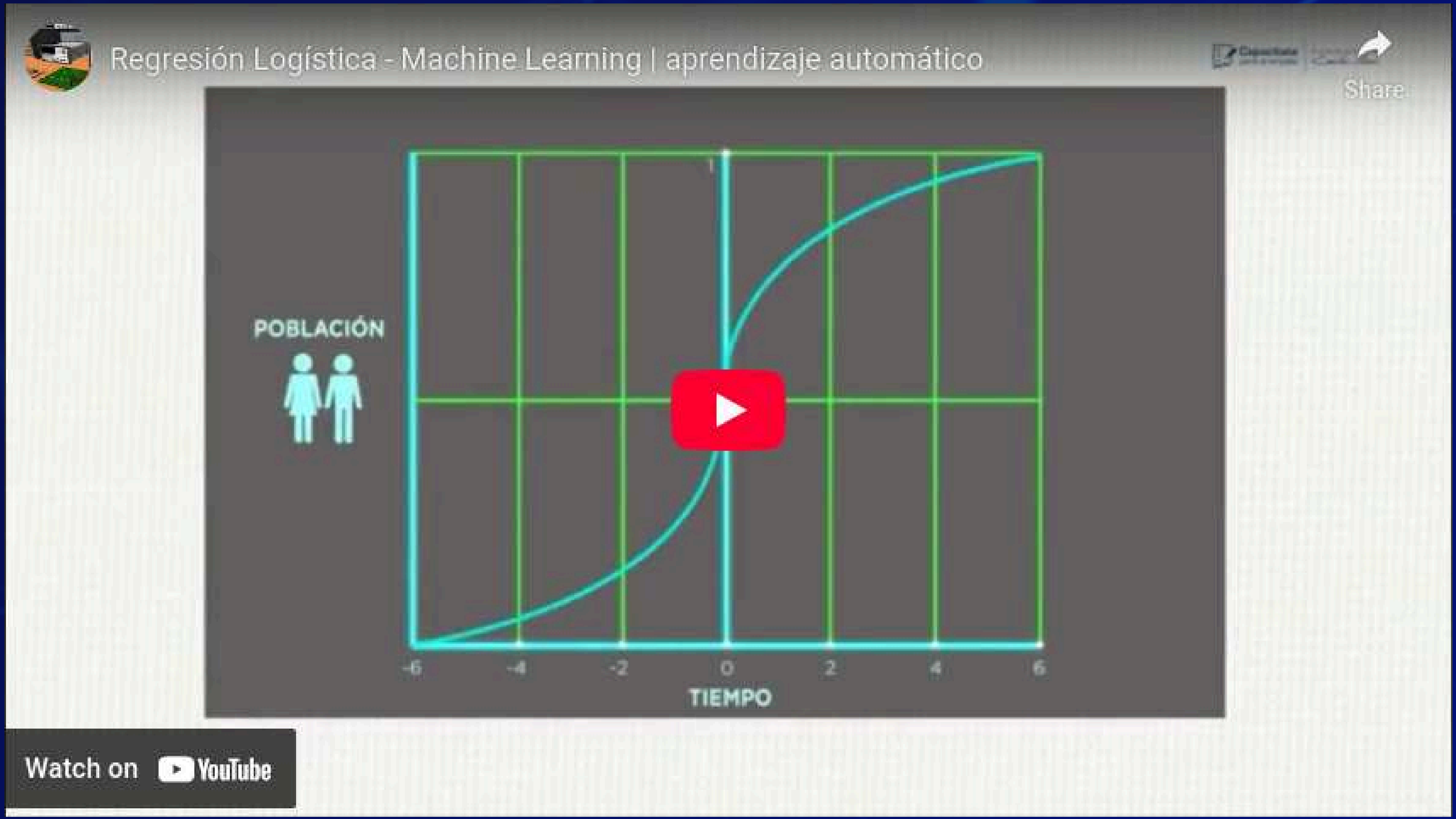


Regresión Logística

La regresión logística es un algoritmo de aprendizaje automático supervisado que se utiliza para tareas de clasificación, cuyo objetivo es predecir la probabilidad de que una instancia pertenezca a una clase determinada.

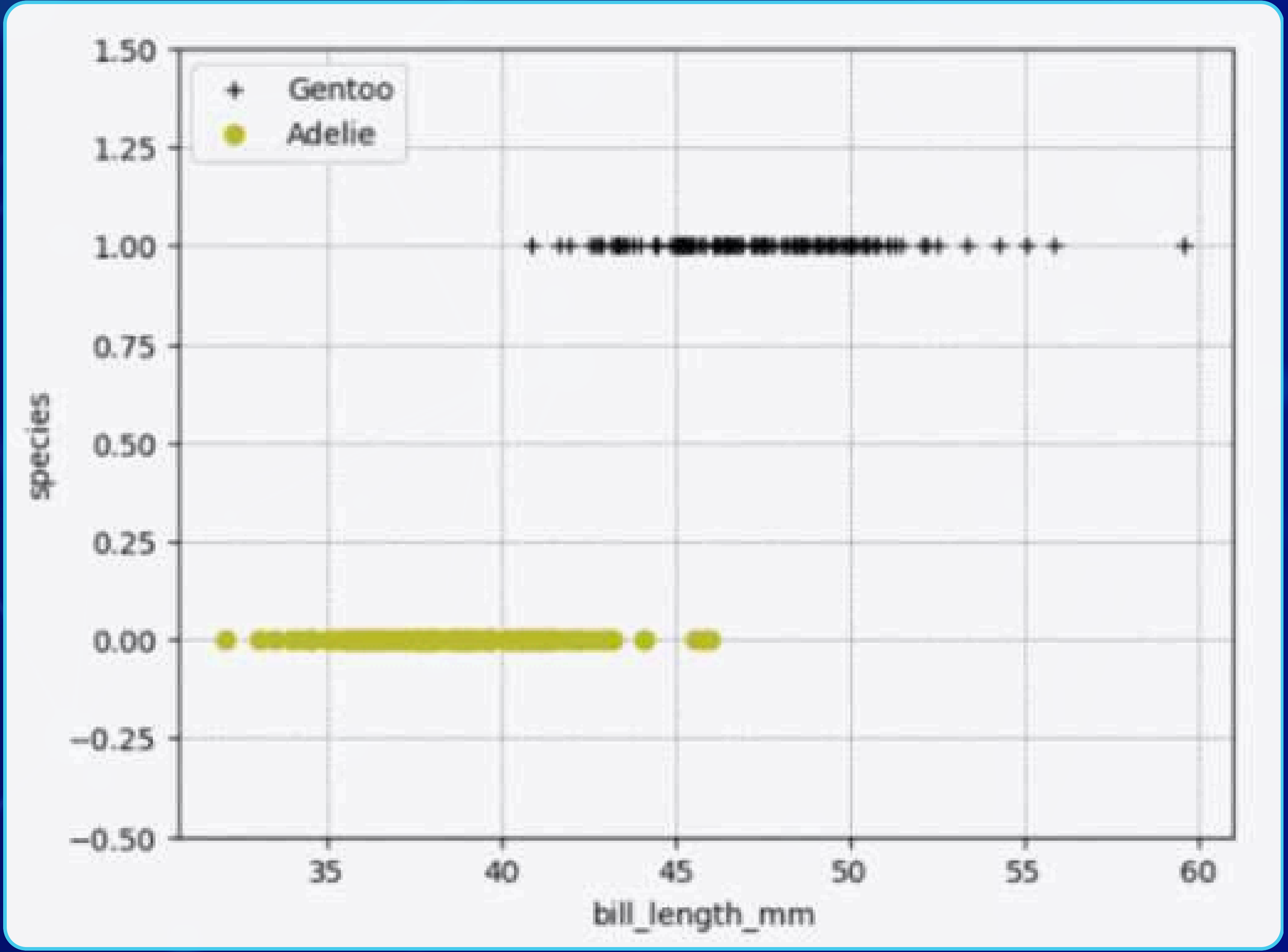


Objetivo: Encontrar β0 y β1 que minimicen el error


Regresión Logística - Machine Learning | aprendizaje automático
Share
POBLACIÓN
-6
-4
-2
0
2
4
6
TIEMPO
Watch on YouTube

Regresión Logística

La regresión logística se modela con una ecuación de la misma forma solo que aplicando la función sigmoidal.
Recordemos que estamos hablando de un modelo de clasificación y su objetivo no es predecir un valor sino encontrar la probabilidad de que una sea clase (0) o de otra (1).



Buscar una ecuación que nos permita predecir la probabilidad de que una observación pertenezca a una de dos clases.

Regresión Logística

Para entender cómo funciona, primero recordemos el caso de la regresión lineal. En regresión lineal, modelamos una salida y como una combinación lineal de las entradas:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \epsilon$$

Sin embargo, en problemas de clasificación binaria, necesitamos que la salida esté entre 0 y 1 para que pueda interpretarse como una probabilidad. Aquí es donde entra en juego la función logística (también conocida como sigmoide).

El Problema: Modelar una Probabilidad Binaria

Queremos predecir la probabilidad p de que un evento ocurra (ej.: $y=1$) dado un predictor X.

Limitación:

- p debe estar entre 0 y 1.
- Una recta $p=\beta 0+\beta 1X$ puede generar valores fuera de [0,1] (ej.: $p=-0.2$ o $p=1.5$), lo cual no tiene sentido.

