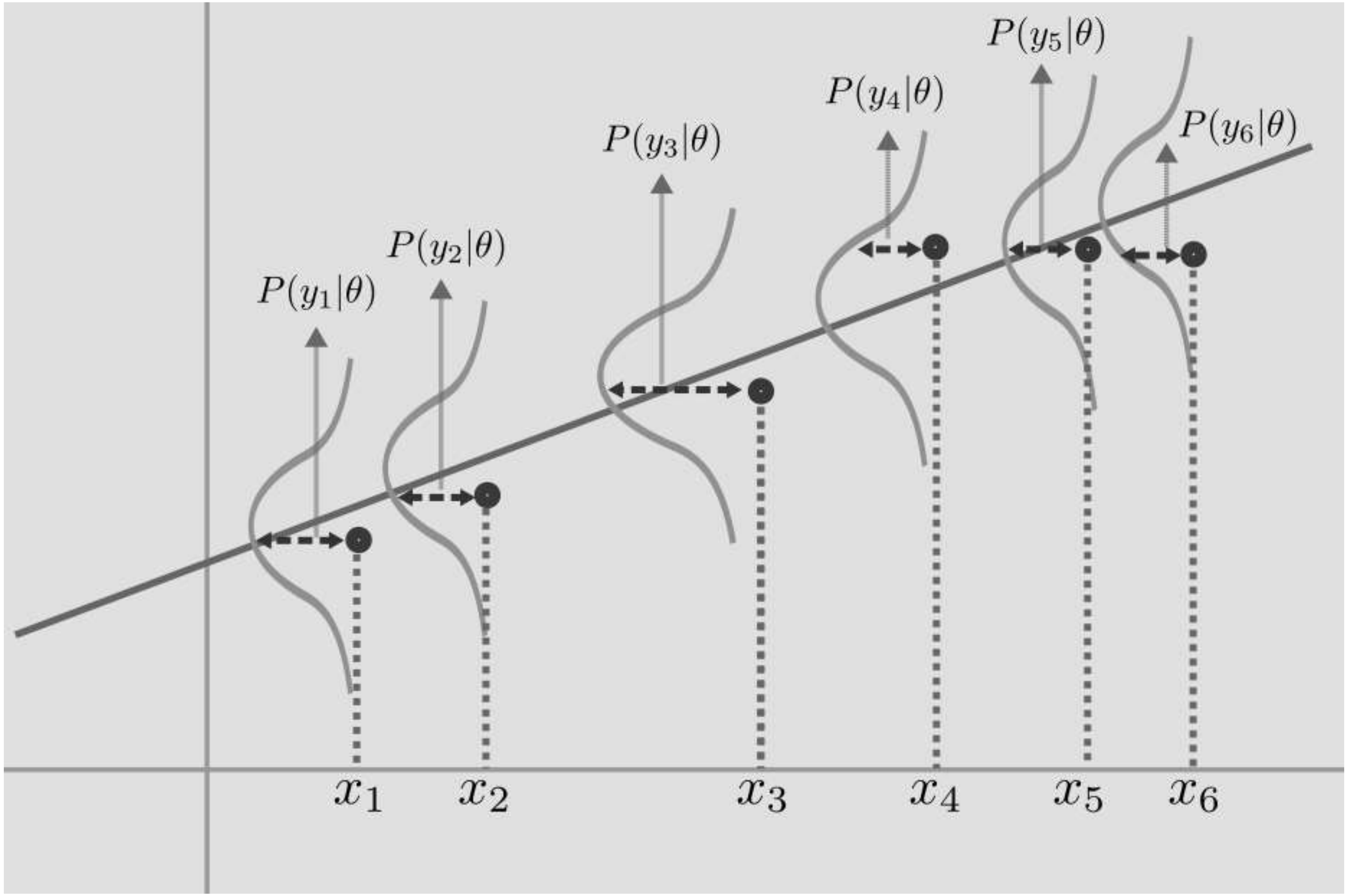
Probabilidad y Regresión Logística

Clasificación

Visión Probabilística de la Regresión

- $y = w_1 x + w_0$
- Hay infinitos modelos
- Elegir el modelo con la mayor probabilidad


$P(y_1|\theta)$
$P(y_2|\theta)$
$P(y_3|\theta)$
$P(y_4|\theta)$
$P(y_5|\theta)$
$P(y_6|\theta)$
x_1
x_2
x_3
x_4
x_5
x_6

MLE (Maximum Likelihood Estimation)

- Estimación de máxima verosimilitud
- Problema de optimización.
- Permite estimar densidades de probabilidad de un conjunto de datos

Probabilidad

$$P(x; w) = L(x; w)$$

$$\max L(x; w) = \max \prod_i P(x_i; w)$$

Tomamos el logaritmo porque valores muy pequeños causan problemas. En lugar de 0.0001 se tiene -9.21…..

$$\max \log L(x; w) = \max \sum_i \log P(x_i; w)$$

Aplicando en regresión

- $y = w_1 x + w_0$

- $P(y|x) \rightarrow \max \sum_i \log P(y_i|x_i; h)$

- h es la función que buscamos
- h no necesariamente debe ser regresión lineal

Equivalencia de MLE y Regresión Lineal

$$\max \sum_{i} \log P(y_i | x_i; h)$$

$$h \rightarrow \mathrm{y} = w_1 x + w_0$$

$$P \rightarrow \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2}$$

$y - \mu$ es el error

En la regresión tradicional

$$\min \sum_{i=1}^{n} (y_i - (w_1 x + w_0))^2$$

$y_i - (w_1 x + w_0)$ es el error

$$\max \sum_i \log \frac{1}{\sigma\sqrt{2\pi}} \times e^{-\frac{1}{2}\left(\frac{y_i-(w_1 x+w_0)}{\sigma}\right)^2}$$

