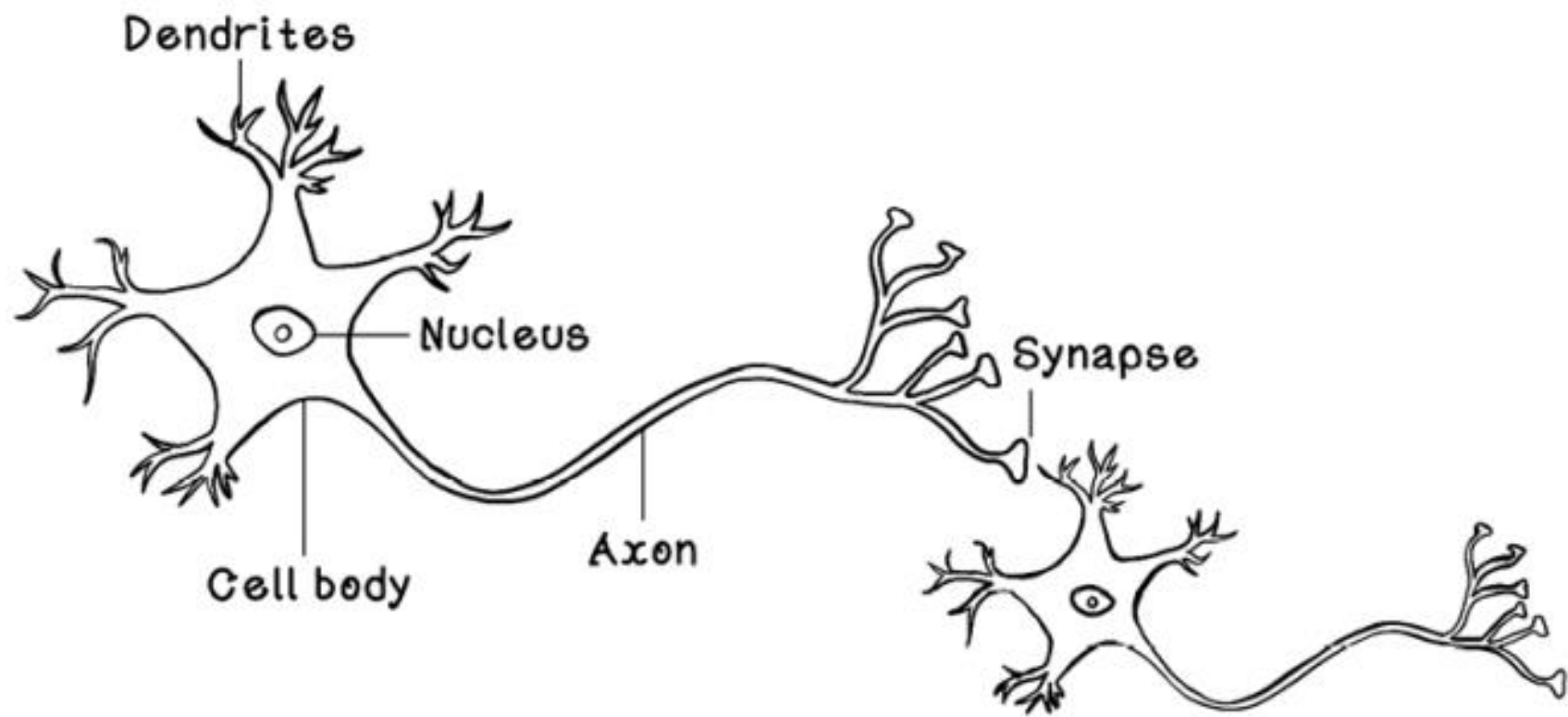
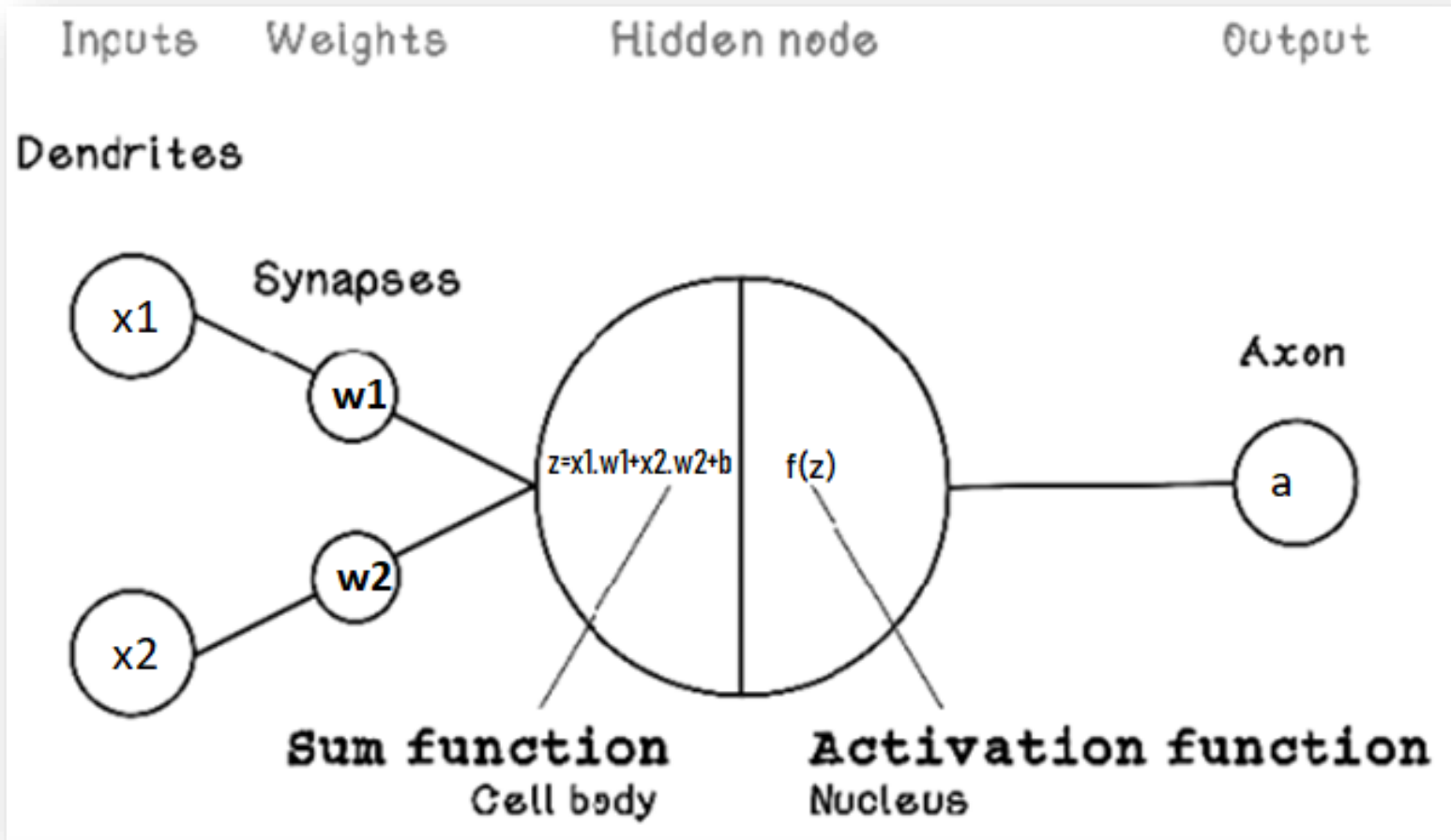


