



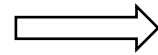
Machine Learning

Introducción a Redes Neuronales

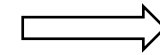
Motivación

- ¿Por qué un nuevo algoritmo de clasificación o regresión si ya tenemos varios?
 - Regresión lineal, polinomial, logística, Naïve Bayes
- Debemos aprender mejores herramientas.
 - Para calcular hipótesis más complejas

Redes Neuronales y Aprendizaje Profundo

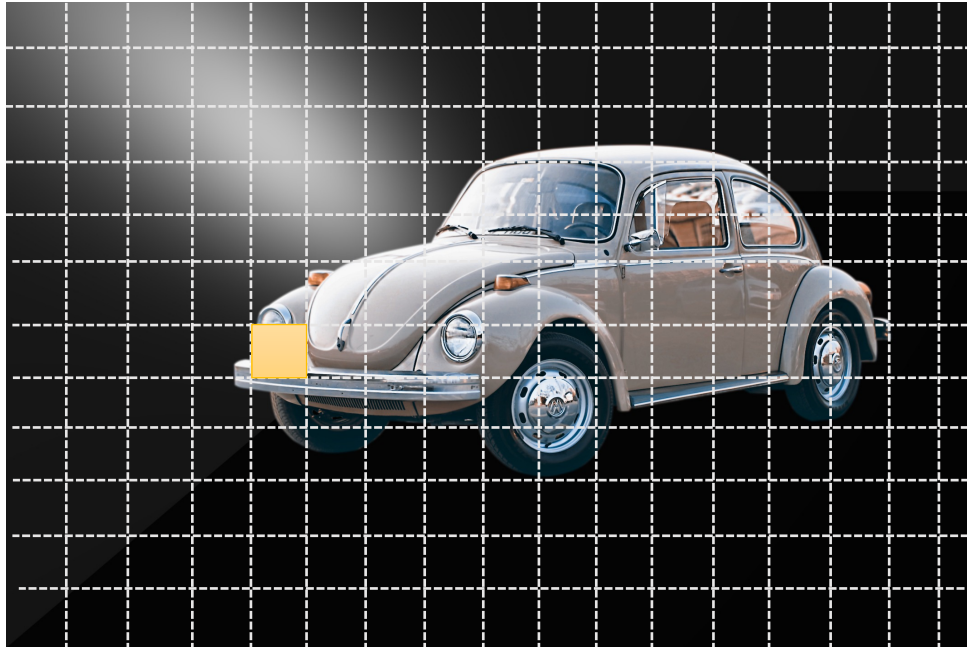


Caja Negra
(Red
Neuronal)



Determinar el
tipo de animal
que se encuentra
en la imagen

Motivación

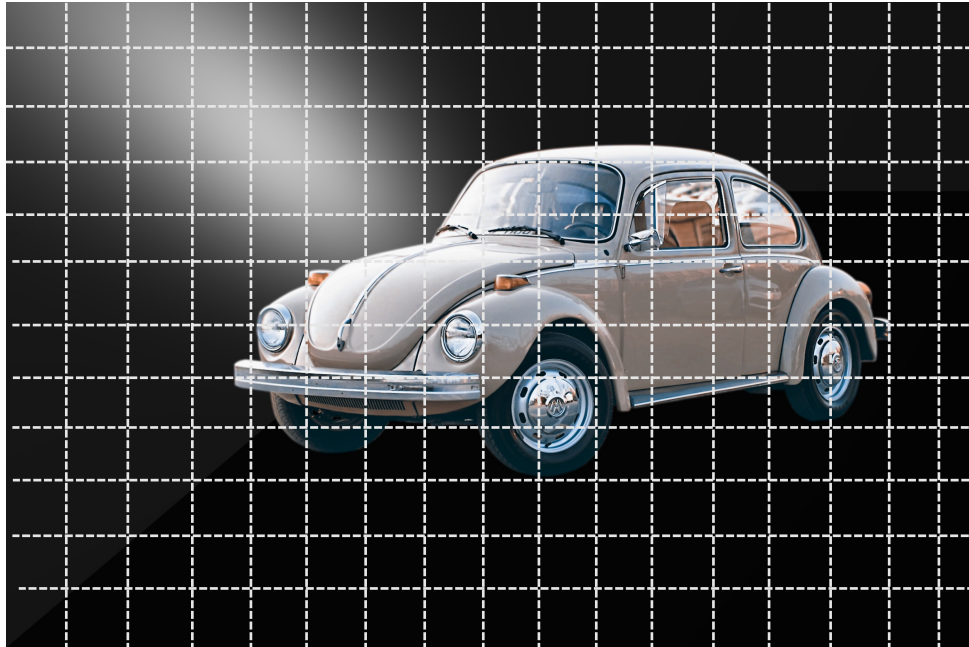


Consideremos el modelo de regresión logística para determinar si una imagen contiene un vehículo o no.

¿Cuáles serían las características?

- Los pixeles de la imagen
- Ahora, si una imagen es de tamaño 50x50, tendríamos 2500 (en blanco y negro)
- Si es RGB, ¡tendríamos 7500!

Motivación



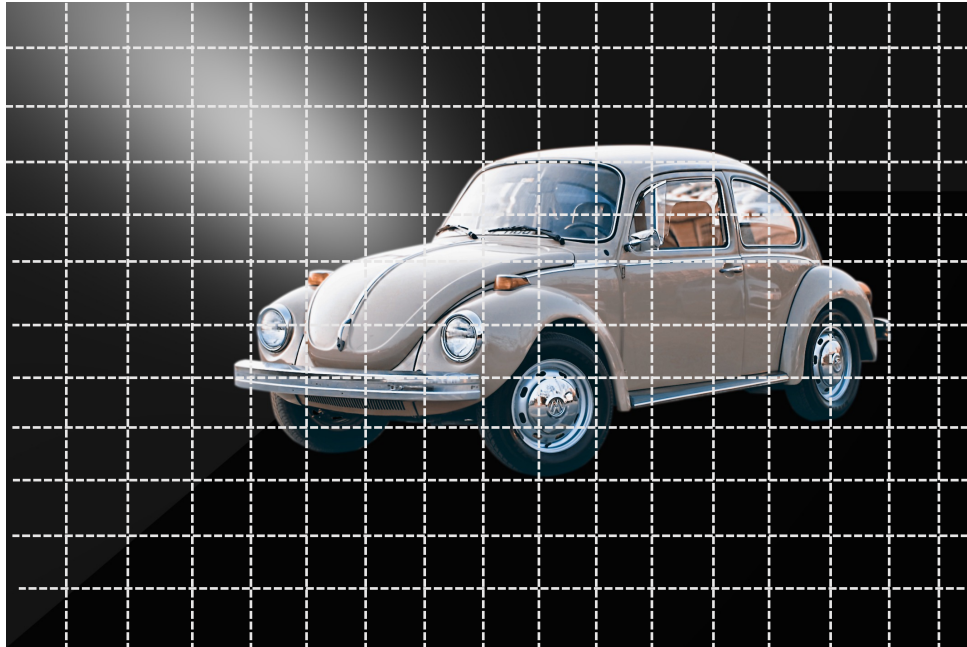
Al realizar el modelo de clasificación

$$g(z)$$

se consideran muchos, muchos parámetros, inclusive en el simple caso del modelo lineal:

$$z = x_0 + \theta_1 x_1 + \dots + \theta_{2500} x_{2500}$$

Motivación



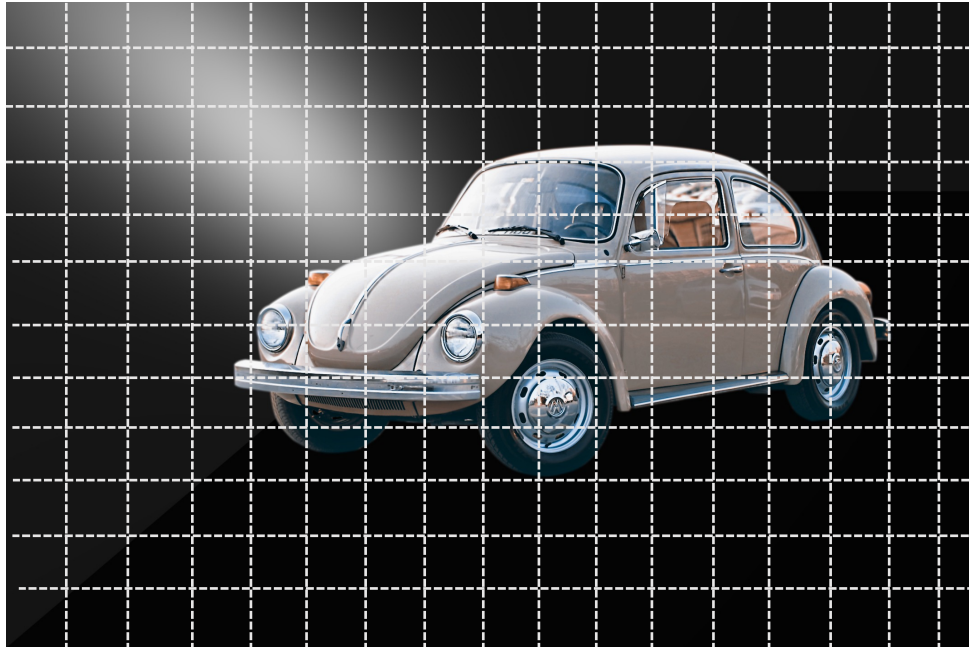
El modelo lineal es limitado, por lo que se puede hacer más complicado. Por ejemplo, podemos agregar como características TODAS las combinaciones que resultan al multiplicar dos pixeles, e.g., para x_1 :

$$x_1^2, x_1x_2, x_1x_3, \dots x_1x_{2500}$$

¡Esto crece, y mucho!

¿Cuántas características tendría el modelo? ¡A calcular!

