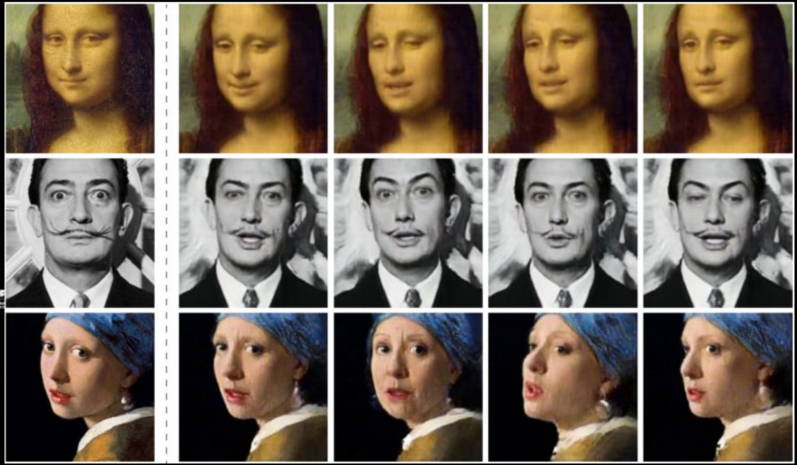
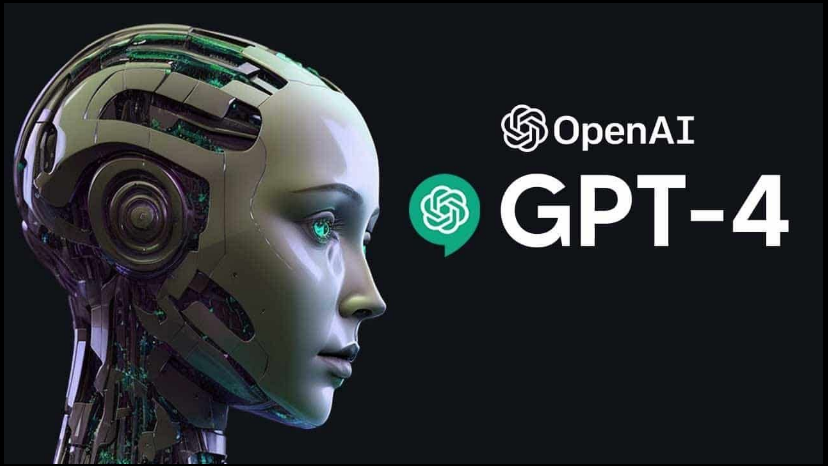
Redes Neuronales

Sistemas de Inteligencia Artificial

Primer Cuatrimestre 2023

Rodrigo Ramele
Eugenia Piñeiro
Alan Pierri
Santiago Reyes
Marina Fuster
Luciano Bianchi



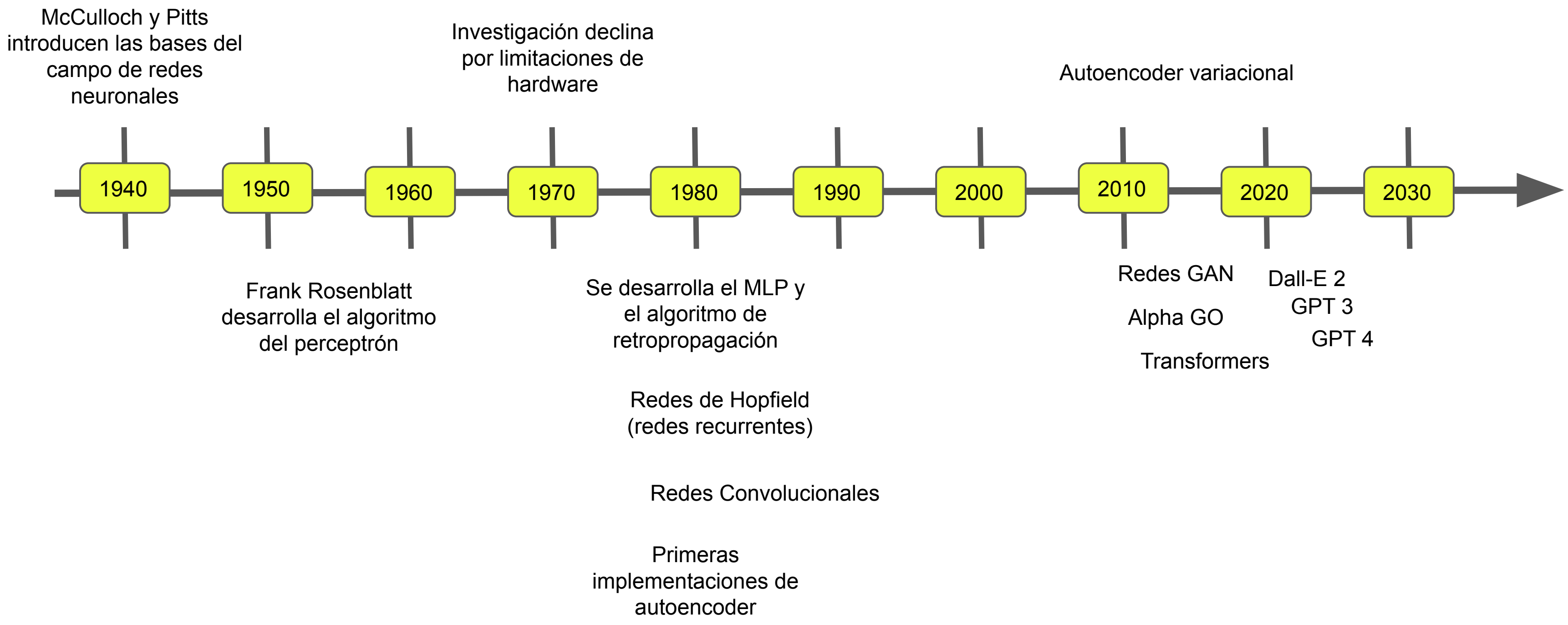


AlphaGo: https://www.youtube.com/watch?v=WXuK6gekU1Y
ChatGPT4: https://www.youtube.com/watch?v=outcGtbnMuQ&t=449s
OpenAI: https://openai.com/https://openai.com/

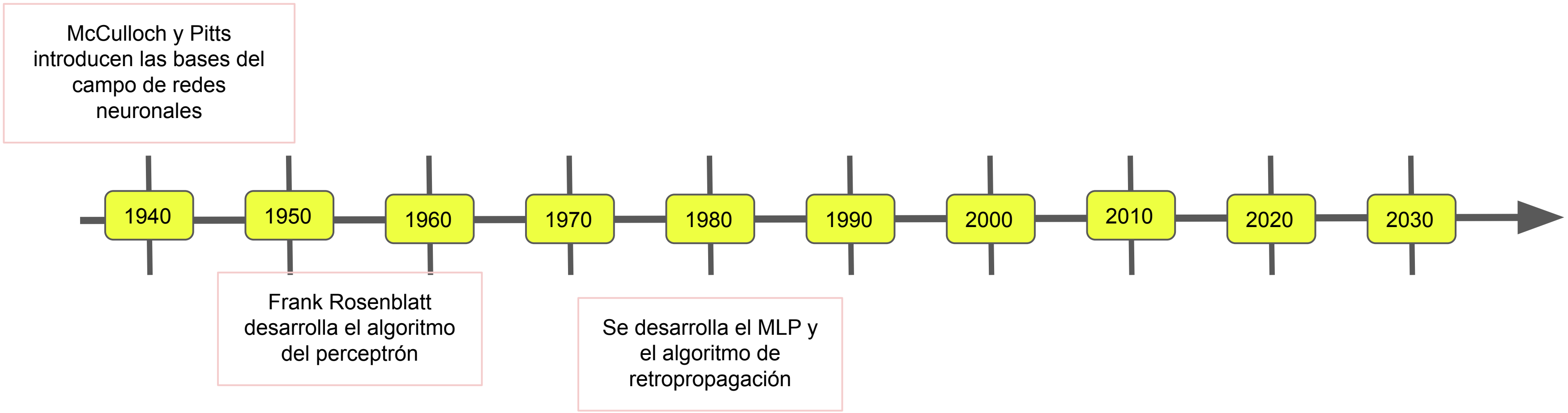
“Una red neuronal es un tipo de algoritmo de aprendizaje automático que está modelado según la estructura y función del cerebro humano. Consiste en una gran cantidad de nodos de procesamiento interconectados (neuronas) que trabajan juntos para procesar y analizar datos complejos. Cada neurona recibe uno o más inputs, los procesa utilizando un conjunto de pesos aprendidos y produce una salida que se transmite a otras neuronas en la red.”