Redes Neuronales



El **Perceptrón**: una representación de una neurona

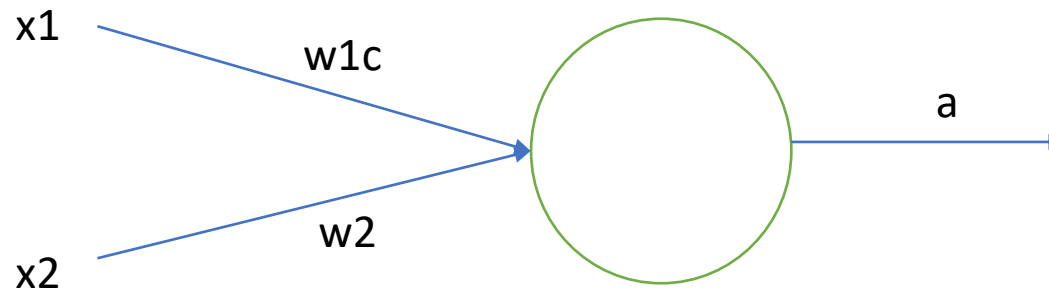
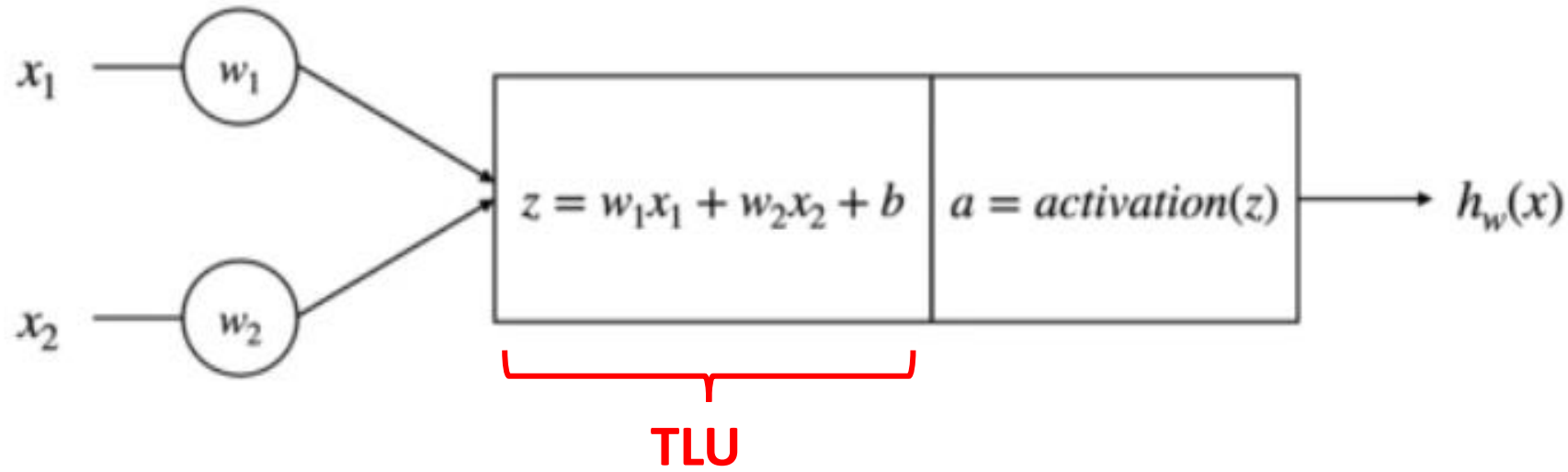


- Los componentes del Perceptrón se describen mediante variables que son útiles para calcular la salida. Pesos modifican las entradas; ese valor es procesado por un nodo oculto; y finalmente, el resultado se proporciona como salida.

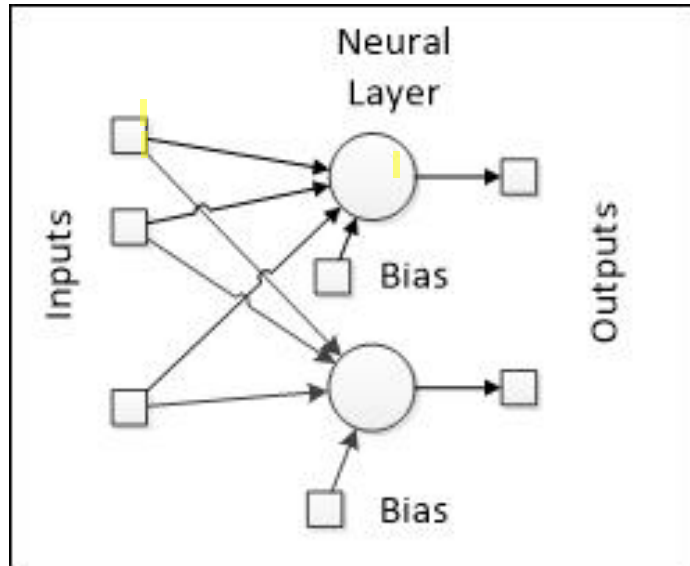
Componentes del Perceptron

- *Entradas* : describen los valores de entrada. En una neurona, estos valores serían una señal de entrada.
- *Pesos* : describa los pesos en cada conexión entre una entrada y el nodo oculto. Los pesos influyen en la intensidad de una entrada y dan como resultado una entrada ponderada. En una neurona, estas conexiones serían las sinapsis.
- *Nodo oculto (suma y activación)*: suma los valores de entrada ponderados y luego aplica una función de activación al resultado sumado. Una función de activación determina la activación/salida del nodo oculto/neurona.
- *Salida* : describe la salida final del Perceptrón.

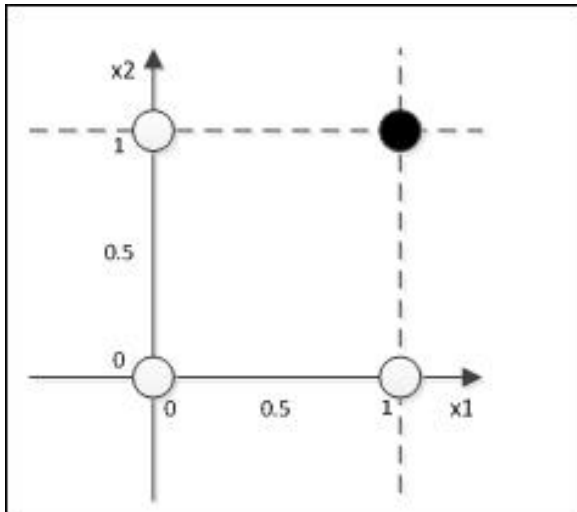
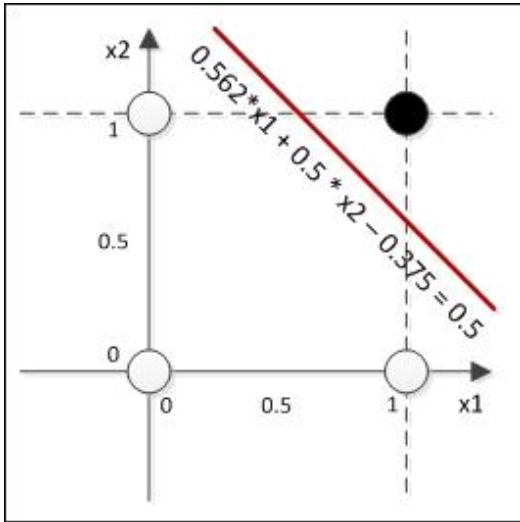
Threshold Logic Unit (TLU)



Arquitectura Red Neuronal Perceptrón



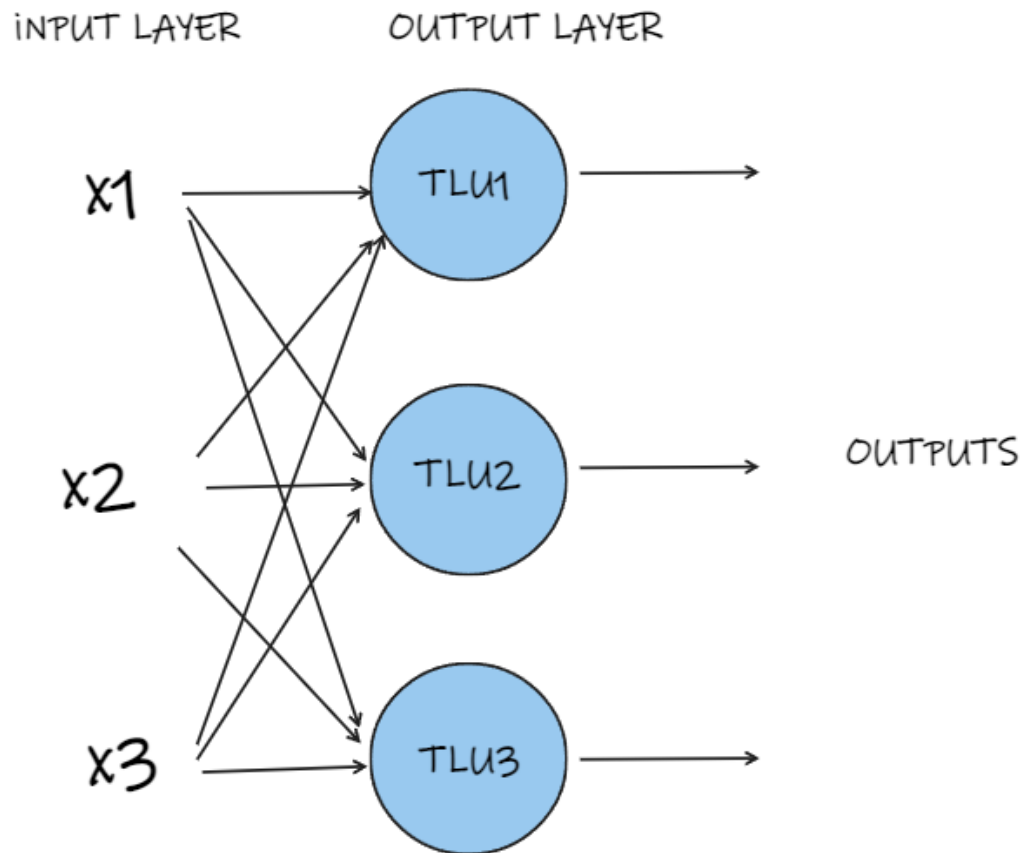
- Perceptrón es la arquitectura de red neuronal mas simple
- Fue propuesta por Frank Rosenblatt en 1957



