



Universidad Autónoma de Coahuila  
Centro de Investigación en Matemáticas Aplicadas



# MODELO DE REGRESIÓN LOGÍSTICA

Autora:  
Angela García

Programación

Tutor:  
Niels Martínez

# ÍNDICE

1. Introducción
2. Algoritmo
3. Modelo manual
4. Evaluación
5. Resultados de Comparación
6. Conclusiones

# INTRODUCCIÓN

La regresión logística es un método que modela la probabilidad de pertenecer a una de dos clases.

A diferencia de la regresión lineal, produce probabilidades que permiten asignar cada caso a la clase más probable.



# ALGORITMO

## Descripción

La probabilidad de un evento se obtiene con una función sigmoide. Primero se calcula una combinación lineal de las variables y, al aplicar la sigmoide a ese resultado, se obtiene la probabilidad de la clase positiva.

La estimación de parámetros se hace minimizando la pérdida logística (log-loss). Y para prevenir el sobreajuste y limitar coeficientes muy grandes, se usa regularización (lambda).

## Fórmulas

- Sigmoide:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

- Z-Value:

$$z = \beta_0 + \sum_{i=1}^n \beta_i x_i$$

- Log-loss:

$$J(\beta) = -\frac{1}{m} \sum_{i=1}^m [y_i \log (\hat{y}_i) + (1 - y_i) \log (1 - \hat{y}_i)]$$

- Lambda Regulurizada:

$$J_{\text{reg}}(\beta) = J(\beta) + \frac{\lambda}{2m} \sum_{j=1}^n \beta_j^2$$

## Descripción

Los parámetros se optimizan con gradiente descendente y una tasa de aprendizaje establecida.

Antes del entrenamiento, se normalizan las variables predictoras; cuando existen clases desbalanceadas, se usan pesos compensatorios por clase.

## Fórmulas

- Gradiente de intercepto:

$$\beta_0 := \beta_0 - \alpha \frac{1}{m} \sum_{i=1}^m (\hat{y}_i - y_i)$$

- Gradiente de predictores:

$$\beta_j := \beta_j - \alpha \frac{\partial J}{\partial \beta_j}, j = 1, \dots, n$$

- Normalización:

$$x'_i = \frac{x_i - \bar{x}_i}{s_{x_i}}$$

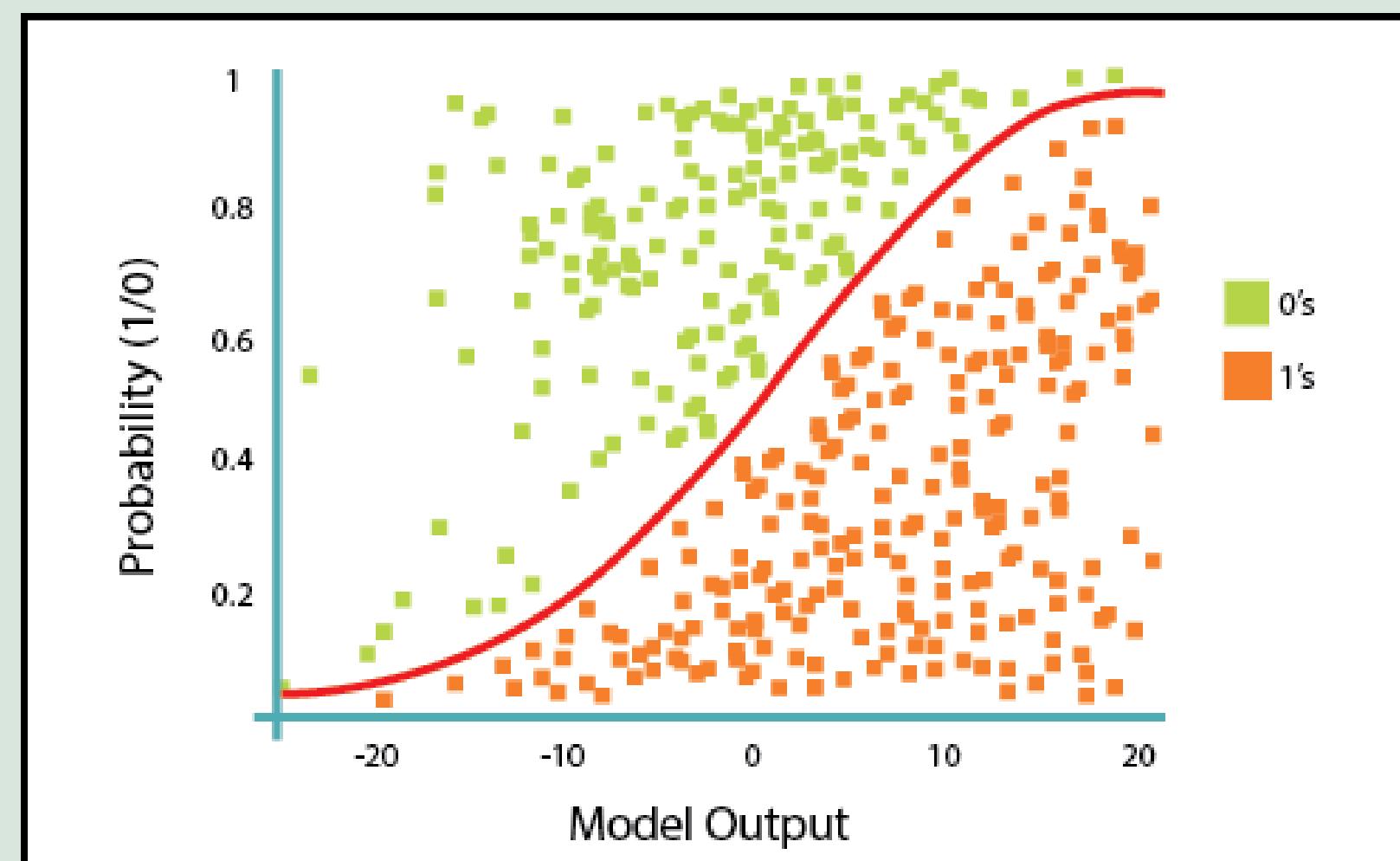
- Penalización:

$$w_1 = \frac{N}{2N_1}, w_2 = \frac{N}{2N_2}$$

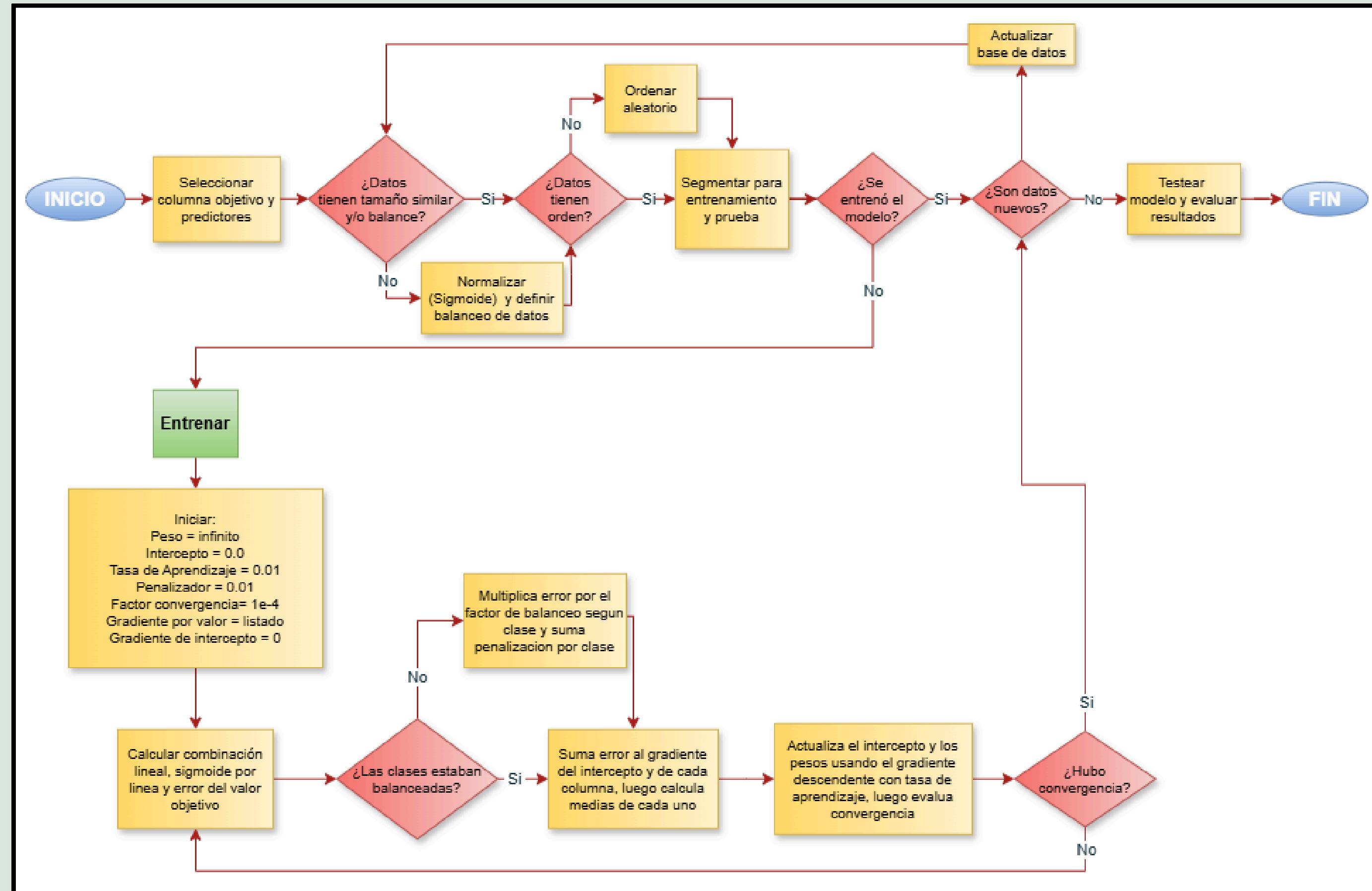
# MODELO

Se implementó un modelo en Python con el uso del Breast Cancer Wisconsin Dataset para distinguir entre tumores malignos y benignos.

- Normalización
- Aleatoriedad de registros
- Pesos iniciales
- Penalización por clase L2
- Gradiente descendiente
- Punto de convergencia