ChatGPT

EVENTOS EN EL ÁREA DE REDES NEURONALES



¿QUÉ IDEAS ESTUDIAMOS DURANTE EL TP3?



- Por el momento no vamos a profundizar el área de "*Deep Learning*" (se ve más adelante en la materia)
- Objetivo: entender qué problemas se pueden resolver, los conceptos básicos que componen a una red neuronal y su entrenamiento

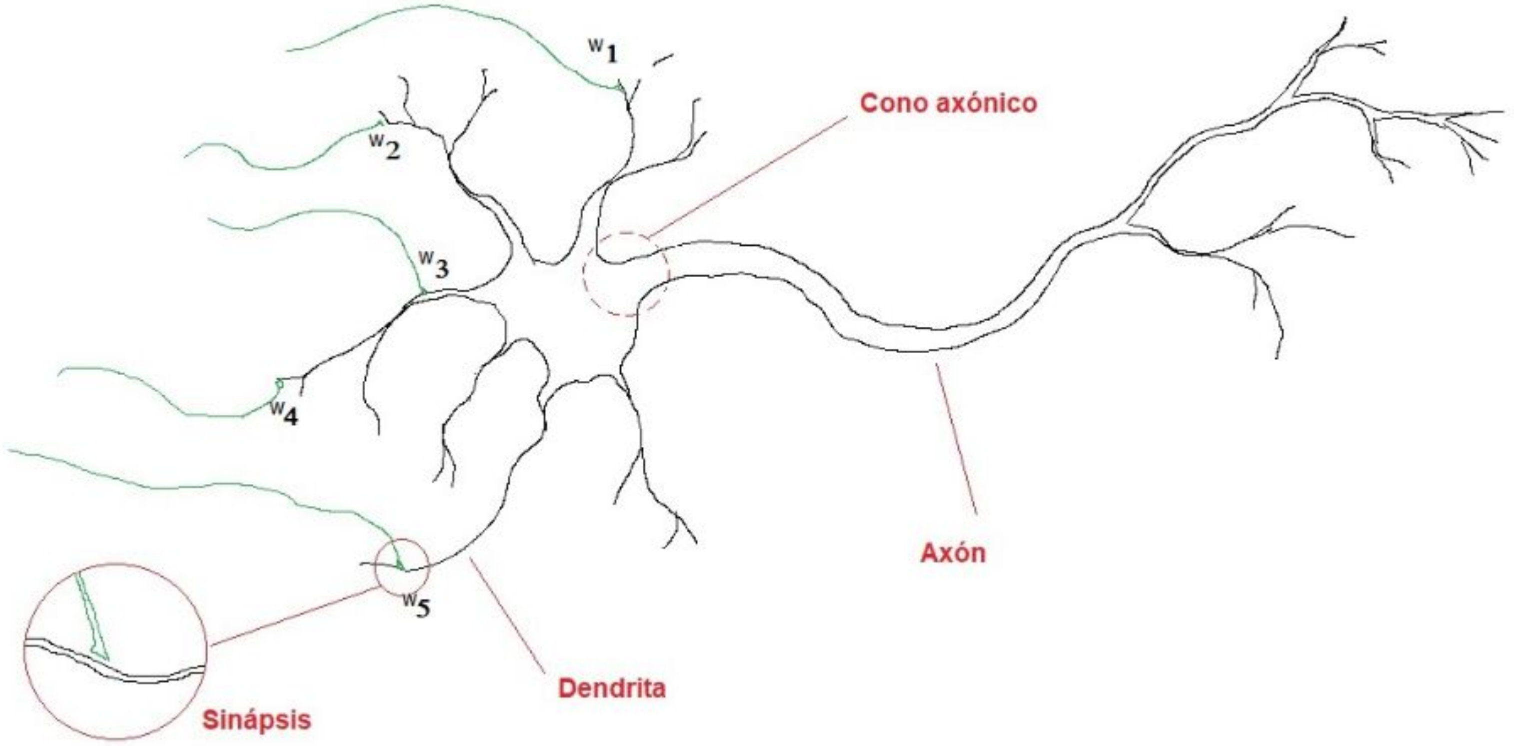
Perceptrón Simple

Sistemas de Inteligencia Artificial

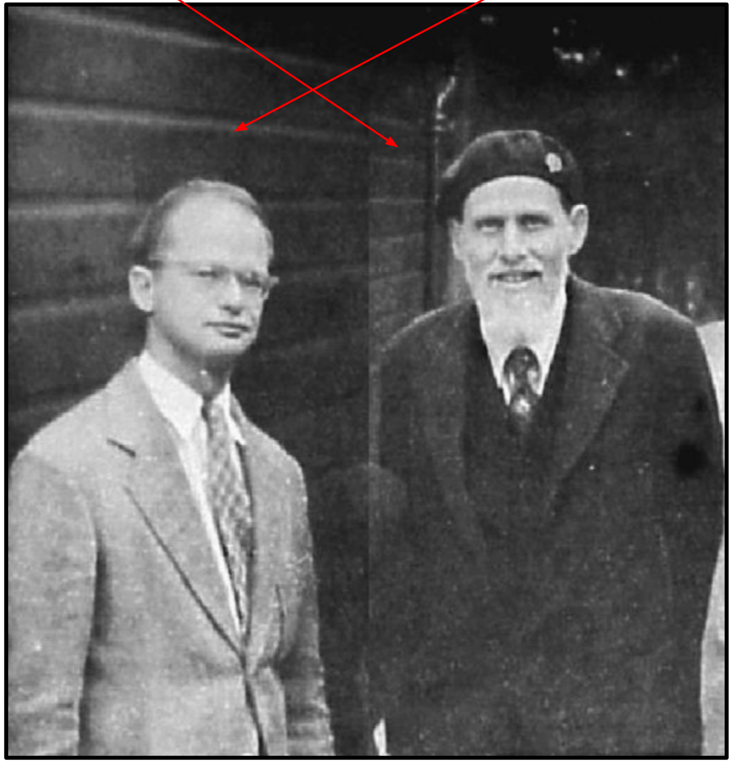
Primer Cuatrimestre 2023

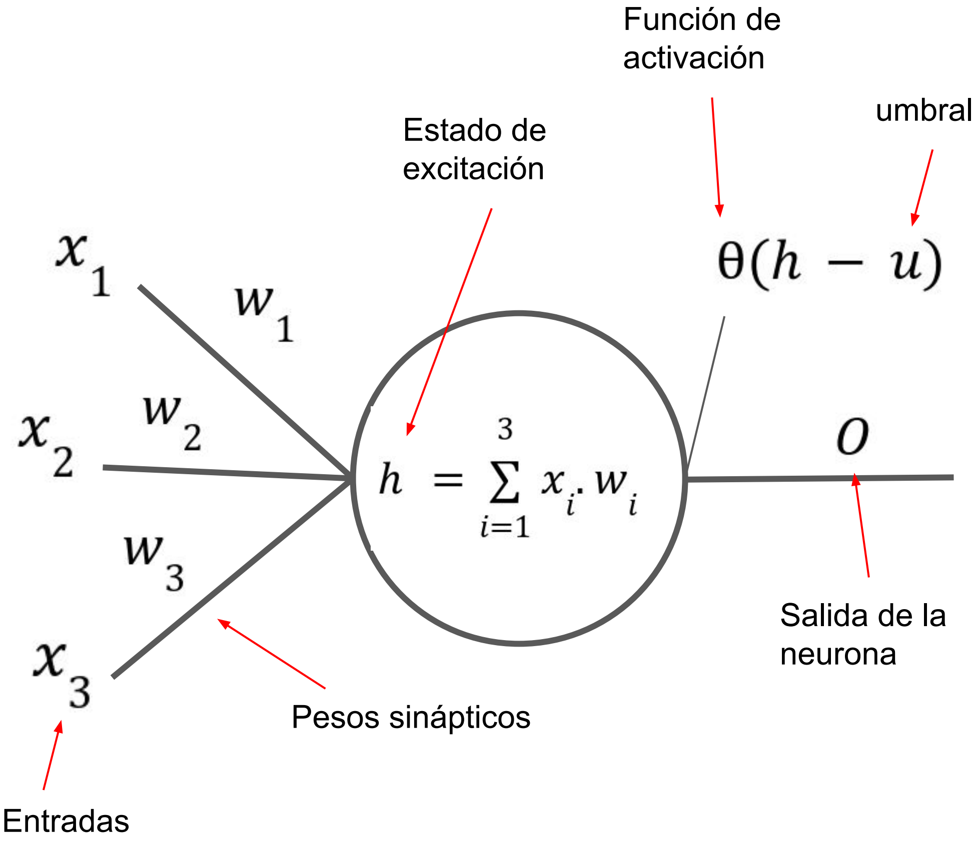
