

# Redes Densas: Fundamentos

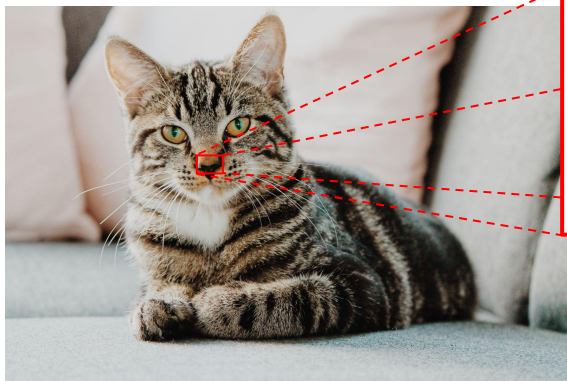
Orlando Ramos Flores

# Contenido

- Introducción
- Neurona
- Modelo de neurona
- Ejemplo
- Conectar más neuronas
- Estructura general

# ¿Qué es esto?

Computadora:



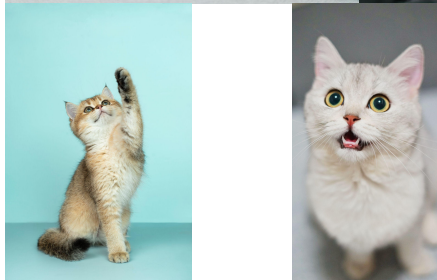
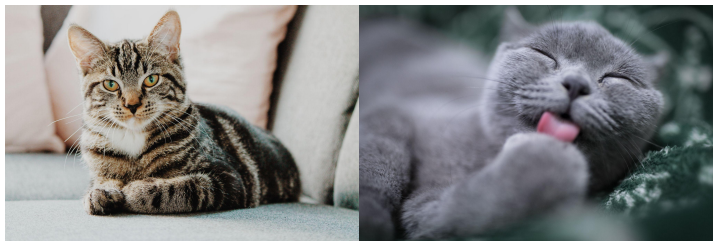
|     |     |     |    |    |    |    |    |
|-----|-----|-----|----|----|----|----|----|
| 234 | 215 | 198 | 40 | 45 | 34 | 25 | 30 |
| 240 | 220 | 199 | 39 | 40 | 27 | 26 | 25 |
| 233 | 225 | 180 | 36 | 39 | 25 | 18 | 20 |
| 245 | 210 | 176 | 30 | 23 | 20 | 16 | 15 |
| 220 | 209 | 167 | 29 | 21 | 19 | 11 | 12 |

cuadrícula de píxeles



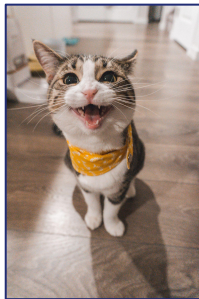
= gato

# Detección de gatos (Clasificador)

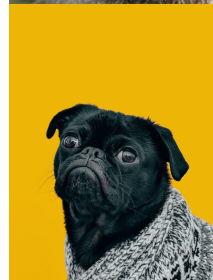


+ gatos

testing



¿ ?



- no gatos

# ¿Qué algoritmo elegir para clasificar gatos?

Regresión  
Logística



pixel 1  
pixel  
(0-255)  
intensidad



pixel 2



+ gatos



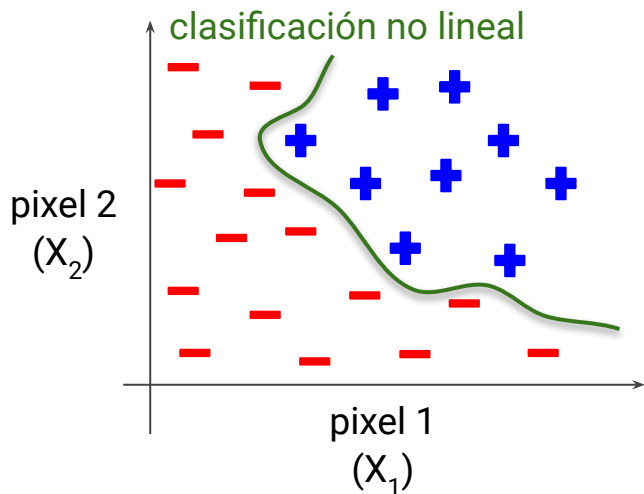
- no gatos

¿Cuál es la dimensión del espacio de las características (píxeles)?

Supongamos: Imágenes de 100 x 100 píxeles = 10000 píxeles

$n = 10000$  (grayscale)

$n = 30000$  (RGB)



$$X = \begin{pmatrix} \text{pixel 1} \\ \text{pixel 2} \\ \text{pixel 3} \\ \dots \\ \text{pixel 10000} \end{pmatrix}$$

Características cuadráticas  
( $X_i * X_j$ ):  $\approx 50$  millones ( $5 \times 10^7$ )

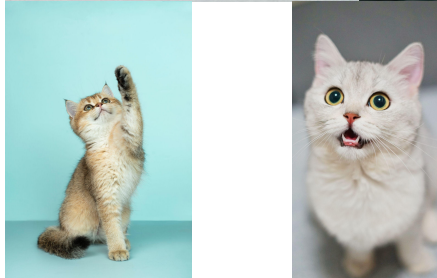
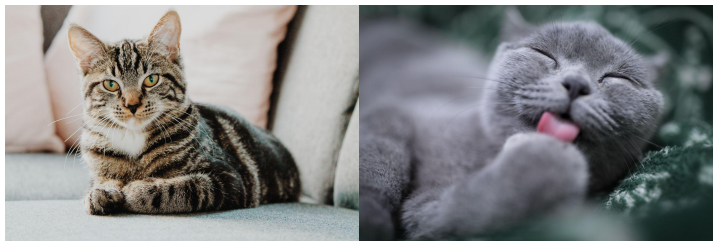
# ¿Qué algoritmo elegir para clasificar gatos?

Regresión  
Logística

Redes Neuronales

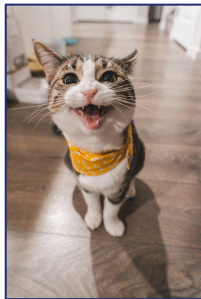


# Detección de gatos (Clasificador)

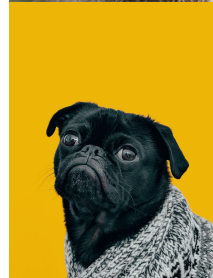


+ gatos

testing



¿ ?