# MODELO

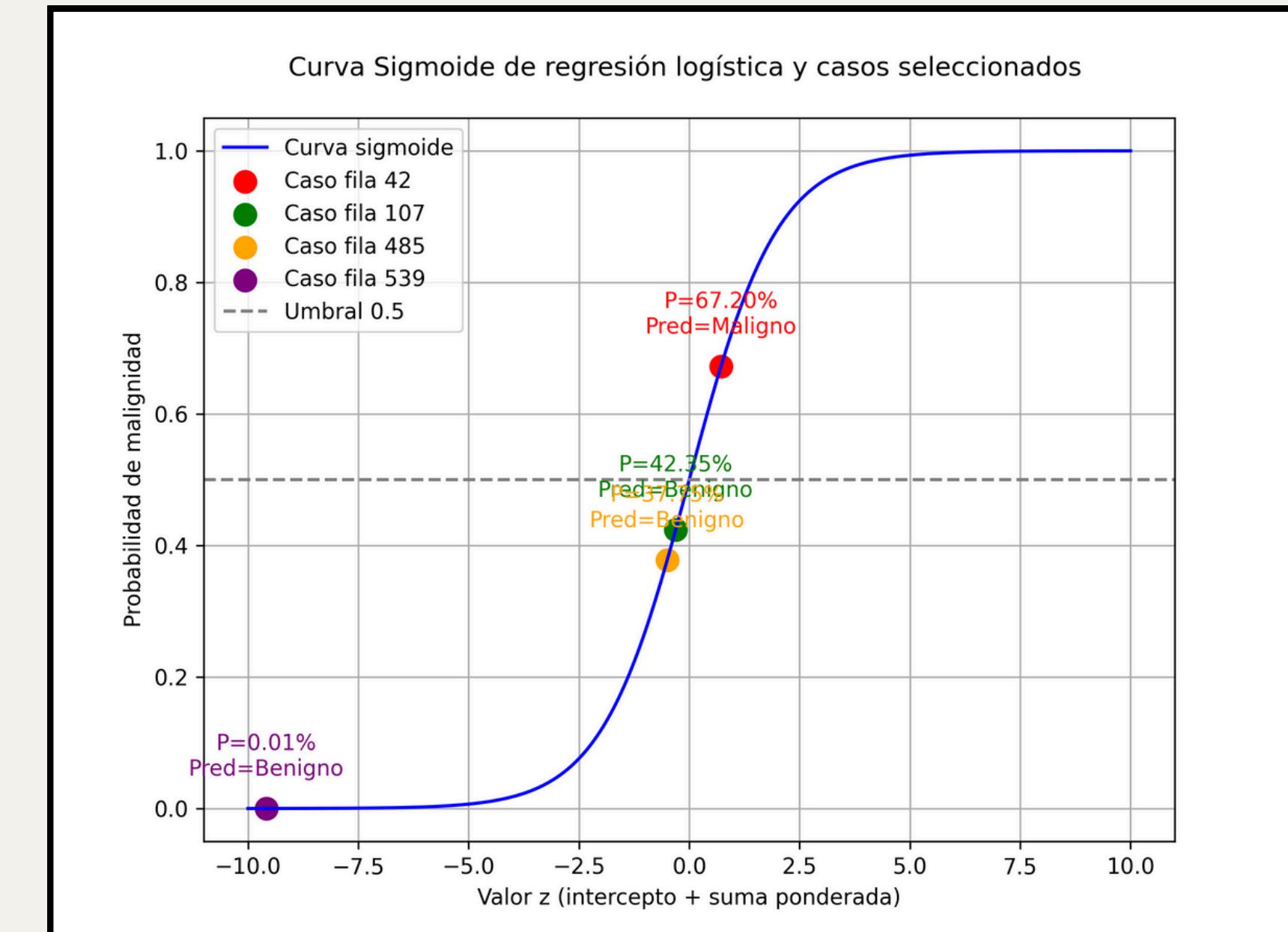
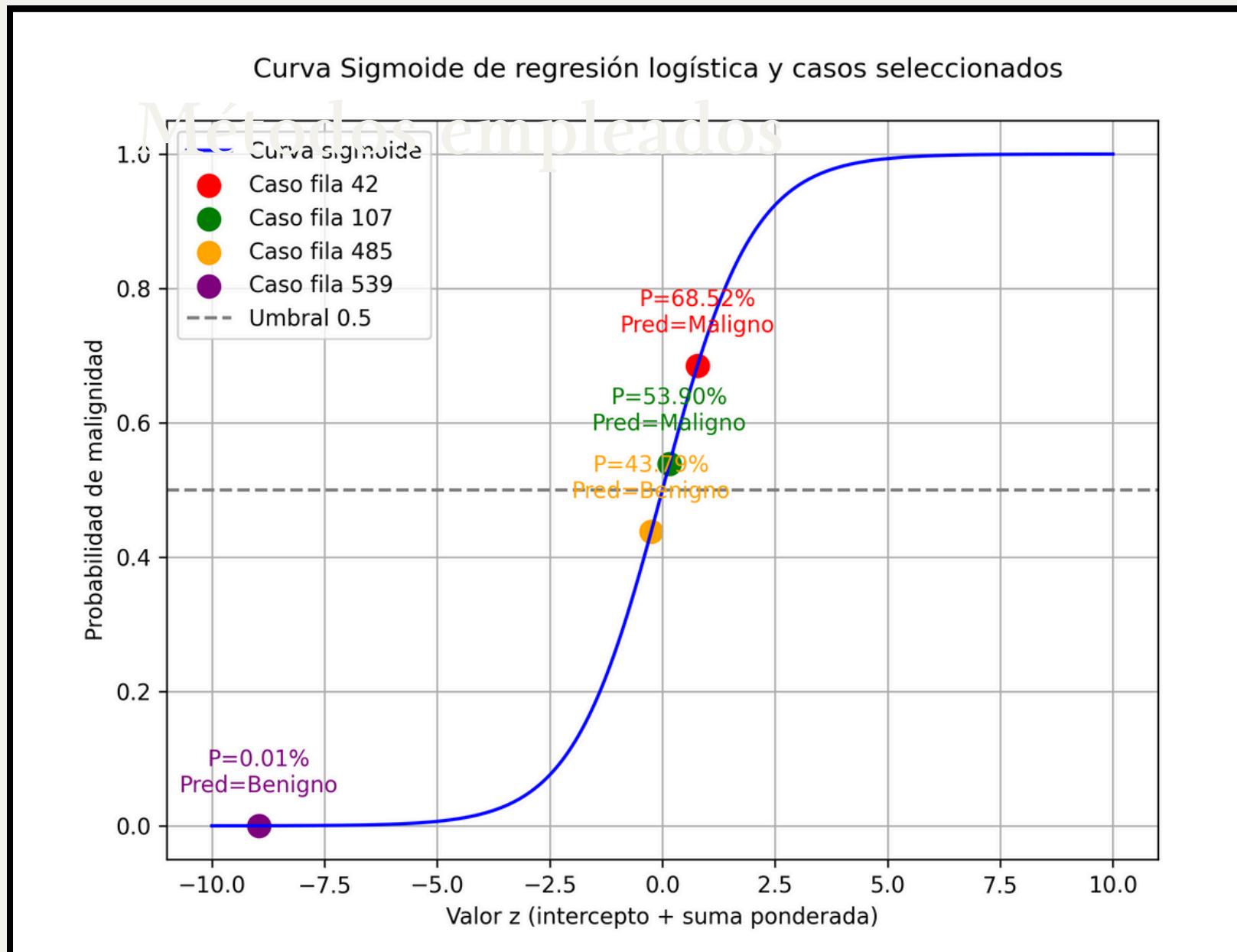


# EVALUACIÓN

Comparación 500 vs 1000 Iteraciones

Caso	Fila	P. Positivo (500)	Predicción	P. Positivo (1000)	Predicción	Clase real
1	42	0.6852	Maligno	0.672	Maligno	Maligno
2	107	0.539	<b>Maligno</b>	0.3775	<b>Benigno</b>	<b>Benigno</b>
3	485	0.4379	Benigno	0.8737	Benigno	Benigno
4	539	0.0001	Benigno	0.0001	Benigno	Benigno

# EVALUACIÓN



# RESULTADOS

	Exactitud Entrenamiento	Exactitud Prueba	Iteraciones	Tiempo de Ejecución
Primer entrenamiento	93.98%	92.40%	500	59.2 s
Último entrenamiento	96.74%	94.15%	1000	122.25 s

# CONCLUSIONES

Desarrollar la regresión logística desde cero permitió examinar con detalle:

- Cómo aprende el modelo
- Cómo influye la regularización
- Por qué importa la normalización

La curva sigmoide facilitó:

- Entender el umbral
- Analizar los errores en zonas límite

# HOW TO CONFUSE MACHINE LEARNING



*i*GRACIAS!