Rodrigo Ramele
Eugenia Piñeiro
Alan Pierri
Santiago Reyes
Marina Fuster
Luciano Bianchi

MODELO DE NEURONA - 1943



Warren McCulloch & Walter Pitts 1943





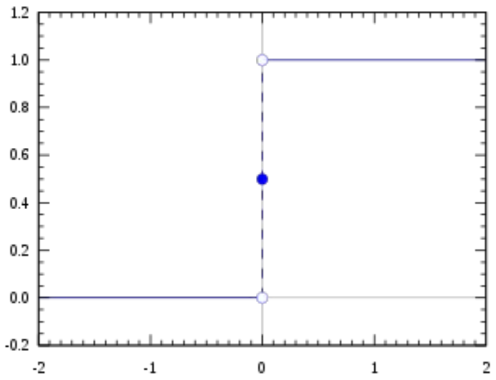
Paper: https://link.springer.com/article/10.1007/BF02478259

FORMALIZACIÓN MATEMÁTICA

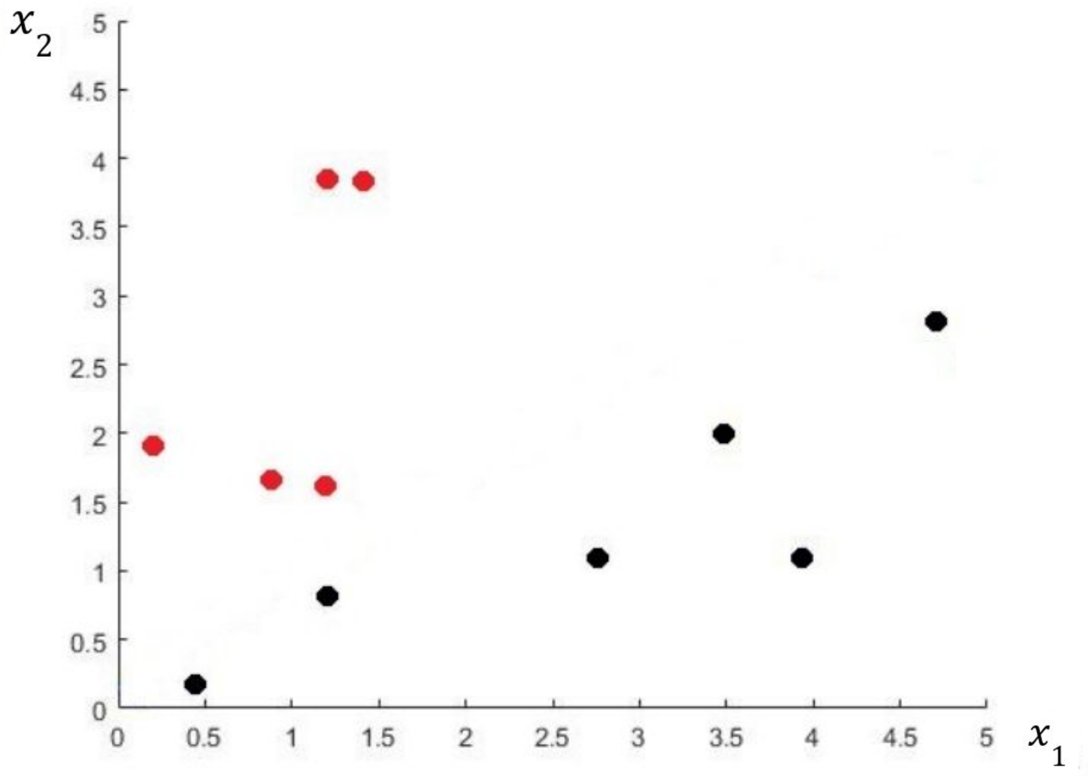
$$O = \theta\left(\sum_{i=1}^{n} x_i . w_i - u\right)$$

$$\theta(x) = \begin{cases} 1 & x \geq 0 \\ 0 & en\ otro\ caso \end{cases}$$

Función de activación ***escalón*** o ***signo***.