Motivación



Conclusión:

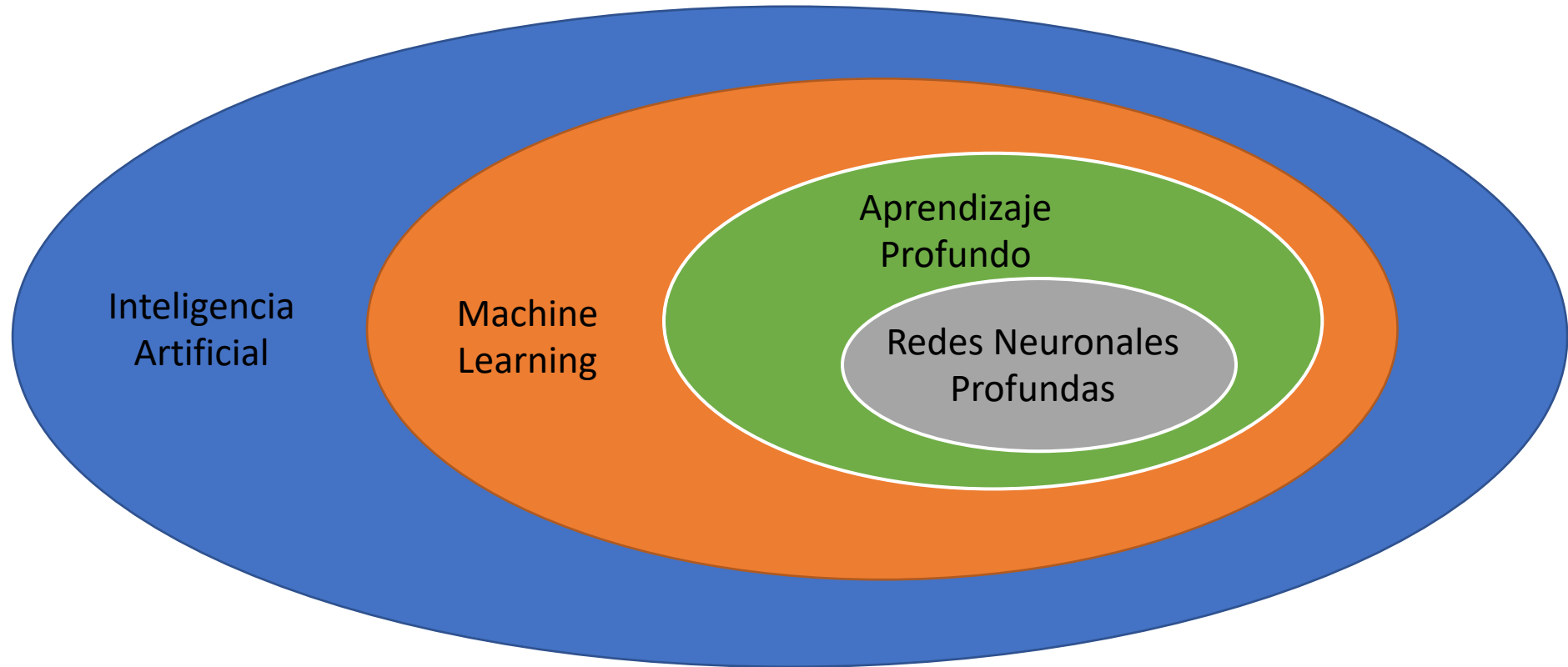
- Para este tipo de problemas más complejos de clasificación, se necesitan modelos que soporten una cantidad asquerosa de características.
- Hasta el momento, lo que tenemos puede resultar poco útil.

Redes Neuronales y Aprendizaje Profundo

Es difícil definir *Deep Learning* (DL):

- Una forma de verlo es que el DL comprende lo que ***NO ES Machine Learning*** (ML).
- Comprende algoritmos de ML que usan múltiples capas de unidades computacionales donde cada capa aprende su propia representación de los datos de entrada, que son combinados en cada capa subsecuente de forma jerárquica → Una red neuronal → Perceptrón multicapa (?)

Redes Neuronales y Aprendizaje Profundo



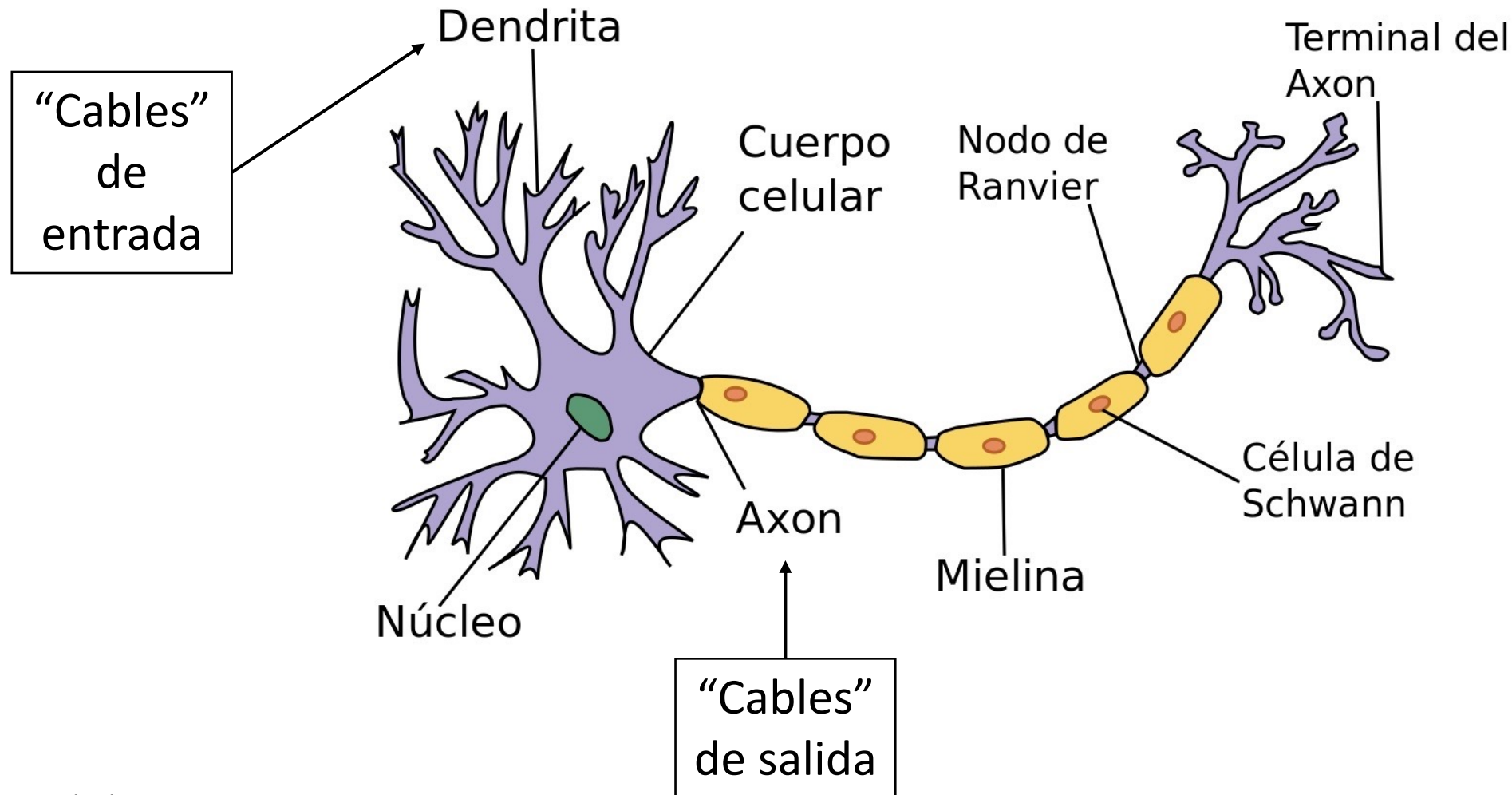
Relación entre inteligencia artificial, Machine Learning, Deep Learning y Redes Neuronales Profundas.

Redes Neuronales

- Se inspiran en el cerebro humano, específicamente en la neurona.
- En la actualidad, es un modelo que representa el estado del arte en la inteligencia artificial.
- Permite atacar problemas complejos, especialmente tareas que realiza el ser humano de manera natural.

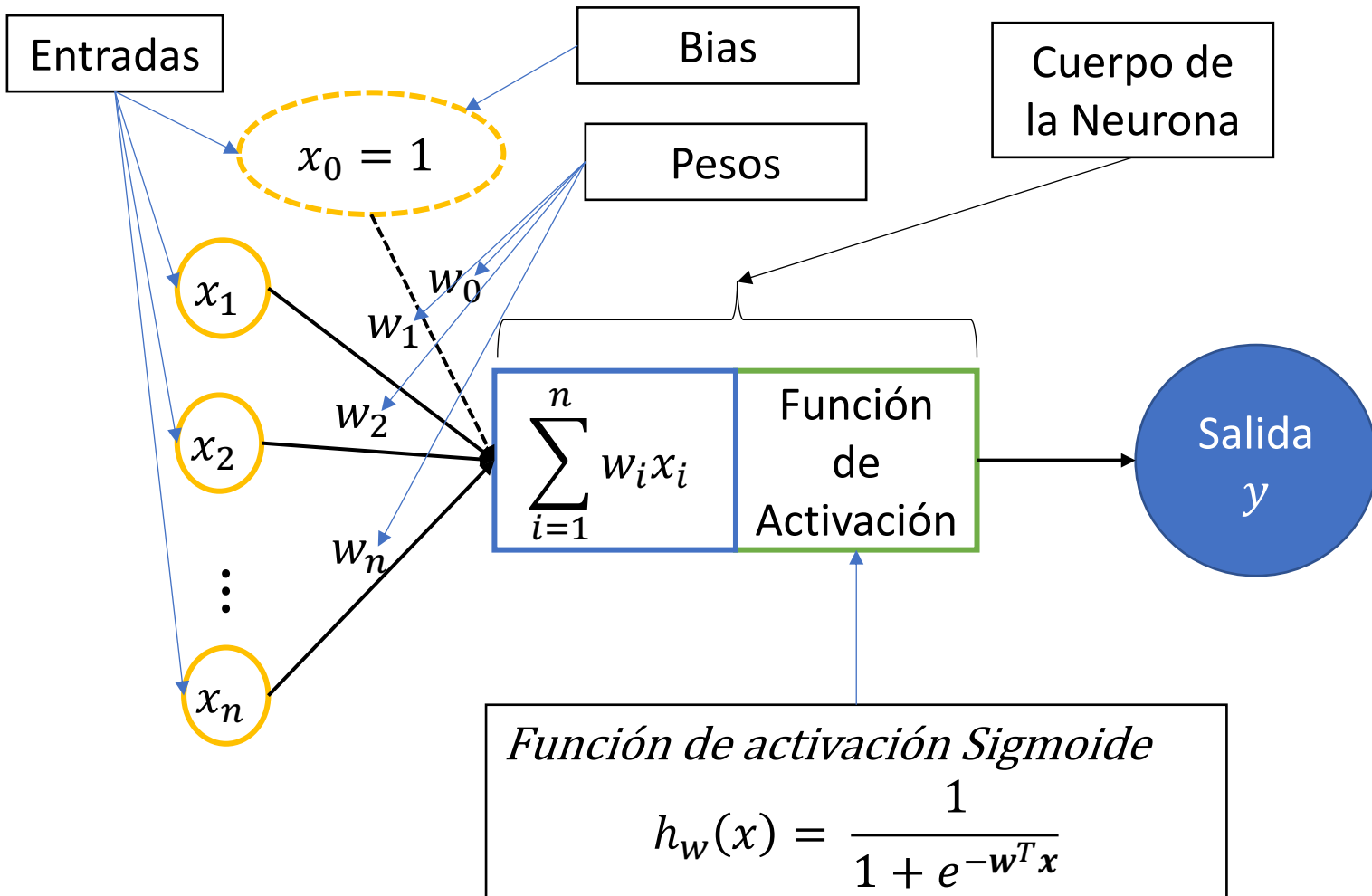


La Neurona Biológica



La neurona se puede ver como una unidad computacional. Entra información por la dendrita y manda señales a otras neuronas por el axón.