Solución: Transformar p para quitar las limitaciones

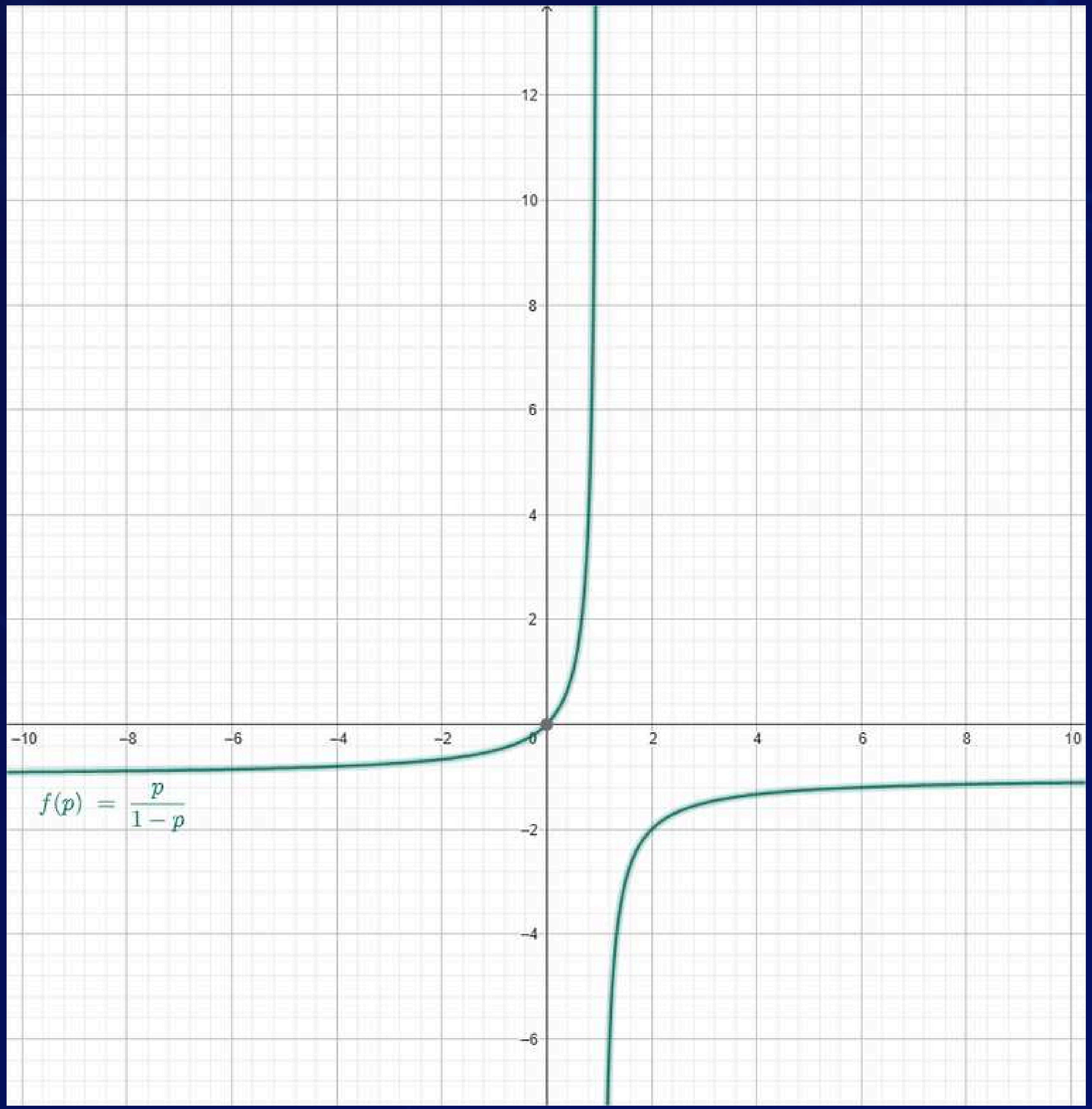
Odds

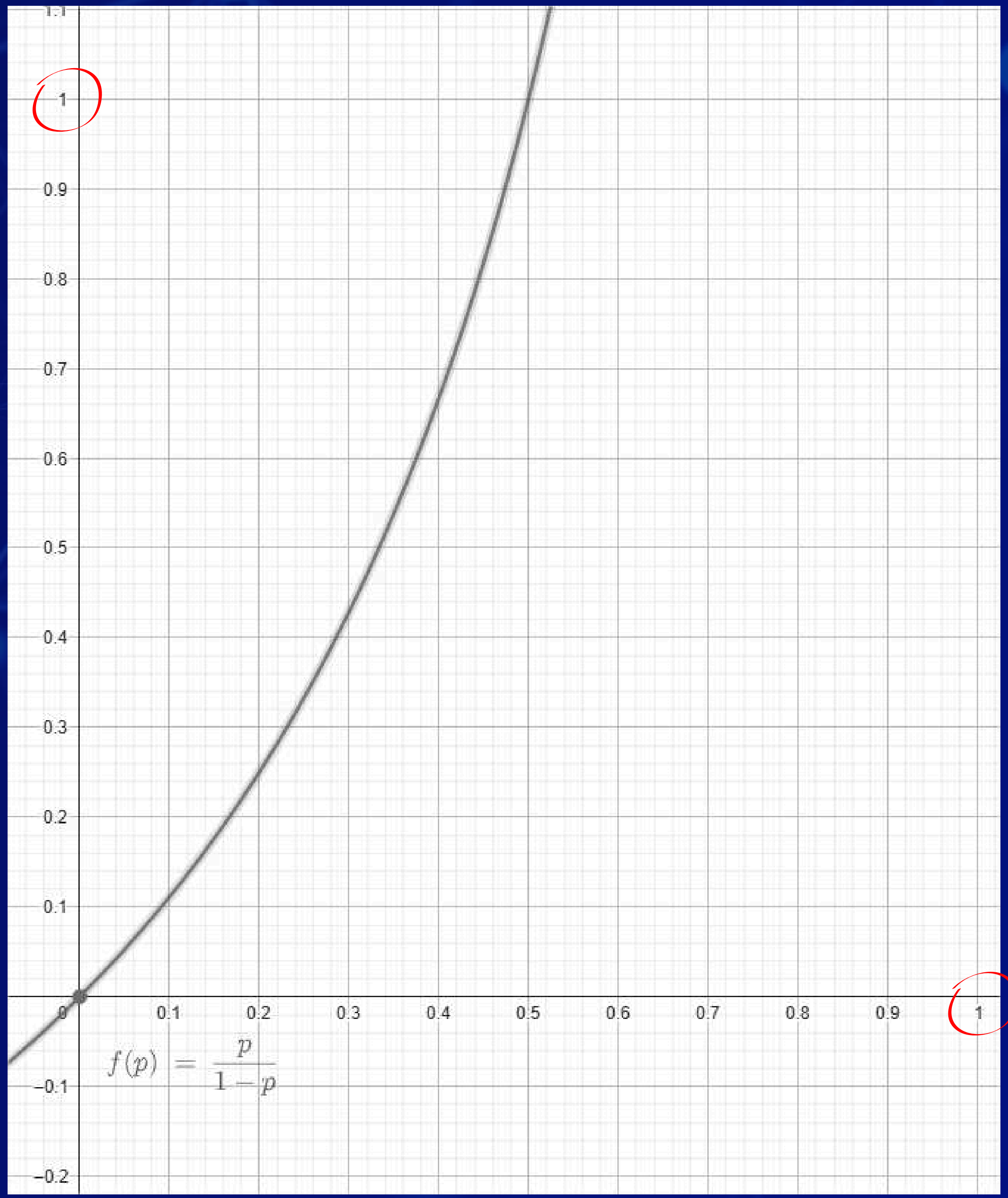
Los odds son una forma alternativa de expresar la probabilidad de un evento. Mientras que la probabilidad (p) indica la proporción de veces que esperamos que ocurra un evento, los odds comparan directamente las posibilidades de que ocurra el evento frente a que no ocurra.

$$Odds = \frac{p}{1-p}$$

- Razón de probabilidades
    - Esto significa que el evento es ciertas veces tan probable como no ocurrir.
    - Para probabilidades estrictamente entre 0 y 1 , es decir, $0<p<1$.

Odds





Logit (Log-Odds)

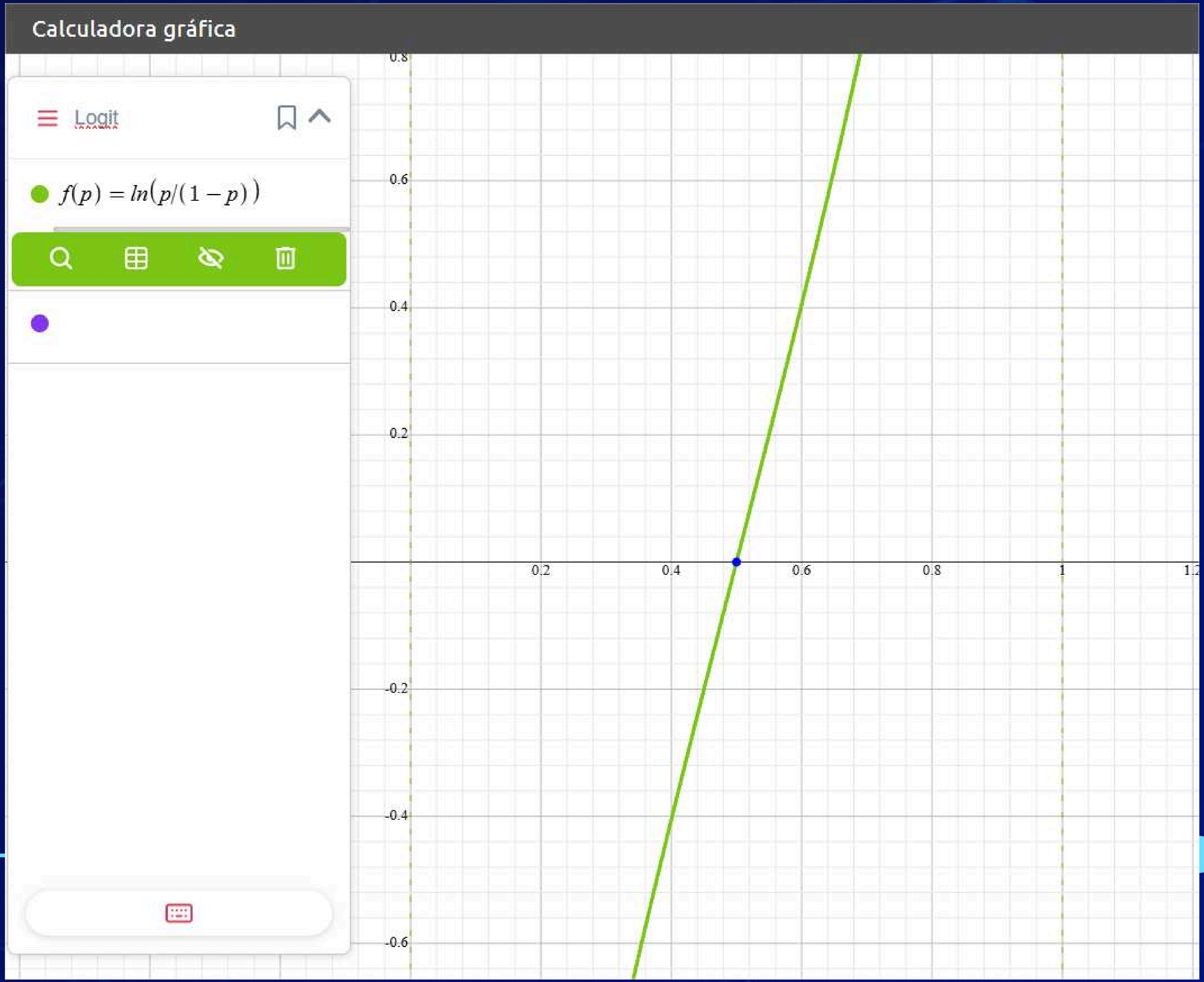
La función logit actúa como un puente entre las probabilidades y los modelos lineales, permitiéndonos trabajar con una estructura matemática más manejable.

$$\ln\left(\frac{p}{1-p}\right)$$

- *Asume un comportamiento lineal*

"Piensa en los odds como una escala elástica: el logit la estira para que una recta pueda ajustarse sin romper las reglas de la probabilidad."

Logit (Log-Odds)



Logit (Log-Odds)

La función logit es clave en la regresión logística porque transforma probabilidades (que están restringidas al intervalo [0,1]) en valores que pueden variar en todo el rango real (-∞ a ∞). Esto permite modelar las probabilidades usando una función lineal.

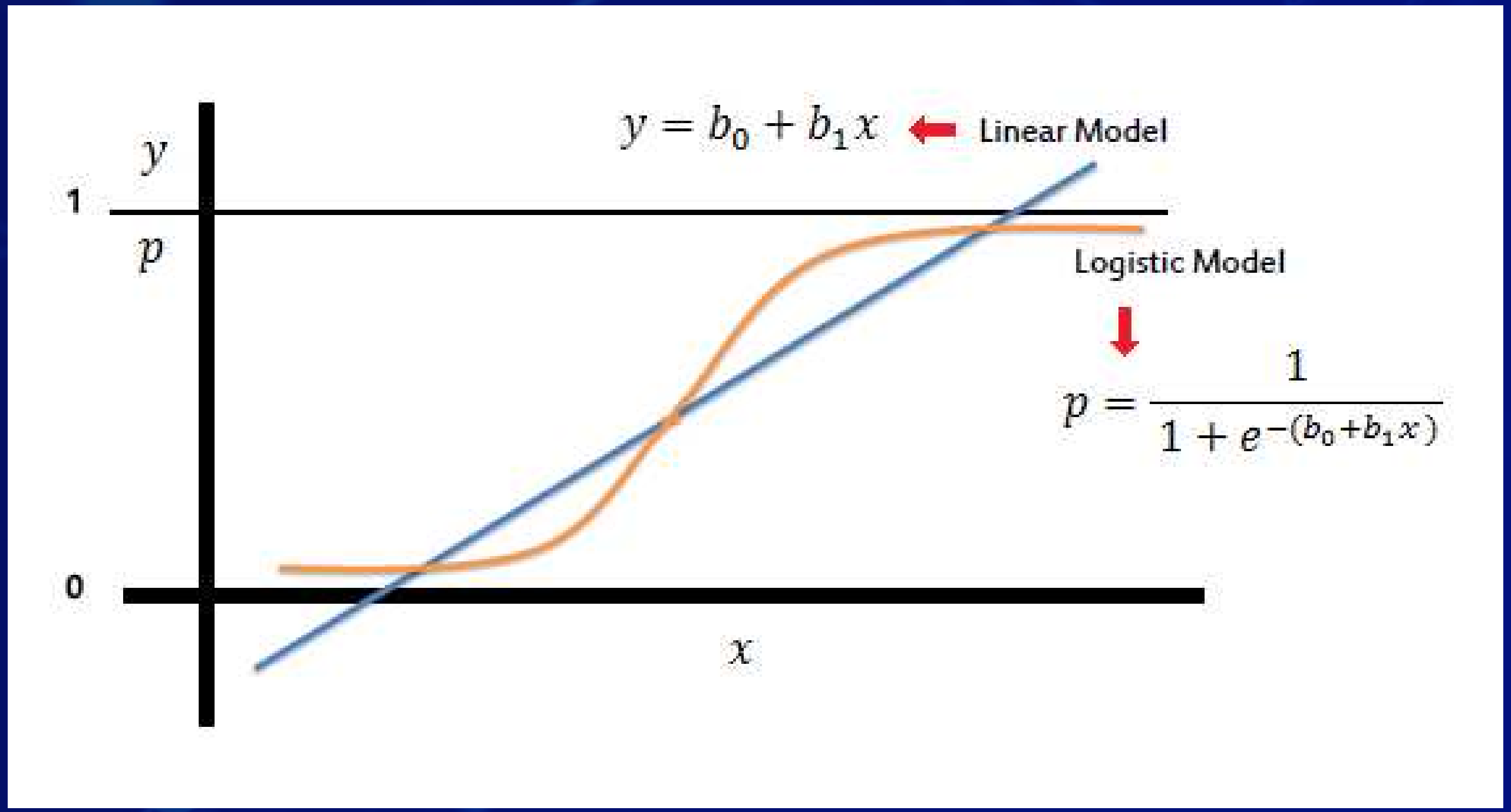
$$\ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X$$