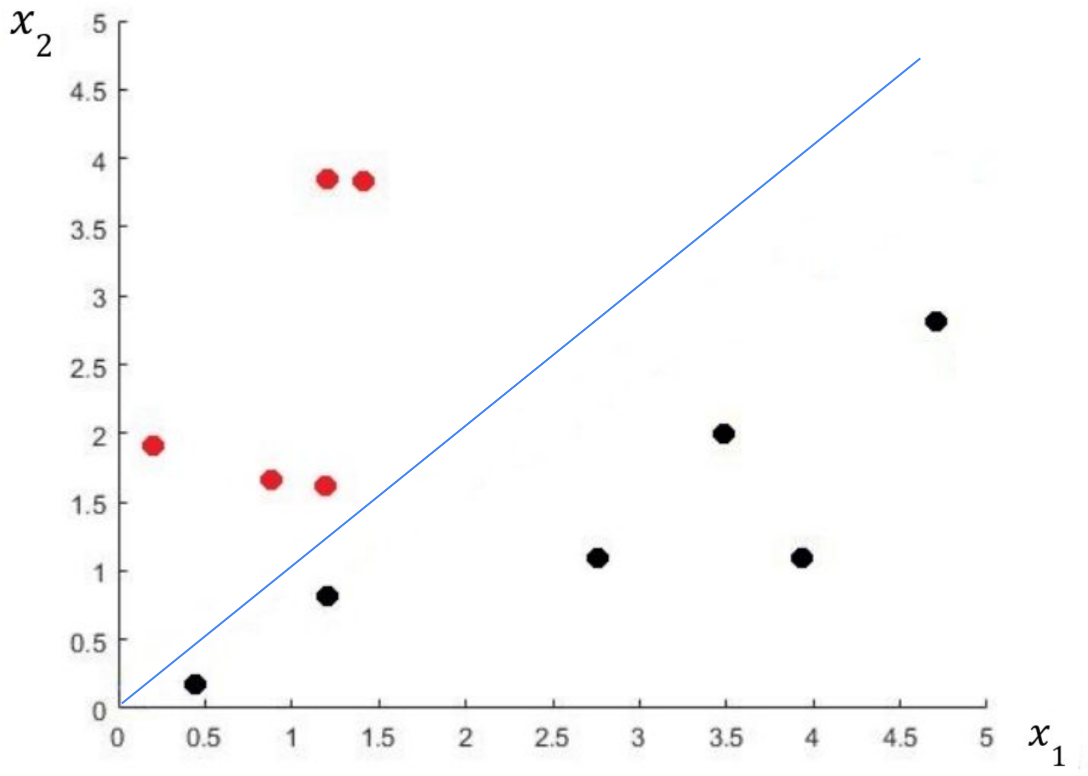


¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?



x_1	x_2	ζ
1.1946	3.8427	1
0.8788	1.6595	1
1.1907	1.6117	1
1.4180	3.8272	1
0.2032	1.9208	1
2.7571	1.0931	-1
4.7125	2.8166	-1
3.9392	1.1032	-1
1.2072	0.8132	-1
3.4799	1.9982	-1
0.4763	0.1020	-1

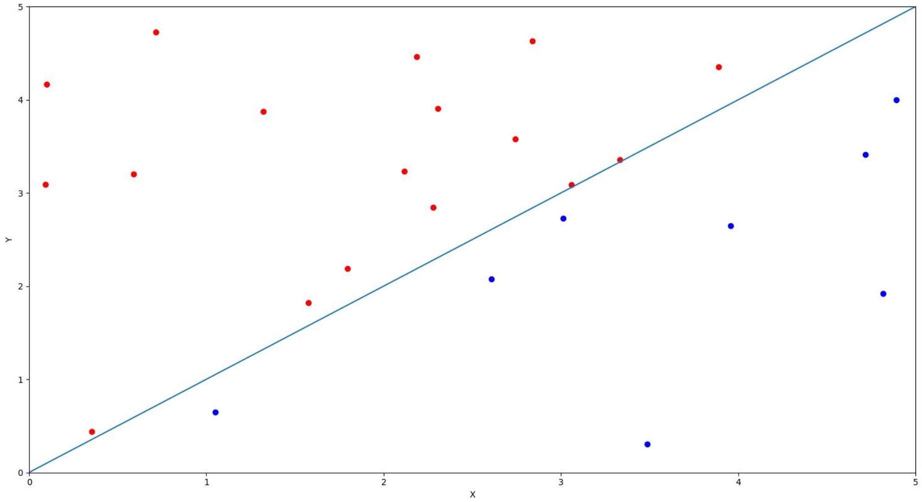
¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?



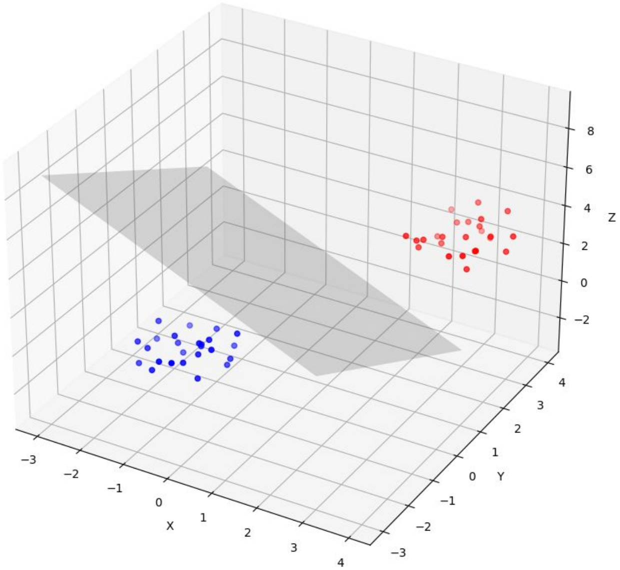
x_1	x_2	ζ
1.1946	3.8427	1
0.8788	1.6595	1
1.1907	1.6117	1
1.4180	3.8272	1
0.2032	1.9208	1
2.7571	1.0931	-1
4.7125	2.8166	-1
3.9392	1.1032	-1
1.2072	0.8132	-1
3.4799	1.9982	-1
0.4763	0.1020	-1

HIPERPLANO DE SEPARACIÓN

$R^2 \quad -> \quad recta\ de\ separación$



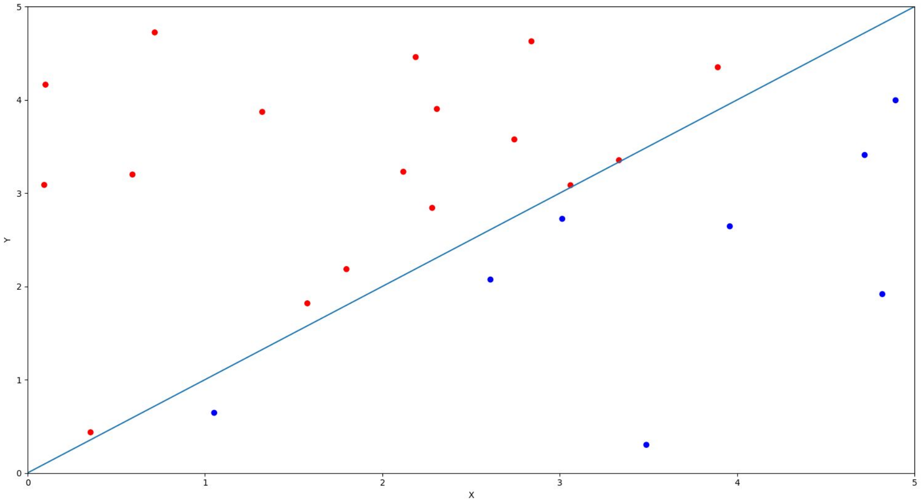
$R^3 \quad -> \quad plano\ de\ separación$



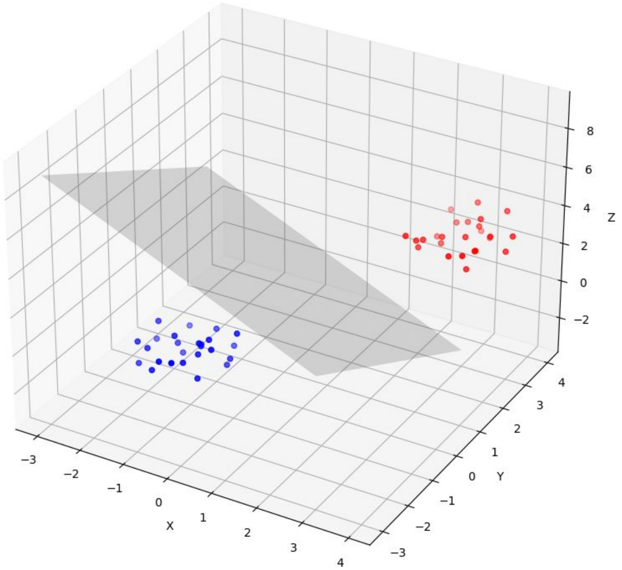
$R^n \quad -> \quad hiplerplano\ de\ separación$