Estructura Típica de una Neurona Artificial



Se calcula la señal que entra a la neurona ponderándolas:

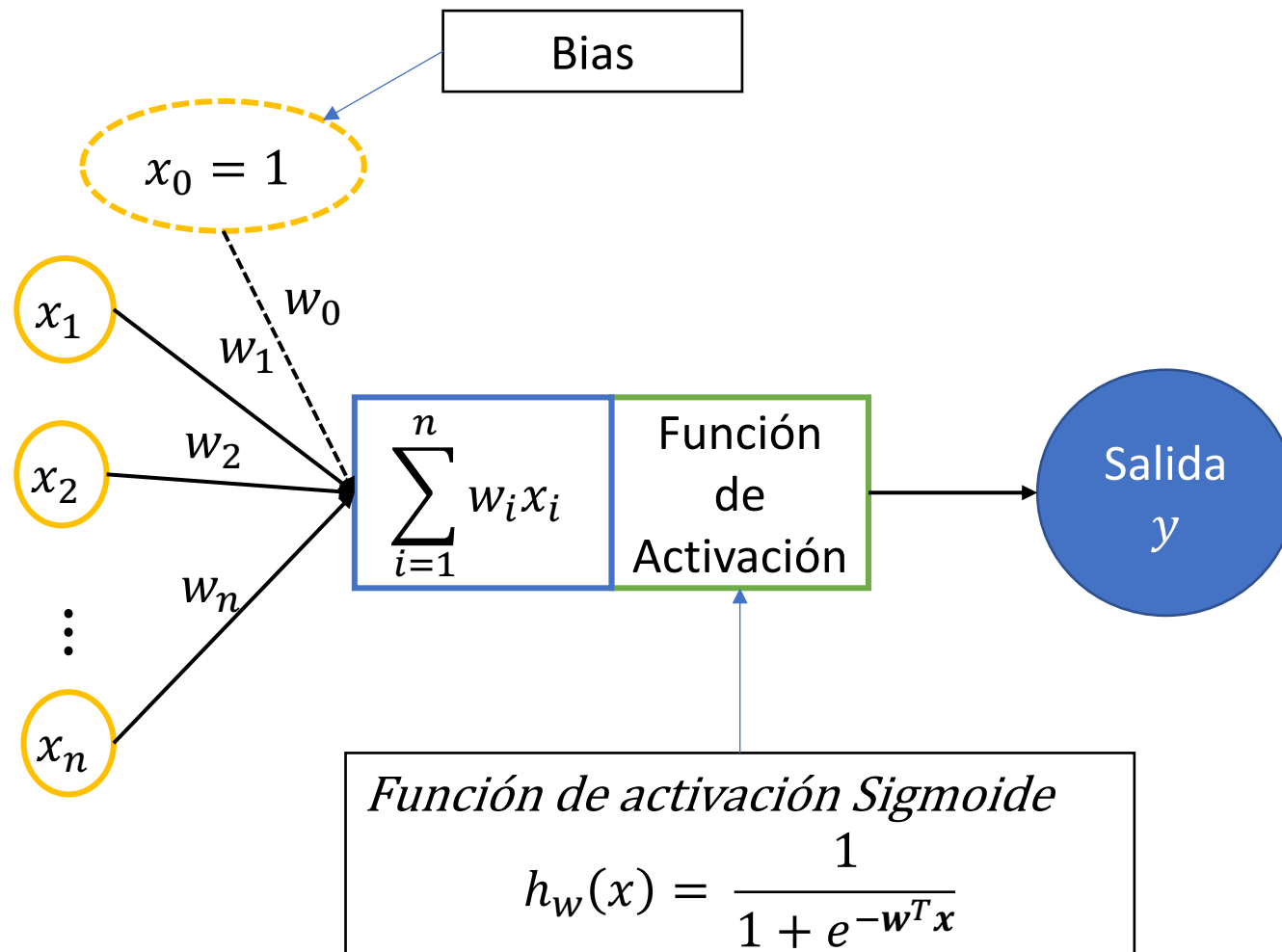
$$w_0x_0 + w_1x_1 + w_2x_2 + \cdots + w_nx_n \\ = \sum_{i=1}^n w_i x_i$$

Si exceden cierto valor de umbral, la neurona se activa:

$$y = h_w \left[\sum_{i=1}^n w_i x_i \right]$$

En el caso anterior, h_w es la función sigmoide. Existen otras, además de esta.

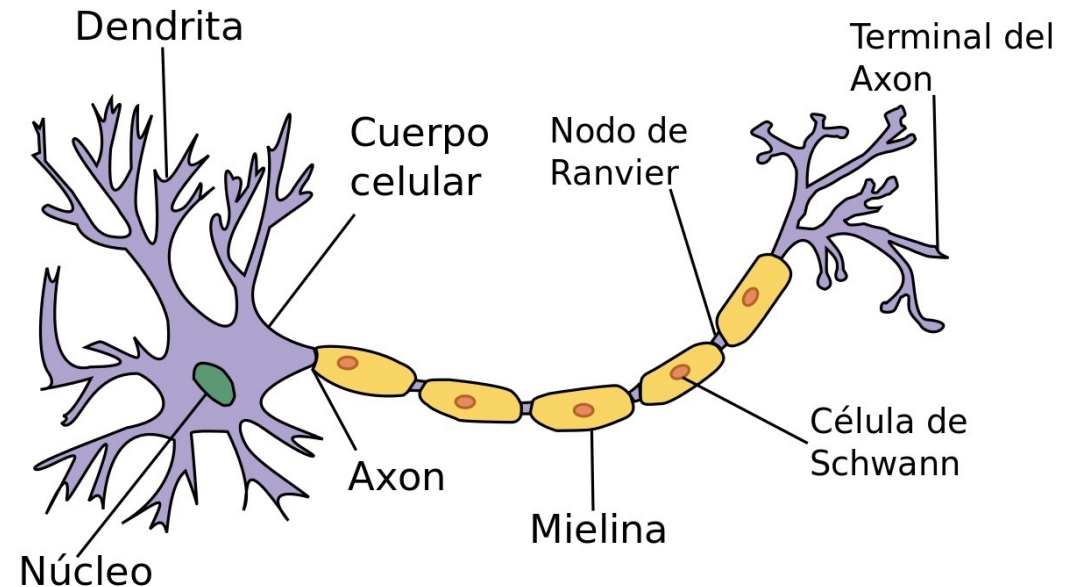
Estructura Típica de una Neurona Artificial



- El Bias, que siempre es constante, es una entrada adicional en cada capa de neuronas que tiene el valor de 1.
- No son influenciadas por las capas anteriores (no entra nada) pero sí mandan señales con sus propios pesos.
- El Bias garantiza que, si todas las entradas son cero, existirá una activación en la neurona.

Neurona de McCulloch y Pitts

- En 1943 se da el primer intento para simular el comportamiento biológico de la neurona humana.
- Proponen un proceso de aprendizaje *complejo* por medio de un modelo matemático.
- Permite clasificar patrones, una imitación del sistema biológico sensorial, por medio de una unidad computacional.