Aplicaciones y Limitaciones | Perceptrón

- Solo es aplicable a tareas simples
- Se dieron cuenta que los perceptrones fallaban cuando analizaban datos complejos
- Analizar un perceptrón con el conector lógico AND
- El perceptrón busca resolver datos que son linealmente separables
- Más adelante se descubren que muchas limitaciones del perceptrón se resuelven añadiendo más capas TLUs, pareciendo las Multi Layer Perceptron

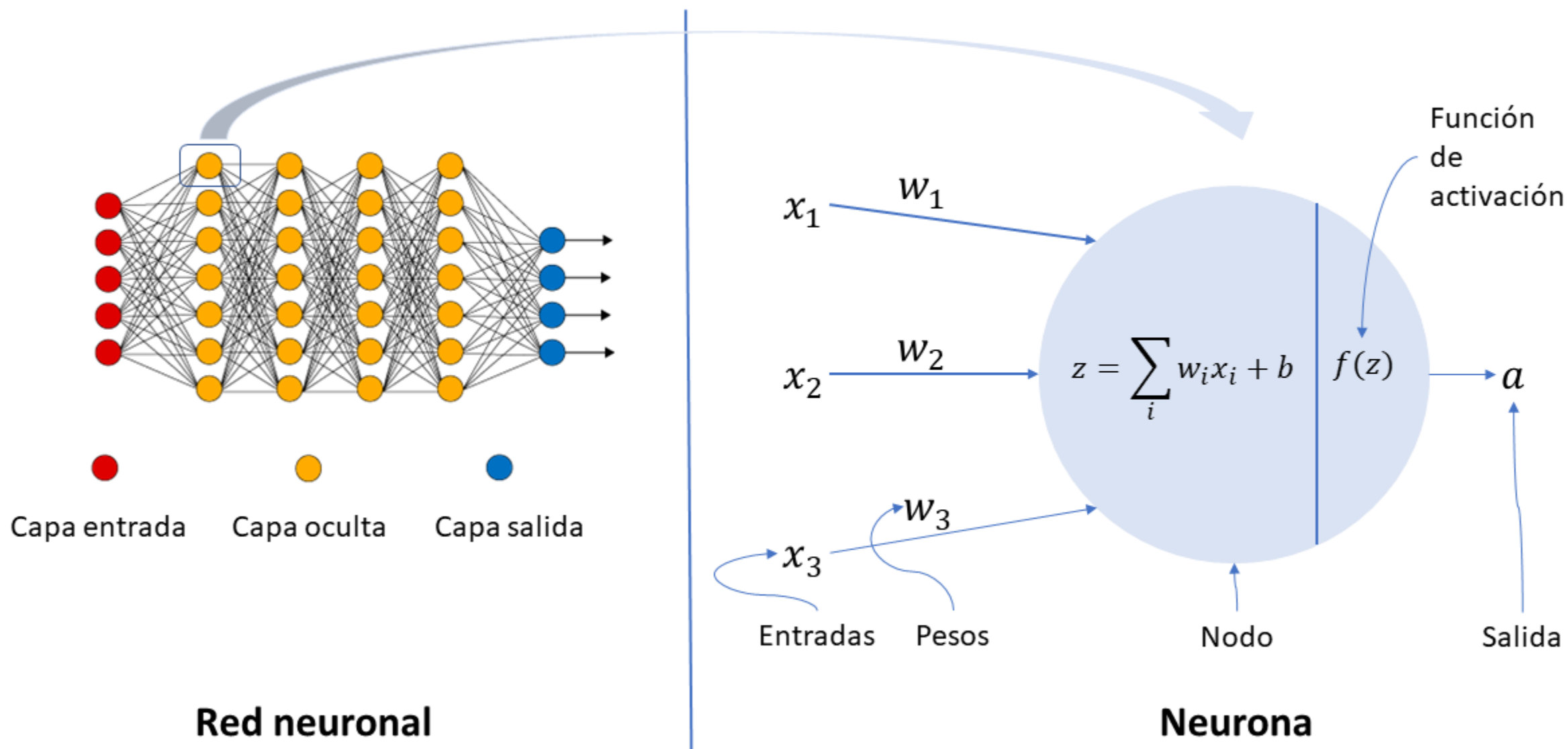
Perceptrón



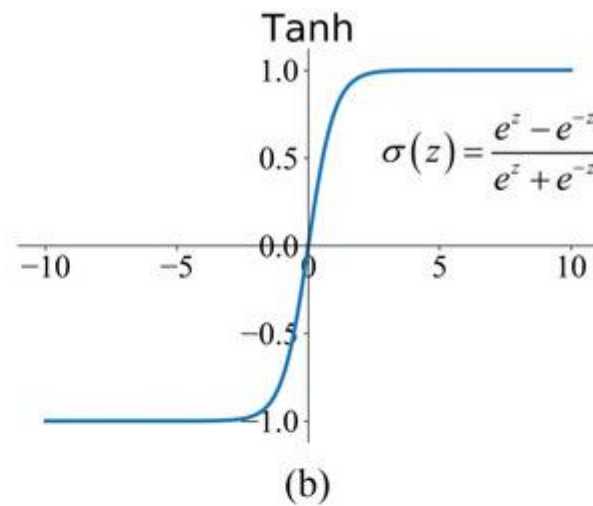
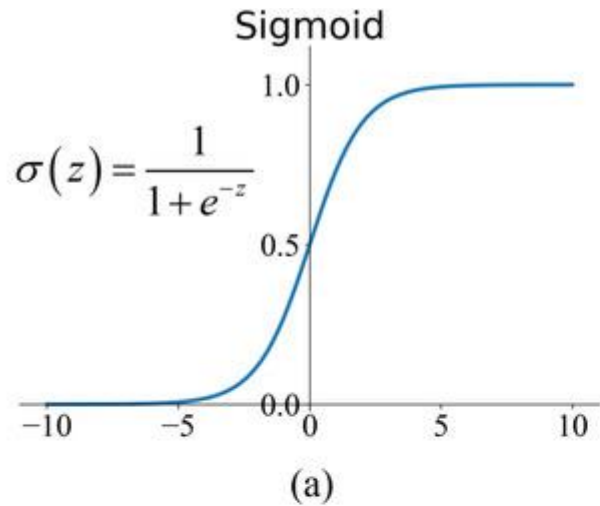
$$\text{TLU1} = Hw(x)1 = a1(w1 x1 + w2x1 + b1)$$