HIPERPLANO DE SEPARACIÓN

$R^2 \quad -> \quad recta\ de\ separación$



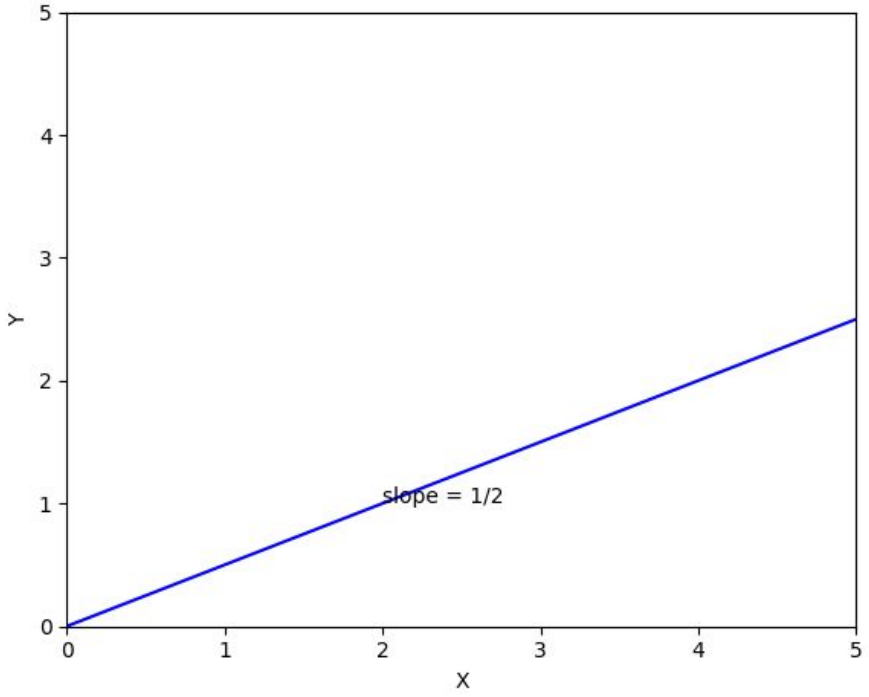
$R^3 \quad -> \quad plano\ de\ separación$