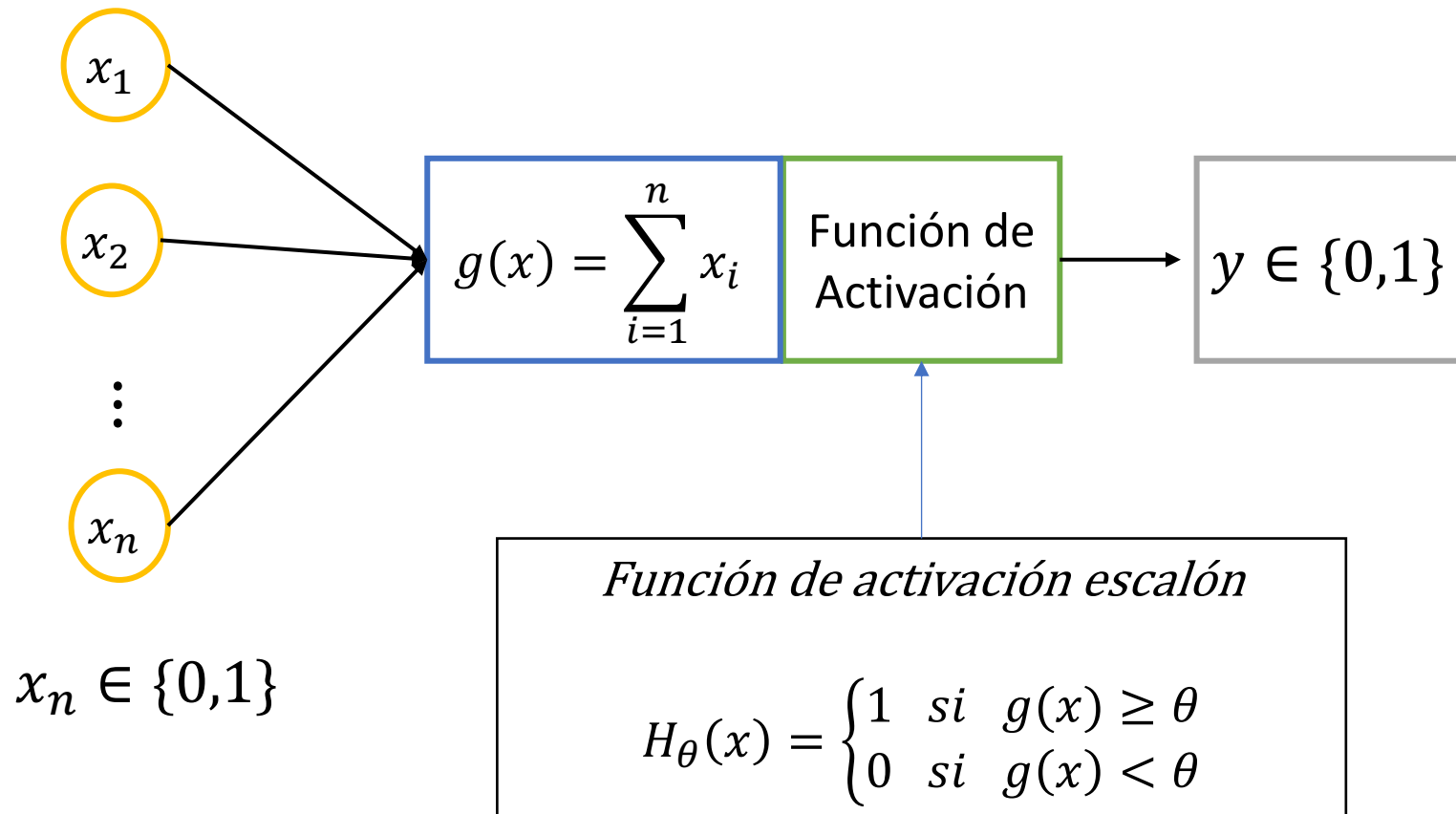


Neurona artificial de McCulloch & Pitts. Cornelio Yáñez Vázquez, Luis Octavio López Leyva y Mario Aldape Pérez. Instituto Politécnico Nacional. Centro de Investigación en Computación, 2007. Disponible en: <https://www.repositoriodigital.ipn.mx/handle/123456789/8640>

Ejemplo: Neurona de McCulloch y Pitts

En el caso de la neurona de McCulloch y Pitts, las salidas y entradas solo pueden ser 1 ó 0.

Es decir, son valores booleanos.



Ejemplo: Neurona de McCulloch y Pitts

Supongamos que queremos predecir la decisión de un amigo de ver un partido de *football* en la televisión.

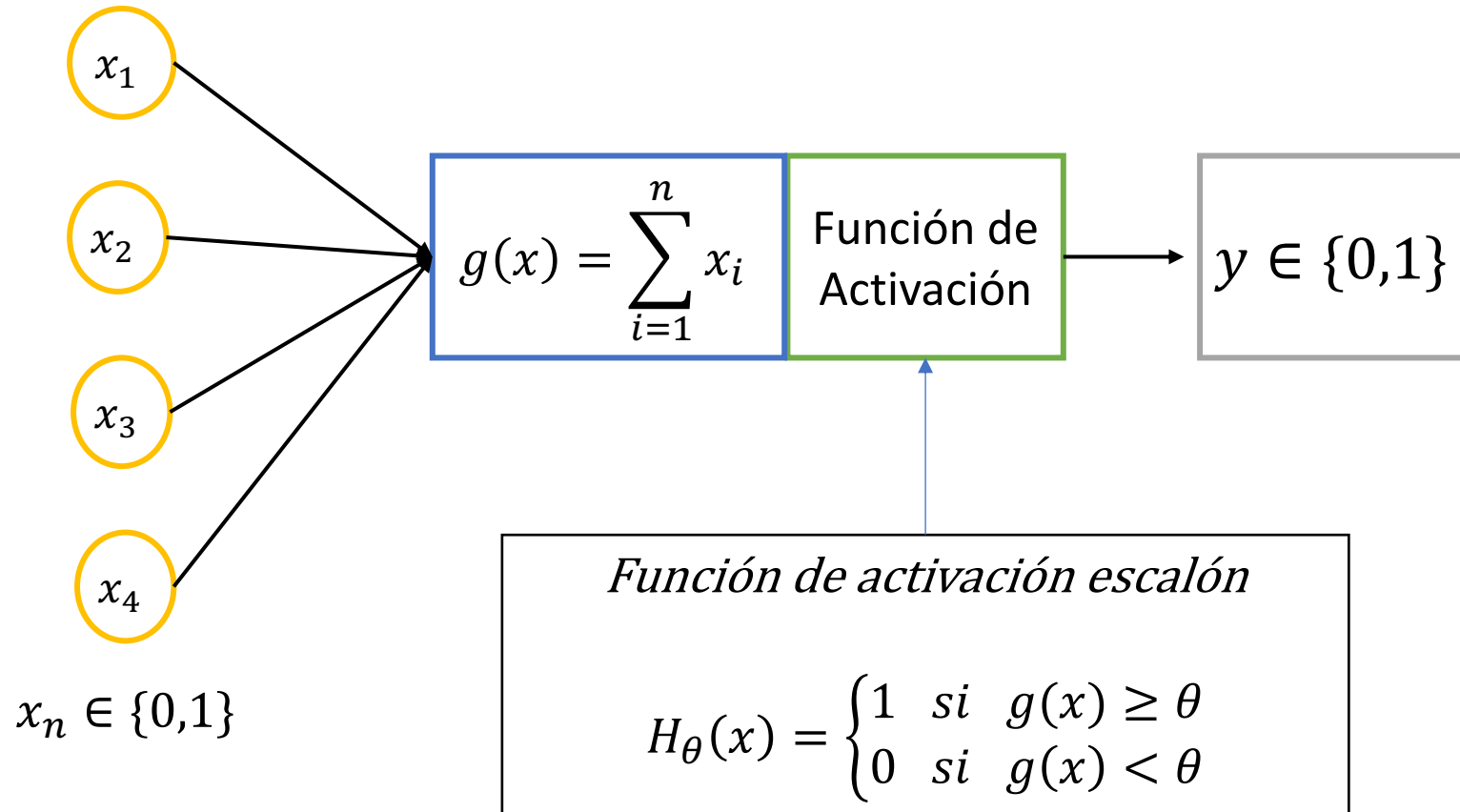
- Las entradas y salidas son valores booleanos (1 = sí, 0 = no)

Vamos a modelar nuestro problema:

- x_1 indica si el partido es de la *Premier League*.
- x_2 indica si es un partido competitivo.
- x_3 indica si el partido se juega en un horario disponible.
- x_4 indica si el partida incluye al equipo favorito.



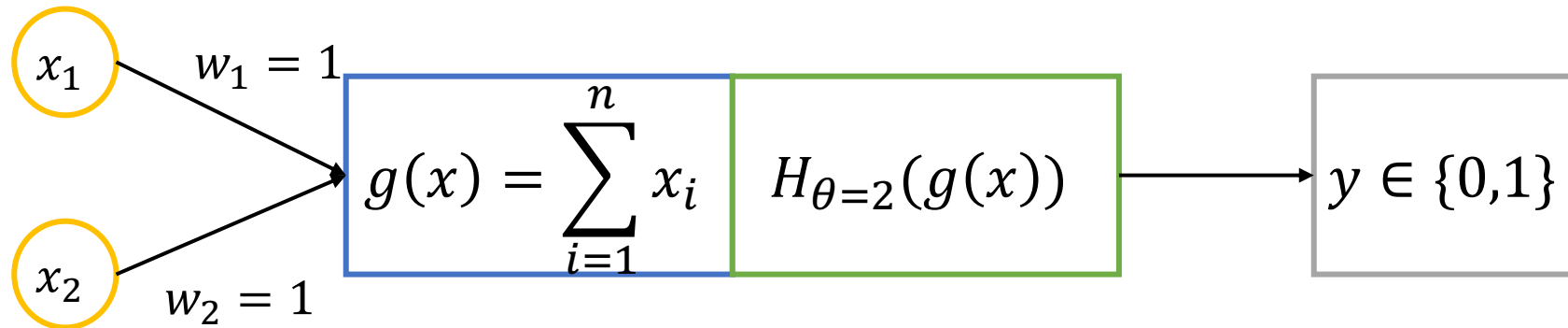
Ejemplo: Neurona de McCulloch y Pitts



En este problema, lo que falta es determinar el valor de θ , el límite de decisión, para que la respuesta sea excitatoria o inhibitoria.

Ejemplo: Neurona de McCulloch y Pitts

Función AND lógica

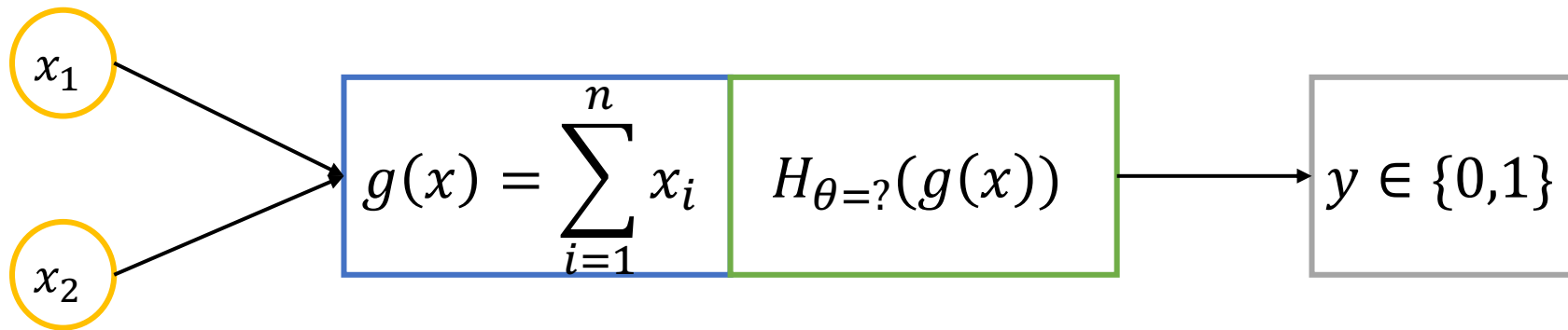


Función de activación escalón

$$H_{\theta}(x) = \begin{cases} 1 & \text{si } g(x) \geq \theta \\ 0 & \text{si } g(x) < \theta \end{cases}$$