- *Es por esta razón que este método lleva de nombre "Regresión" Logística*

Función Sigmoide

Al despejar para *p*, la ecuación queda:

$$p = \frac{e^{(\beta_0+\beta_1 x)}}{1+e^{(\beta_0+\beta_1 x)}}$$



Objetivo: Encontrar β0 y β1 que minimicen el error

Función Sigmoide

Divide el numerador y denominador de la ecuación por $e^{\beta_0+\beta_1 x}$

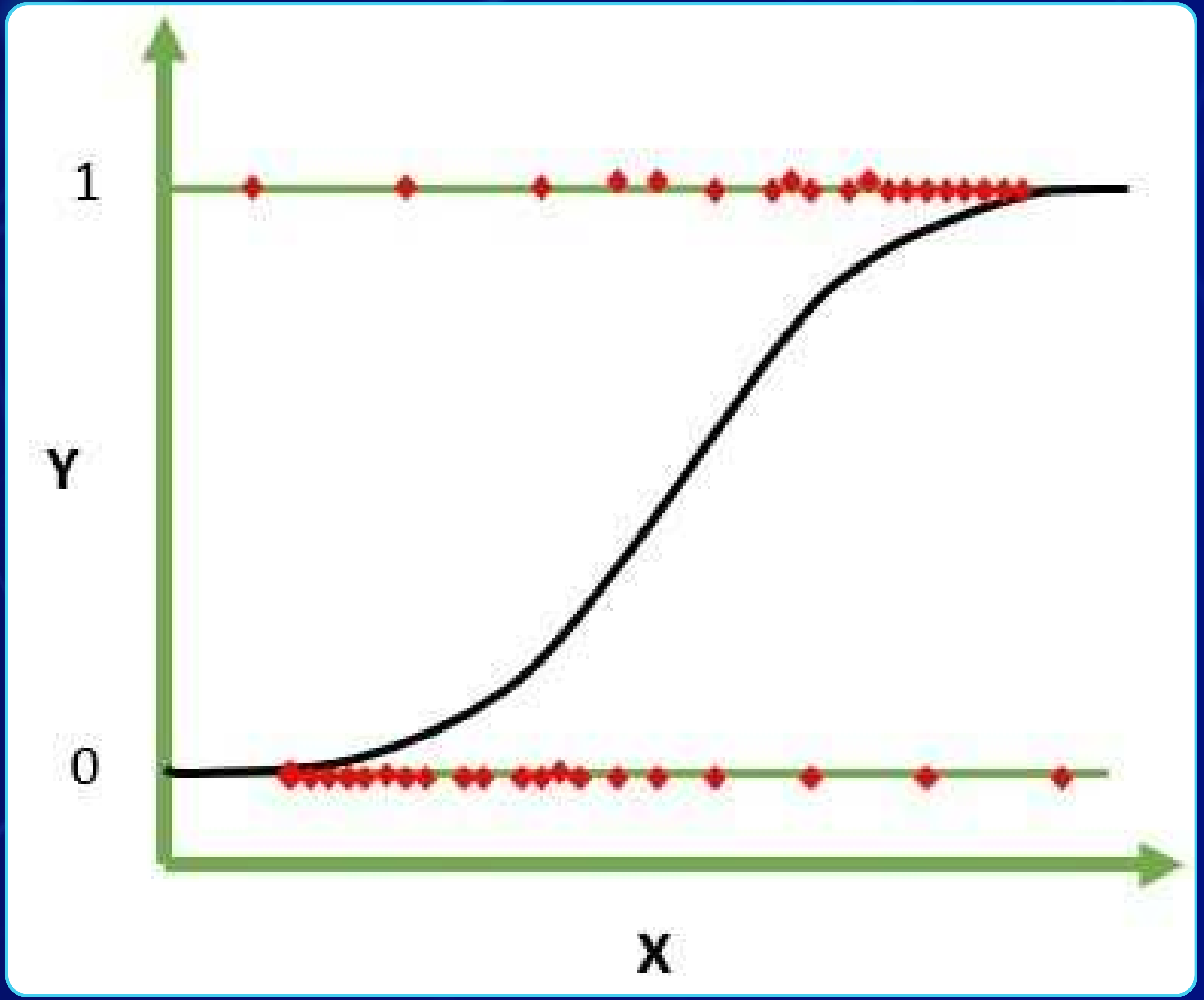
$$p = \frac{e^{(\beta_0+\beta_1 x)} \div e^{(\beta_0+\beta_1 x)}}{\left(1 + e^{(\beta_0+\beta_1 x)}\right) \div e^{(\beta_0+\beta_1 x)}}$$

Función Sigmoide

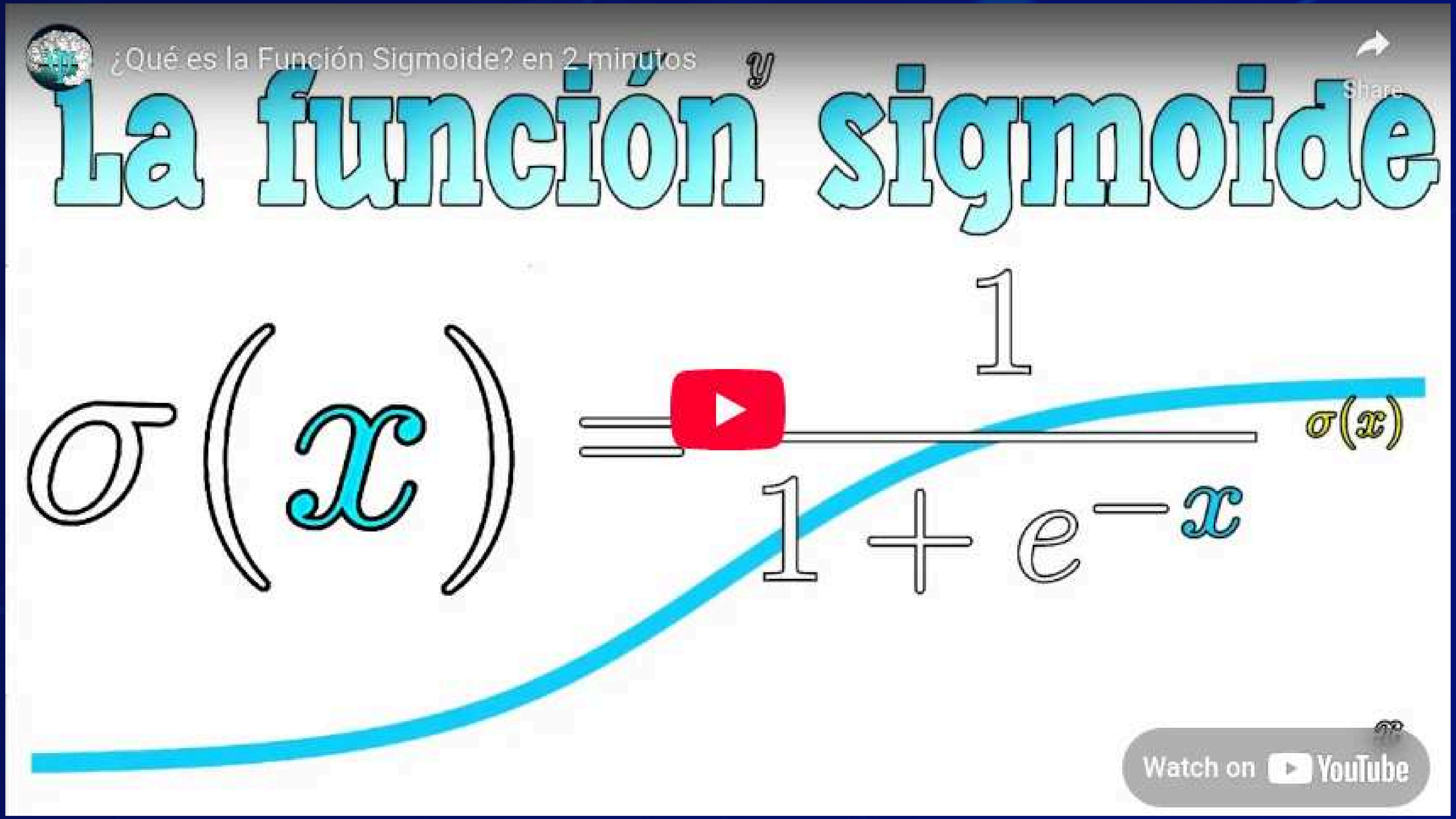
Ecuación:

$$p = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$$

Siendo *p* la probabilidad de que pertenecer a la clase 1 y el valor de la ecuación de la regresión lineal en *x*.



- Mostrar que:
  - Si *x* + , P(y=1) 1.
  - Si *x* - , P(y=1) 0.
  - En *x*=0, P(y=1)=0.5 (punto de decisión).



¿Qué es la Función Sigmoide? en 2 minutos
La función sigmoide
$\sigma(x) = \frac{1}{1+e^{-x}}$
$\sigma(x)$
Share
Watch on YouTube

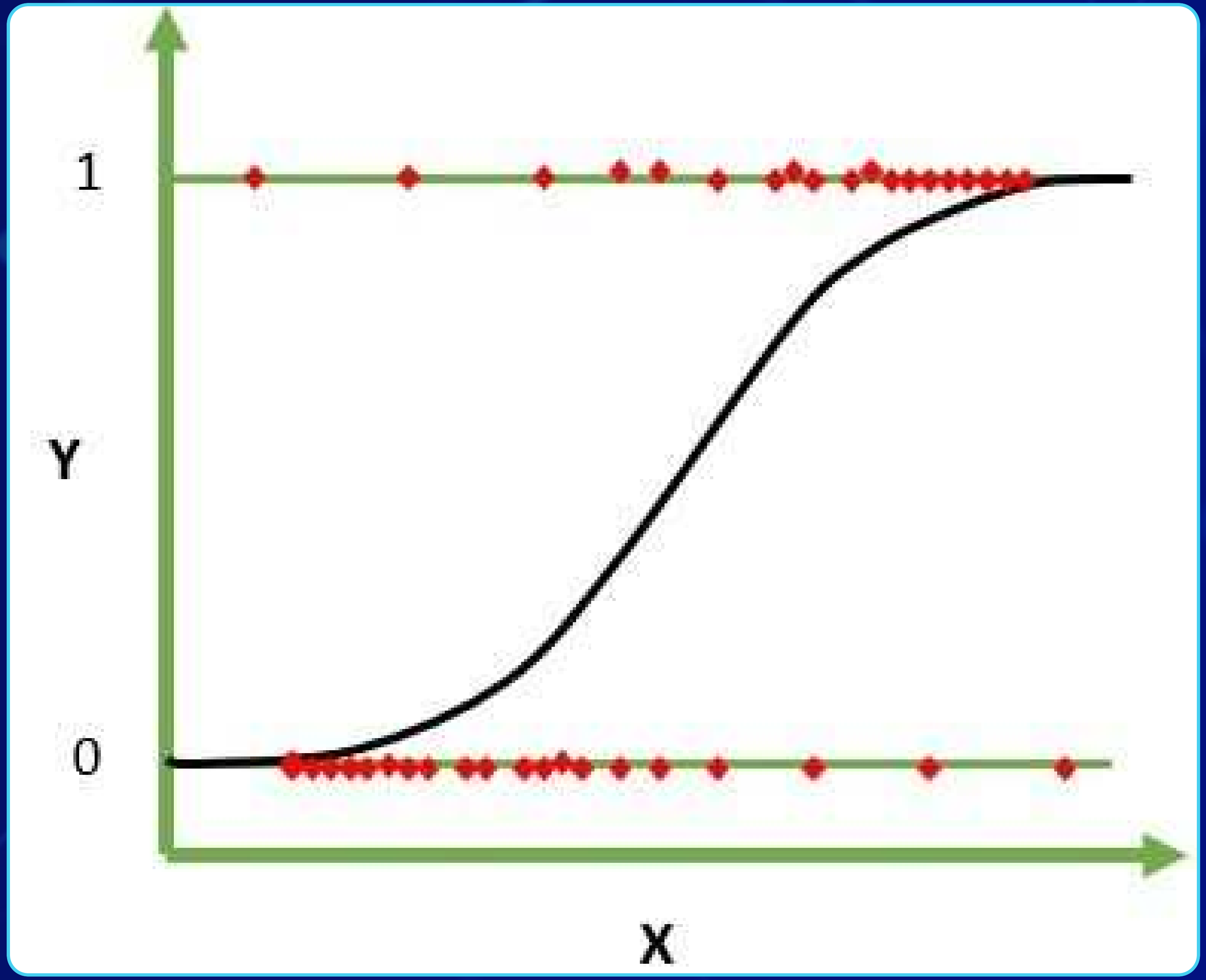
Función de Costo

Ecuación:

$$J(\beta) = \frac{1}{N} \sum_{i=1}^{N} L\left(y_i, \hat{y}_i\right)$$

Donde:

- y_i es la etiqueta real del ejemplo (0 o 1).
- $\hat{y}_i$ es la predicción del modelo para ese ejemplo (una probabilidad entre 0 y 1).