$R^n \quad -> \quad hiplerplano\ de\ separación$

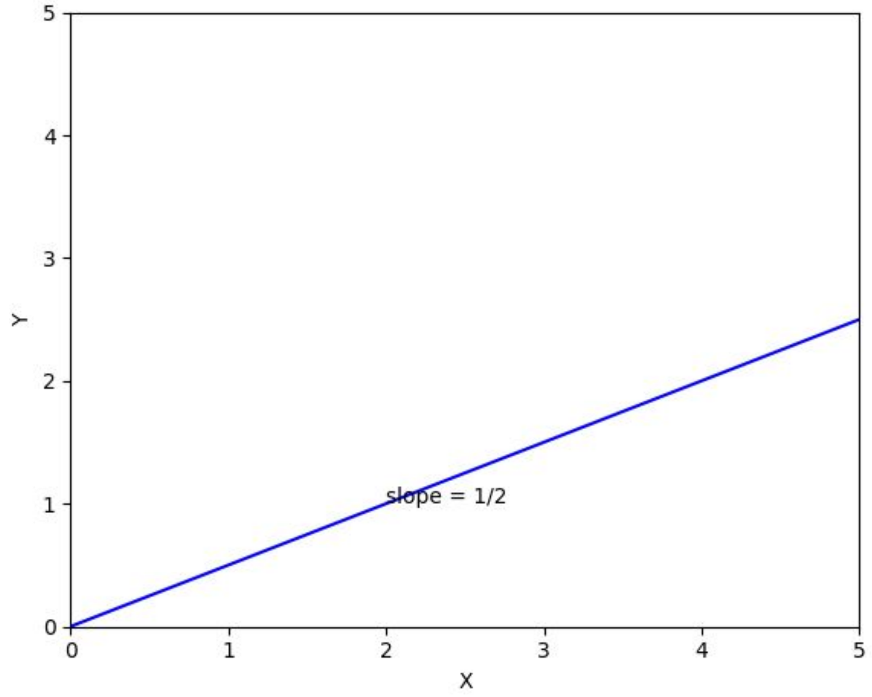
Problema de separabilidad lineal

REPRESENTACIÓN DEL HIPERPLANO



$$y = mx + b \longrightarrow y = \frac{1}{2}x$$

REPRESENTACIÓN DEL HIPERPLANO

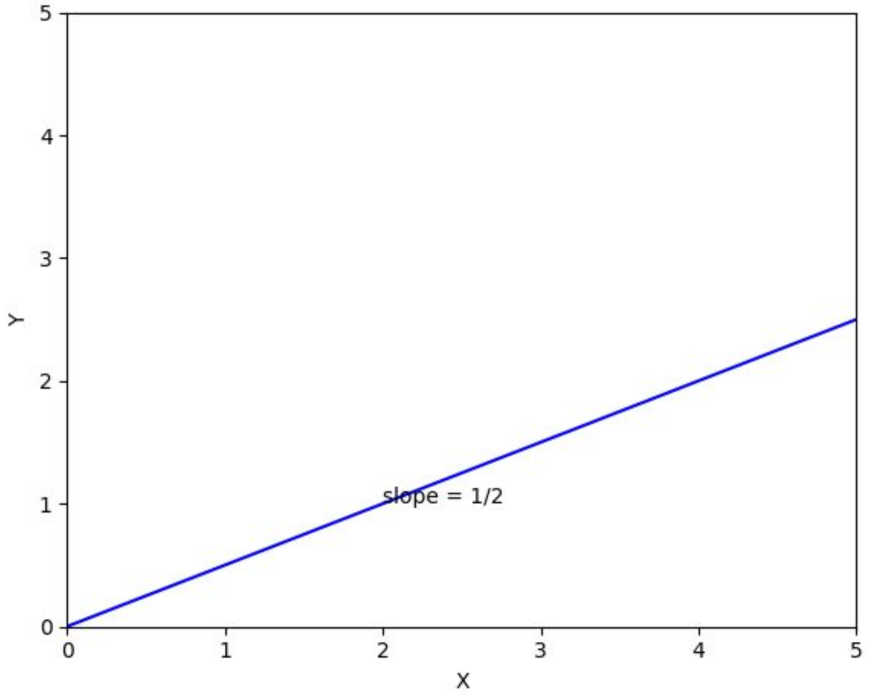


$$y = mx + b \longrightarrow y = \frac{1}{2}x$$

$$\downarrow$$

$$ax + by + c = 0$$

REPRESENTACIÓN DEL HIPERPLANO



$$y = mx + b \longrightarrow y = \frac{1}{2}x$$

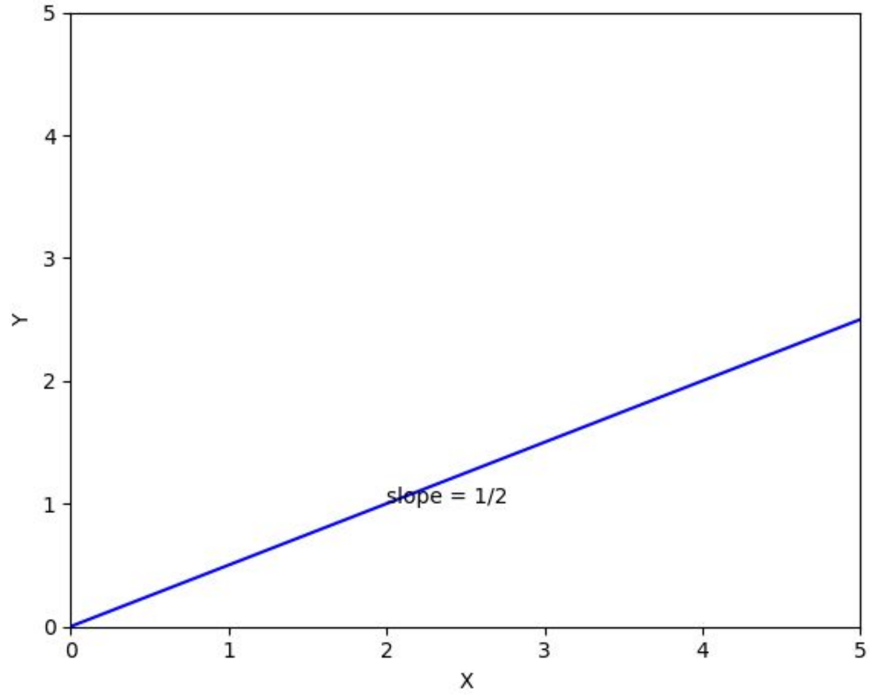
$$\downarrow$$

$$ax + by + c = 0$$

$$\downarrow$$

$$mx + (-1)y + b = 0$$

REPRESENTACIÓN DEL HIPERPLANO



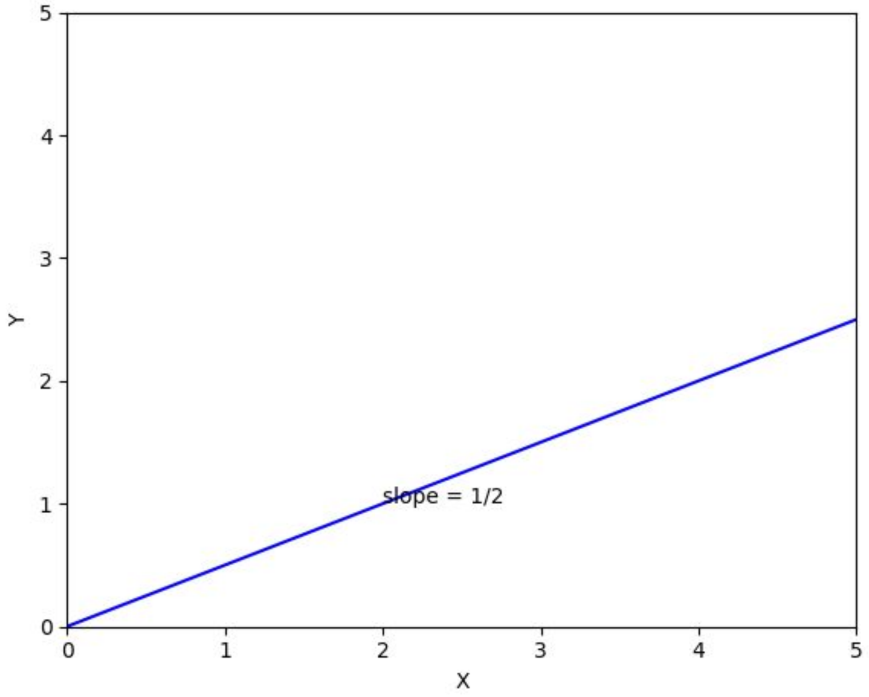
$$y = mx + b \longrightarrow y = \frac{1}{2}x$$