- no gatos



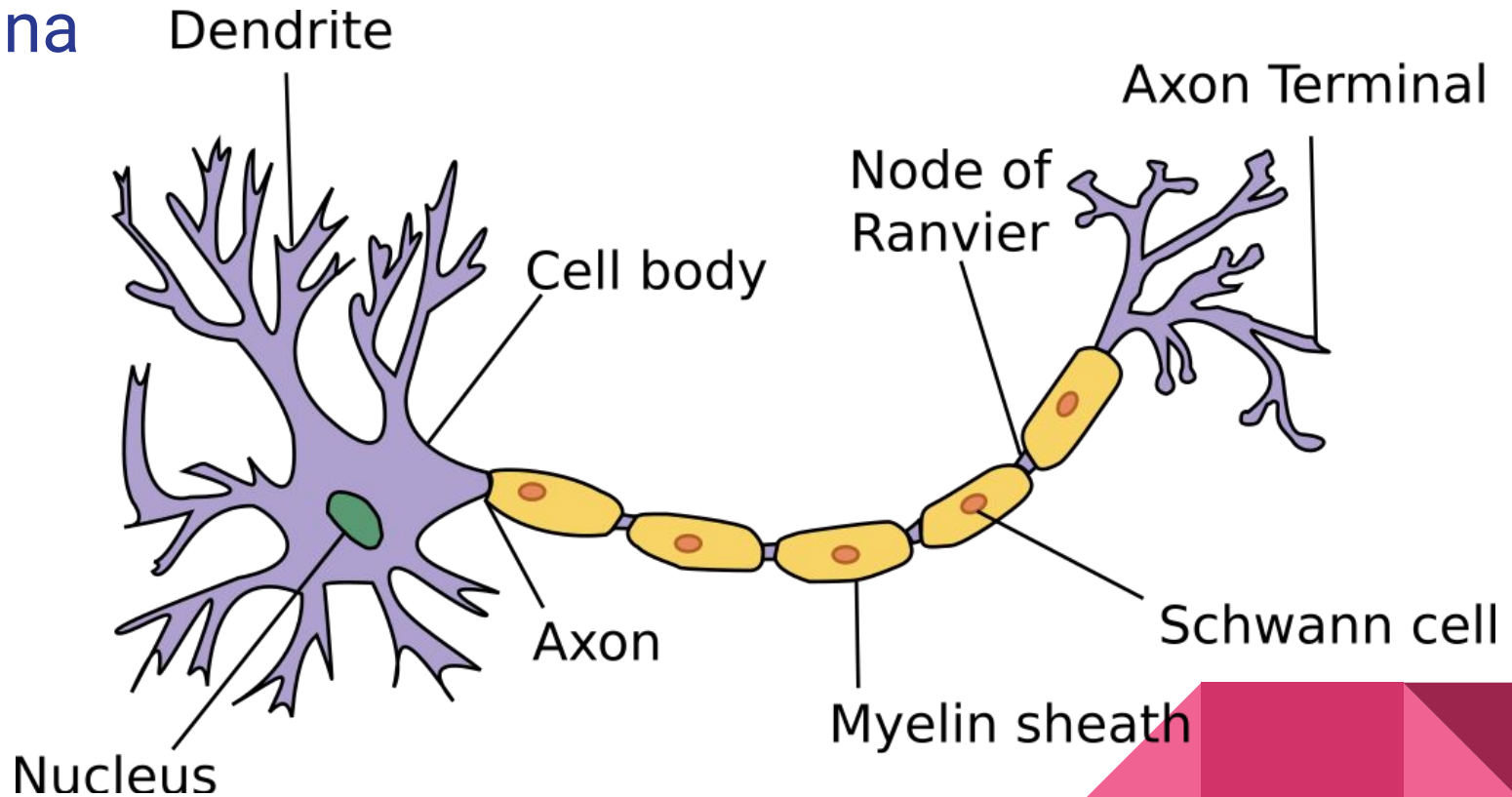
# Red Neuronal: Historia

Entre las décadas de 1950 y 1960 un tipo de neurona artificial llamada **perceptrón** fue desarrollada por el científico Frank Rosenblatt, su construcción fue de acuerdo con principios biológicos y mostró la capacidad para aprender. El perceptrón se desarrolló a partir de las ideas de fortalecimiento sináptico de Hebb y la neurona McCulloch-Pitts. La idea clave fueron las variaciones del descenso de gradiente estocástico [1,5,7].

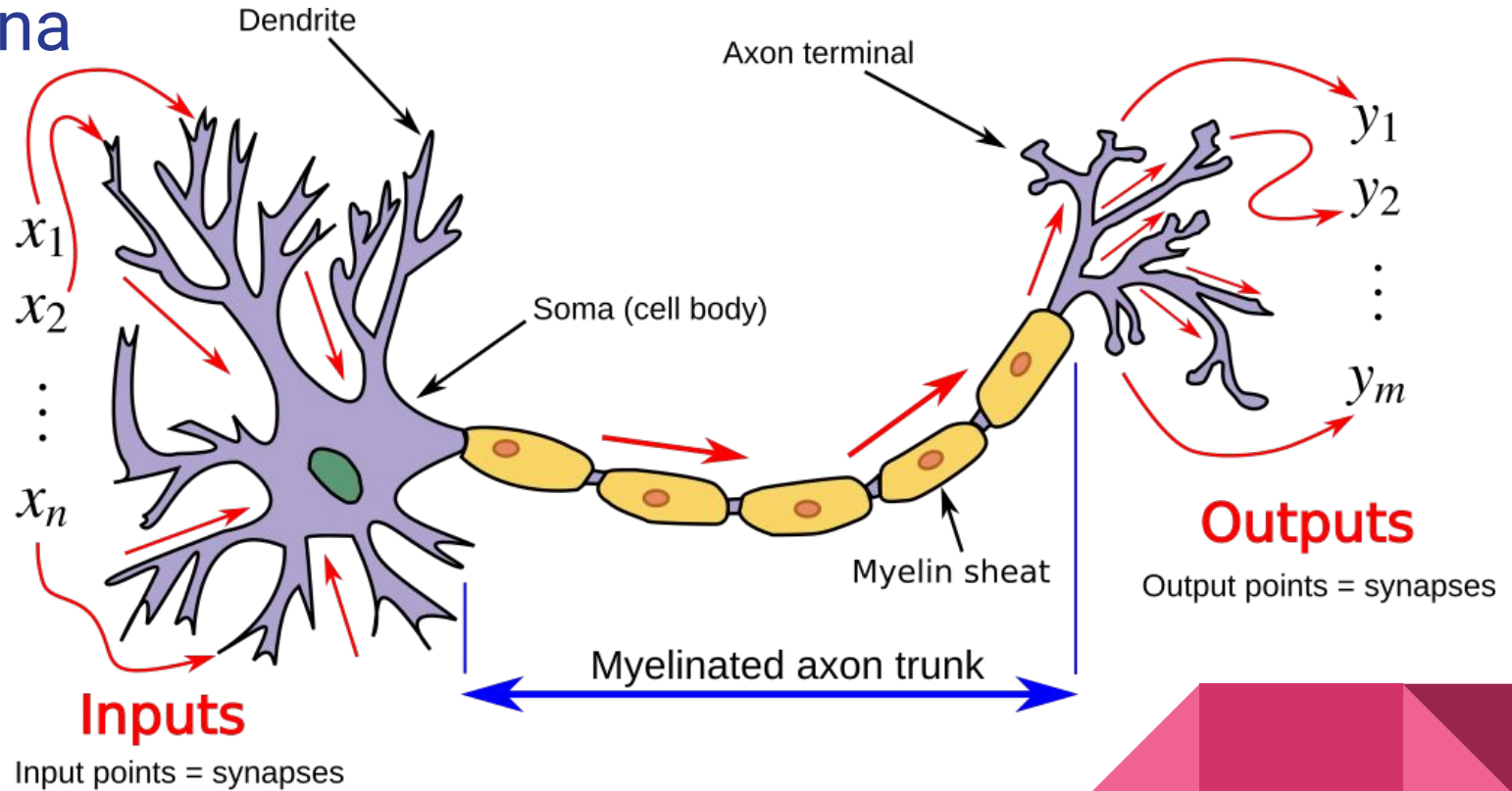


Fuente: [https://en.wikipedia.org/wiki/Frank\\_Rosenblatt](https://en.wikipedia.org/wiki/Frank_Rosenblatt)