En la regresión tradicional

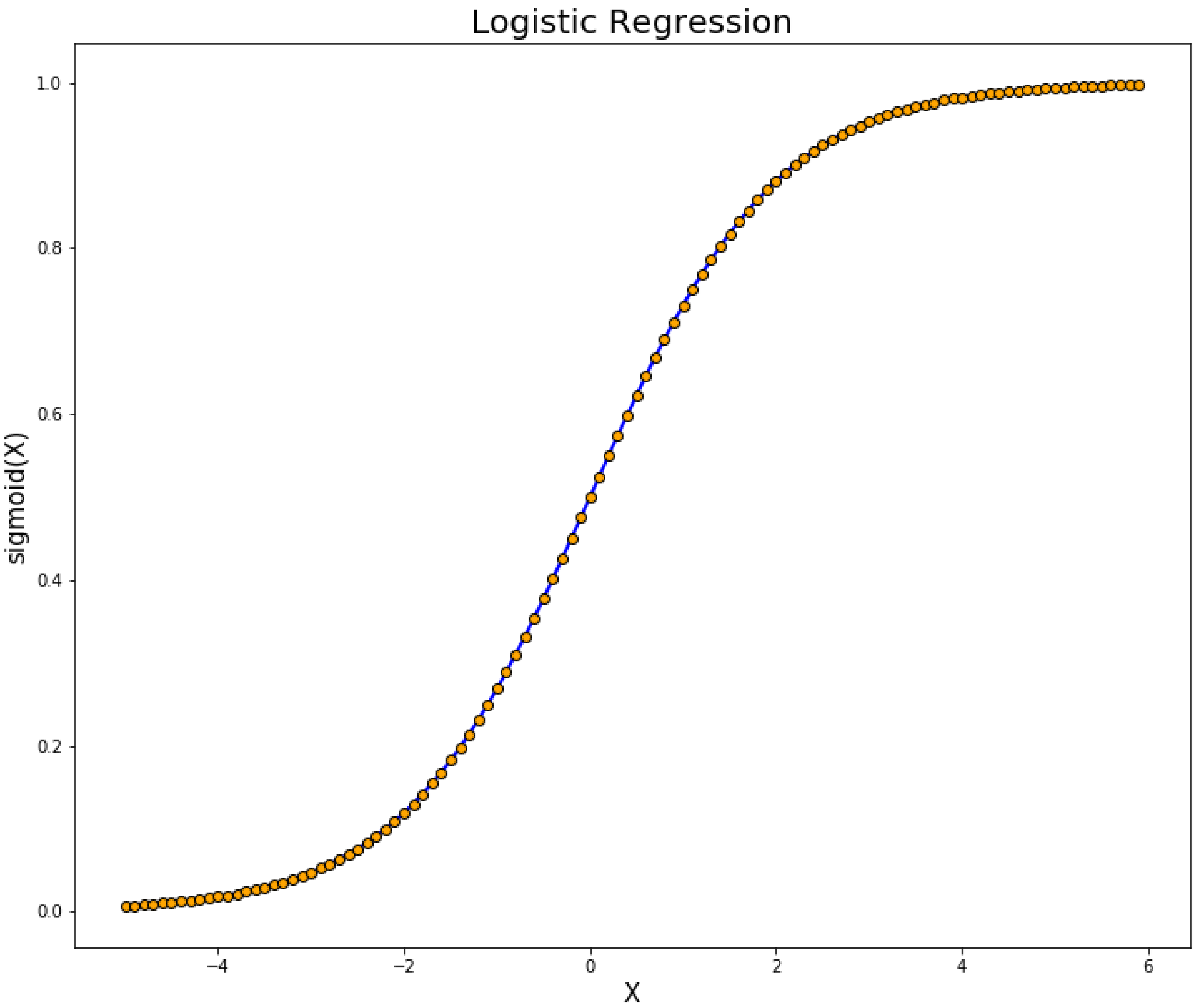
$$\min \sum_{i=1}^{n} (y_i - (w_1 x + w_0))^2$$

$$\max \sum_i \log \frac{1}{\sigma\sqrt{2\pi}} + \log e^{-\frac{1}{2}\left(\frac{y_i-(w_1 x+w_0)}{\sigma}\right)^2}$$