$L(y_i, \hat{y}_i)$ Es la función de pérdida.

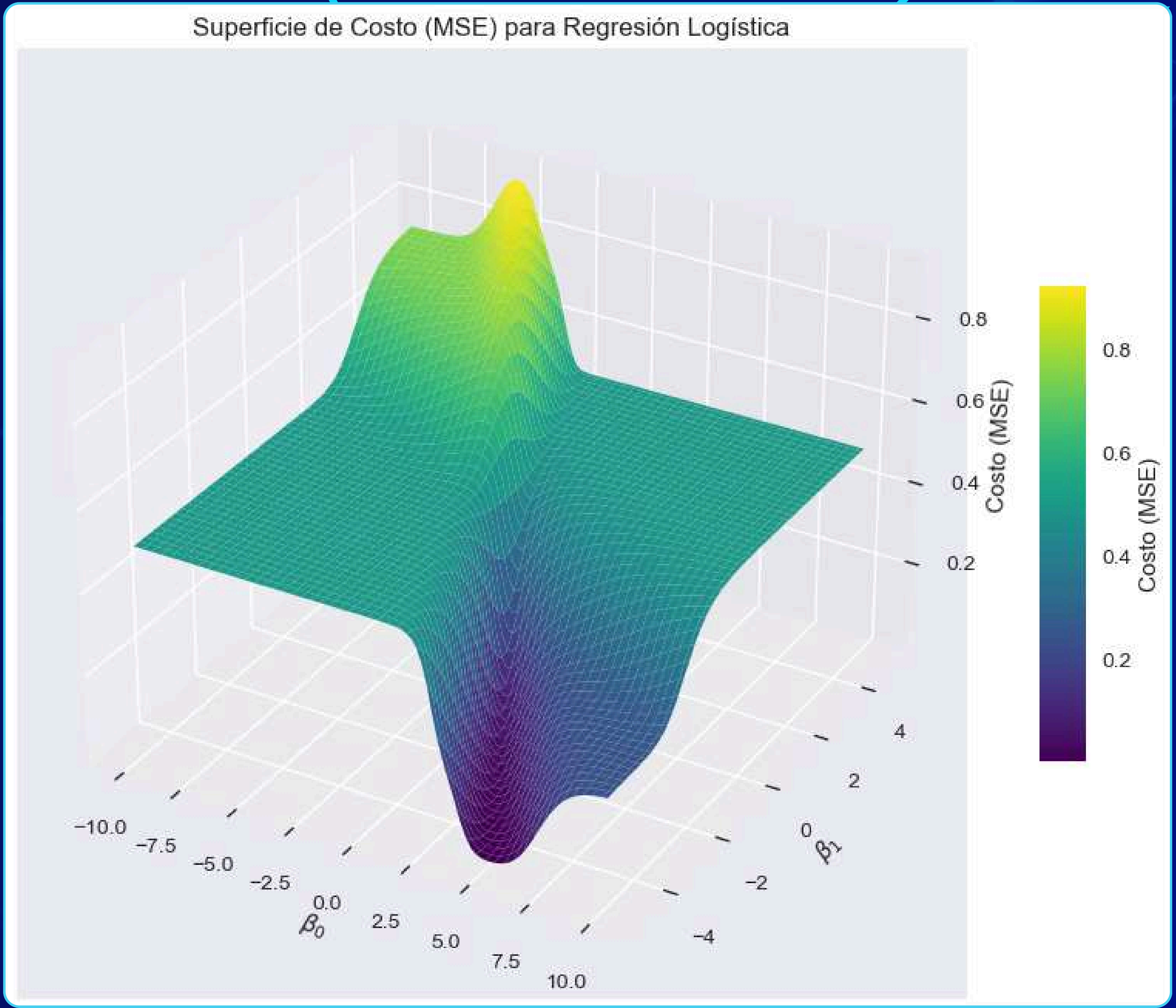
Función de Costo

Buscar una función de pérdida adecuada:

$$L(y_i, \hat{y}_i)$$

No podemos usar las funciones de Regresión Lineal, porque:

- La superficie tiene múltiples valles y crestas, lo que indica que existen varios mínimos locales (puntos bajos donde el algoritmo podría estancarse).

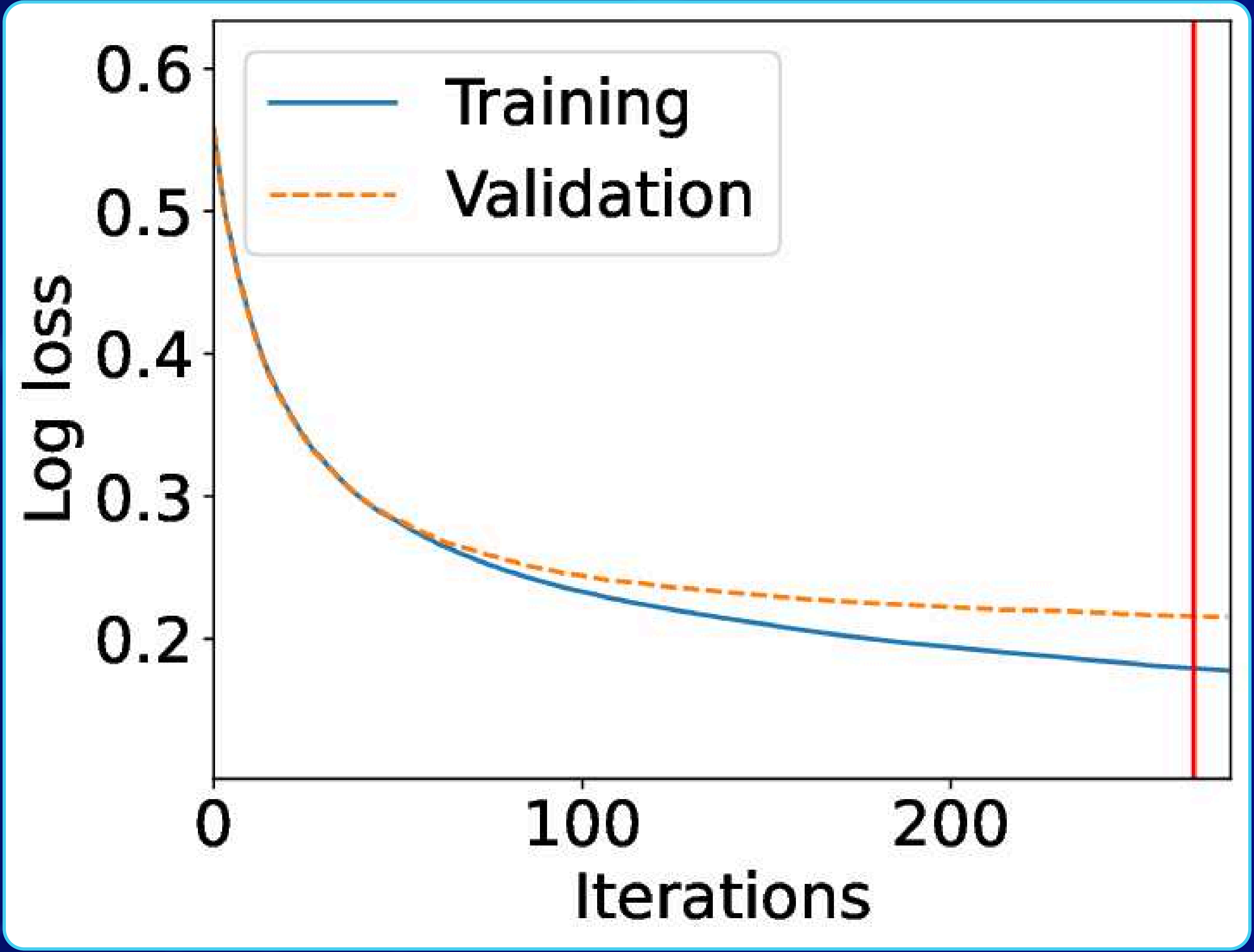


El diagrama muestra claramente que el MSE es inadecuado para regresión logística debido a su no convexidad. La Entropía Cruzada es la alternativa óptima, ya que garantiza encontrar la mejor solución.

Función de Pérdida

Log-Loss:

$$L = -[y_i ln(p_i) + (1 - y_i) ln(1 - p_i)]$$

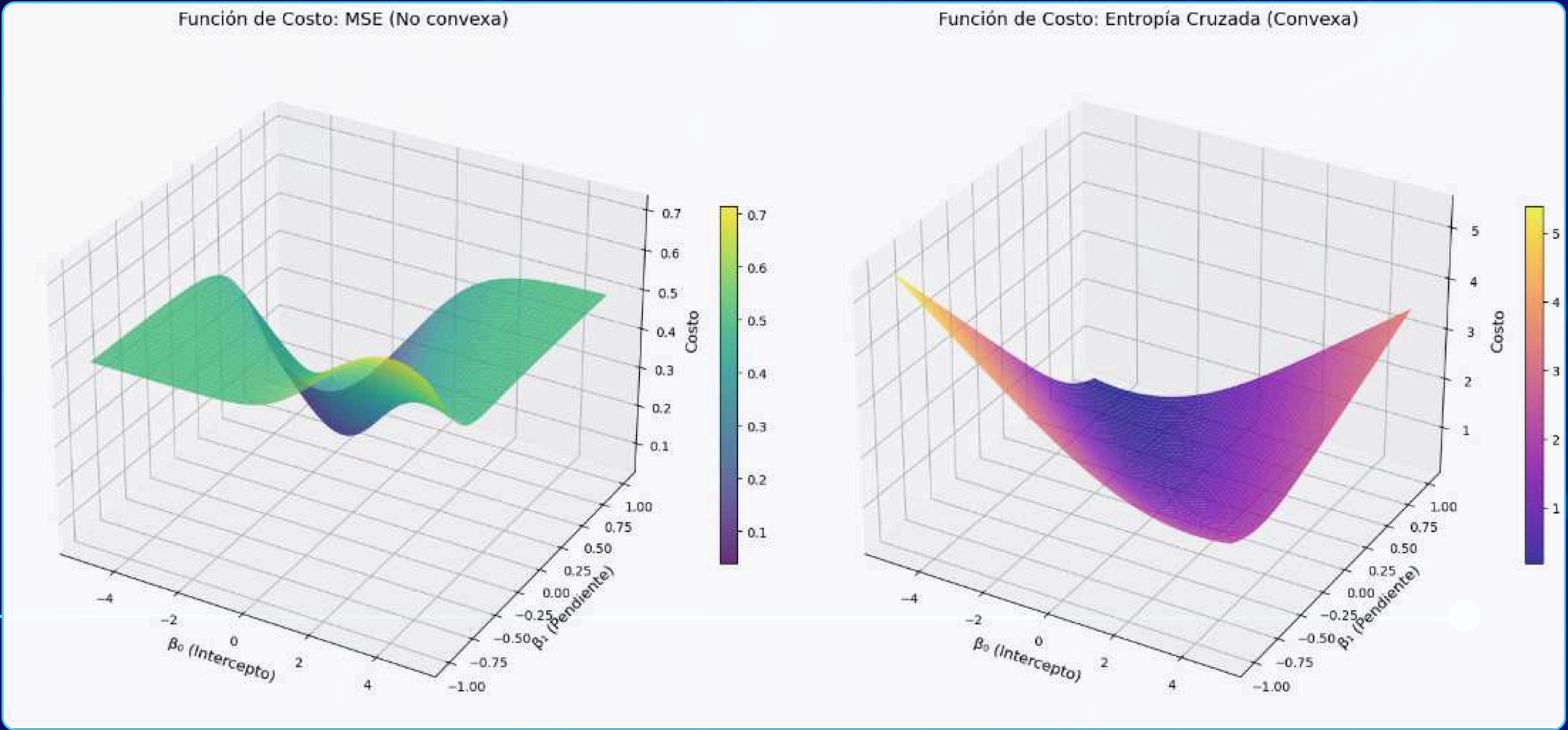


donde:

- *yi* {0,1}: Etiqueta real.
- *pi*: Probabilidad predicha de que ***yi*=1** (salida de la sigmoide).