$$\text{TLU1} = Hw(x)2 = a2(w2 x1 + w2x2 + b2)$$

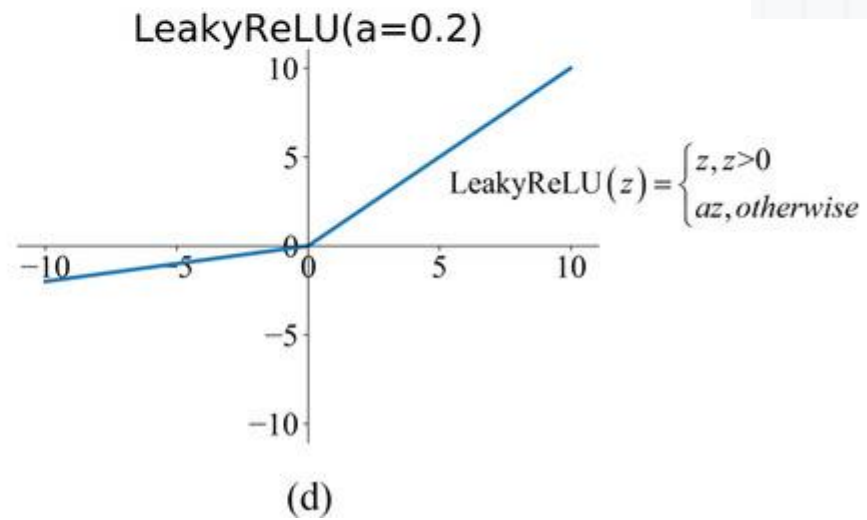
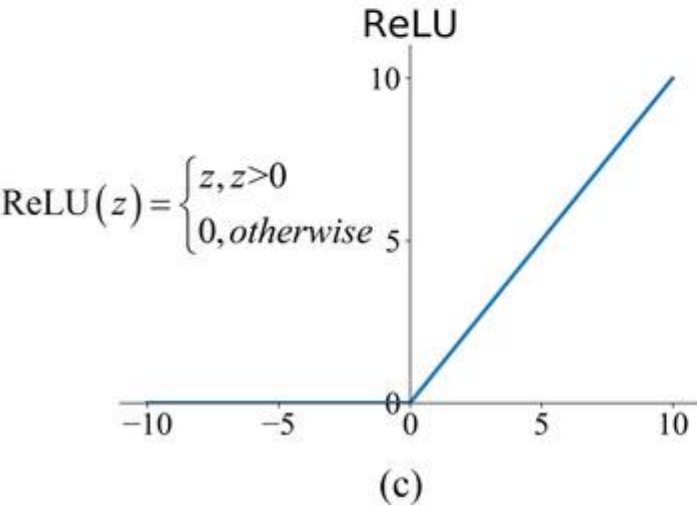
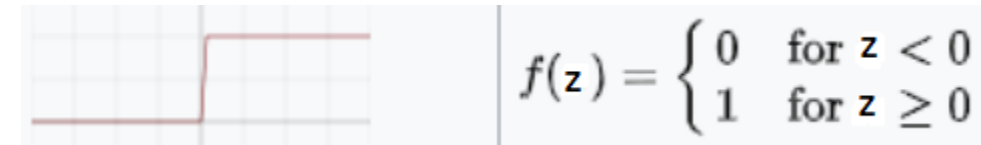
$$\text{TLU1} = Hw(x)3 = a3(w3 x1 + w3x2 + b3)$$



Funciones de Activación

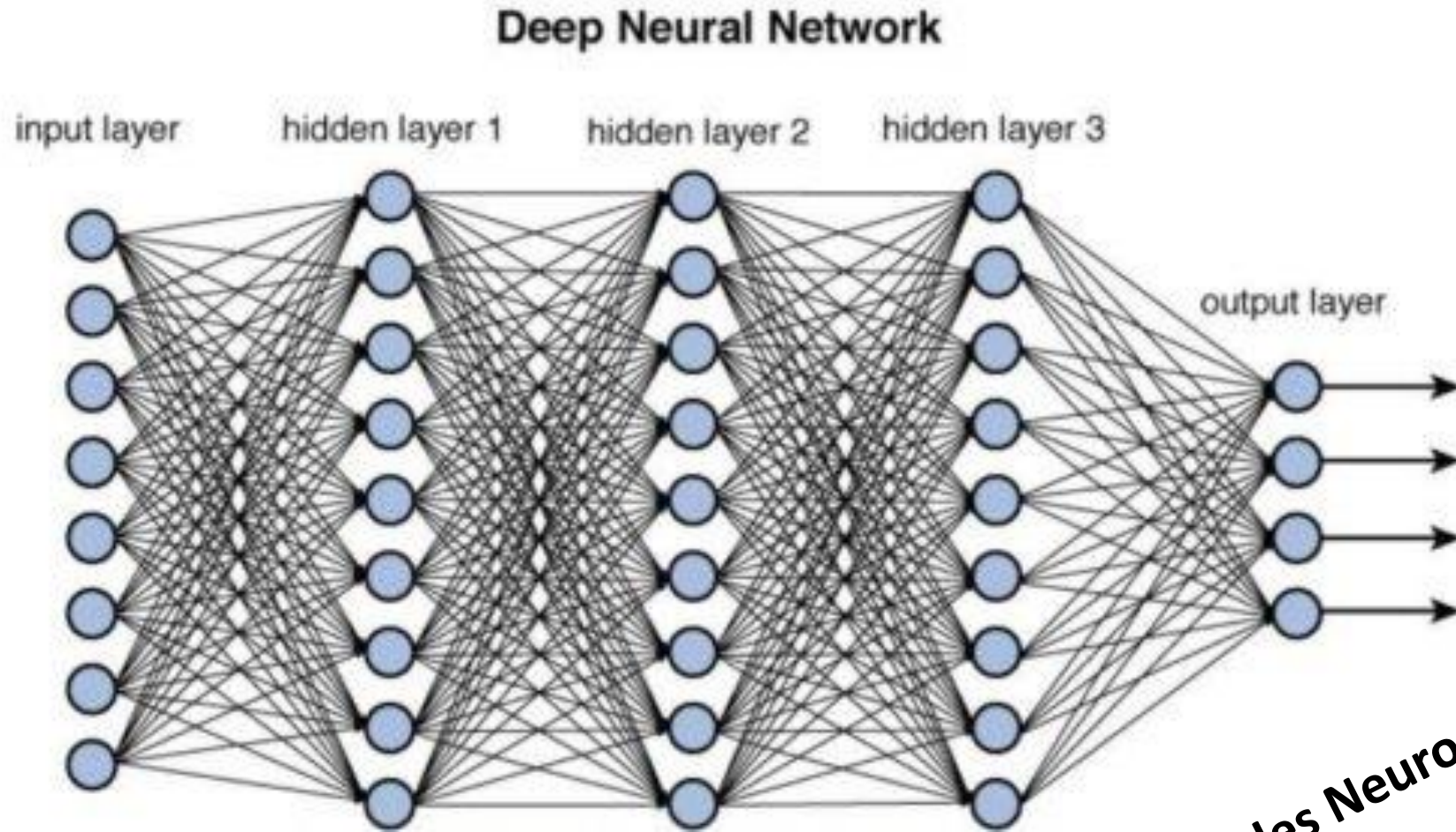


Binary Step-RED



Donde “z” representa el resultado de la suma de las señales de las entradas a la neurona $\Rightarrow z = (x_1w_1 + x_2w_2 + \dots)$

Deep Learning



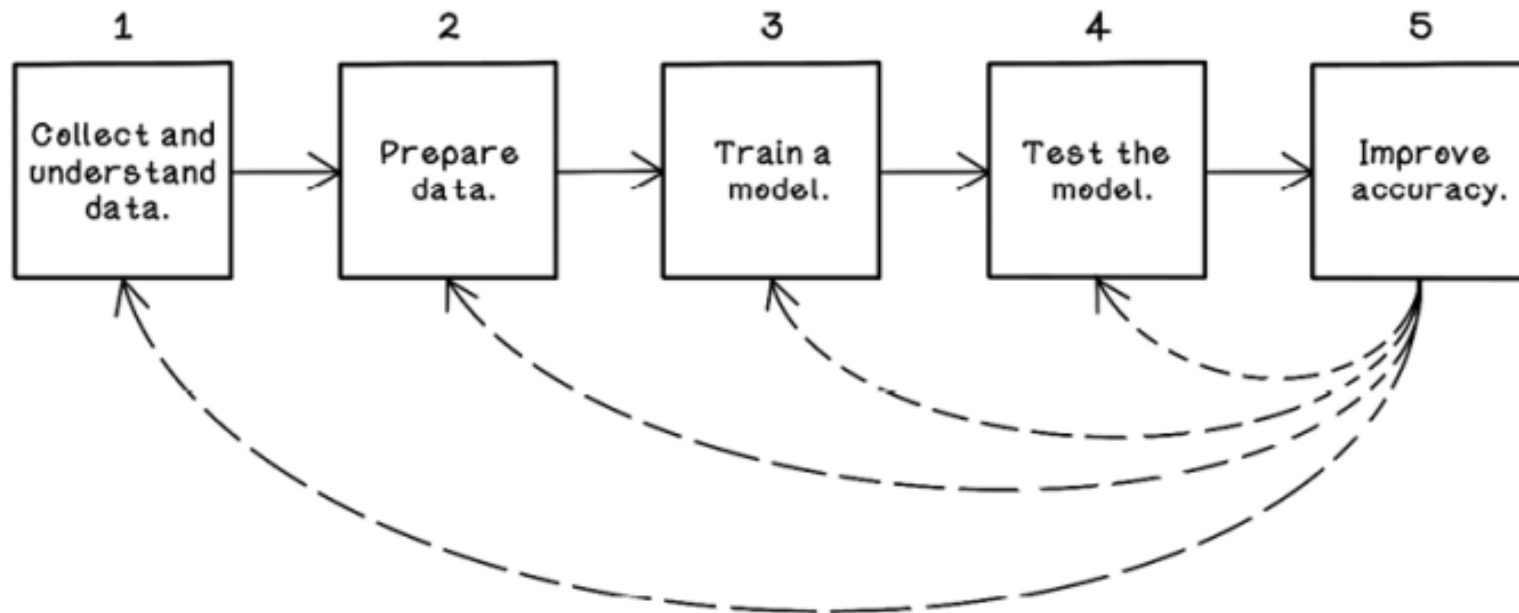
Redes Neuronales Artificiales

ANN: Artificial Network Neuronal

- *Las redes neuronales artificiales (ANN)* son herramientas poderosas en el conjunto de herramientas de aprendizaje automático, que se utilizan de diversas maneras para lograr objetivos como el reconocimiento de imágenes, el procesamiento del lenguaje natural y los juegos.
- Las ANN aprenden de manera similar a otros algoritmos de aprendizaje automático: mediante el uso de datos de entrenamiento. Se adaptan mejor a los datos no estructurados donde es difícil entender cómo se relacionan las características entre sí.