$$\max \sum_i -\frac{1}{2}\left(\frac{y_i-(w_1 x+w_0)}{\sigma}\right)^2$$

$$\min \frac{1}{2\sigma^2} \sum_i (y_i - (w_1 x + w_0))^2$$

Regresión Logistica

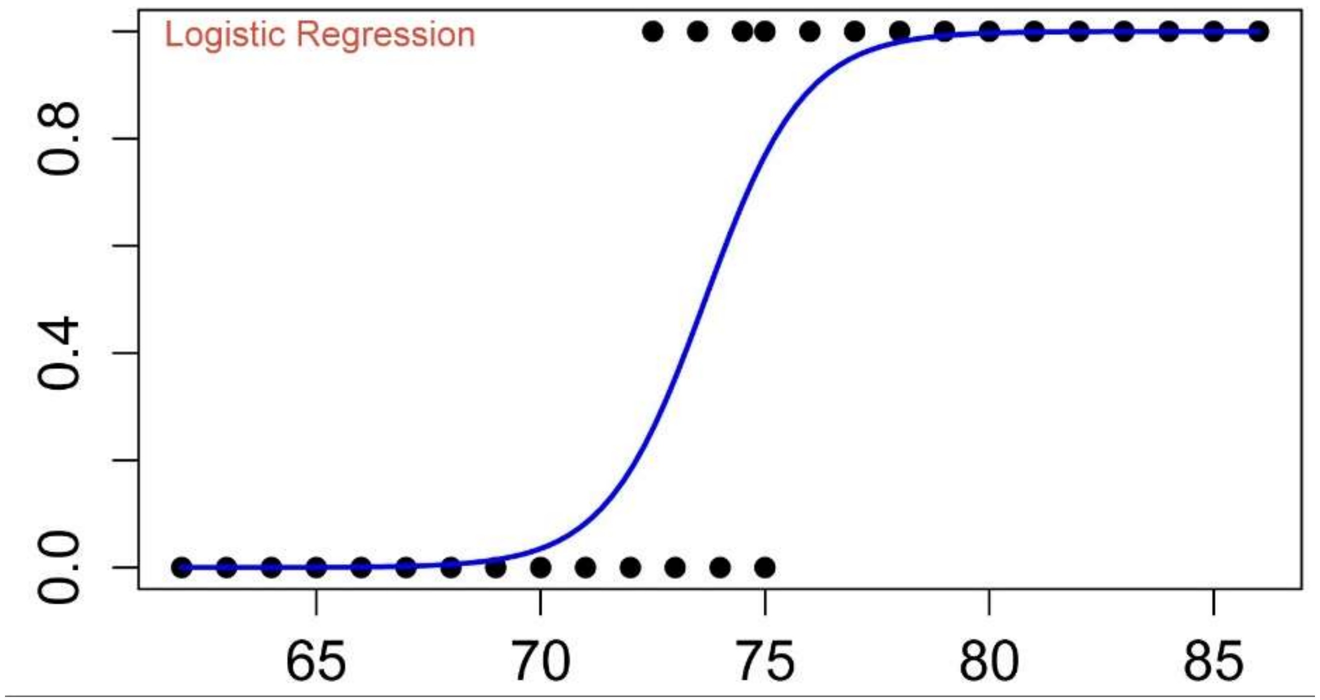


Regresión Logística

Variables independientes: sin cambios a la regresión lineal

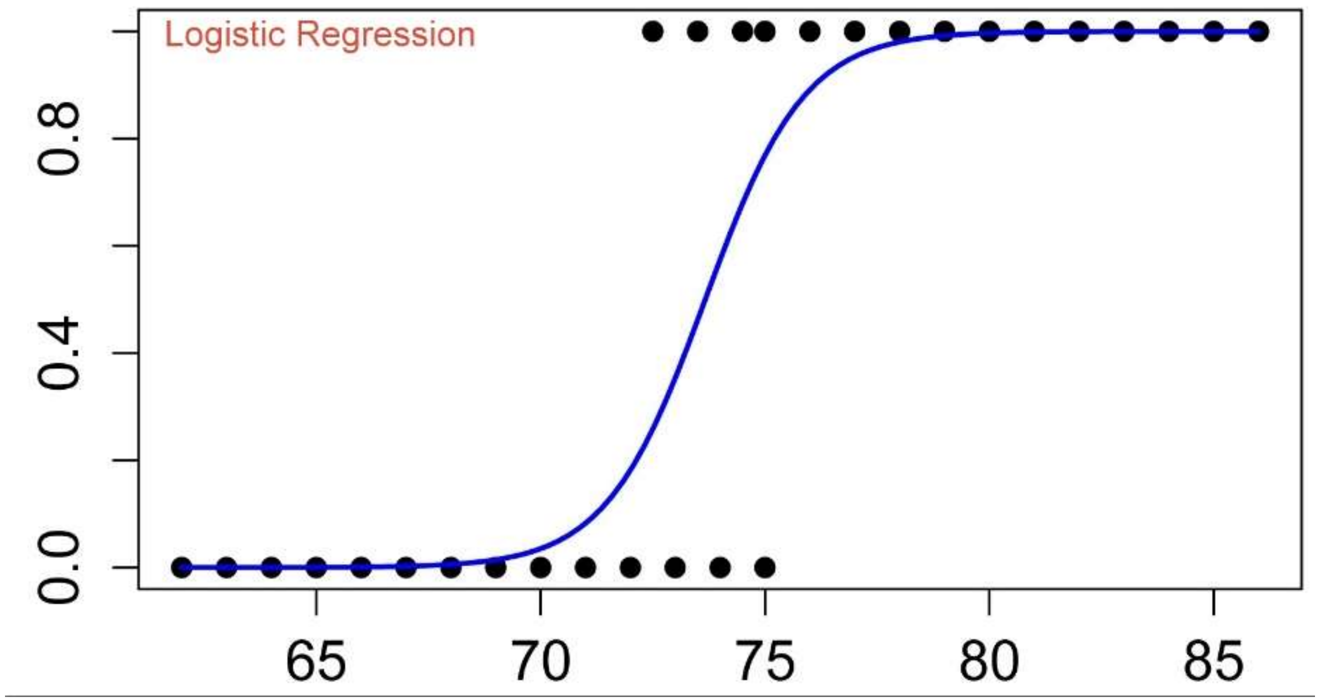
Variable dependiente: Categórica

La función



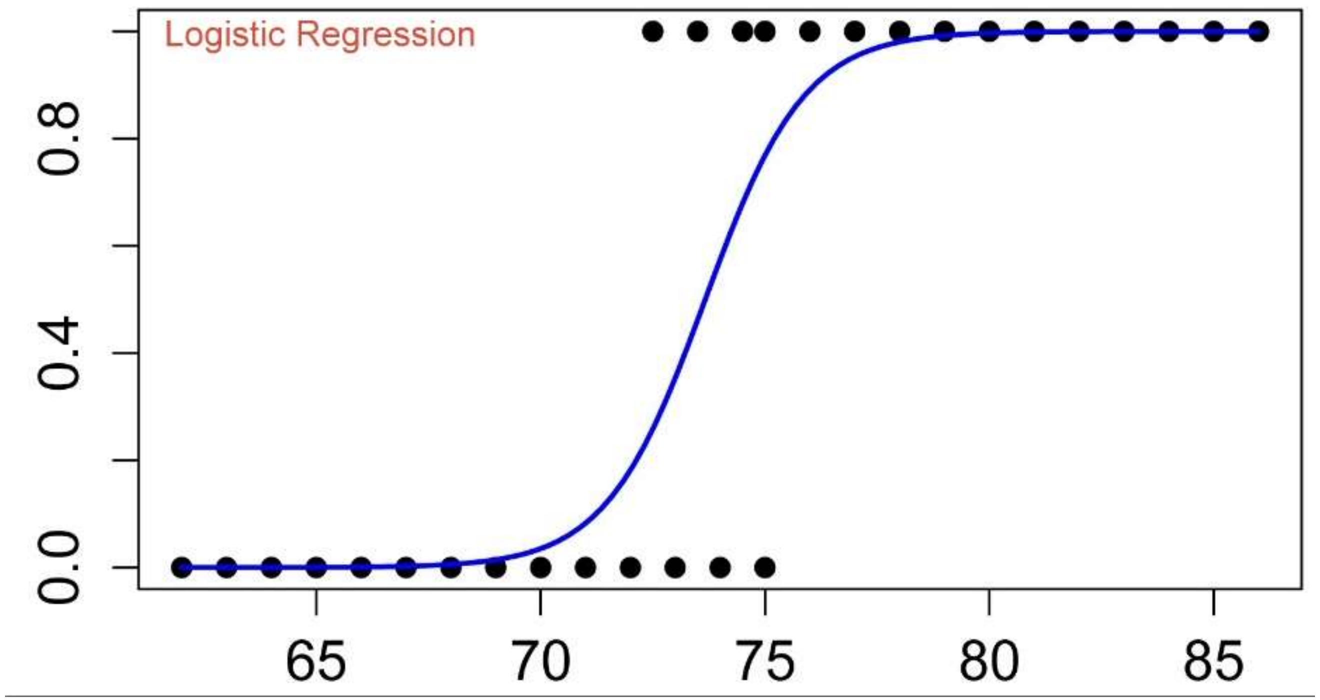
- Binaria (0,1)
- Usa la función logística
- $\sigma(z) = \frac{1}{1+e^{-z}}$
- Forma sigmoide