$$ax + by + c = 0$$

$$mx + (-\ 1)y + b = 0$$

$$\frac{1}{2}x + (-\ 1)y + 0 = 0$$

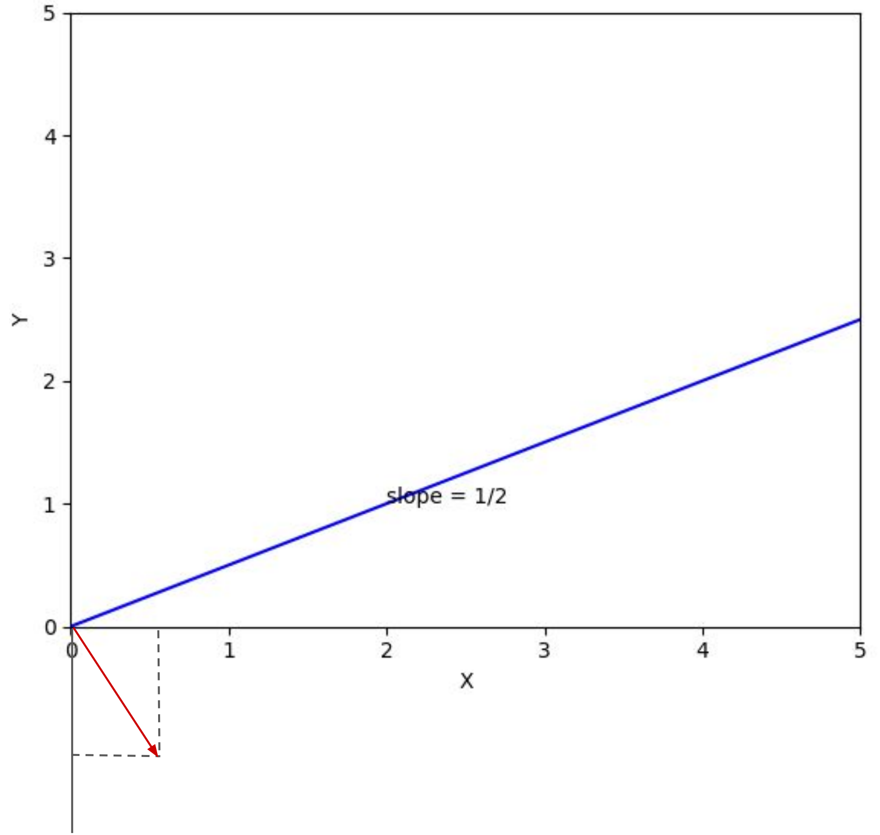
REPRESENTACIÓN DEL HIPERPLANO



$$\frac{1}{2}x + (-\ 1)y + 0 = 0$$

$$w = (\frac{1}{2},\ \ -\ 1)$$

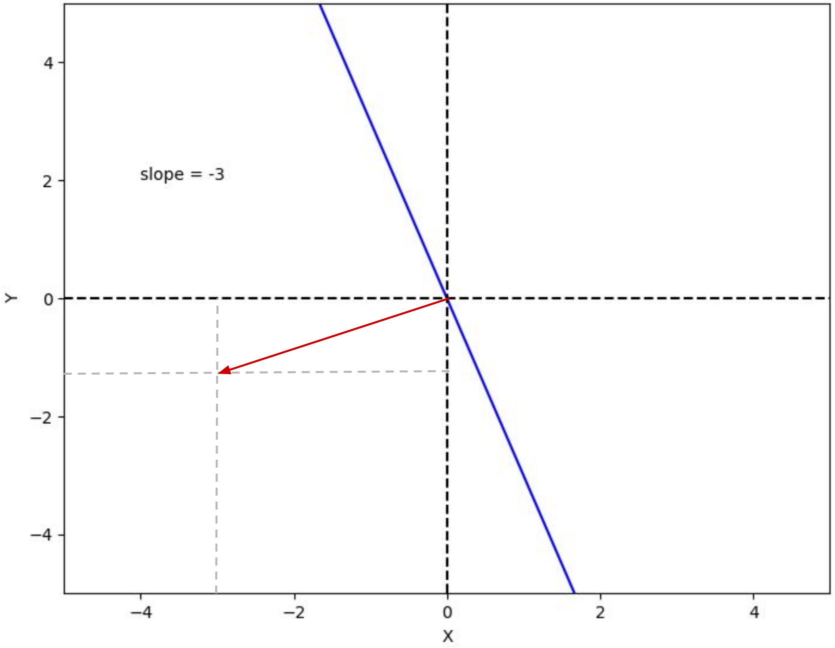
REPRESENTACIÓN DEL HIPERPLANO



$$\frac{1}{2}x + (-\ 1)y + 0 = 0$$

$$w = (\frac{1}{2},\ \ -\ 1)$$

REPRESENTACIÓN DEL HIPERPLANO



$$y = (-3)x$$

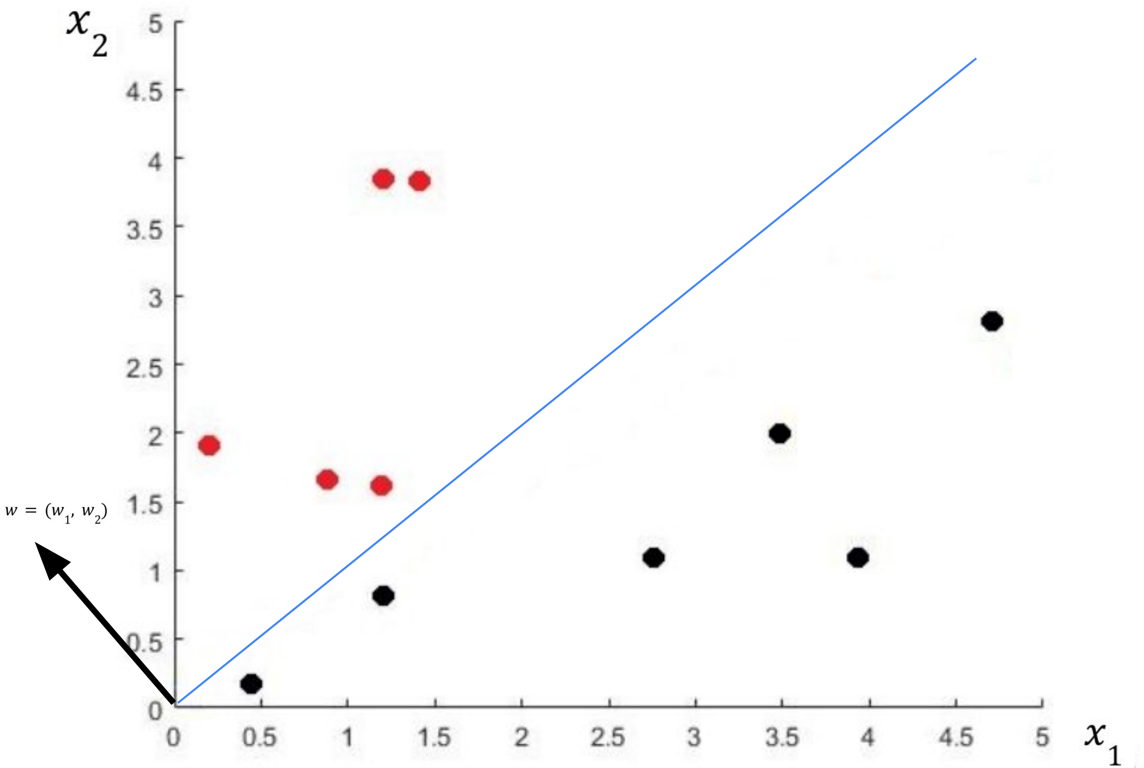
$$\downarrow$$

$$(-3)x + (-1)y + 0 = 0$$

$$\downarrow$$

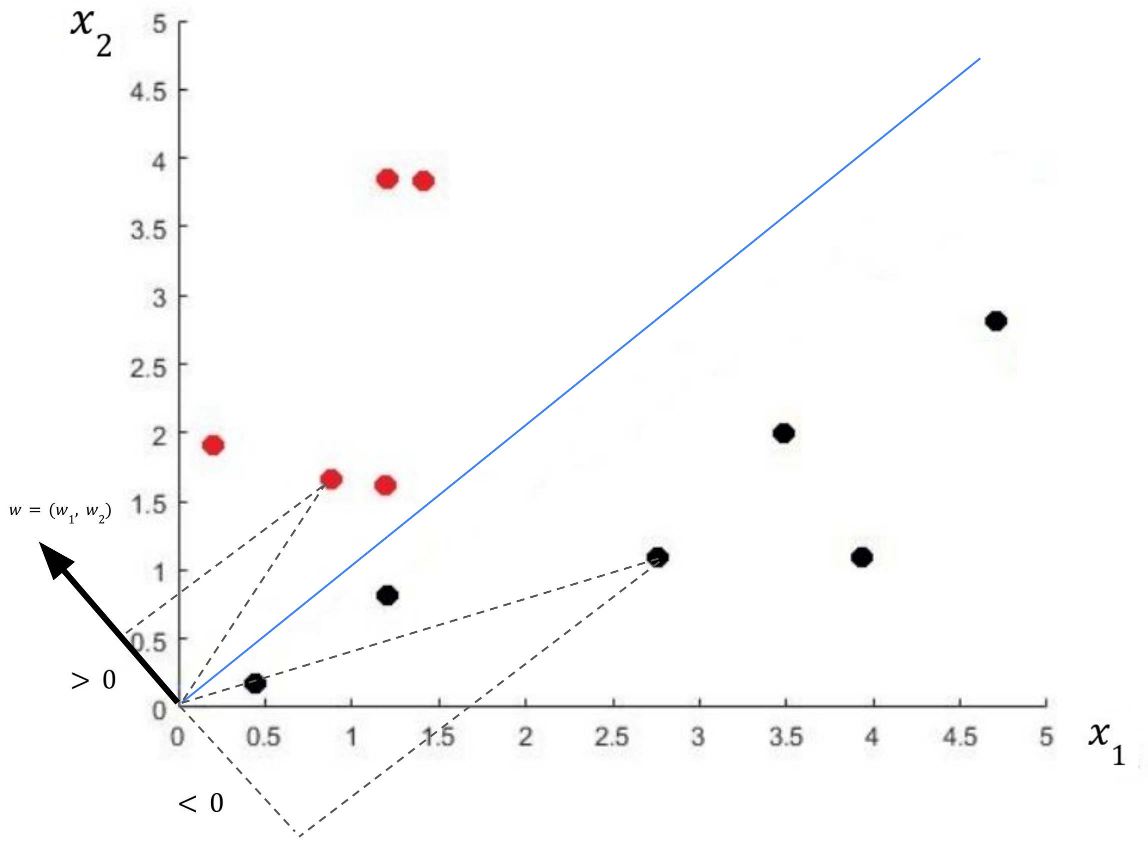
$$w = (-3, -1)$$

¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?



x_1	x_2	ζ
1.1946	3.8427	1
0.8788	1.6595	1
1.1907	1.6117	1
1.4180	3.8272	1
0.2032	1.9208	1
2.7571	1.0931	-1
4.7125	2.8166	-1
3.9392	1.1032	-1
1.2072	0.8132	-1
3.4799	1.9982	-1
0.4763	0.1020	-1

¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?