- Mide el error para una sola observación.
- En regresión logística, se conoce como entropía cruzada binaria (binary cross-entropy).

Función de Pérdida



- A diferencia de la Entropía Cruzada (que tiene forma de tazón), el MSE no es convexo. Esto dificulta encontrar el óptimo global.

Explicación Log-Loss

Probabilidad de *yi=1*

$$P(y_i = 1|X_i) = p_i = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_{i1} + \ldots + \beta_k x_{ik})}}$$

Probabilidad de *yi=0*

$$P(y_i = 0|X_i) = 1 - p_i$$

Explicación Log-Loss

Maximum Likelihood Estimation:
Es un método estadístico utilizado para encontrar los parámetros de un modelo que maximizan la probabilidad de que se observen los datos dados.

La idea es encontrar los valores de β que maximicen la probabilidad de observar los datos que tenemos:

$$L(\beta) = \prod_{i=1}^{N} P(y_i|X_i) \qquad P(y_i \mid X_i) = \begin{cases} p_i & \text{si } y_i = 1, \\ 1 - p_i & \text{si } y_i = 0, \end{cases}$$

Explicación Log-Loss

Maximum Likelihood Estimation:
Para evitar escribir casos separados, usamos trucos matemáticos con exponentes:

$$P(y_i) = p_i^{y_i}(1 - p_i)^{1-y_i}$$

Cómo funciona:

- Si yi=1:

$$p_i^{1}(1 - p_i)^{0} = p_i \times 1 = p_i$$

- Si yi=0:

$$p_i^{0}(1 - p_i)^{1} = 1 \times (1 - p_i) = 1 - p_i$$

Ventaja:

- Compacta la expresión para todas las observaciones en un solo producto.

Explicación Log-Loss

Maximum Likelihood Estimation:
La idea es encontrar los valores de β que maximicen la probabilidad de observar los datos que tenemos:

$$L(\beta) = \prod_{i=1}^{N} \left(P(y_i = 1|X_i)^{y_i} \times P\left(y_i = 0|X_i\right)^{1-y_i}\right) = \prod_{i=1}^{N} {p_i}^{y_i}(1 - p_i)^{1-y_i}$$

Forma resumida y equivalente, usando
propiedades de las potencias

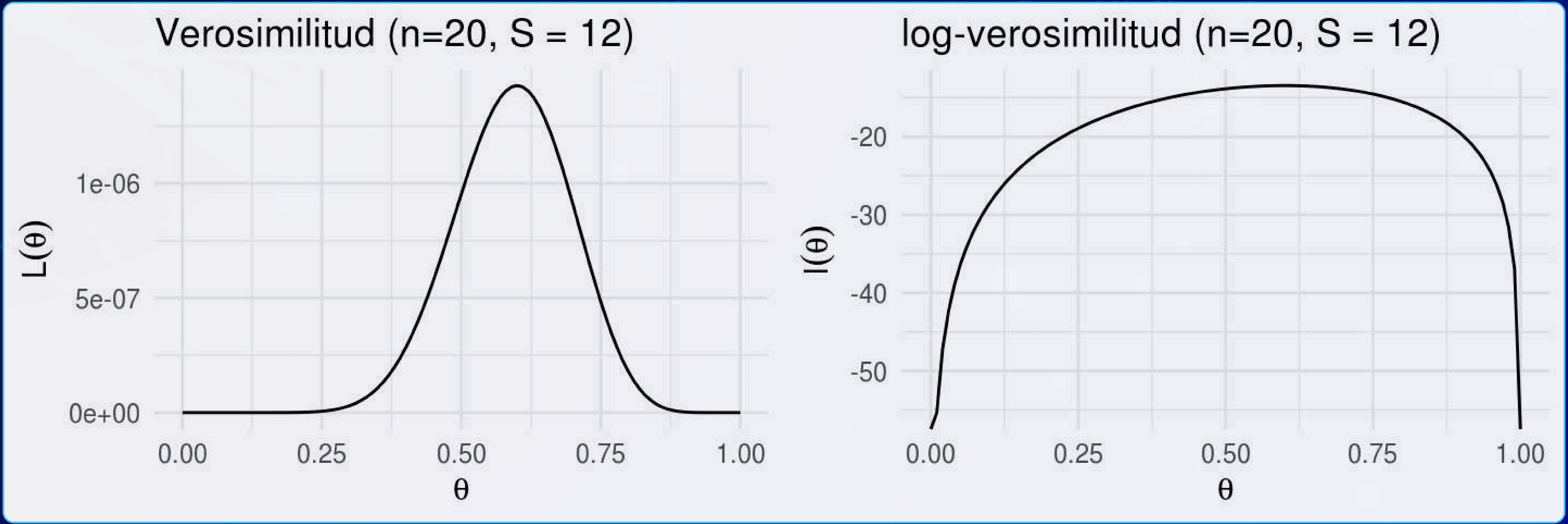
Optimización numérica (gradiente)

Log-verosimilitud:
donde l(β) representa la log-verosimilitud $\ell(\beta) = ln(L(\beta))$

$$\ell(\beta) = \sum_{i=1}^{N} \left[y_i \ln\left(p_i\right) + \left(1 - y_i\right) \ln\left(1 - p_i\right) \right]$$

Trabajar con el logaritmo de la verosimilitud simplifica los cálculos (los productos se convierten en sumas)

Optimización (gradiente)



Función de Costo

El objetivo es minimizar *J(β)*, lo que equivale a ajustar los parámetros *β0,β1...* para hacer las predicciones lo más cercanas posible a las etiquetas reales *yi*.

- Se basa en la log-loss, que penaliza predicciones incorrectas con alta confianza.
- Es diferenciable y adecuada para optimización mediante gradiente descendente.
- Nos permite ajustar los parámetros *β0, β1* para mejorar el desempeño del modelo.

$$J(\beta) = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \ln(p_i) + (1 - y_i)\ln(1 - p_i)\right]$$

Estimación de los Coeficientes β

Log-verosimilitud:
La idea es encontrar los valores de β que maximicen la probabilidad de observar los datos que tenemos:

$$\ell(\beta) = \sum_{i=1}^{N} [y_i \ln\left(\frac{1}{1+e^{-z_i}}\right) + (1-y_i)\ln\left(\frac{1}{1+e^{-z_i}}\right)]$$

Donde:

$$z_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_k x_{ik}$$

Se despeja la función de Log-Verosimilitud para los *Betas* y se derivan las funciones

Estimación de los Coeficientes β

Gradiente Descendente: Actualización de Coeficientes

El gradiente descendente actualiza los coeficientes iterativamente:

$$\beta_j := \beta_j - \alpha \cdot \frac{\partial J}{\partial \beta_j}$$

donde:

- α: Tasa de aprendizaje (en este ejemplo se usará α=0.1).
- $\frac{\partial J}{\partial \beta_j}$: Derivada parcial de la función de costo respecto a βj.

$$\frac{\partial J}{\partial \beta_0} = \frac{1}{N} \sum_{i=1}^{N} (p_i - y_i)$$

$$\frac{\partial J}{\partial \beta_1} = \frac{1}{N} \sum_{i=1}^{N} (p_i - y_i) \cdot x_i$$

Estimación de los Coeficientes β

Aplicar método del Gradiente Descendente:

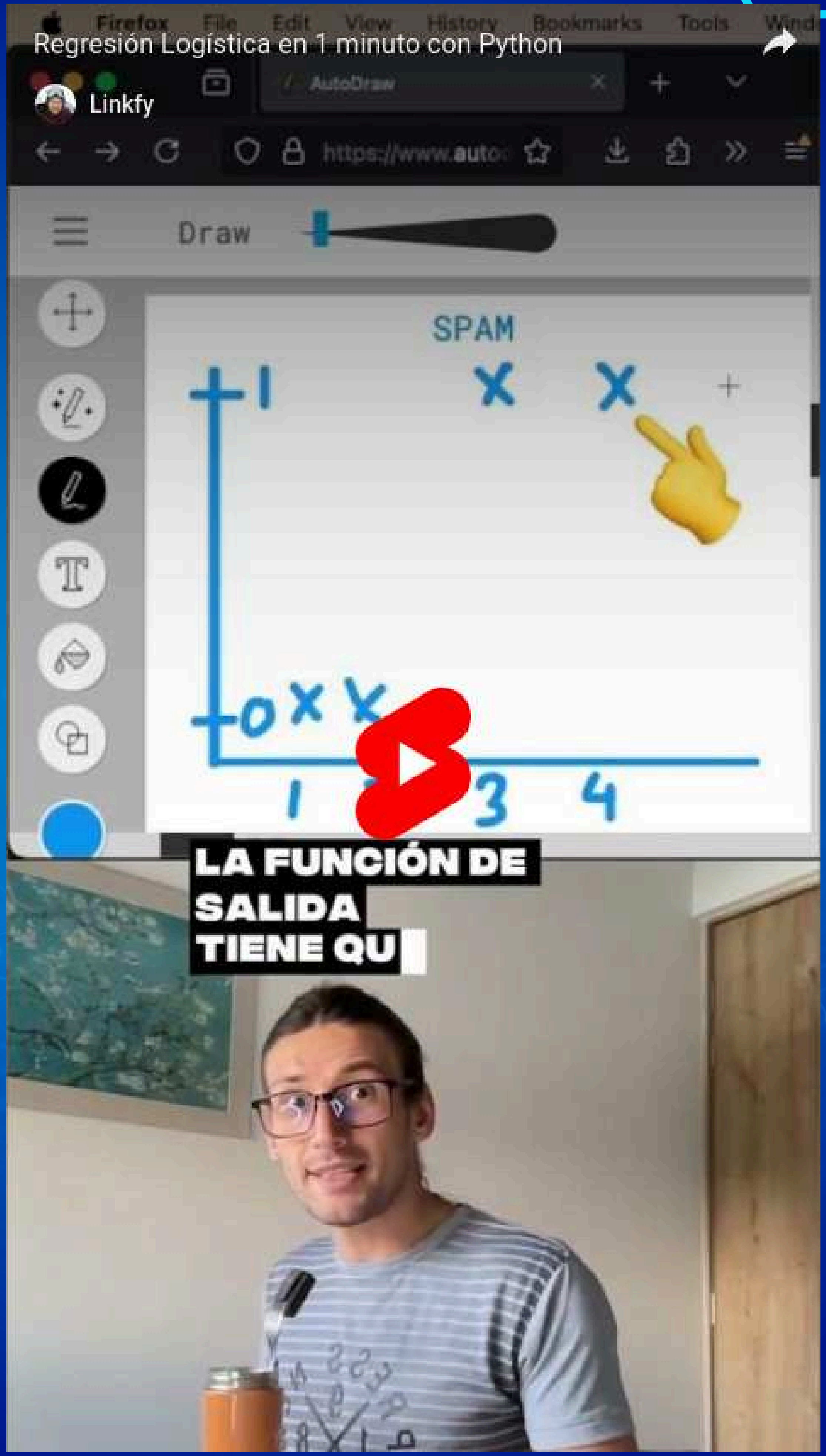
1. Inicializar β con valores aleatorios.
2. Calcular el gradiente $\frac{\partial J}{\partial \beta_j}$
3. Actualizar β en dirección del gradiente