La función



- Encaja bien con el objetivo de clasificar
- Interpretable como probabilidad

La función



- Objetivo de la regresión: Ajustar la función para que todos los datos de un conjunto estén cerca al 0, y los del otro conjunto, al 1

Recordemos

- Queremos clasificar en base a atributos
- $\mathrm{x} = \{x_1, x_2 \dots, x_n\}$
- $y = \{0,1\}$

- Agregamos pesos a los atributos
- $X = w_1 x_1 + w_2 x_2 \dots + w_n x_n$

Transformando la función

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

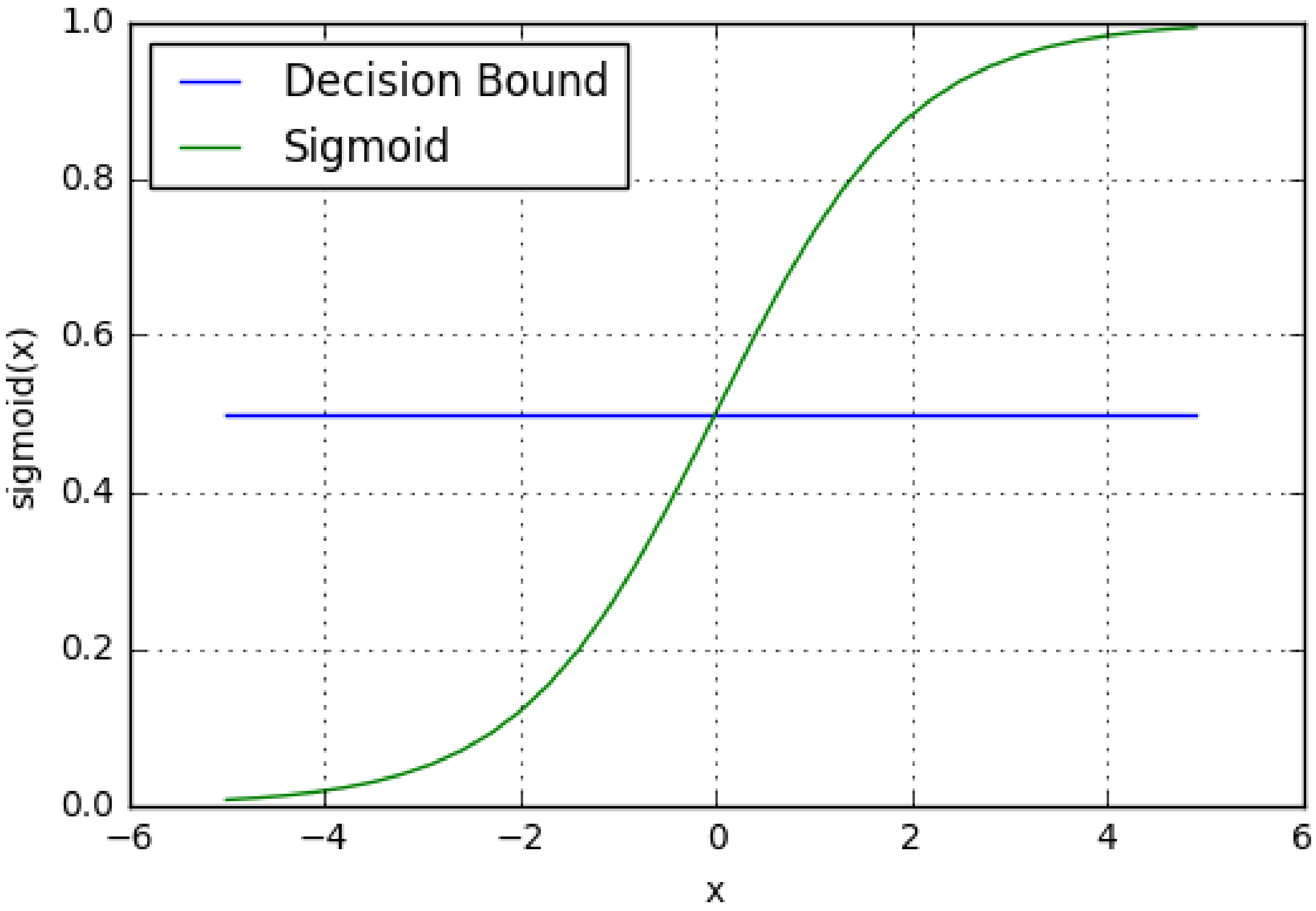
z es la suma ponderada de las variables independientes