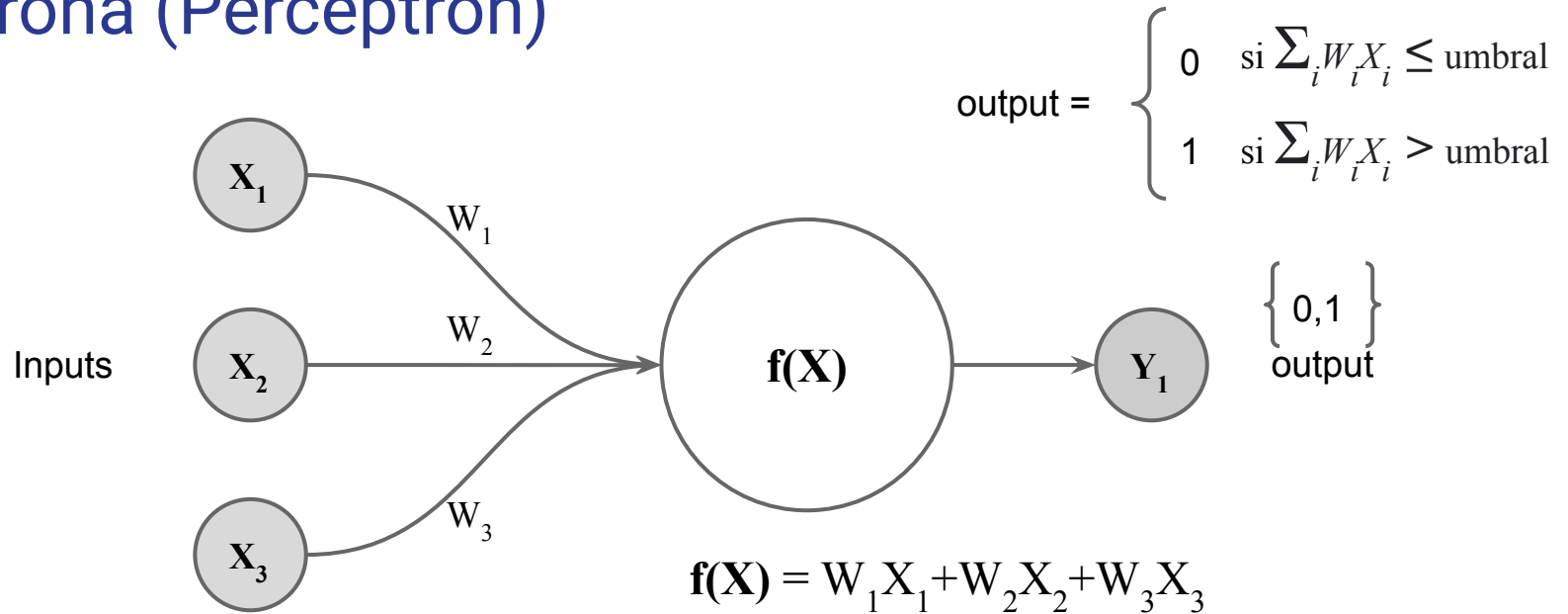
# Neurona



# Neurona



# Neurona (Perceptrón)



# Ejemplo: Descripción

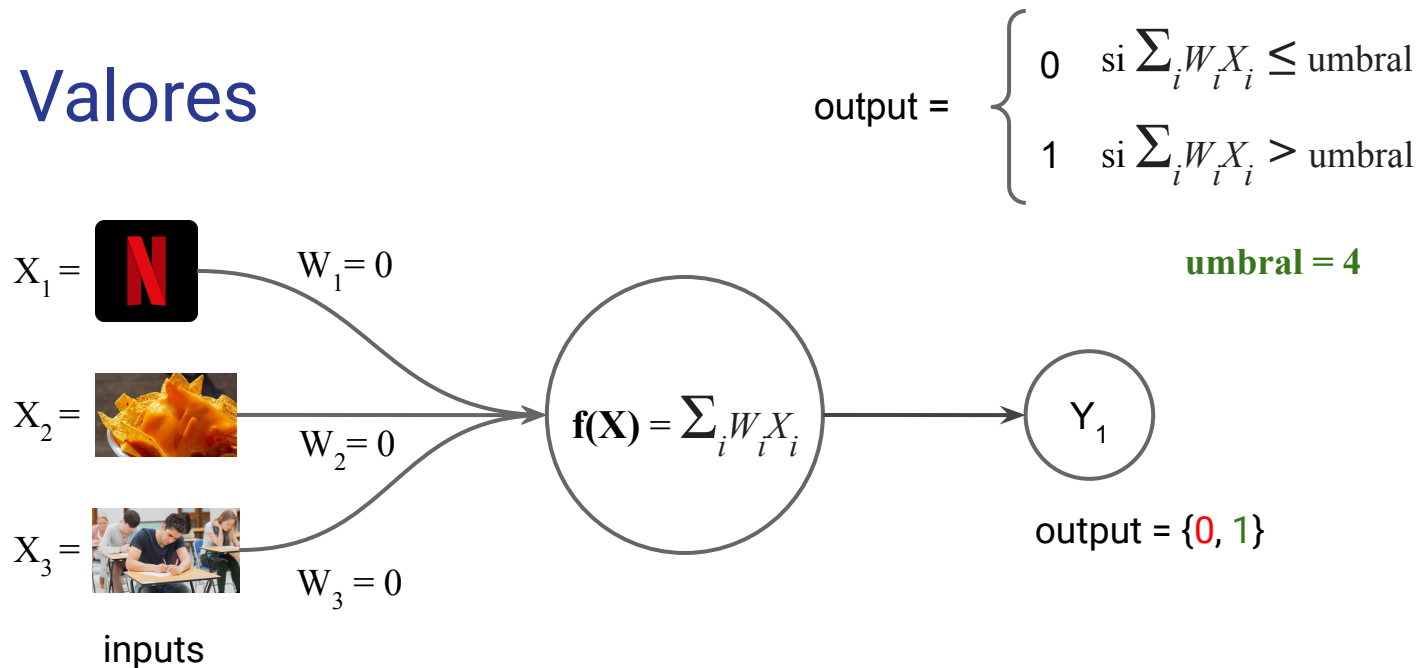
**Problema:** El fin de semana vas a ver el final de tu serie favorita de **Netflix**<sup>1</sup>, para ello tienes planeado comer **nachos**<sup>2</sup> con mucho queso. Además tienes que **estudiar**<sup>3</sup> para **aprobar** el examen parcial de Aprendizaje Profundo del día lunes.