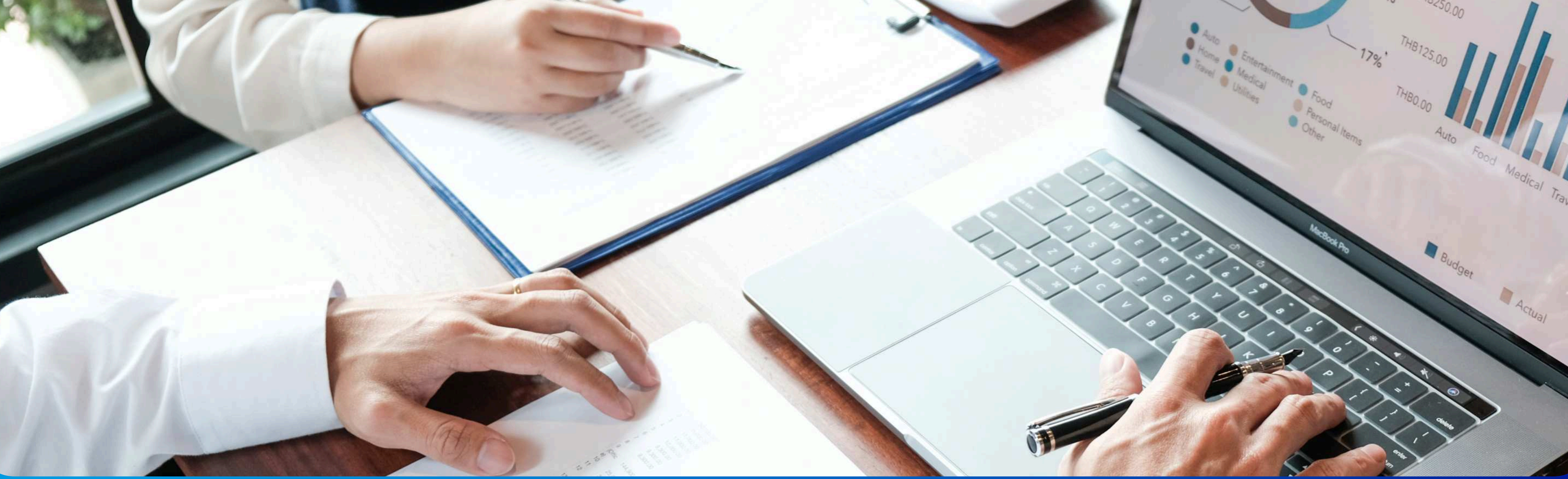
$$\beta_j := \beta_j - \alpha \cdot \frac{\partial J}{\partial \beta_j}$$

El algoritmo se detiene hasta que:

- El valor de J es suficientemente pequeño.
- Se alcanzan las iteraciones máximas permitidas.

Regularización





Regresión Logística: Ejemplo Paso a Paso

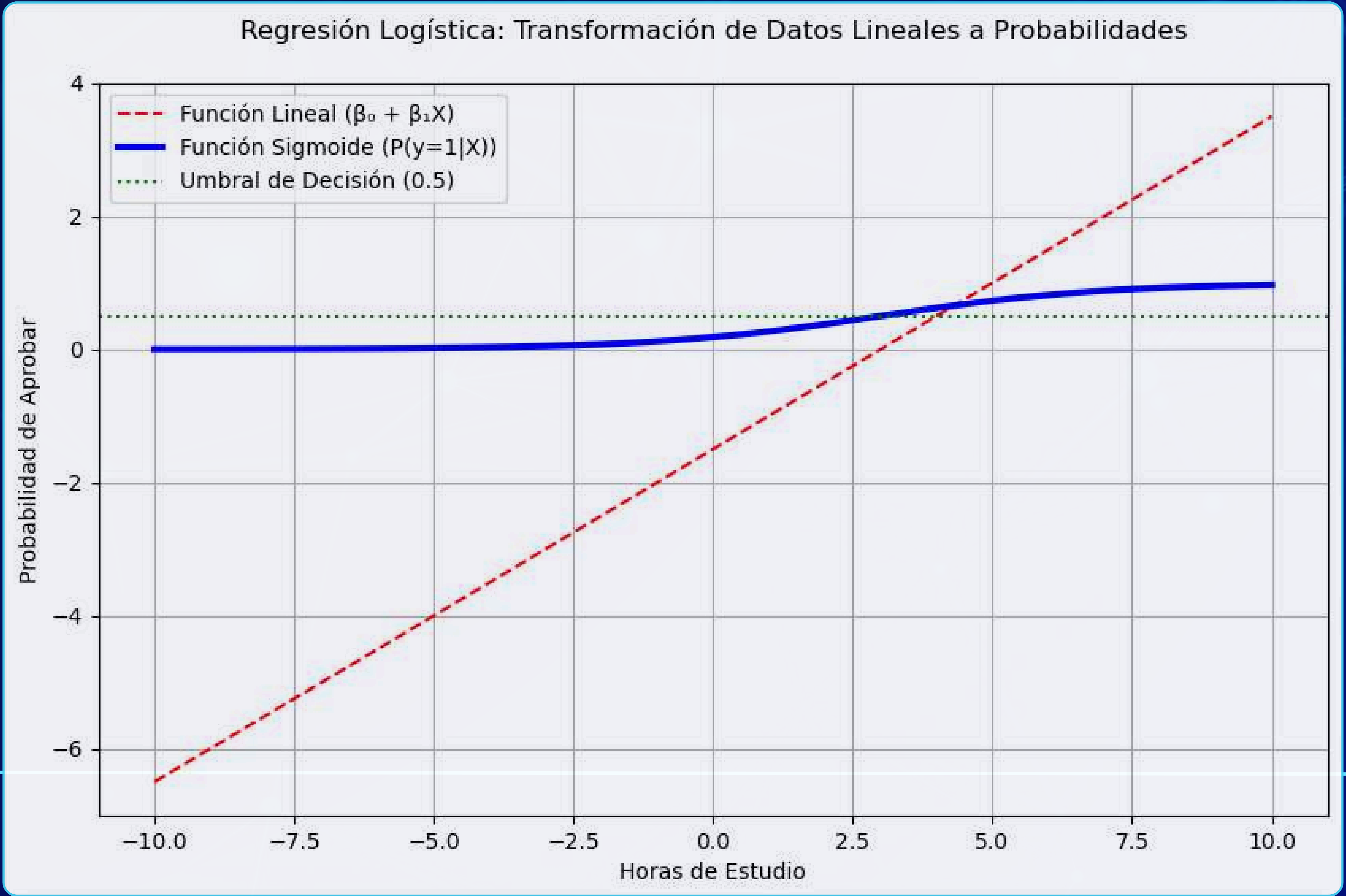
Regresión Logística: Ejemplo Paso a Paso

Clasificación de Alumnos Aprobados o Reprobados

- Queremos predecir si un estudiante aprueba (y=1) o reprueba (y=0) un examen.
- Basamos la predicción en una característica: Horas de estudio (X) .
- Usaremos un modelo de regresión logística simple.

Horas de Estudio (X)	Aprobado (y)
1	0
3	0
5	1
7	1

Regresión Logística: Ejemplo Paso a Paso



- Muestra cómo una relación lineal se transforma en probabilidades mediante la sigmoide (Debido a esa transformación, se tiene el nombre de Regresión).
- Es complicado modelar el problema con Regresión Lineal, por eso se usa Regresión Logística.
- El umbral de decisión (0.5) separa las clases.

Regresión Logística: Ejemplo Paso a Paso

Modelo de Regresión Logística

Fórmula general:

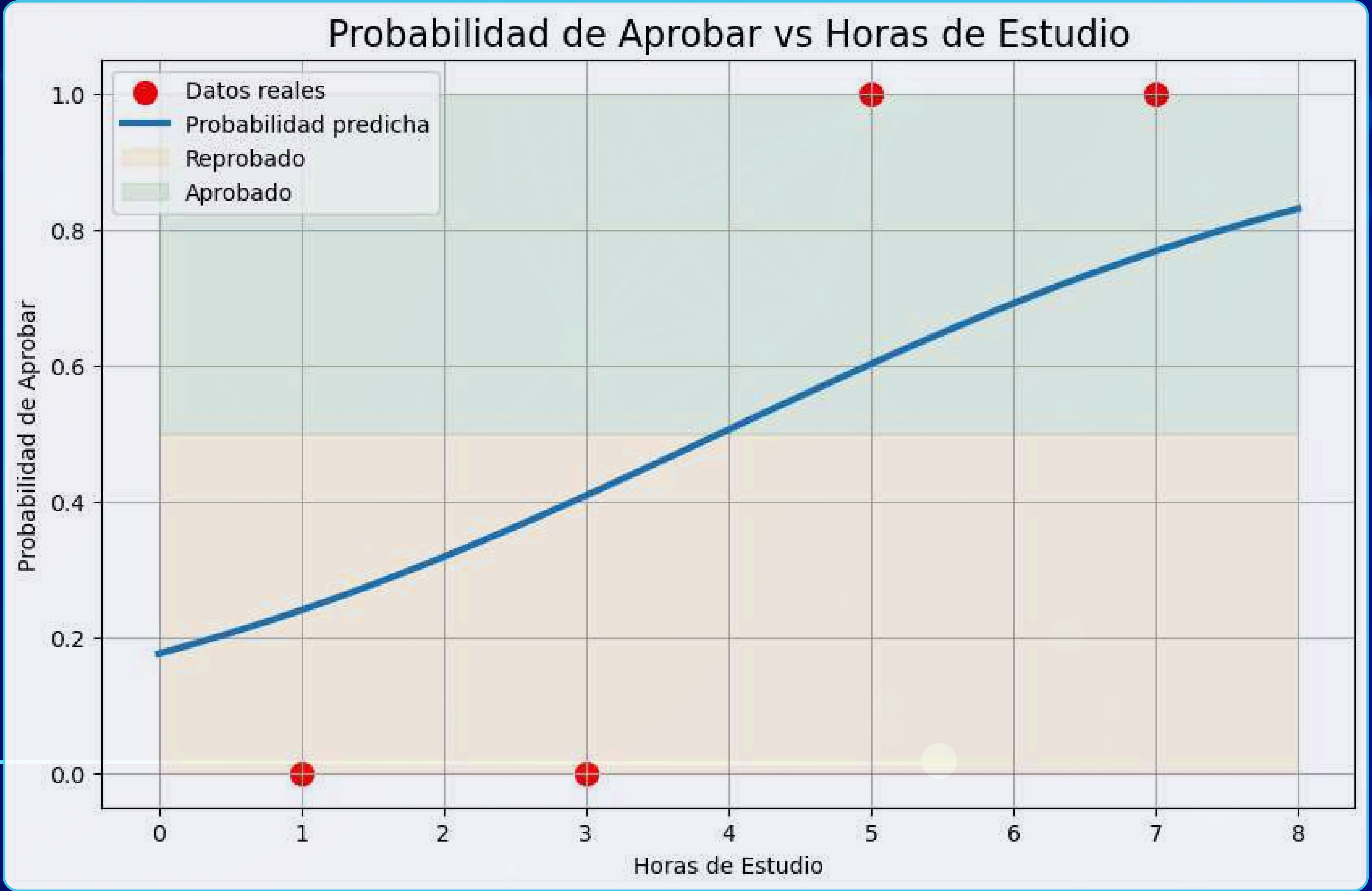
$$P(y_i = 1|X_i) = \frac{1}{1 + e^{-z}}$$

donde:

$$z = \beta_0 + \beta_1 X$$

Regresión Logística: Ejemplo Paso a Paso

- Encontrar un modelo que asigne probabilidades a nuevos datos.
- Las áreas sombreadas representan las zonas de decisión.