$$z = w^T x$$

$$P(clase = 1) = \frac{1}{1 + e^{-w^T x}}$$

Limite de Decisión

- $p \geq 0.5, clase = 1$
- $p \leq 0.5, clase = 0$



Usando MLE para definir la función de pérdida

$$\max \sum_i \log P(y_i | x_i; h)$$

$$h \rightarrow \hat{y} = \mathrm{f(x;w)} = \frac{1}{1 + e^{-w^T x}}$$

$$L_i = \hat{y}_i \times y_i + (1 - \hat{y}_i) \times (1 - y_i)$$

Distribución Bernouli

$$P = \begin{cases} p \rightarrow y = 1 \\ 1 - p \rightarrow y = 0 \end{cases}$$

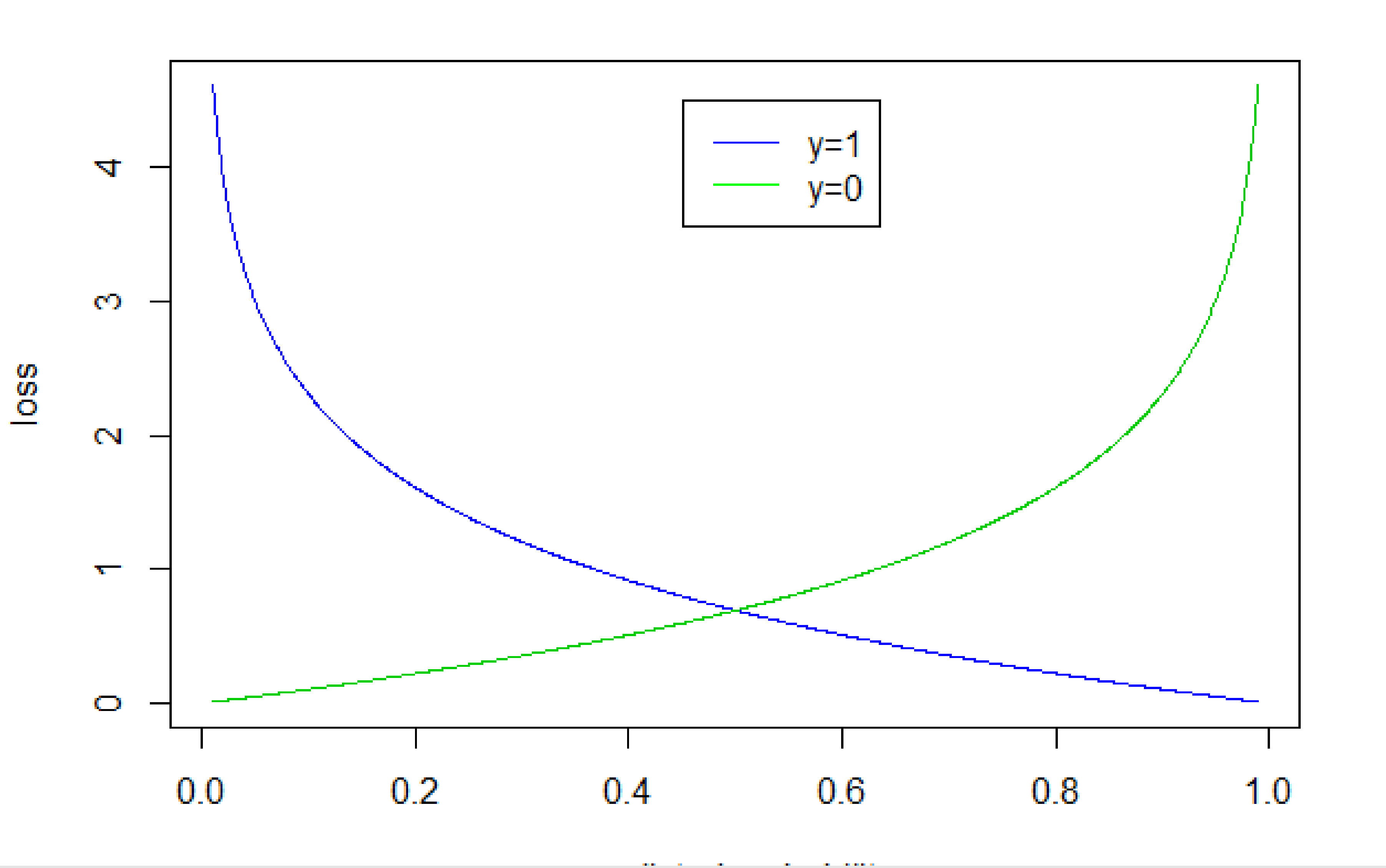
$$\bar{p} = p \times 1 + (1 - p) \times 0$$

Cross-Entropy

$$\min Costo = CE = -\frac{1}{m}\sum_i y_i \times \log\big(\mathrm{f}(x_i; w)\big) + (1 - y_i) \times \log\big(1 - \mathrm{f}(x_i; w)\big)$$

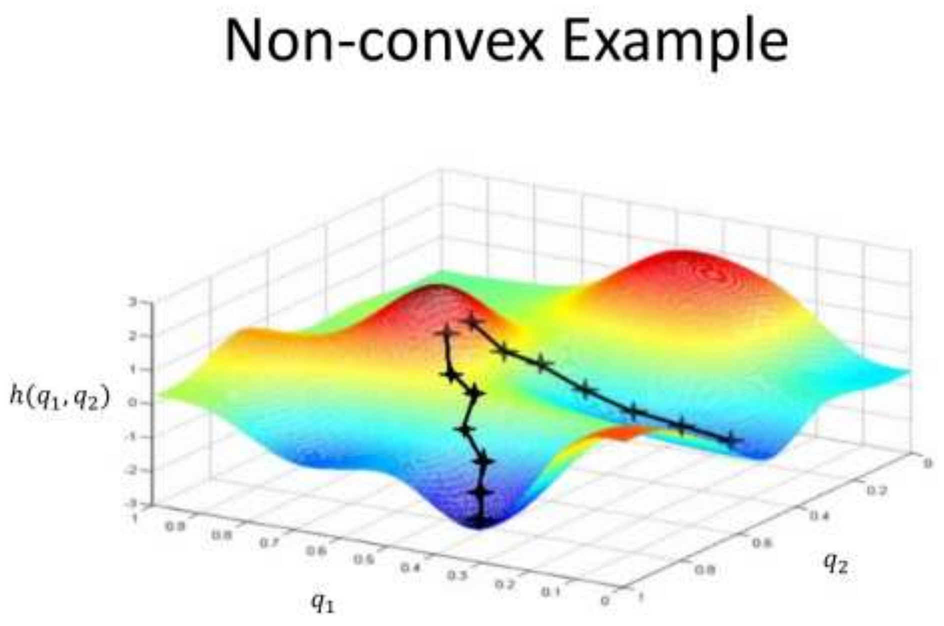
En vectores

$$\min -y^T \times \log\big(\mathrm{f}(x; w)\big) + (1 - y^T) \times \log\big(1 - \mathrm{f}(x; w)\big)\ /m$$


y=1
y=0
loss
0
1
2
3
4
0.0
0.2
0.4
0.6
0.8
1.0

Para encontrar el mínimo

- Debemos usar gradient descent
- Puede que hayan varias respuestas (mínimos locales)
- Hay otros algoritmos para leer