Ejemplo: Neurona de McCulloch y Pitts

Función OR lógica



Función de activación escalón

$$H_{\theta}(x) = \begin{cases} 1 & \text{si } g(x) \geq \theta \\ 0 & \text{si } g(x) < \theta \end{cases}$$

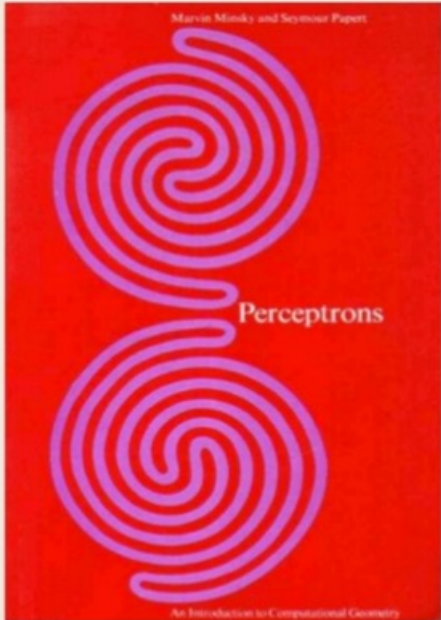
$\theta = ?$

$\theta = 1$

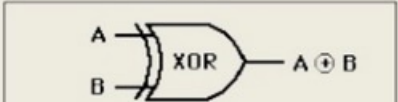
Neurona de McCulloch y Pitts

- La neurona de McCulloch y Pitts (y en general, el perceptrón) tiene sus limitantes.
- No pueden representar funciones muy complejas.
- Esto ocasionó, en 1969, el primer invierno de la IA, principalmente por el libro de Minsky y Papert.
- Sin embargo, esto se resuelve si se combinan múltiples neuronas.

1969: Perceptrons can't do XOR!




<http://www.i-programmer.info/images/stories/BabBag/AI/book.jpg>



A	B	Out
0	0	0
0	1	1
1	0	1
1	1	0

<http://hyperphysics.phy-astr.gsu.edu/hbase/electronic/ietron/xor.gif>



Minsky & Papert

<https://constructingkids.files.wordpress.com/2013/05/minsky-papert-71-csolomon-x640.jpg>

Ejemplo: Neurona de McCulloch y Pitts

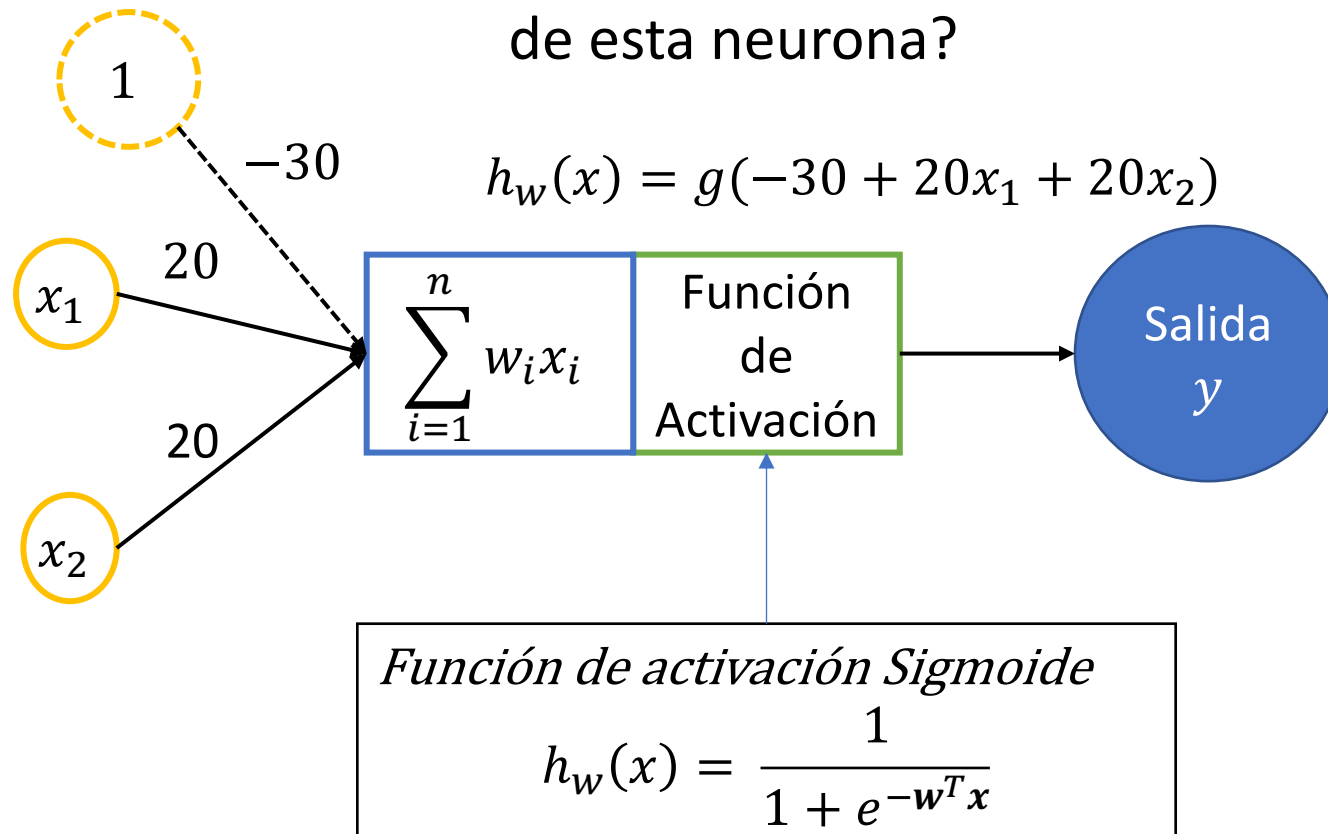
Tarea:

1. Investiguen el problema de determinar XOR con la neurona de McCulloch – Pitts.
2. ¿Cómo se ve gráficamente el XOR?
3. ¿Cómo se resuelve el problema del XOR con la neurona de McCulloch-Pitts?
4. ¿Qué causó este problema en el campo de las redes neuronales y, en general, en la inteligencia artificial?

Primera Intuición: ¿Qué hace una neurona?

Función AND

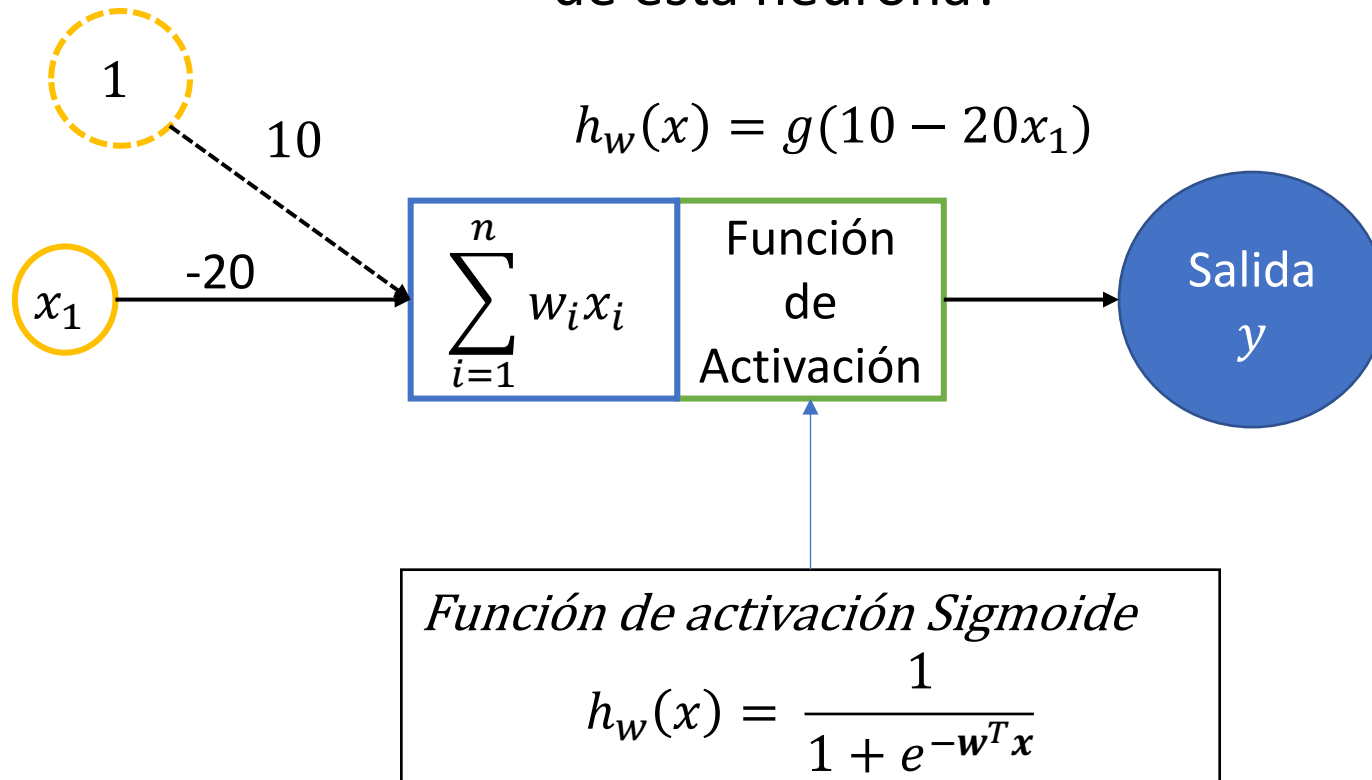
Ejercicio: ¿Qué saldría de esta neurona?



x_1	x_2	$h_w(x)$
1	1	
1	0	
0	1	
0	0	

Primera Intuición: ¿Qué hace una neurona?

Ejercicio: ¿Qué saldría de esta neurona?