Ciclo

- El aprendizaje profundo, incluidas las ANN, se puede utilizar para resolver problemas de aprendizaje supervisado, aprendizaje no supervisado y aprendizaje por refuerzo.



REDES NEURONALES ARTIFICIALES

- Corresponde al componente principal del Deep Learning
- Se introducen por primera vez en 1943 por el neuropsicólogo Warren McCulloch y el matemático Walter Pitts
- En los 90 pierden interés a favor de Machine Learning como Support Vector Machine
- En 2012 Deep Learning surge con fuerza junto con Geoffrey Hinton que gana el reto de ImageNet con una red convolucional.

Inputs

Weights

Hidden node

Output

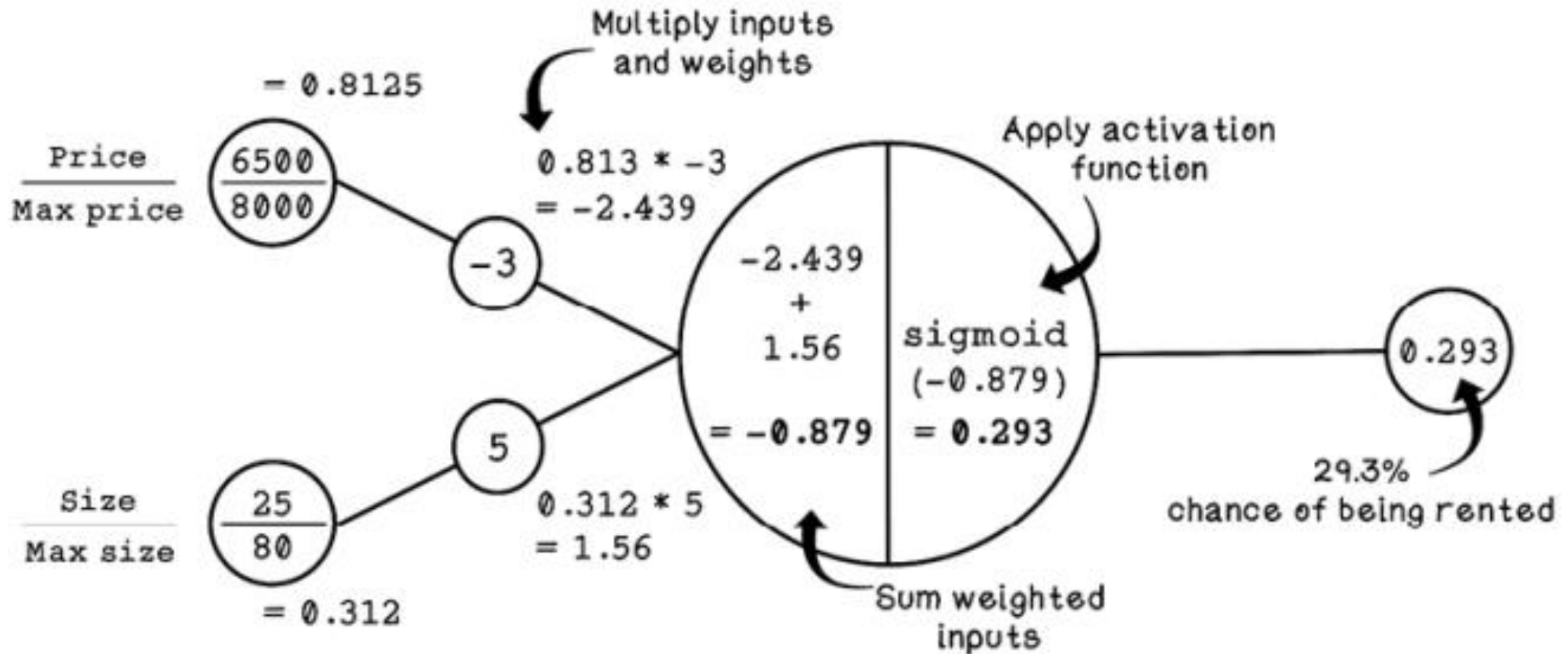
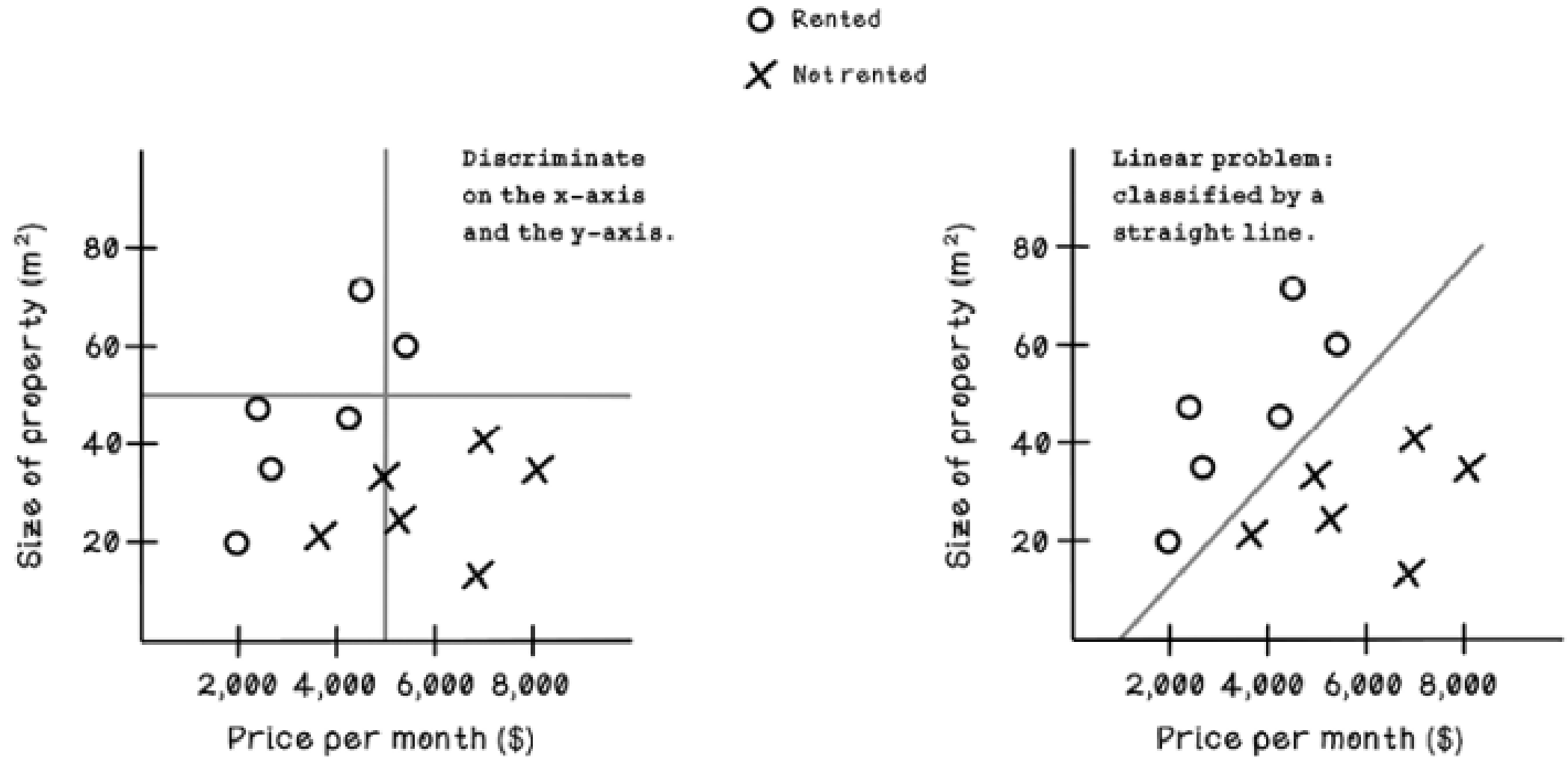
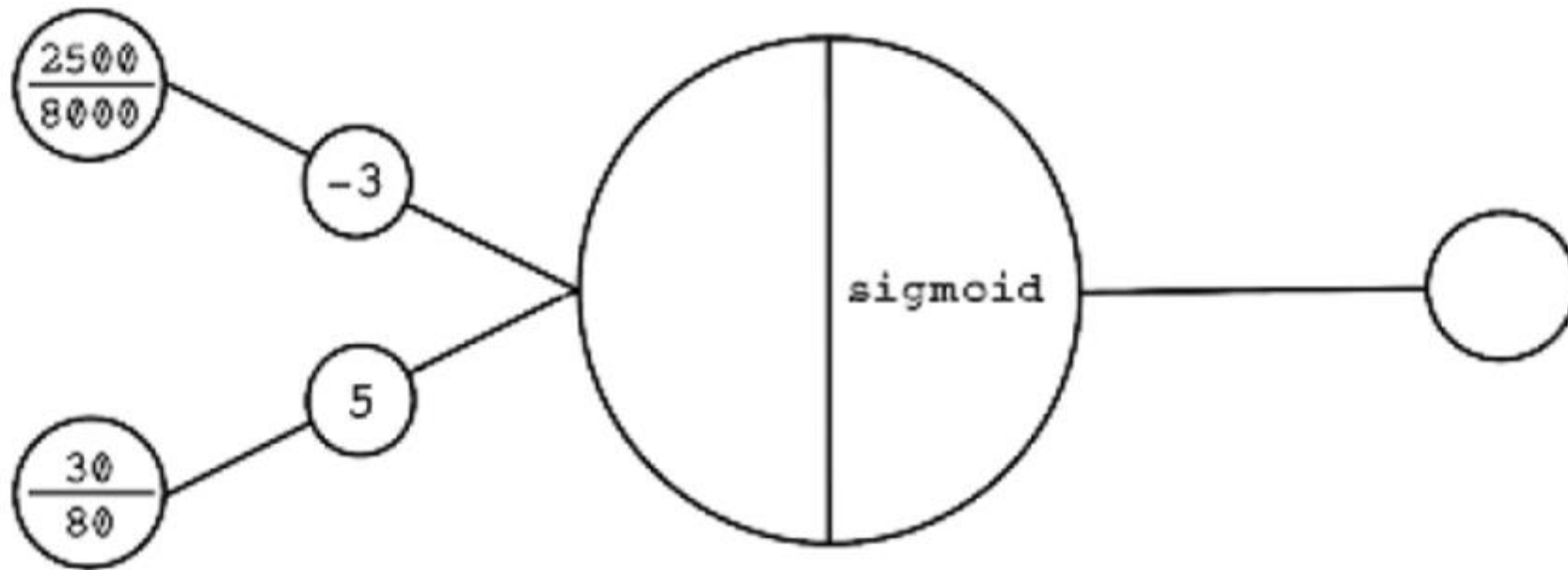