La derivada de la sigmoide

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

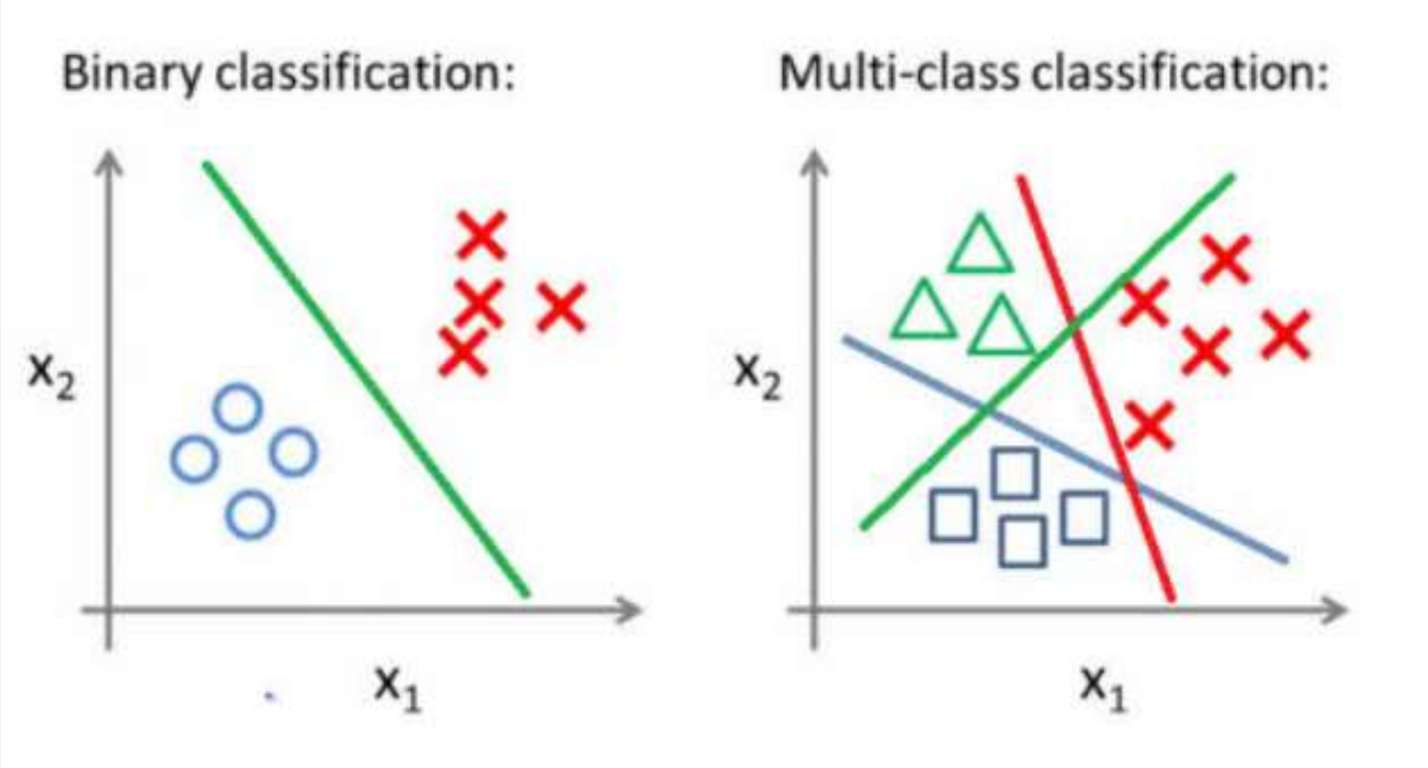
$$\frac{d\sigma}{dz} = \sigma(z)(1 - \sigma(z))$$