$Y_1 = \{0: \text{reprobar}, 1: \text{aprobar}\}$

1. [https://es.m.wikipedia.org/wiki/Archivo:Netflix\\_icon.svg](https://es.m.wikipedia.org/wiki/Archivo:Netflix_icon.svg)
2. <https://www.divinacocina.es/nachos-con-queso/>
3. <https://elcomercio.pe/respuestas/como/examen-lleva-tilde-escribe-diccionario-palabras-rae-espanol-ortografia-nnda-nnlt-noticia-542390-noticia/>

# Ejemplo: Valores

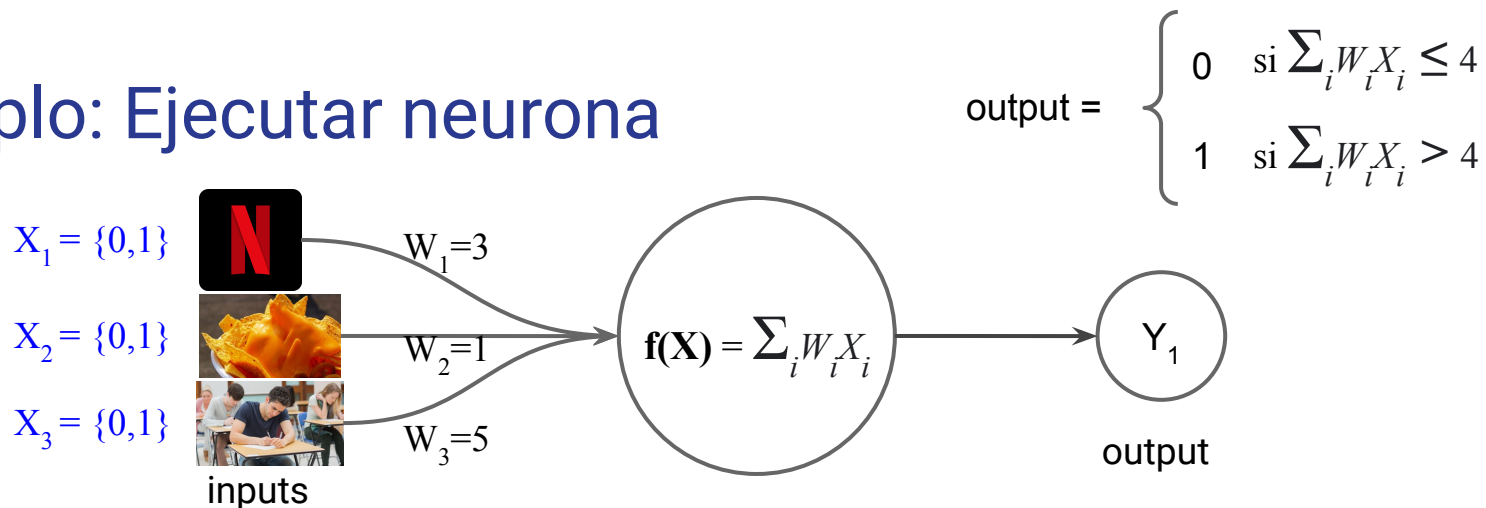


$X_1 = \{0, 1\}$ ; 1 si ves tu serie, 0 si no la ves

$X_2 = \{0, 1\}$ ; 1 comes nachos, 0 no hay nachos

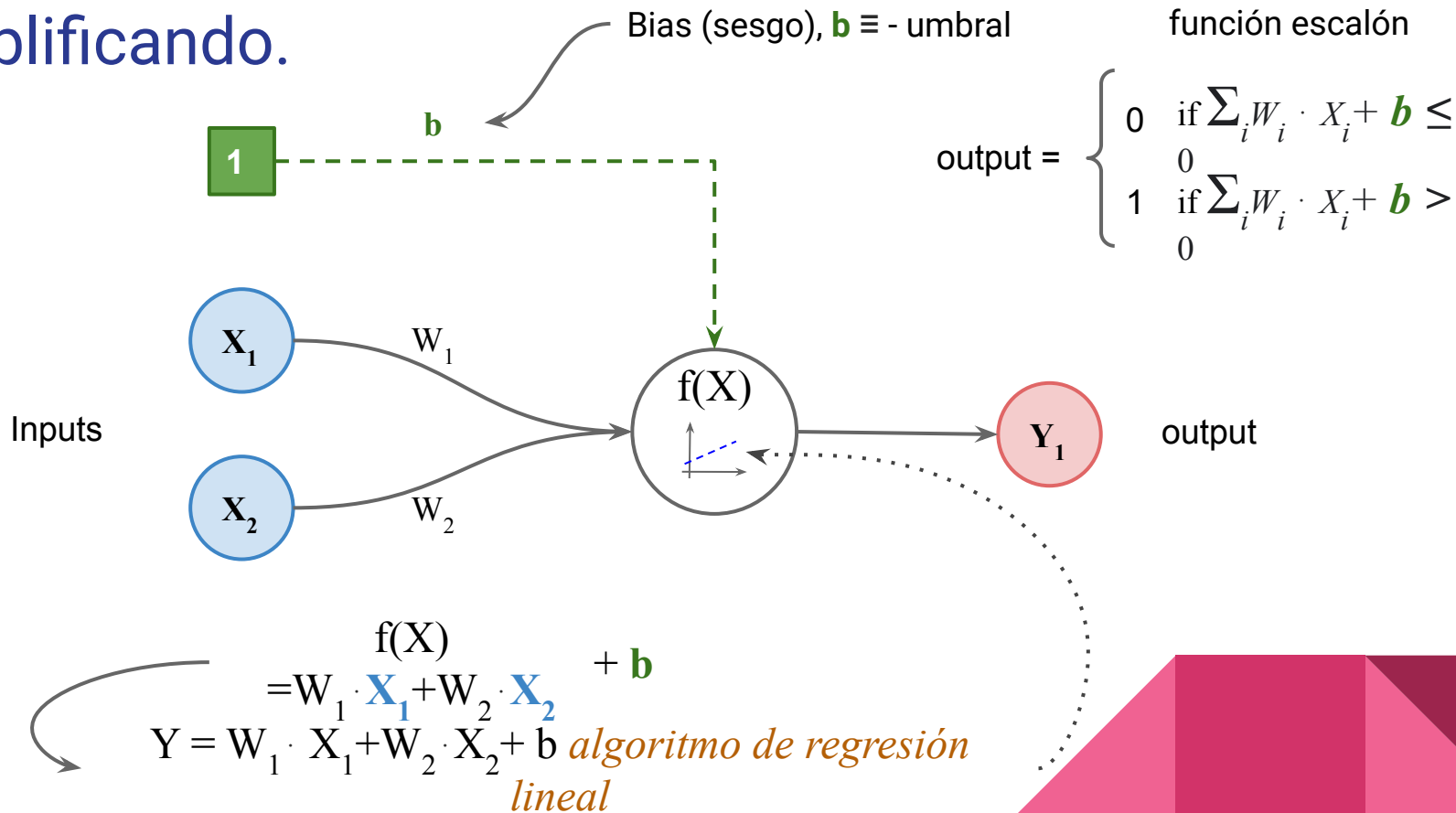
$X_3 = \{0, 1\}$ ; 1 estudias para el examen, 0 no estudias para el examen