Regresión Logística: Ejemplo Paso a Paso

Función de Costo

$$J(\beta_0, \beta_1) = -\frac{1}{4}\sum_{i=1}^{4}[y_i \ln(p_i) + (1 - y_i)\ln(1 - p_i)])$$

Usando la ecuación Sigmoide:

$$p_i = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_i)}}$$

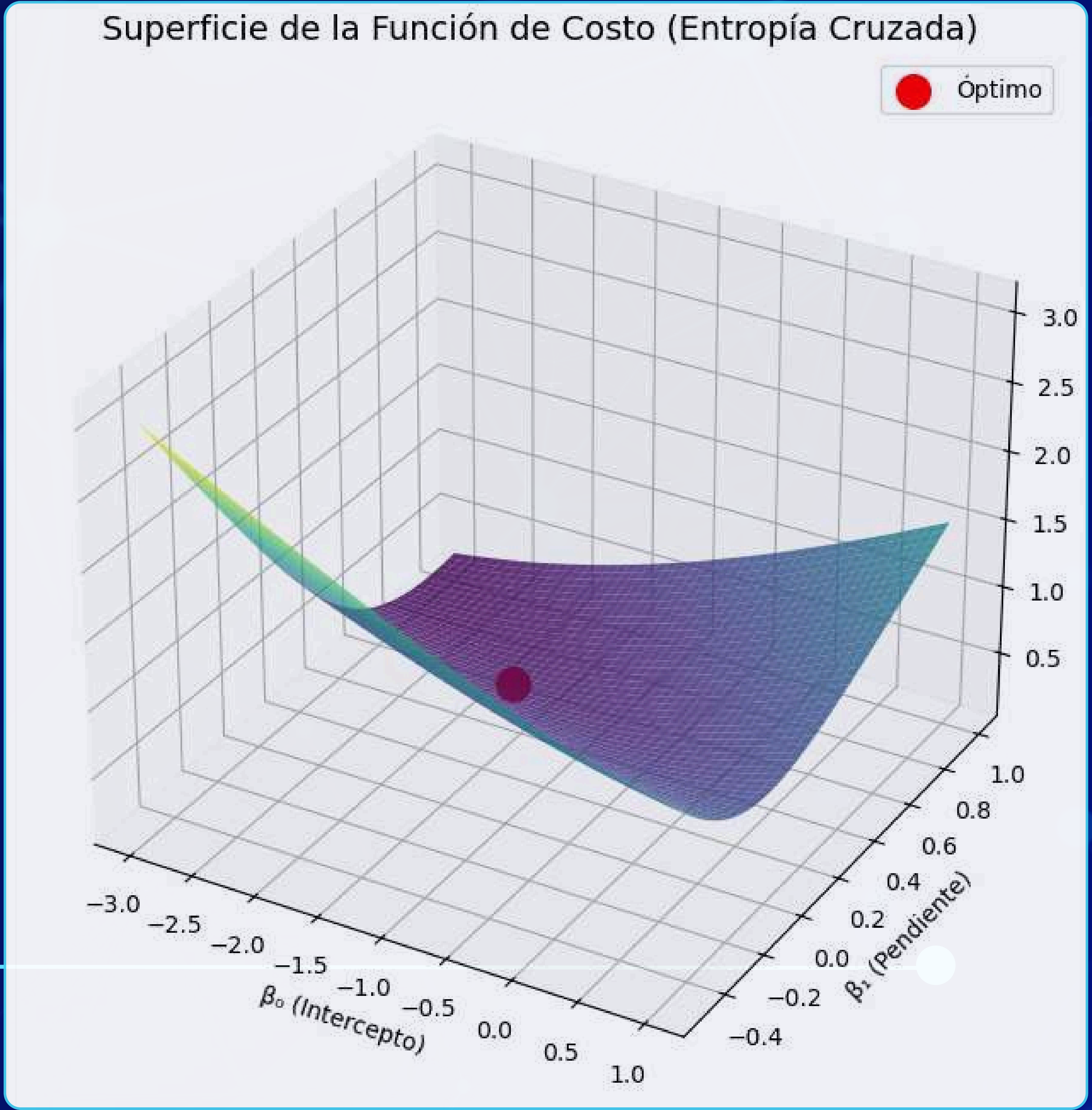
Calcular *pi* para cada dato:
Con *β0*=0 y *β1*=0:

$$p_i = \frac{1}{1 + e^{-(0 + 0X_i)}}$$

Regresión Logística: Ejemplo Paso a Paso

Función de Costo

- Visualiza cómo el costo varía con diferentes valores de β0 y β1.
- El punto rojo marca el mínimo encontrado por el modelo.



Regresión Logística: Ejemplo Paso a Paso

Iteración #1

Iteraciones del Gradiente Descendente

Calcular las derivadas parciales:

- Para β0:

$$\frac{\partial J}{\partial \beta_0} = \frac{1}{4}\sum_{i=1}^{4}(p_i - y_i)$$

Sustituyendo:

$$\frac{\partial J}{\partial \beta_0} = \frac{1}{4}[(0.5-0)+(0.5-0)+(0.5-1)+(0.5-1)] = 0$$

Horas de Estudio (X)	pi	Aprobado (y)
1	0.5	0
3	0.5	0
5	0.5	1
7	0.5	1

Regresión Logística: Ejemplo Paso a Paso

Iteración #1

Iteraciones del Gradiente Descendente

Calcular las derivadas parciales:

- Para β1:

$$\frac{\partial J}{\partial \beta_1} = \frac{1}{4}\sum_{i=1}^{4}(p_i - y_i)\cdot x_i$$

Sustituimos:

$$\frac{\partial J}{\partial \beta_1} = \frac{1}{4}[(0.5-0)\,(1) + (0.5-0)\,(3) + (0.5-1)\,(5) + (0.5-1)\,(7)] = -1.0$$

Horas de Estudio (X)	pi	Aprobado (y)
1	0.5	0
3	0.5	0
5	0.5	1
7	0.5	1

Regresión Logística: Ejemplo Paso a Paso

Actualizar coeficientes (con tasa de aprendizaje α=0.1)

$$\beta_0^{nuevo} = 0 - 0.1 \cdot 0 = 0$$

$$\beta_1^{nuevo} = 0 - 0.1 \cdot (-1.0) = 0.1$$

Calcular el Costo (Usar Función de Entropía Cruzada/Log-Loss

$$J(\beta) = -\frac{1}{4}\sum_{i=1}^{4}[y_i \ln(p_i) + (1 - y_i)\ln(1 - p_i))] \approx 0.693\ldots$$

Resultados después de Iteración 1:

β0=0,β1=0.1, J=0.6931

Regresión Logística: Ejemplo Paso a Paso

Iteración #2

Calcular nuevas *pi*

$$p_i = \frac{1}{1 + e^{-(0+0.1X_i)}}$$

Horas de Estudio (X)	pi	Aprobado (y)
1	0.5249...	0
3	0.5744...	0
5	0.6224...	1
7	0.6681...	1

Regresión Logística: Ejemplo Paso a Paso

Iteración #2

Iteraciones del Gradiente Descendente

Calcular las derivadas parciales:

- Para β0:

$$\frac{\partial J}{\partial \beta_0} = \frac{1}{4}\sum_{i=1}^{4}(p_i - y_i)$$

Horas de Estudio (X)	pi	Aprobado (y)
1	0.5249...	0
3	0.5744...	0
5	0.6224...	1
7	0.6681...	1

Sustituyendo:

$$\frac{\partial J}{\partial \beta_0} = \frac{1}{4}[(0.525 - 0) + (0.574 - 0) + (0.622 - 1) + (0.668 - 1)] \approx -0.09725$$

Regresión Logística: Ejemplo Paso a Paso

Iteración #2

Iteraciones del Gradiente Descendente

Calcular las derivadas parciales:

- Para β1:

$$\frac{\partial J}{\partial \beta_1} = \frac{1}{4}\sum_{i=1}^{4}(p_i - y_i)\cdot x_i$$

Sustituimos:

$$\frac{\partial J}{\partial \beta_1} = \frac{1}{4}[(0.525-0)(1)+(0.574-0)(3)+(0.622-1)(5)+(0.668-1)(7)] \approx -0.49175$$

Horas de Estudio (X)	pi	Aprobado (y)
1	0.5249...	0
3	0.5744...	0
5	0.6224...	1
7	0.6681...	1

Regresión Logística: Ejemplo Paso a Paso

Actualizar coeficientes (con tasa de aprendizaje α=0.1)

$$\beta_0{}^{nuevo} = 0 - 0.1 \cdot (0.09725) \approx -0.0097$$
$$\beta_1{}^{nuevo} = 0.1 - 0.1 \cdot (-0.49175) \approx 0.1492$$

Calcular el Costo (Usar Función de Entropía Cruzada/Log-Loss

$$J(\beta) = -\frac{1}{4}\sum_{i=1}^{4}[y_i \ln(p_i) + (1-y_i)\ln(1-p_i))] \approx 0.619\ldots$$

Resultados después de Iteración 2:

β0=-0.0097, β1=0.1492, J=0.6190

Regresión Logística: Ejemplo Paso a Paso

Iteración #3

Calcular nuevas *pi*

$$p_i = \frac{1}{1 + e^{-(-0.0097+0.1492X_i)}}$$

Horas de Estudio (X)	pi	Aprobado (y)
1	0.5348...	0
3	0.6077...	0
5	0.6761...	1
7	0.7378...	1

Regresión Logística: Ejemplo Paso a Paso

Iteración #3

Iteraciones del Gradiente Descendente

Calcular las derivadas parciales:

- Para β0:

$$\frac{\partial J}{\partial \beta_0} = \frac{1}{4}\sum_{i=1}^{4}(p_i - y_i)$$

Horas de Estudio (X)	pi	Aprobado (y)
1	0.5348...	0
3	0.6077...	0
5	0.6761...	1
7	0.7378...	1

Sustituyendo:

$$\frac{\partial J}{\partial \beta_0} = \frac{1}{4}[(0.534 - 0) + (0.607 - 0) + (0.676 - 1) + (0.737 - 1)] \approx 0.1385$$

Regresión Logística: Ejemplo Paso a Paso

Iteración #3

Iteraciones del Gradiente Descendente

Calcular las derivadas parciales:

- Para β1:

$$\frac{\partial J}{\partial \beta_1} = \frac{1}{4}\sum_{i=1}^{4}(p_i - y_i)\cdot x_i$$

Horas de Estudio (X)	pi	Aprobado (y)
1	0.5348...	0
3	0.6077...	0
5	0.6761...	1
7	0.7378...	1

Sustituimos:

$$\frac{\partial J}{\partial \beta_1} = \frac{1}{4}[(0.534-0)(1)+(0.607-0)(3)+(0.676-1)(5)+(0.737-1)(7)] \approx -0.2765$$

Regresión Logística: Ejemplo Paso a Paso

Actualizar coeficientes (con tasa de aprendizaje α=0.1)

$$\beta_0^{nuevo} = -0.0097 - 0.1 \cdot (0.1385) \approx 0.0236$$

$$\beta_1^{nuevo} = 0.1492 - 0.1 \cdot (-0.2765) \approx 0.176$$

Calcular el Costo (Usar Función de Entropía Cruzada/Log-Loss

$$J(\beta) = -\frac{1}{4}\sum_{i=1}^{4}[y_i \ln(p_i) + (1 - y_i)\ln(1 - p_i))] \approx 0.599\ldots$$

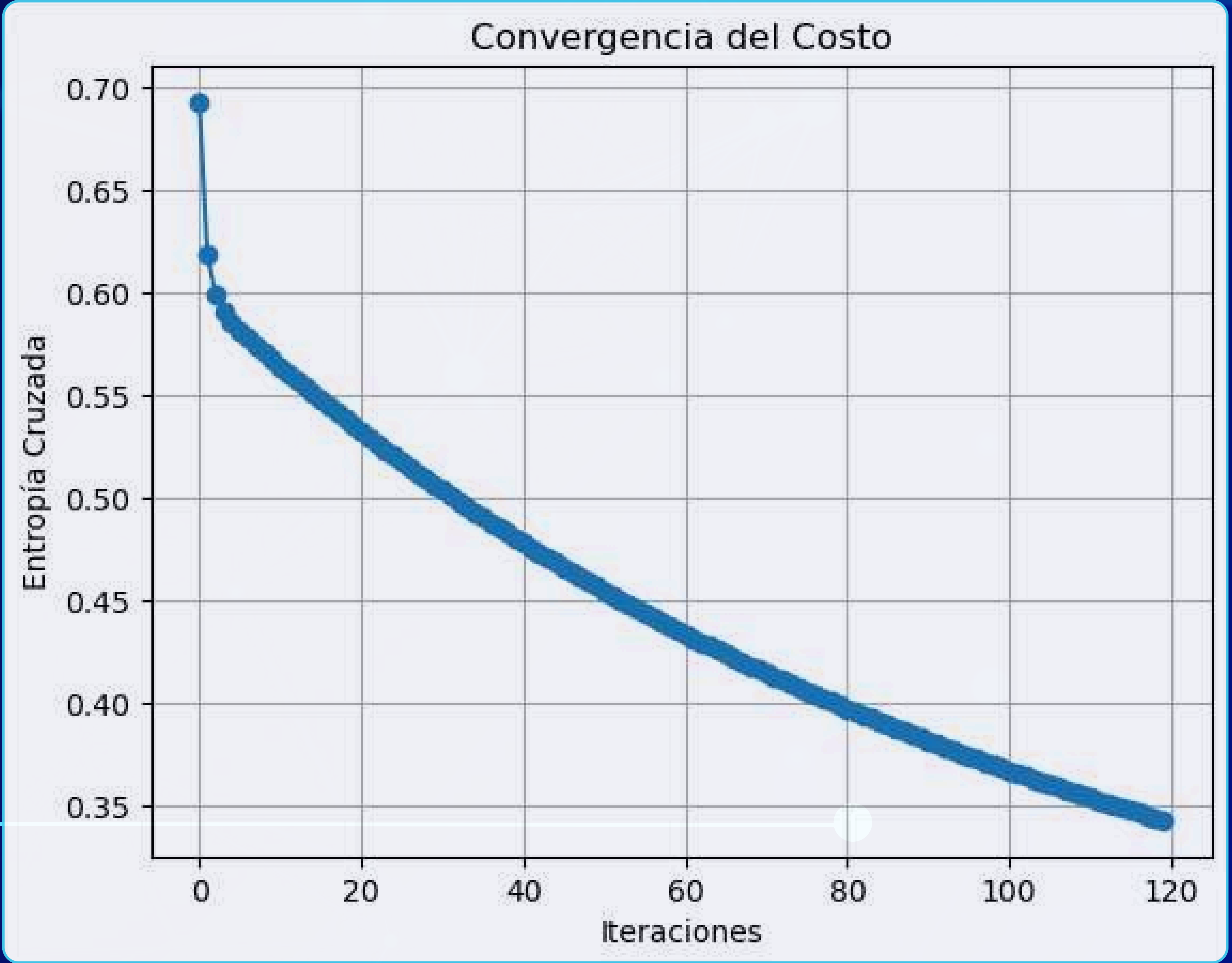
Resultados después de Iteración 3:

β0=0.0236, β1=0.1769, J=0.5992

Regresión Logística: Ejemplo Paso a Paso

Función de Costo

- Visualiza cómo el costo varía va disminuyendo en cada iteración
- En las primeras iteraciones, la entropía cruzada disminuye rápidamente, indicando que el modelo está aprendiendo de manera eficiente y mejorando sus predicciones.
- Después de muchas iteraciones (en este caso, alrededor de 120), el costo se estabiliza alrededor de un valor cercano a 0.35. Esto ocurre porque el modelo ha encontrado una configuración de parámetros que minimiza la entropía cruzada en el conjunto de datos de entrenamiento.



Regresión Logística: Ejemplo Paso a Paso

Iteración #100
Calcular nuevas *pi*

$$p_i = \frac{1}{1 + e^{-(-1.4266 + 0.5097 X_i)}}$$

Horas de Estudio (X)	pi	Aprobado (y)
1	0.2855...	0
3	0.5256...	0
5	0.7543...	1
7	0.8948...	1

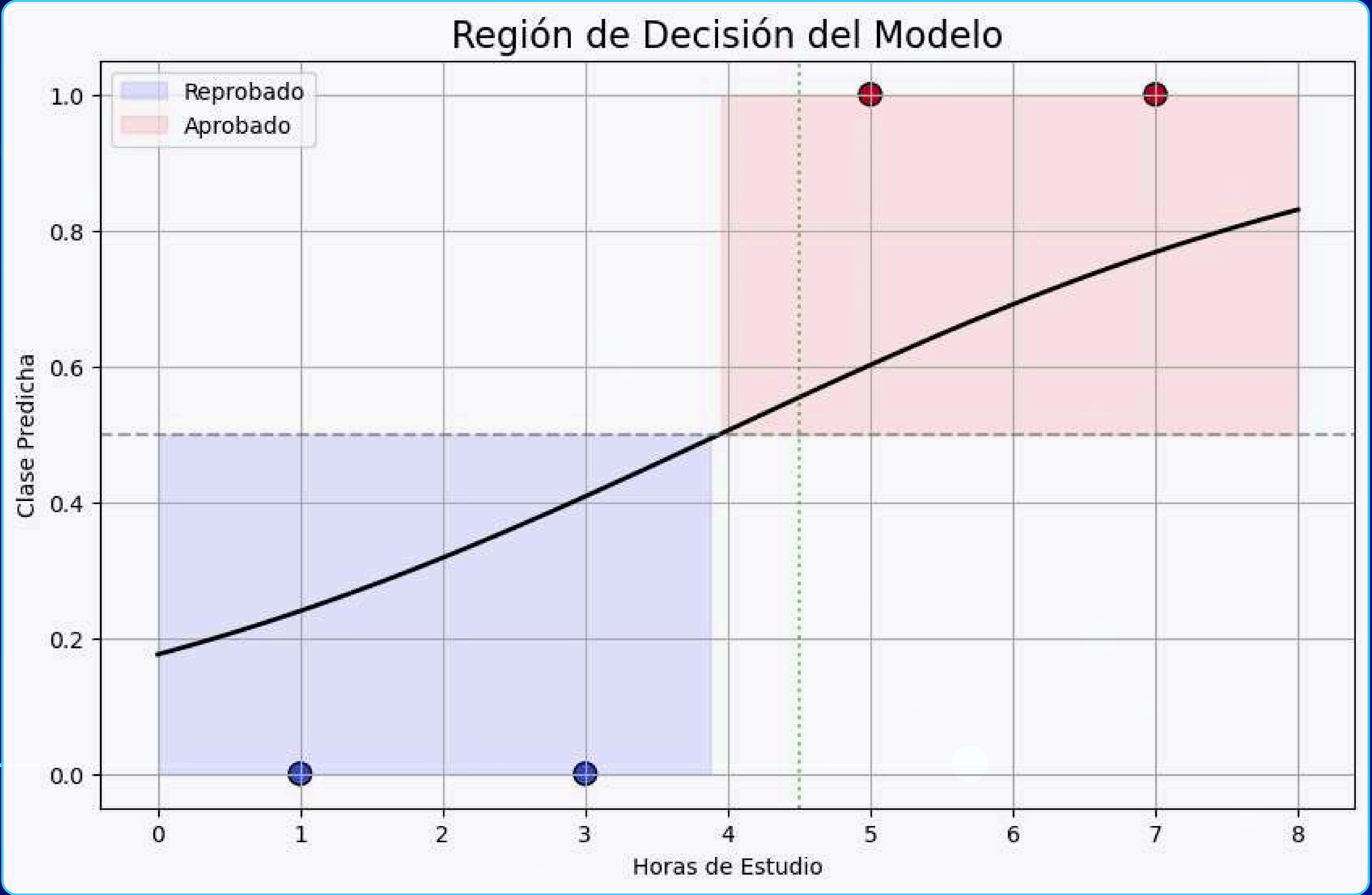
Resultados después de Iteración 100:

$\beta 0$=-1.4381, $\beta 1$=0.5123, J=0.3688

Regresión Logística: Ejemplo Paso a Paso

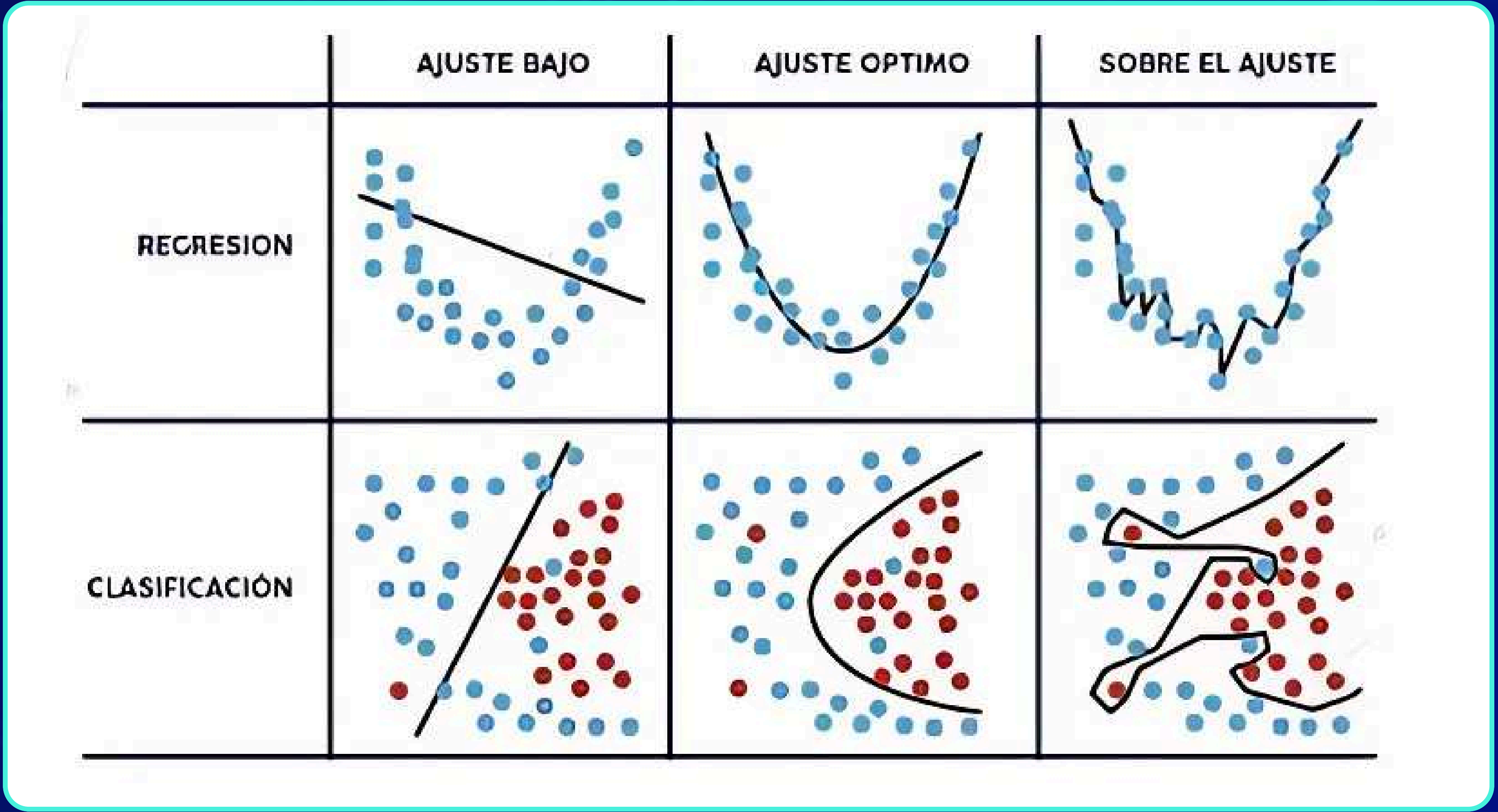
Función de Costo

- Muestra cómo el modelo divide el espacio en zonas de clasificación.
- El punto de cruce (≈4.5 horas) es donde P(y=1)=0.5.



Sobreajuste y subajuste

- El subajuste ocurre cuando el modelo presenta un sesgo muy alto y no puede capturar los patrones complejos de los datos. Esto genera mayores errores de entrenamiento y validación, ya que el modelo no es lo suficientemente complejo como para clasificar los datos subyacentes.
- El sobreajuste es lo opuesto, ya que el modelo es demasiado complejo (o superior) y captura incluso el ruido de los datos.



Matriz de Confusión

Entropía Cruzada

Estas métricas evalúan cómo bien el modelo predice las probabilidades de pertenecer a cada clase.

Entropía Cruzada Binaria (Log Loss)

La entropía cruzada binaria mide la discrepancia entre las probabilidades predichas (P(yi=1)) y las etiquetas reales (yi).

- Menor es mejor: Cuanto más cercano sea el valor de L a cero, mejor ajusta el modelo.
- Ideal para probabilidades: Es la métrica preferida cuando el interés está en evaluar la calidad de las probabilidades predichas.

Curva ROC y AUC

La curva ROC es una herramienta gráfica que mide la capacidad de un modelo para discriminar entre clases. Representa la relación entre:

- True Positive Rate (TPR) : La sensibilidad o recall. Es la proporción de casos positivos correctamente identificados.
- False Positive Rate (FPR) : La tasa de falsos positivos. Es la proporción de casos negativos incorrectamente identificados como positivos.

La curva ROC traza los valores de TPR contra FPR para diferentes umbrales de clasificación. Un buen modelo tendrá una curva que se acerca a la esquina superior izquierda del gráfico, donde:

- **TPR = 1** : Todos los casos positivos están correctamente identificados.
- **FPR = 0** : Ningún caso negativo está incorrectamente identificado.

Curva ROC y AUC

- El AUC es el área bajo la curva ROC. Mide la capacidad del modelo para discriminar entre clases. Un AUC cercano a 1 indica un excelente modelo, mientras que un valor cercano a 0.5 indica que el modelo no discrimina mejor que el azar.