Por lo tanto la derivada de la función de costo es…

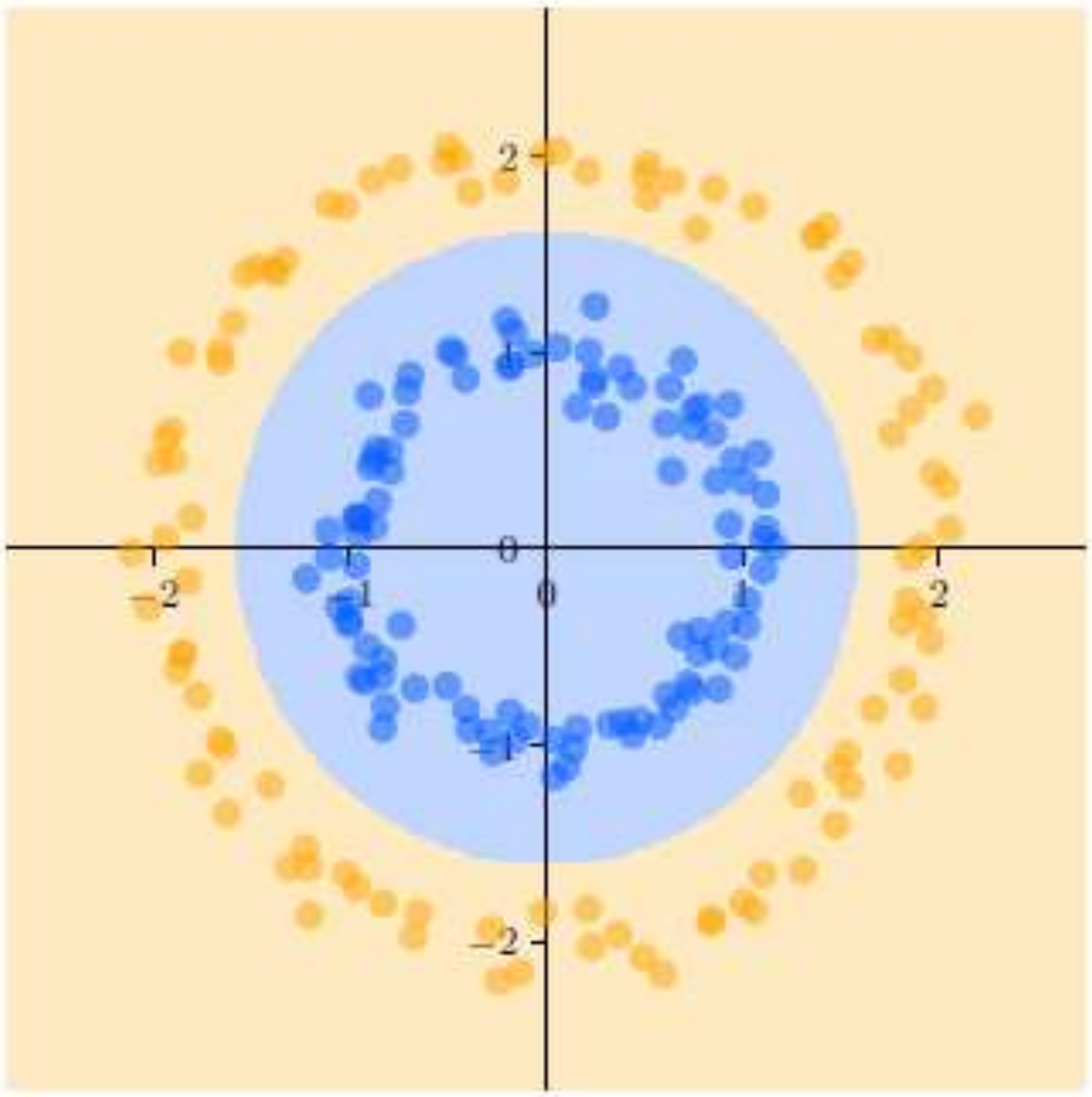
$$\frac{\partial CE}{\partial w_j} = \left(\mathrm{f}\left(x_j; w_j\right) - y\right)x_j$$

Para múltiples variables

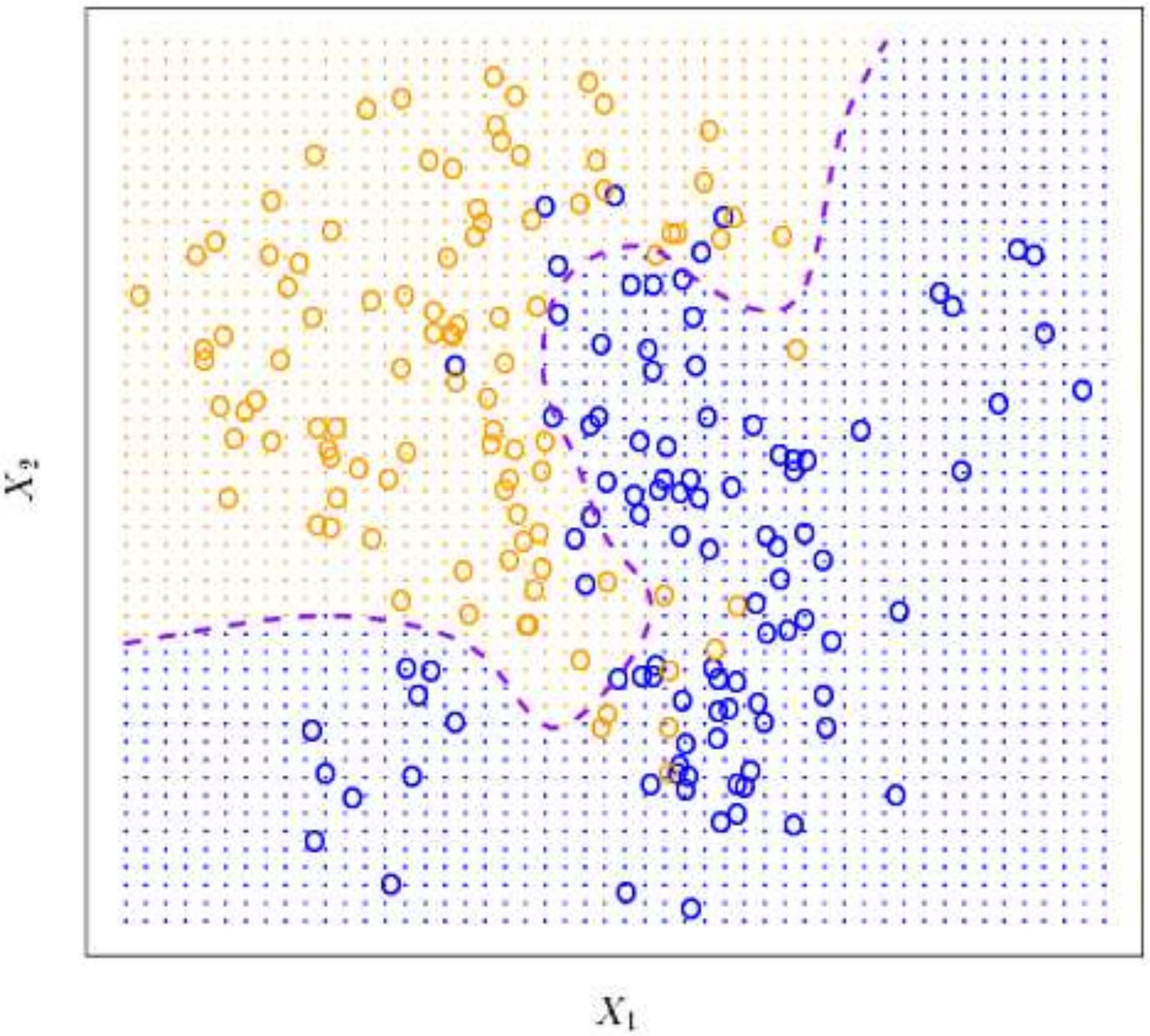
Uno vs el resto (one-vs-rest)



¿Y si no es lineal?