# Ejemplo: Ejecutar neurona



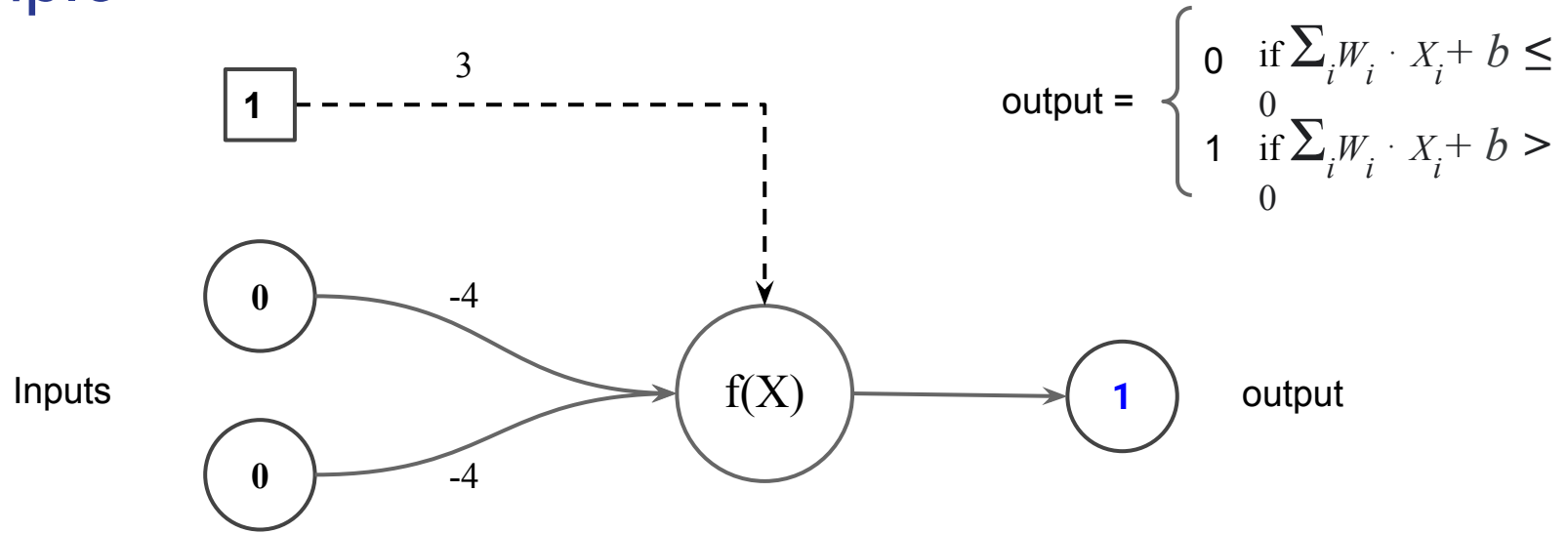
|                                      |                    |                          |
|--------------------------------------|--------------------|--------------------------|
| $f_1(X) = 0 * 3 + 0 * 1 + 0 * 5 = 0$ | $Y_1 = \mathbf{0}$ | ---                      |
| $f_2(X) = 1 * 3 + 0 * 1 + 0 * 5 = 3$ | $Y_1 = \mathbf{0}$ | serie                    |
| $f_3(X) = 0 * 3 + 1 * 1 + 0 * 5 = 1$ | $Y_1 = \mathbf{0}$ | nachos                   |
| $f_4(X) = 1 * 3 + 1 * 1 + 0 * 5 = 4$ | $Y_1 = \mathbf{0}$ | serie y nachos           |
| $f_5(X) = 0 * 3 + 0 * 1 + 1 * 5 = 5$ | $Y_1 = \mathbf{1}$ | estudiar                 |
| $f_6(X) = 1 * 3 + 0 * 1 + 1 * 5 = 6$ | $Y_1 = \mathbf{1}$ | nachos y estudiar        |
| $f_7(X) = 0 * 3 + 1 * 1 + 1 * 5 = 8$ | $Y_1 = \mathbf{1}$ | serie y estudiar         |
| $f_8(X) = 1 * 3 + 1 * 1 + 1 * 5 = 9$ | $Y_1 = \mathbf{1}$ | serie, nachos y estudiar |

# Simplificando.





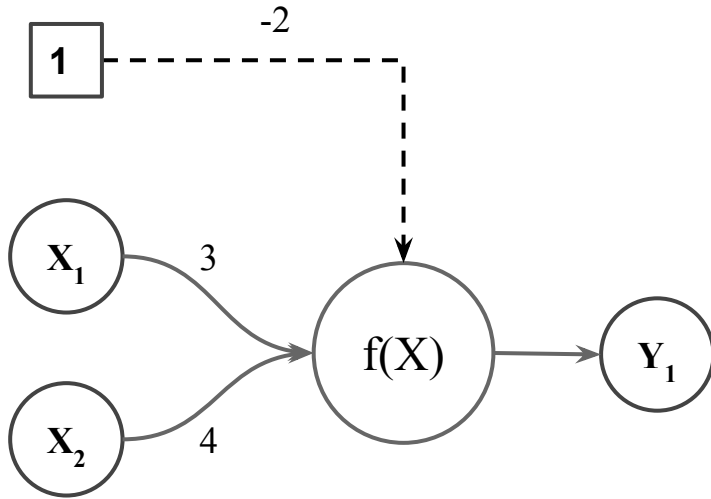
# Ejemplo



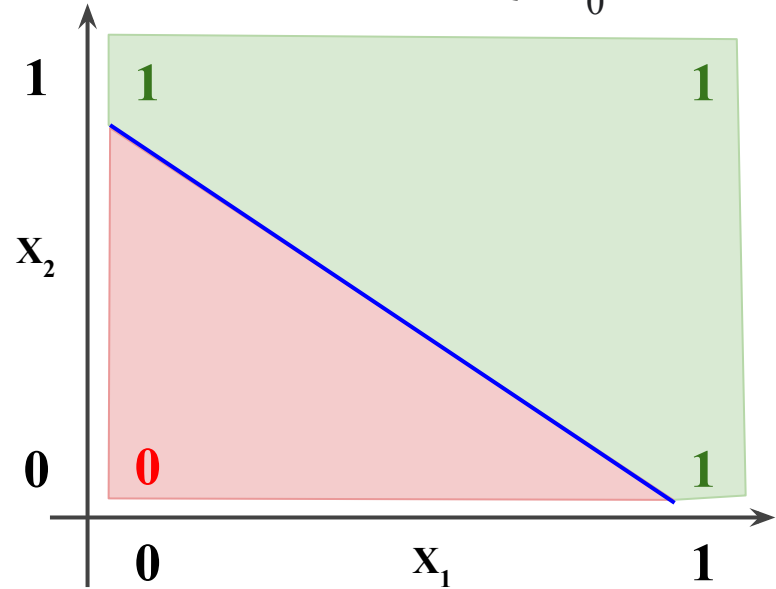
$$f(X) = (-4) * (0) + (-4) * (0) + (1 * 3) = 3$$