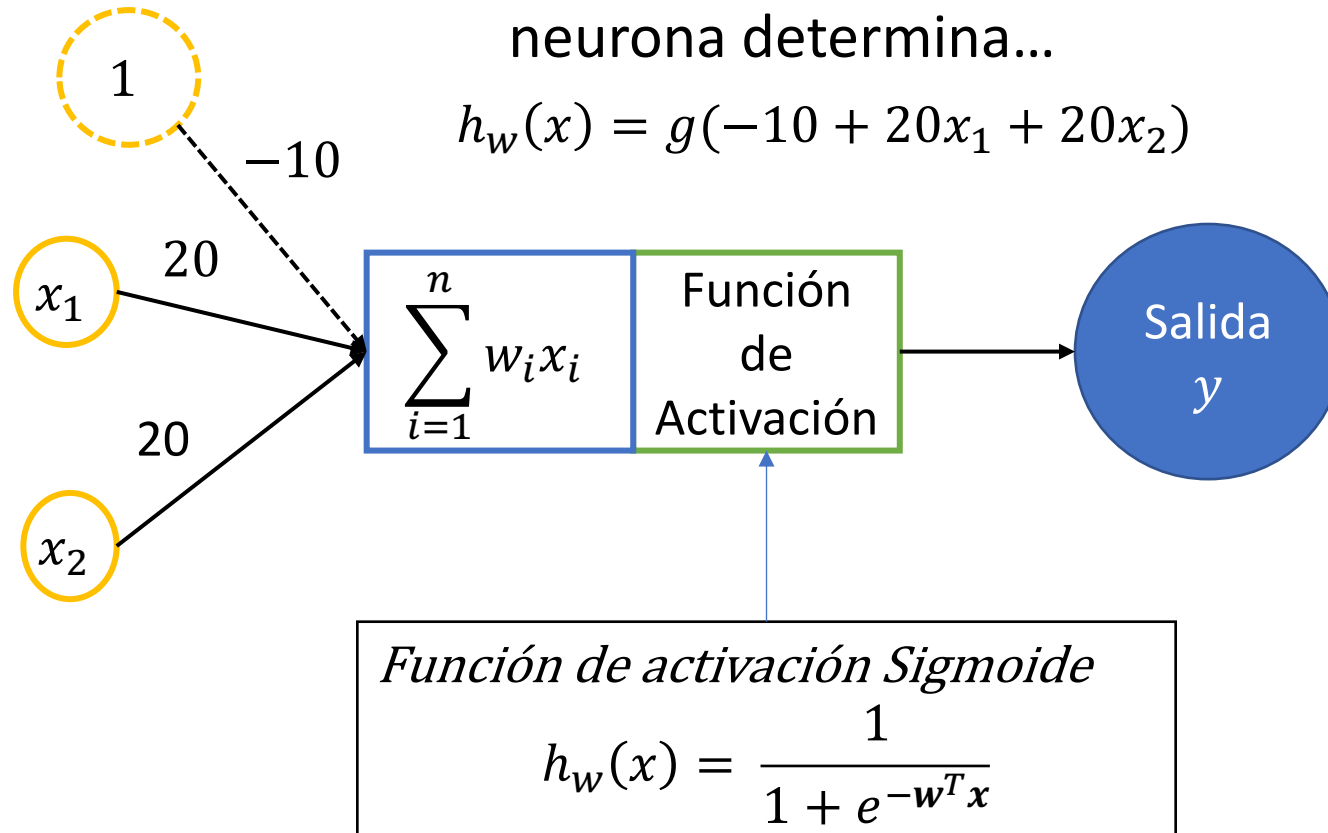
Función NOT

x_1	$h_w(x)$
1	
0	

Primera Intuición: ¿Qué hace una neurona?

Ejercicio: Verificar que esta neurona determina...

$$h_w(x) = g(-10 + 20x_1 + 20x_2)$$

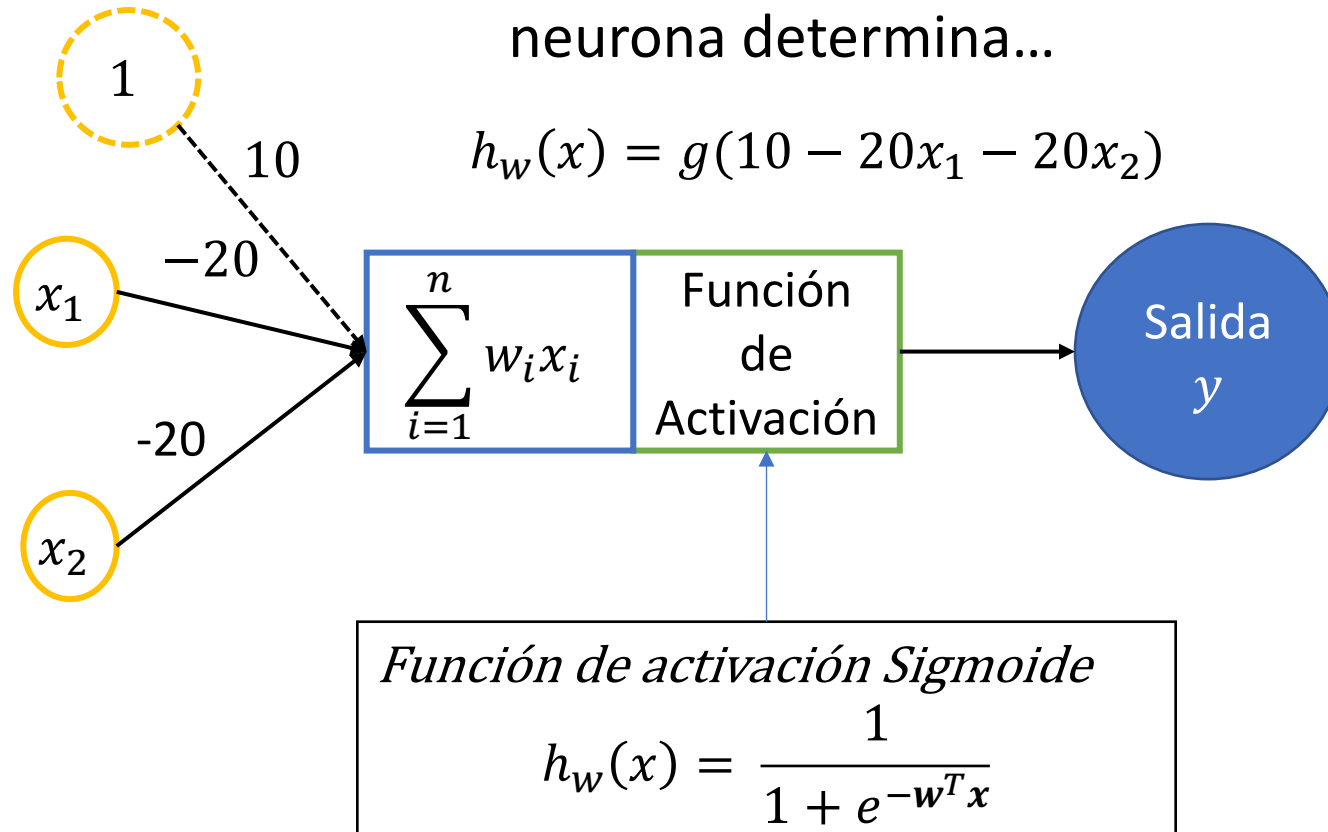


Función OR

x_1	x_2	$h_w(x)$
1	1	
1	0	
0	1	
0	0	

Primera Intuición: ¿Qué hace una neurona?

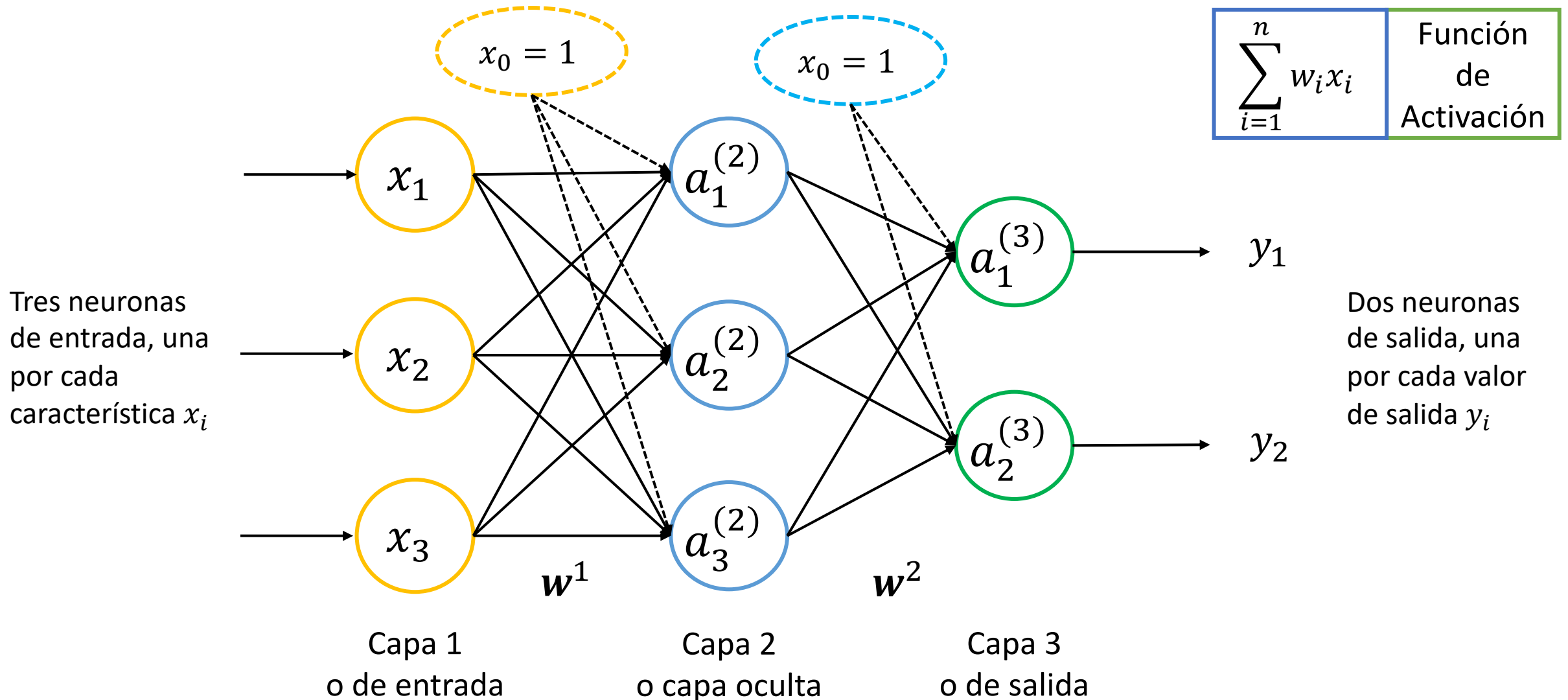
Ejercicio: Verificar que esta neurona determina...