Figura 9.8 Ejemplo de un problema de clasificación lineal

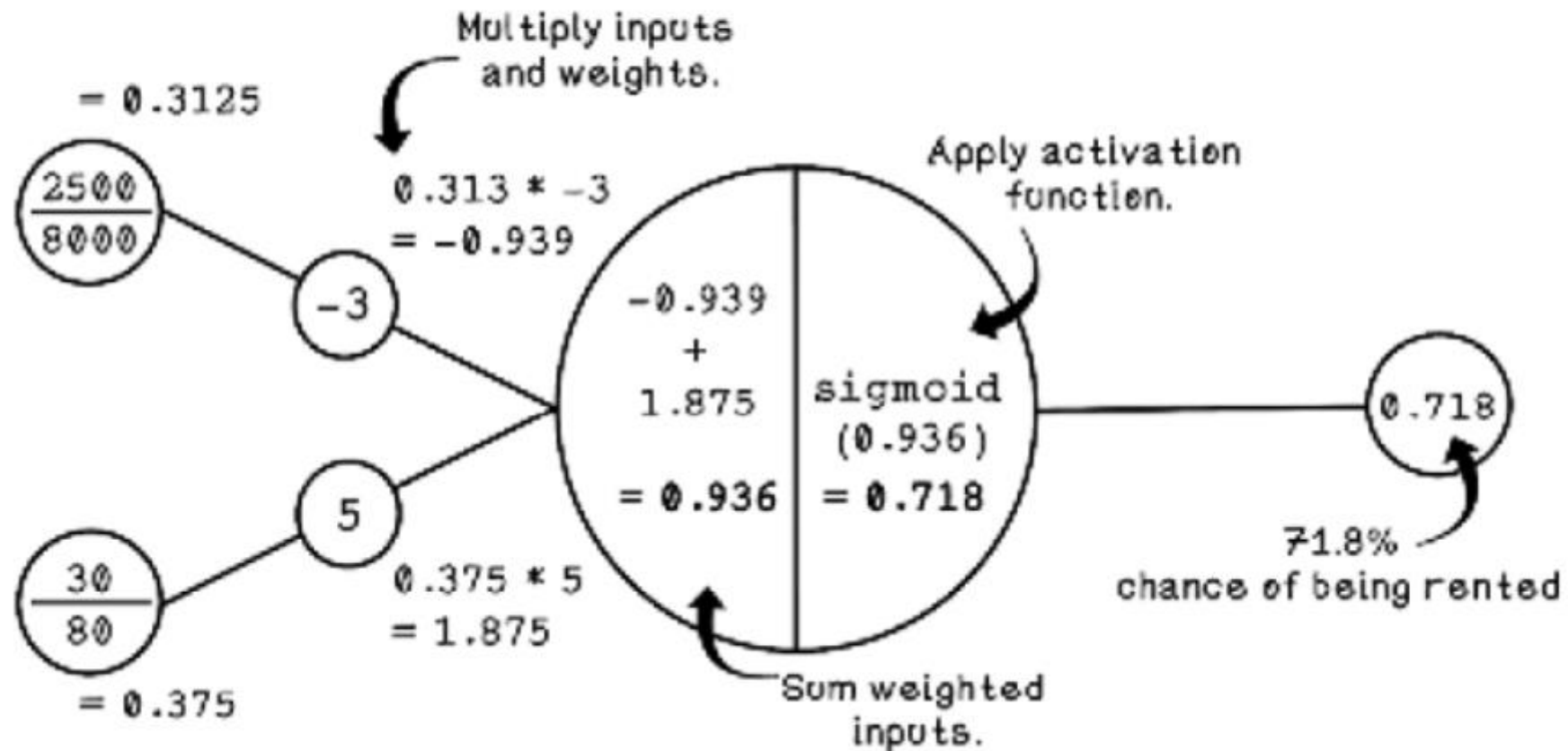


EJERCICIO: CALCULAR LA SALIDA DE LA SIGUIENTE ENTRADA PARA EL PERCEPTRON

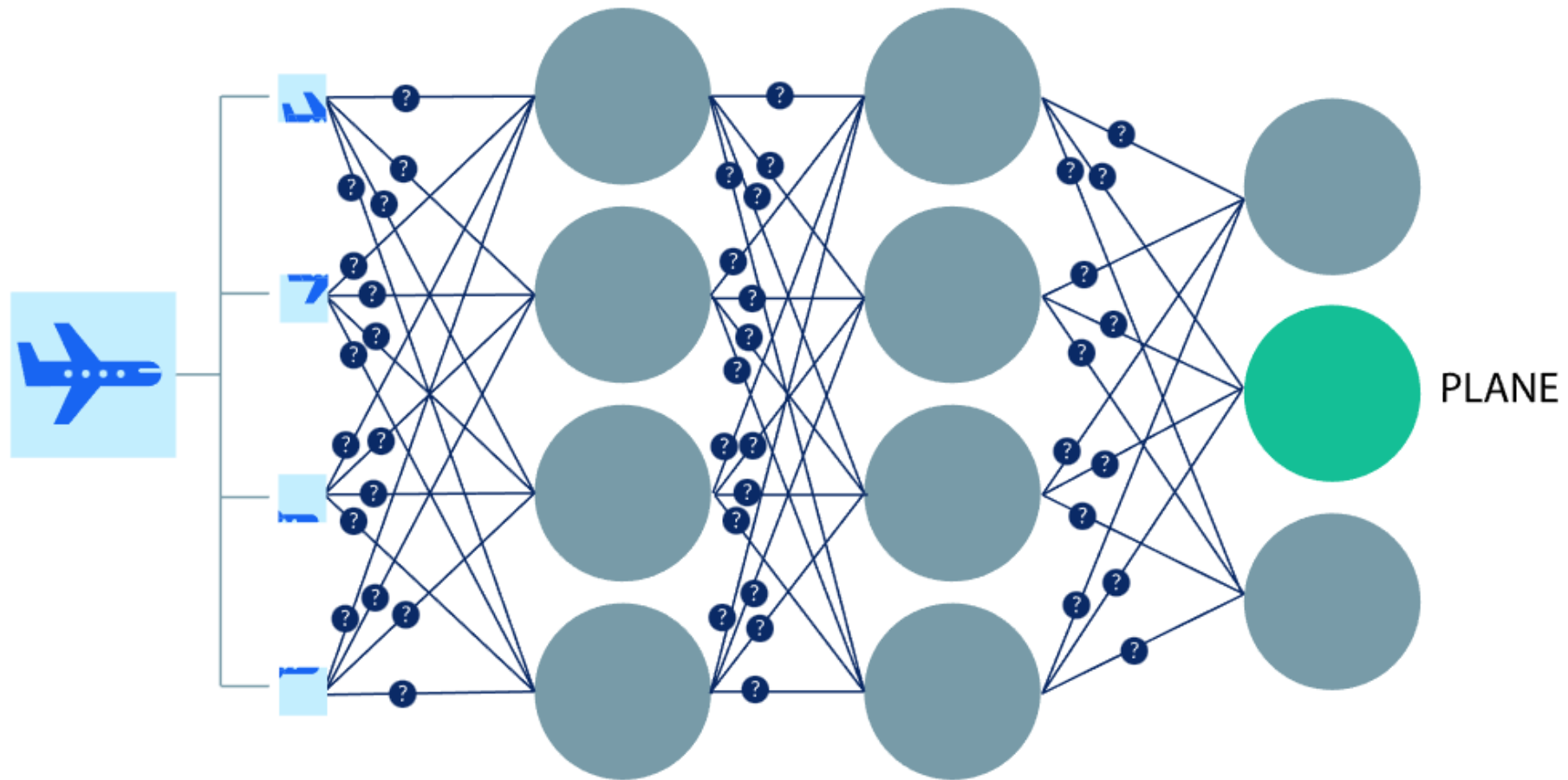
Usando su conocimiento de cómo funciona el Perceptrón, calcule la salida para el siguiente:



SOLUCIÓN: CALCULAR LA SALIDA DE LA SIGUIENTE ENTRADA PARA EL PERCEPTRON



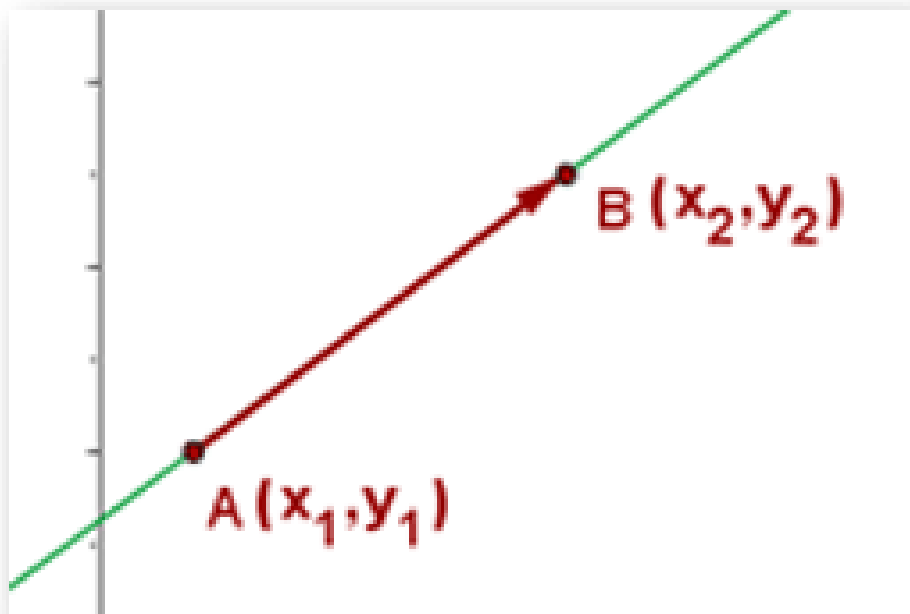
El objetivo de la fase de entrenamiento es determinar los pesos para las conexiones entre neuronas, que clasificarán correctamente los datos de entrenamiento.