# Funciones lógicas OR

$$\text{output} = \begin{cases} 0 & \text{if } \sum_i w_i \cdot X_i + b \leq 0 \\ 1 & \text{if } \sum_i w_i \cdot X_i + b > 0 \end{cases}$$

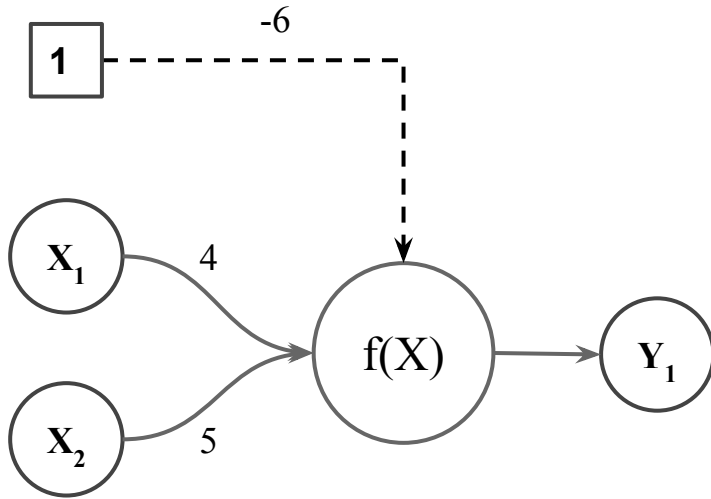


$$f(X) = W_1 \cdot X_1 + W_2 \cdot X_2 + b$$

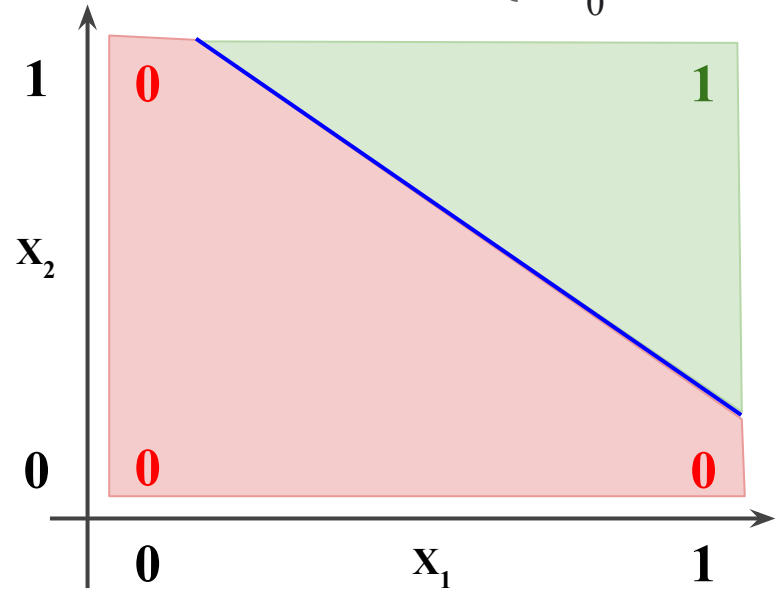


# Funciones lógicas AND

$$\text{output} = \begin{cases} 0 & \text{if } \sum_i w_i \cdot X_i + b \leq 0 \\ 1 & \text{if } \sum_i w_i \cdot X_i + b > 0 \end{cases}$$

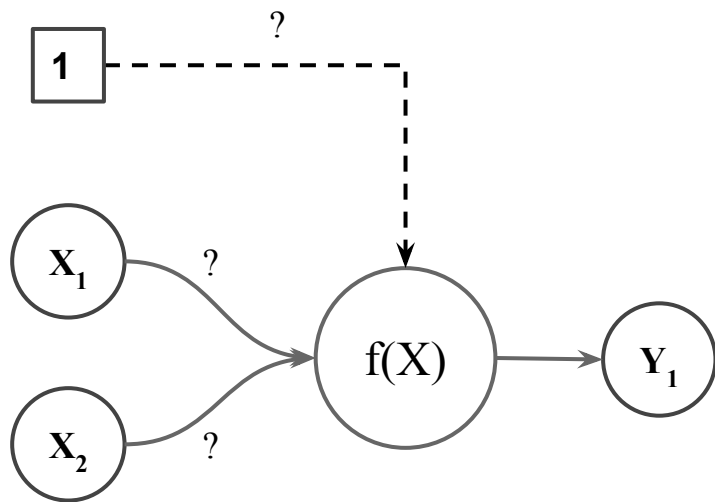


$$f(X) = w_1 \cdot X_1 + w_2 \cdot X_2 + b$$

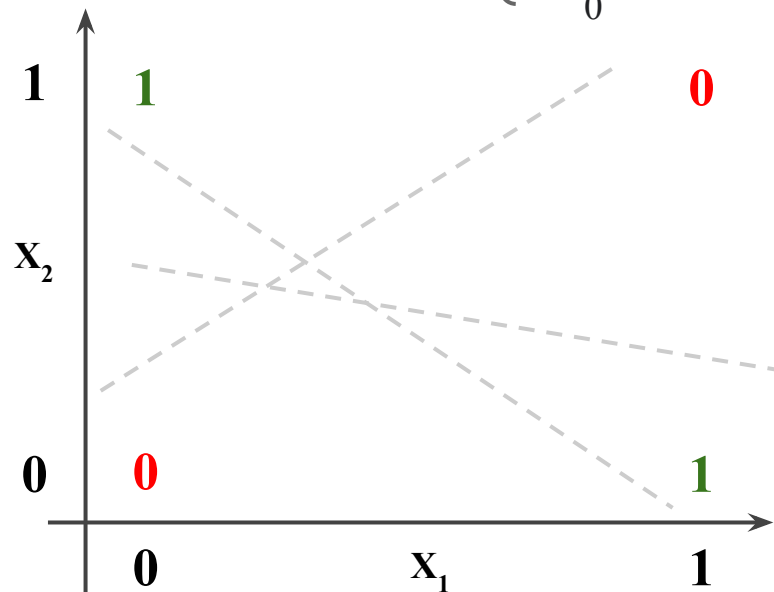


# Funciones lógicas XOR

$$\text{output} = \begin{cases} 0 & \text{if } \sum_i w_i \cdot X_i + b \leq 0 \\ 1 & \text{if } \sum_i w_i \cdot X_i + b > 0 \end{cases}$$