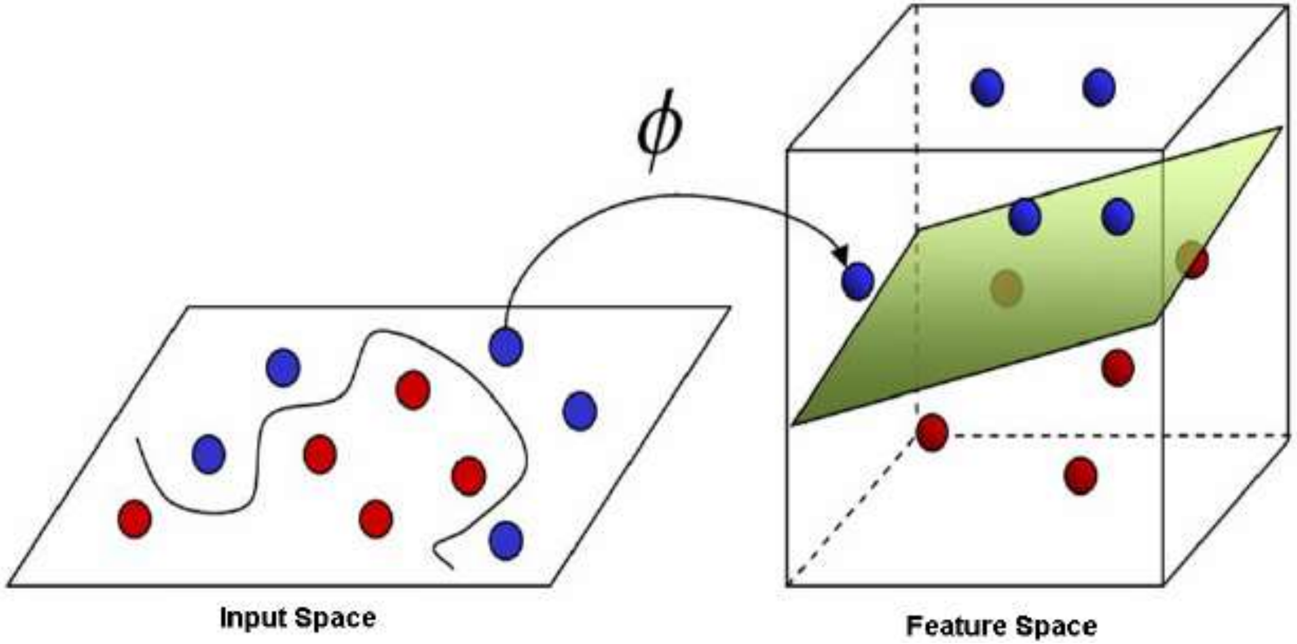


Non-linear Decision Boundary



Decision Boundary de grado 2…3…4

- $f = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2$ …
- $W^T \Phi(x)$
- ¿Regularización?



Regularización

Ridge $CE + \lambda w^2$

Lasso $CE + \lambda||w||$

Modelando en Python

Evaluando nuestro modelo

Matriz de Confusión

	Positivo real	Negativo real
Predicción positiva	Verdadero positivo (TP)	Falso positivo (FP) Error tipo I
Predicción negativa	Falso negativo (FN) Error tipo II	Verdadero negativo (TN)

Evaluando nuestro modelo (Accuracy)

	Positivo real	Negativo real
Predicción positiva	Verdadero positivo (TP)	Falso positivo (FP)
Predicción negativa	Falso negativo (FN)	Verdadero negativo (TN)

- Accuracy (Precision)
- $\frac{TP+TN}{TP+TN+FP+FN}$

- Métrica más intuitiva
- No es el más adecuado si la data no está balanceada
- (10 de una clase 0 y 90 de la clase 1)

Evaluando nuestro modelo (Recall)

	Positivo real	Negativo real
Predicción positiva	Verdadero positivo (TP)	Falso positivo (FP)
Predicción negativa	Falso negativo (FN)	Verdadero negativo (TN)

- Recall (Sensitivity)
- $\frac{TP}{TP+FN}$

- Para datos médicos, este valor es importante

Evaluando nuestro modelo (Specificity)

	Positivo real	Negativo real
Predicción positiva	Verdadero positivo (TP)	Falso positivo (FP)
Predicción negativa	Falso negativo (FN)	Verdadero negativo (TN)

- Specificity
- $\frac{TN}{TN+FP}$

Evaluando nuestro modelo (Precision)

	Positivo real	Negativo real
Predicción positiva	Verdadero positivo (TP)	Falso positivo (FP)
Predicción negativa	Falso negativo (FN)	Verdadero negativo (TN)

- Precision
- $\frac{TP}{TP+FP}$

Evaluando nuestro modelo (F1 Score)

	Positivo real	Negativo real
Predicción positiva	Verdadero positivo (TP)	Falso positivo (FP)
Predicción negativa	Falso negativo (FN)	Verdadero negativo (TN)

- F1 Score
- $F1 = 2\frac{precision \times recall}{precision + recall}$
- $\frac{TP}{TP+\frac{1}{2}(FP+FN)}$
- Métrica ponderada de Verdaderos positivos
- Útil cuando se le quiere dar más peso a una clase o cuando hay desbalance

Evaluando nuestro modelo (Resumen)

		Predicted Class		
		Positive	**Negative**	
Actual Class	**Positive**	True Positive (TP)	False Negative (FN) **Type II Error**	**Sensitivity** $\frac{TP}{(TP + FN)}$
	Negative	False Positive (FP) **Type I Error**	True Negative (TN)	**Specificity** $\frac{TN}{(TN + FP)}$
		Precision $\frac{TP}{(TP + FP)}$	**Negative Predictive Value** $\frac{TN}{(TN + FN)}$	**Accuracy** $\frac{TP + TN}{(TP + TN + FP + FN)}$

Más el F-score

Resumen de regresión

- Definir la función de pérdida
- Derivar para encontrar la gradiente
- Encontrar el mínimo de la función de pérdida
    - Si se aplica GD y variantes
        - Calcular la gradiente
        - Actualizar pesos
        - Repetir hasta converger en un mínimo
- Evaluar el modelo con data no vista