La red neuronal comienza con todos los pesos inicializados a valores aleatorios, por lo que sus clasificaciones iniciales están muy lejos de ser las correctas. Sin embargo, aprende de sus errores, y finalmente encuentra un conjunto de pesos que realizan el mejor trabajo posible para clasificar todos los datos de entrenamiento.



TIP



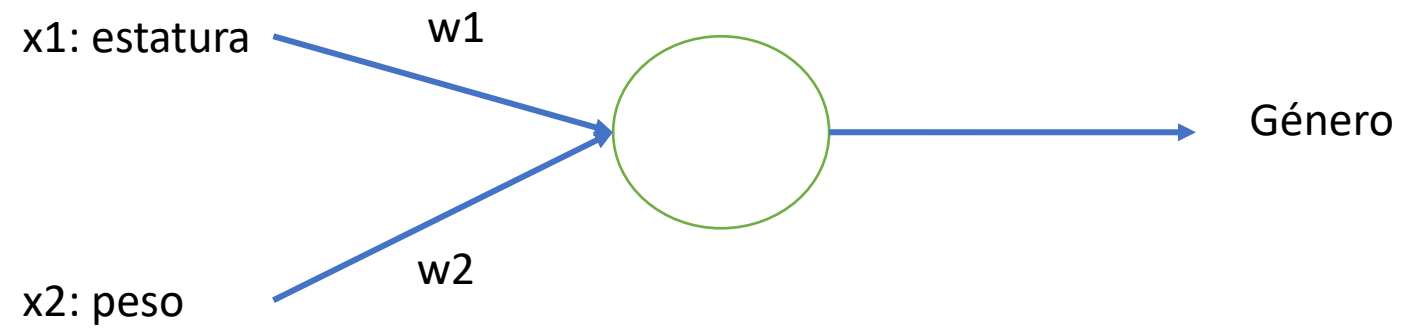
Recta entre 2 puntos

$$\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1}$$

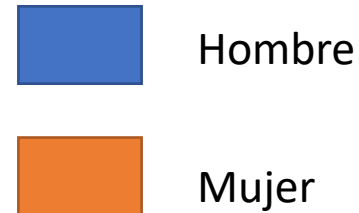
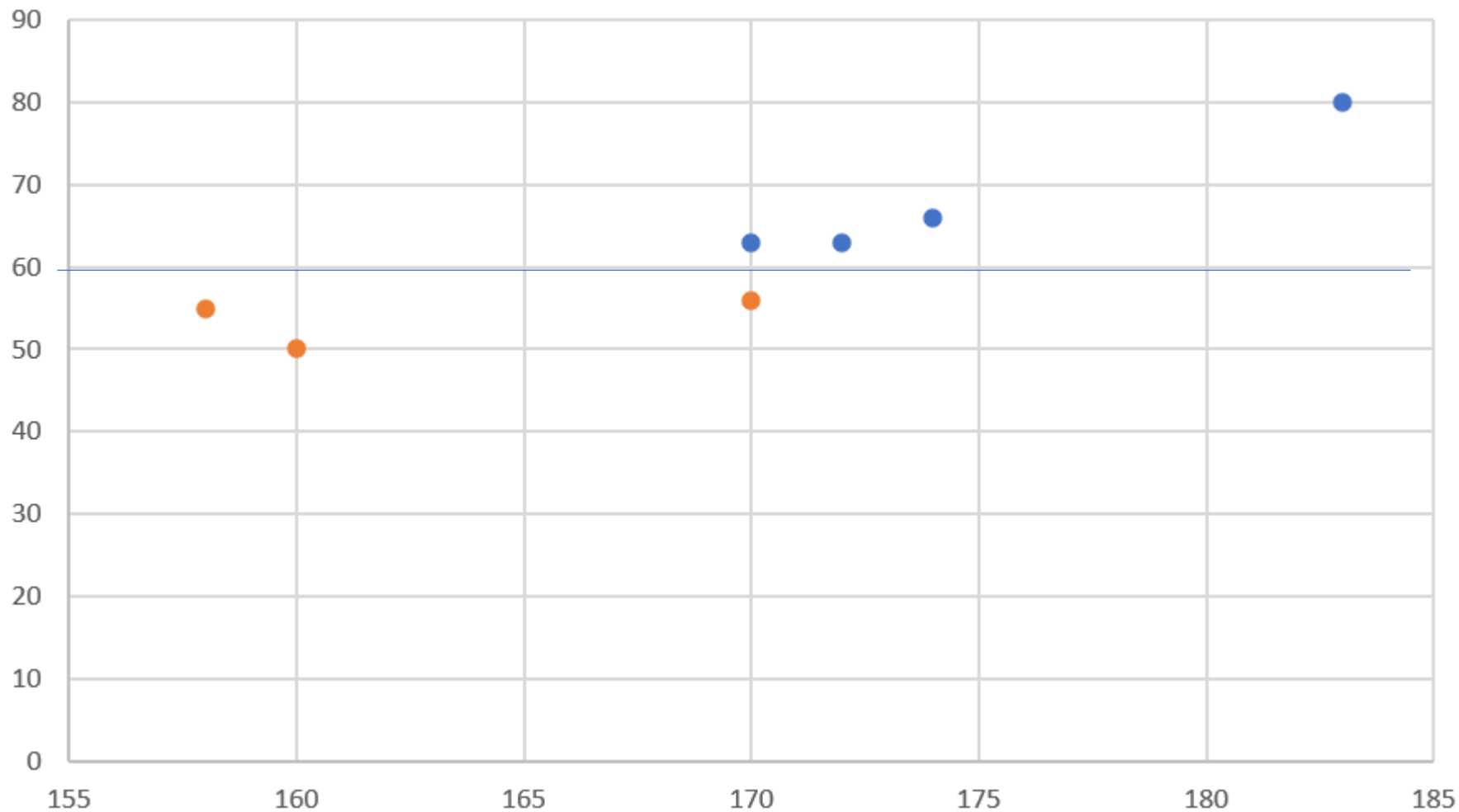
Ejercicio1:

<u>Estatura</u>	<u>Peso</u>	<u>Género</u>
<u>170</u>	<u>56</u>	<u>Mujer</u>
<u>172</u>	<u>63</u>	<u>Hombre</u>
<u>160</u>	<u>50</u>	<u>Mujer</u>
<u>170</u>	<u>63</u>	<u>Hombre</u>
<u>174</u>	<u>66</u>	<u>Hombre</u>
<u>158</u>	<u>55</u>	<u>Mujer</u>
<u>183</u>	<u>80</u>	<u>Hombre</u>
<u>182</u>	<u>70</u>	<u>Hombre</u>
<u>165</u>	<u>54</u>	<u>Mujer</u>

- Diseñar un sistema basado en redes neuronales de tipo perceptrón, para clasificar hombres y mujeres según su peso y su estatura. Utilice los siguientes datos para el desarrollo.



Peso



Estatura	Peso	Género
170	56	Mujer
172	63	Hombre
160	50	Mujer
170	63	Hombre
174	66	Hombre
158	55	Mujer
183	80	Hombre
182	70	Hombre
165	54	Mujer

Estatura

$$\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1}$$

$$0x + y - 60 = 0$$

Ejercicio2

Queremos construir un sistema automático que coloque las manzanas en el depósito verde y las piñas en el depósito amarillo. Usamos dos sensores para determinar el fruto, una galga extensiométrica para medir el peso P1 del fruto y una cámara para determinar el color promedio P2 del fruto. Diseñe una red neuronal perceptrón que realice la clasificación entre piñas y manzanas. Considere que los sensores entregan los siguientes niveles de voltaje par las siguientes 3 piñas típicas:

$$\mathbf{p}_1 = \begin{bmatrix} 1.5 \\ -0.3 \end{bmatrix} \quad \mathbf{p}_2 = \begin{bmatrix} 0.9 \\ 0.05 \end{bmatrix} \quad \mathbf{p}_3 = \begin{bmatrix} 2.1 \\ 0.2 \end{bmatrix}$$

Y para las siguientes 3 manzanas típicas:

$$\mathbf{p}_4 = \begin{bmatrix} 0.24 \\ -0.87 \end{bmatrix} \quad \mathbf{p}_5 = \begin{bmatrix} 0.45 \\ -0.60 \end{bmatrix} \quad \mathbf{p}_6 = \begin{bmatrix} 0.15 \\ -0.43 \end{bmatrix}$$

El primer valor de cada vector hace referencia al peso, por ej. en P1 el **peso** de la piña típica es de 1.5. El segundo valor representa el **color** promedio. Por ej. En P1 -0.3 es el **color** promedio de la piña.

Diseñar una red neuronal perceptrón que decida cuando el fruto es piña y cuando es manzana.

Piñas

$$p_1 = \begin{bmatrix} 1.5 \\ -0.3 \end{bmatrix}$$

$$p_2 = \begin{bmatrix} 0.9 \\ 0.05 \end{bmatrix}$$

$$p_3 = \begin{bmatrix} 2.1 \\ 0.2 \end{bmatrix}$$

Manzanas

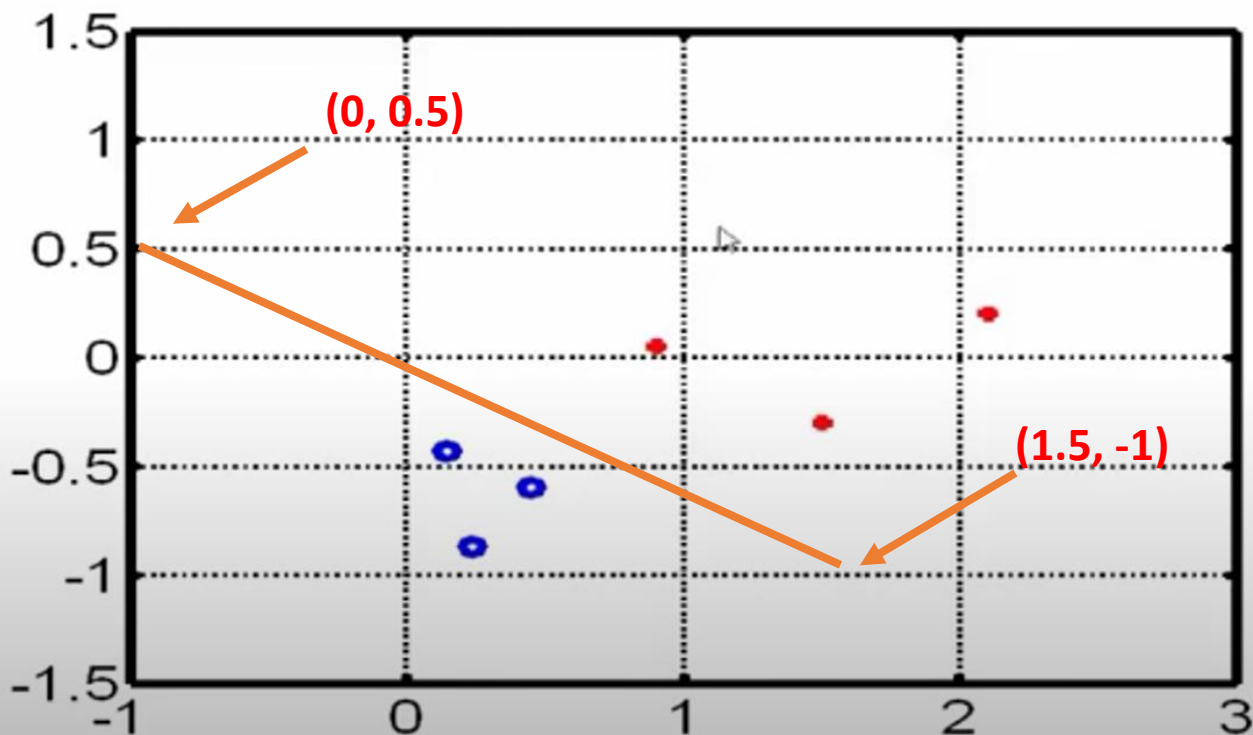
$$p_4 = \begin{bmatrix} 0.24 \\ -0.87 \end{bmatrix}$$

$p_5 = \begin{bmatrix} 0.45 \\ -0.60 \end{bmatrix}$

$$p_6 = \begin{bmatrix} 0.15 \\ -0.43 \end{bmatrix}$$

0

1



Peso	Color	Fruta
1	0.3	Piña
0.9	0.005	Piña
2.1	0.2	Piña
0.24	-0.87	Manzana
0.45	-0.6	Manzana
0.15	0.43	Manzana