$$f(X) = W_1 \cdot X_1 + W_2 \cdot X_2 + b$$

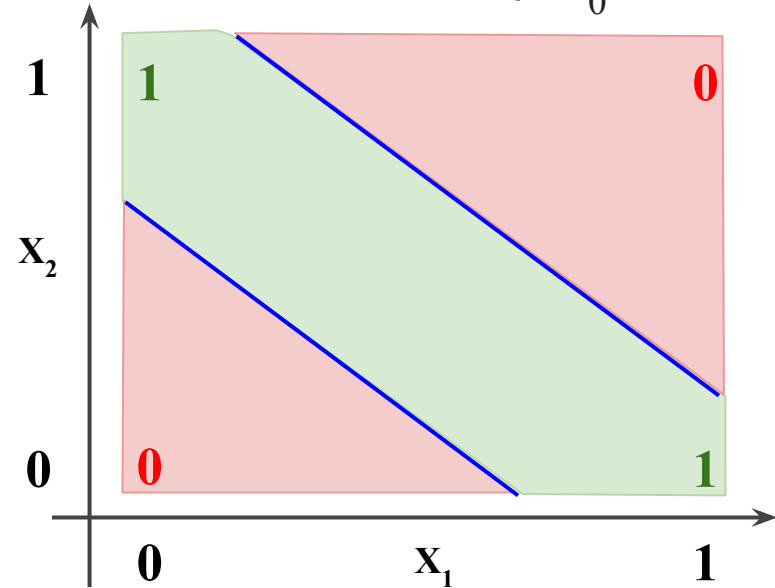
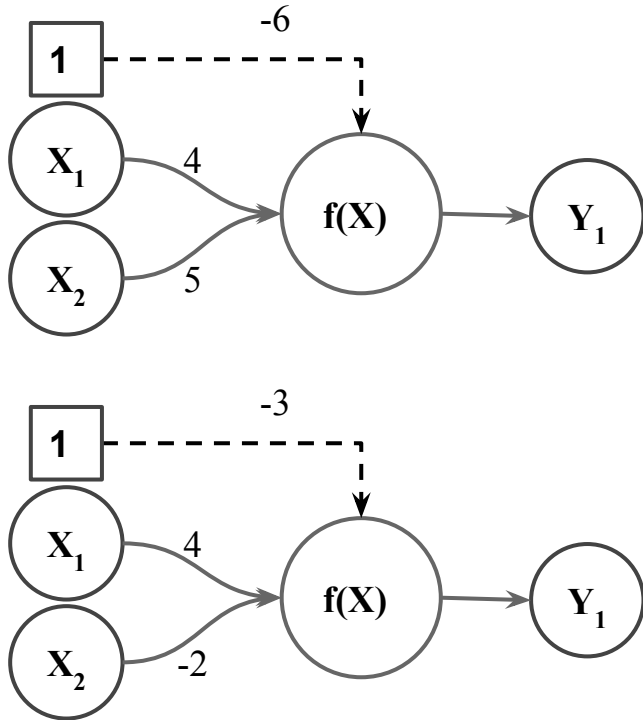


**No se puede solucionar!**

# XOR: solución

¡Agregar una neurona!

$$\text{output} = \begin{cases} 0 & \text{if } \sum_i w_i \cdot X_i + b \leq 0 \\ 1 & \text{if } \sum_i w_i \cdot X_i + b > 0 \end{cases}$$