Función
 $\neg x_1 \text{ AND } \neg x_2$

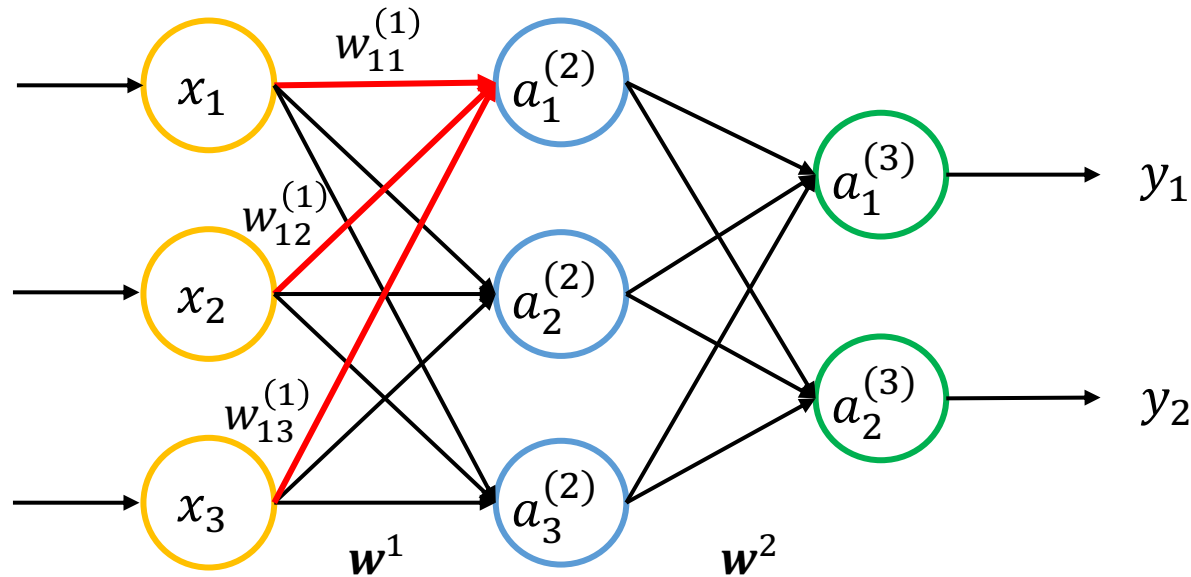
x_1	x_2	$h_w(x)$
1	1	
1	0	
0	1	
0	0	

De la Neurona a una Red Neuronal



De la Neurona a una Red Neuronal

$\sum_{i=1}^n w_i x_i$	Función de Activación
------------------------	-----------------------------



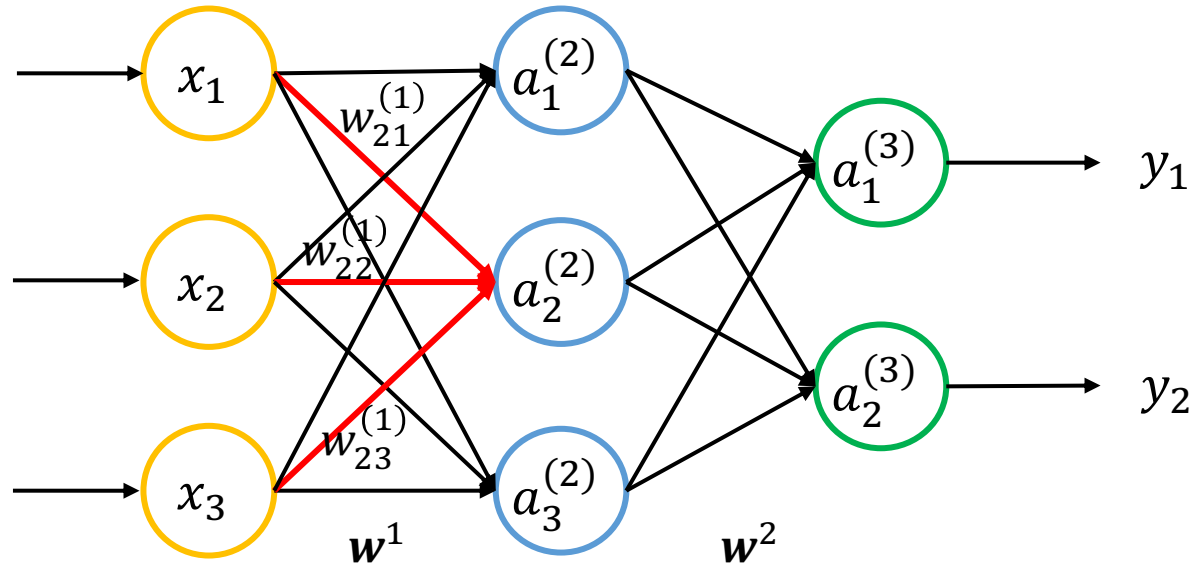
$a_i^{(j)}$ = activación o valor de salida de la neurona i en la capa j

w^j = matriz de pesos de la capa j a la capa $j+1$

$$a_1^{(2)} = g(w_{10}^{(1)} x_0 + w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2 + w_{13}^{(1)} x_3)$$

De la Neurona a una Red Neuronal

$\sum_{i=1}^n w_i x_i$	Función de Activación
------------------------	-----------------------------



$a_i^{(j)}$ = activación o valor de salida de la neurona i en la capa j

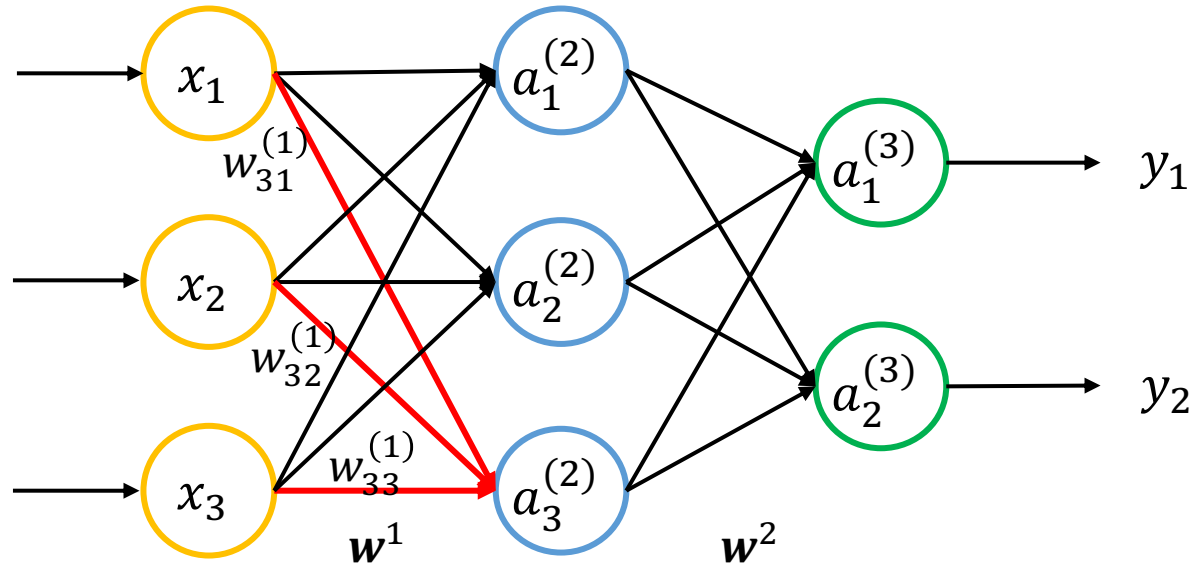
w^j = matriz de pesos de la capa j a la capa $j+1$

$$a_1^{(2)} = g(w_{10}^{(1)} x_0 + w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2 + w_{13}^{(1)} x_3)$$

$$a_2^{(2)} = g(w_{20}^{(1)} x_0 + w_{21}^{(1)} x_1 + w_{22}^{(1)} x_2 + w_{23}^{(1)} x_3)$$

De la Neurona a una Red Neuronal

$\sum_{i=1}^n w_i x_i$	Función de Activación
------------------------	-----------------------------



$a_i^{(j)}$ = activación o valor de salida de la neurona i en la capa j

w^j = matriz de pesos de la capa j a la capa $j+1$

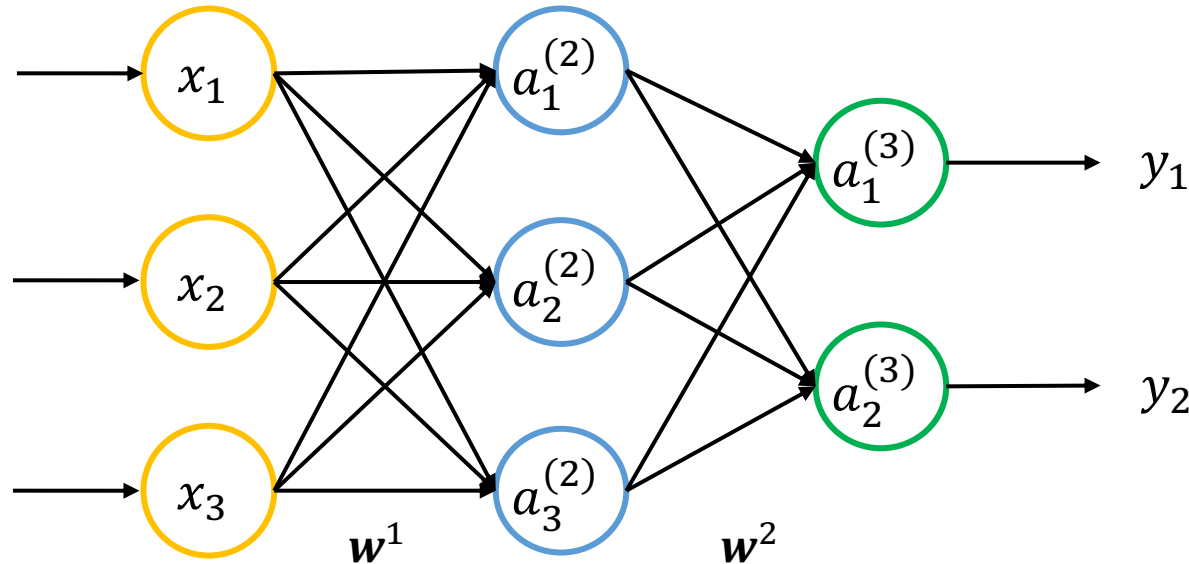
$$a_1^{(2)} = g(w_{10}^{(1)} x_0 + w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2 + w_{13}^{(1)} x_3)$$

$$a_2^{(2)} = g(w_{20}^{(1)} x_0 + w_{21}^{(1)} x_1 + w_{22}^{(1)} x_2 + w_{23}^{(1)} x_3)$$

$$a_3^{(2)} = g(w_{30}^{(1)} x_0 + w_{31}^{(1)} x_1 + w_{32}^{(1)} x_2 + w_{33}^{(1)} x_3)$$