$$f(X) = W_1 \cdot X_1 + W_2 \cdot X_2 + b$$

# Estructura general de una red neuronal

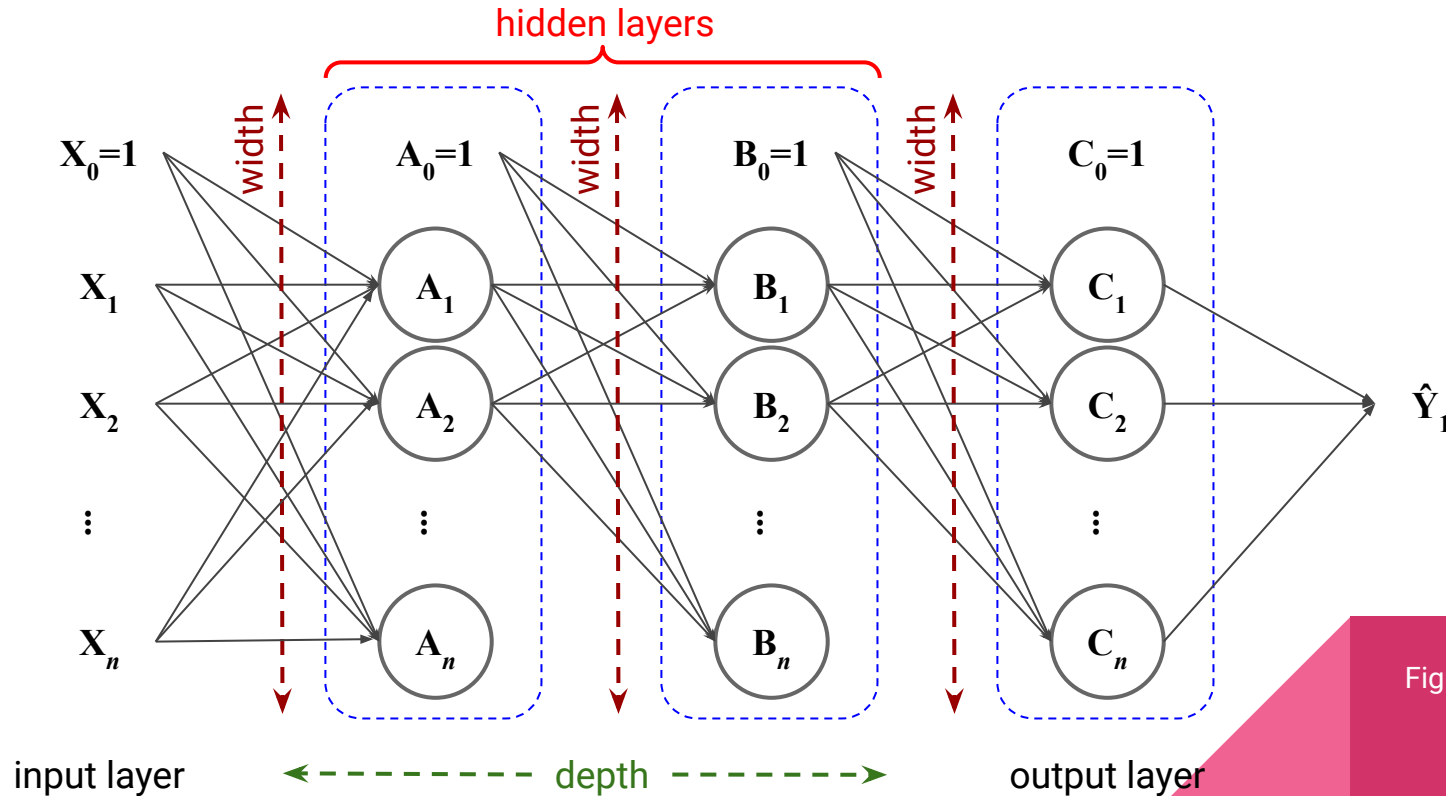
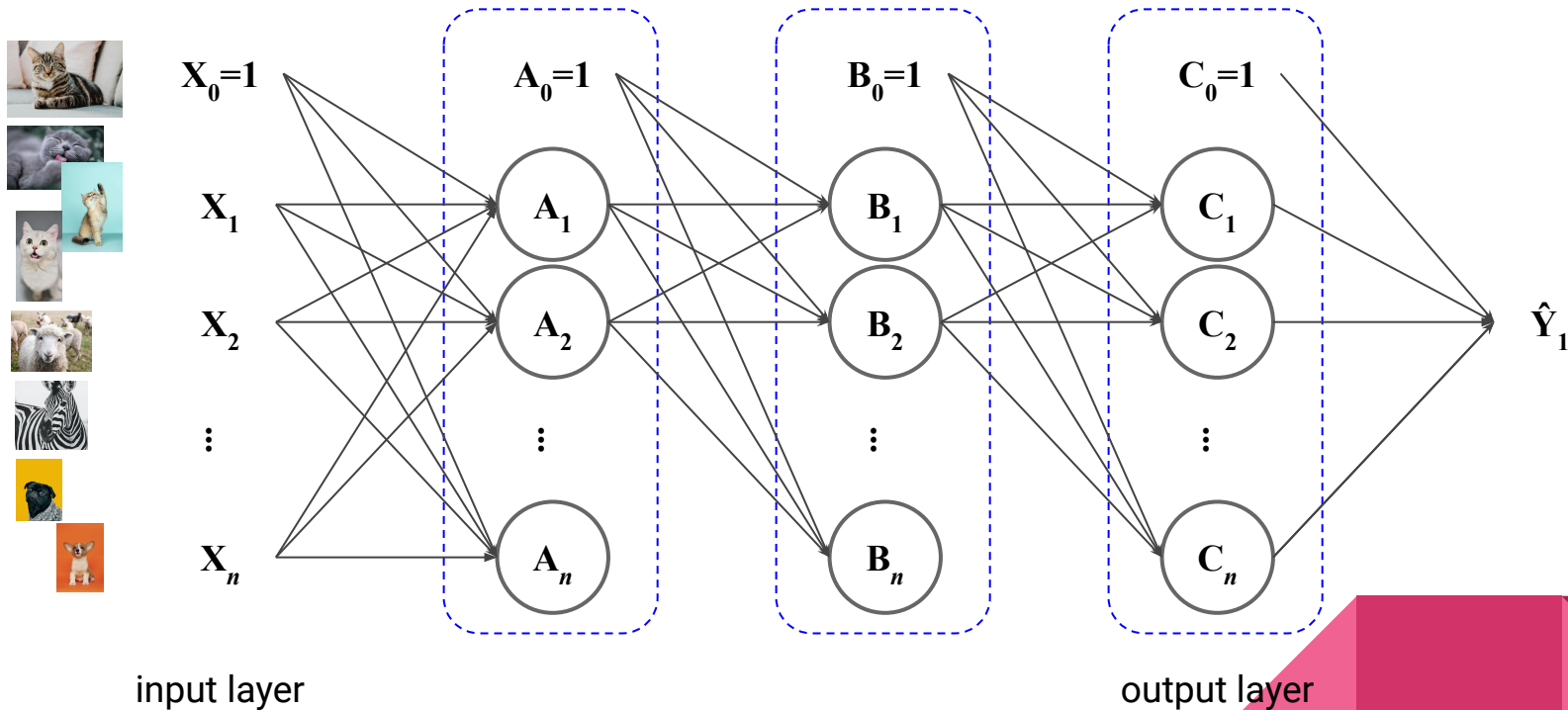
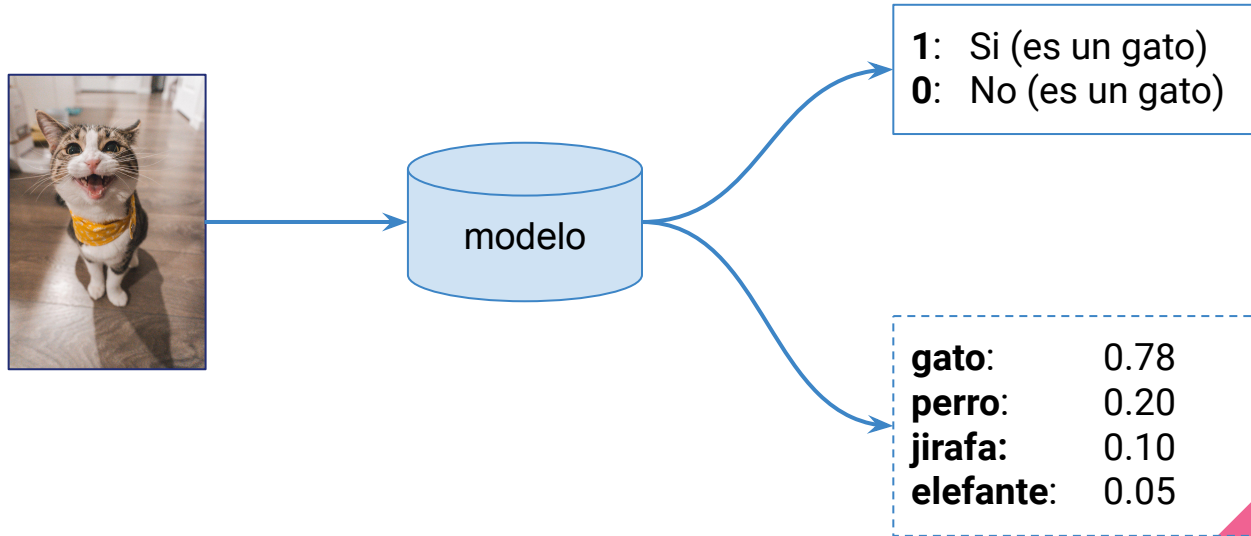


Figura adaptada de [5]

# Clasificador de gatos. Entrenar modelo



# Clasificador de gatos





# Referencias

1. Michael Nielsen. (2019, December). *Neural Networks and Deep Learning: Why are deep neural networks hard to train?*. <http://neuralnetworksanddeeplearning.com/chap5.html>
2. Andrew Ng. (n.d.). *Coursera: Machine Learning, Non-linear Hypotheses*. <https://www.coursera.org/learn/machine-learning/lecture/OAOhO/non-linear-hypotheses>
3. Murphy, K. P. (2022). *Probabilistic Machine Learning: An introduction*. MIT Press.
4. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.
5. Ketkar, N. (2017). *Deep learning with Python* (Vol. 1). Berkeley, CA: Apress.
6. Andrew Ng. (n.d.). *Coursera: Machine Learning, Neurons and the Brain*. <https://www.coursera.org/learn/machine-learning/lecture/IPmzw/neurons-and-the-brain>
7. Tappert, C. C. (2019, December). Who Is the Father of Deep Learning?. In 2019 International Conference on Computational Science and Computational Intelligence (CSCI) (pp. 343-348). IEEE.