De la Neurona a una Red Neuronal

$\sum_{i=1}^n w_i x_i$	Función de Activación
------------------------	-----------------------



$a_i^{(j)}$ = activación o valor de salida de la neurona i en la capa j

w^j = matriz de pesos de la capa j a la capa $j+1$

$$w^1 \in R^{3 \times 4}$$

$$a_1^{(2)} = g(w_{10}^{(1)} x_0 + w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2 + w_{13}^{(1)} x_3)$$

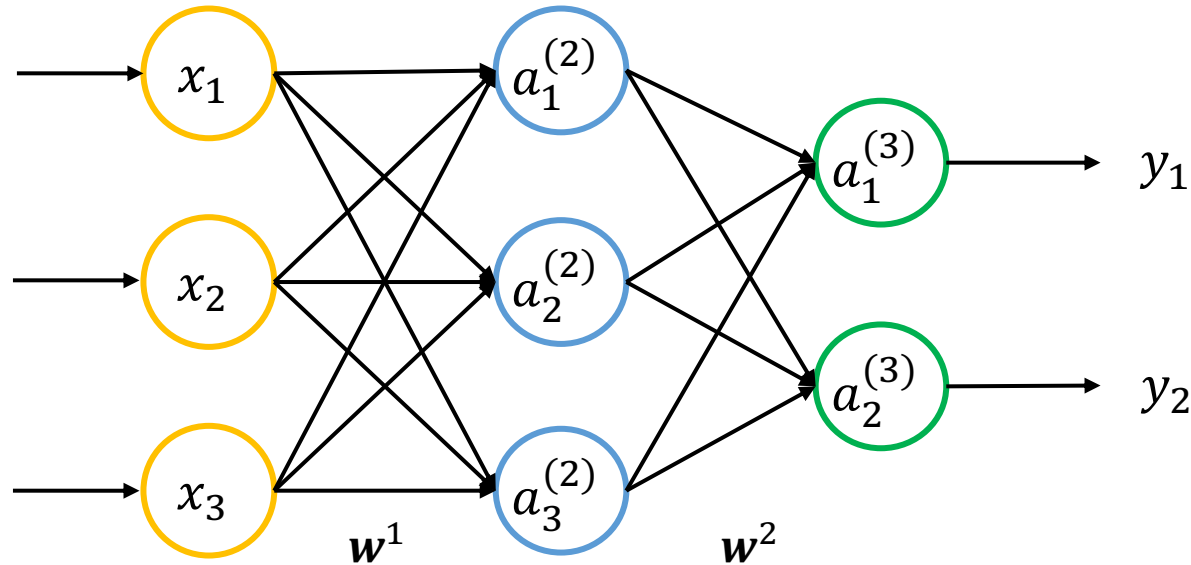
$$a_2^{(2)} = g(w_{20}^{(1)} x_0 + w_{21}^{(1)} x_1 + w_{22}^{(1)} x_2 + w_{23}^{(1)} x_3)$$

$$a_3^{(2)} = g(w_{30}^{(1)} x_0 + w_{31}^{(1)} x_1 + w_{32}^{(1)} x_2 + w_{33}^{(1)} x_3)$$

¿Cómo quedarían las expresiones para la tercera capa?

De la Neurona a una Red Neuronal

$\sum_{i=1}^n w_i x_i$	Función de Activación
------------------------	-----------------------------



$a_i^{(j)}$ = activación o valor de salida de la neurona i en la capa j

w^j = matriz de pesos de la capa j a la capa $j+1$

Esta parte se le conoce como **forward propagation**.

$$a_1^{(2)} = g(w_{10}^{(1)} x_0 + w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2 + w_{13}^{(1)} x_3)$$

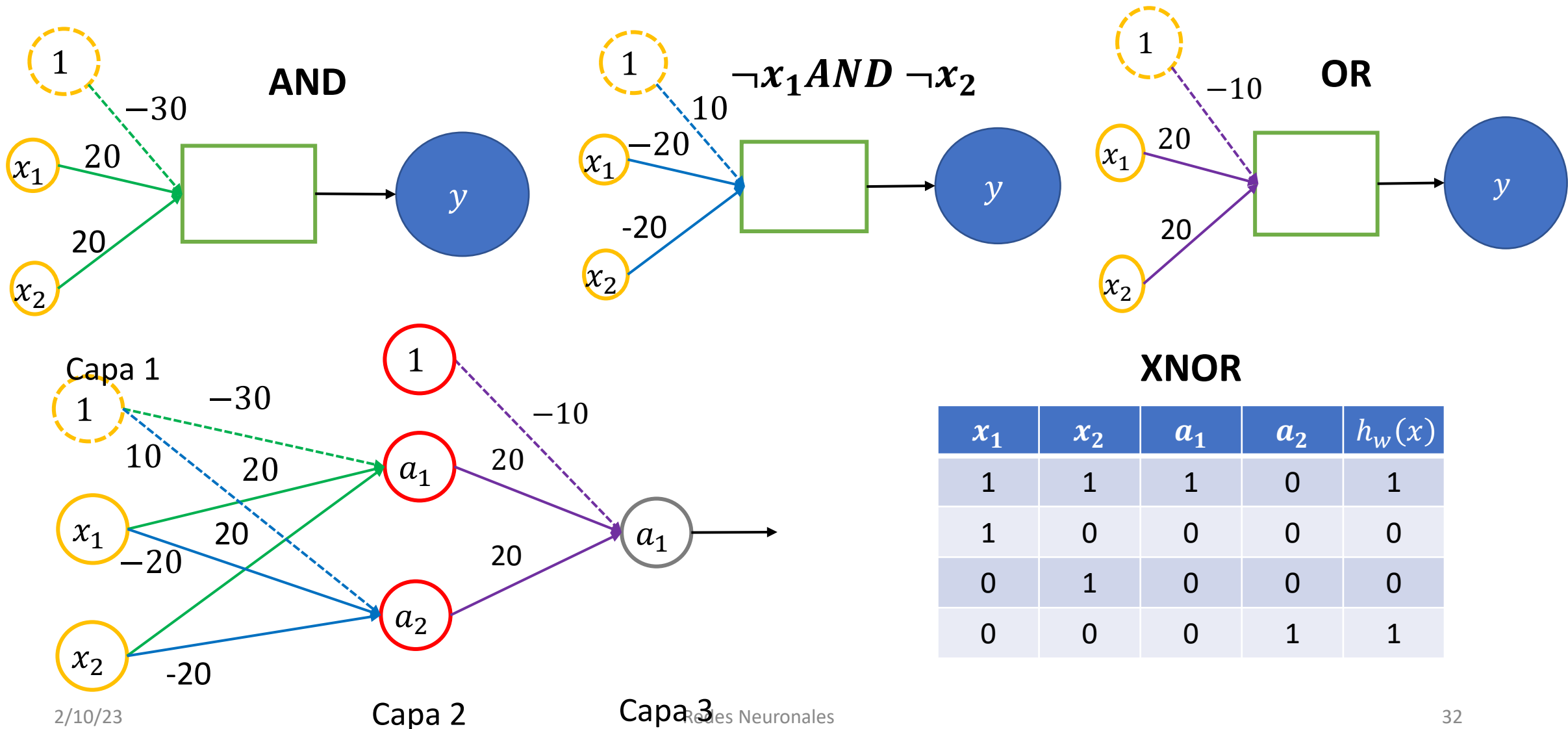
$$a_2^{(2)} = g(w_{20}^{(1)} x_0 + w_{21}^{(1)} x_1 + w_{22}^{(1)} x_2 + w_{23}^{(1)} x_3)$$

$$a_3^{(2)} = g(w_{30}^{(1)} x_0 + w_{31}^{(1)} x_1 + w_{32}^{(1)} x_2 + w_{33}^{(1)} x_3)$$

$$a_1^{(3)} = g(w_{10}^{(2)} a_0^{(2)} + w_{11}^{(2)} a_1^{(2)} + w_{12}^{(2)} a_2^{(2)} + w_{13}^{(2)} a_3^{(2)})$$

$$a_2^{(3)} = g(w_{20}^{(2)} a_0^{(2)} + w_{21}^{(2)} a_1^{(2)} + w_{22}^{(2)} a_2^{(2)} + w_{23}^{(2)} a_3^{(2)})$$

Red Neuronal: Segunda Intuición



Red Neuronal: Segunda Intuición

Tarea

- Verificar manualmente que la red anterior es una implementación de la función XNOR.

Redes Neuronales – Primeras Conclusiones

- En general, permiten calcular diferentes funciones.
- Conforme se van agregando capas, se pueden calcular funciones más y más complejas.
- En el caso del XNOR, se pasa de un margen de decisión lineal a uno no lineal.
- Un poco más del campo de la interpretación, cada capa permite extraer «ideas» más complejas y abstractas de las características x_i .
- Pueden manejar «fácilmente» muchas características.

Redes Neuronales Entran al Ruedo





Fin de la presentación

¡Gracias por su atención!