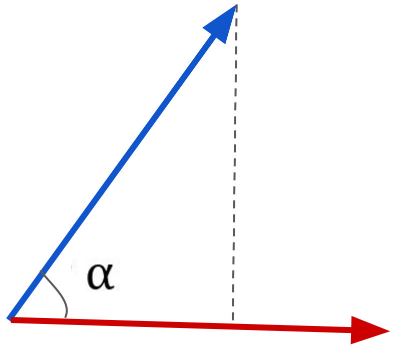


x_1	x_2	ζ
1.1946	3.8427	1
0.8788	1.6595	1
1.1907	1.6117	1
1.4180	3.8272	1
0.2032	1.9208	1
2.7571	1.0931	-1
4.7125	2.8166	-1
3.9392	1.1032	-1
1.2072	0.8132	-1
3.4799	1.9982	-1
0.4763	0.1020	-1

PROYECCIÓN VECTORIAL

La proyección del vector a sobre el vector b puede definirse como:

$$||a||\ cos\ \alpha$$



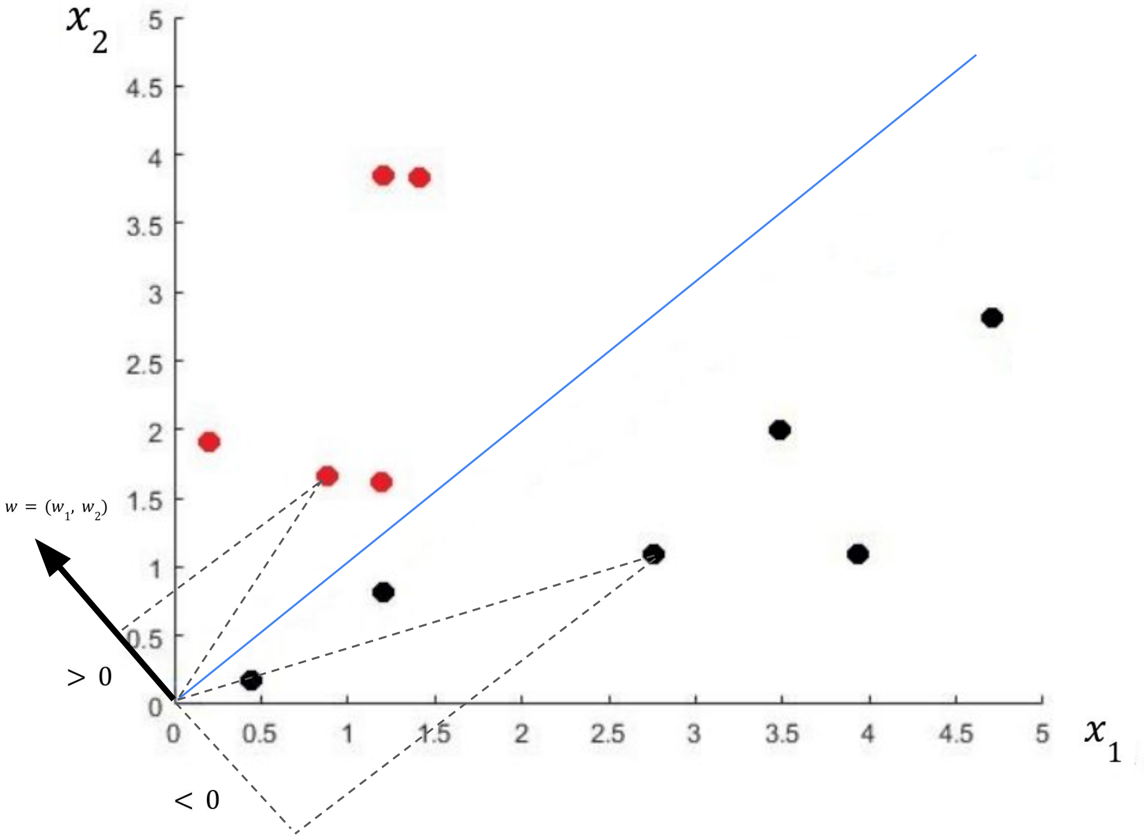
A su vez, esto equivale al producto interno:

$$||a||\ ||b||\ cos\ \alpha\ =\ a.b$$

Definición de producto interno:

$$a.b\ =\ \sum_{i=1}^{n} a_i \,.\, b_i$$

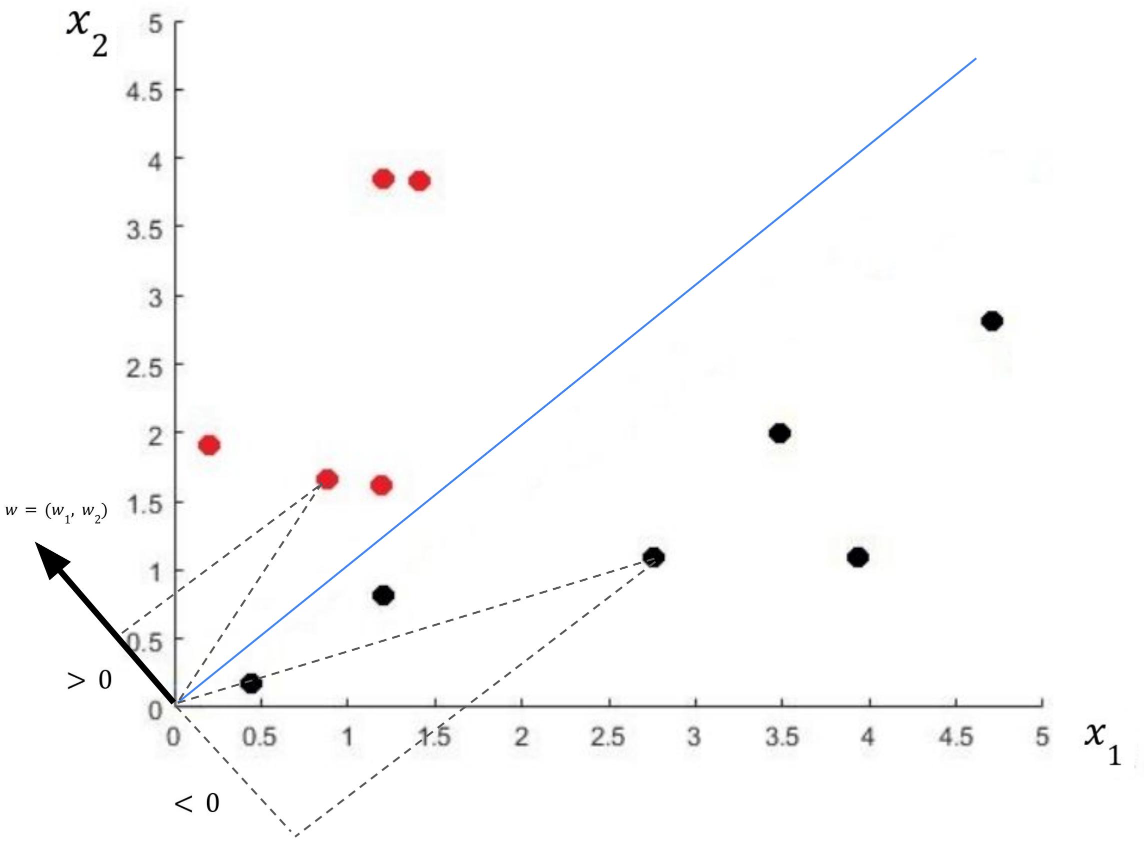
PROYECCIÓN DEL DATO SOBRE EL VECTOR *W*



Para el vector ***w*** que se corresponde a la recta **que pasa por el origen**:

$$proyección = \sum_{i=1}^{2} x_i \, . \, w_i$$

PROYECCIÓN DEL DATO SOBRE EL VECTOR *W*

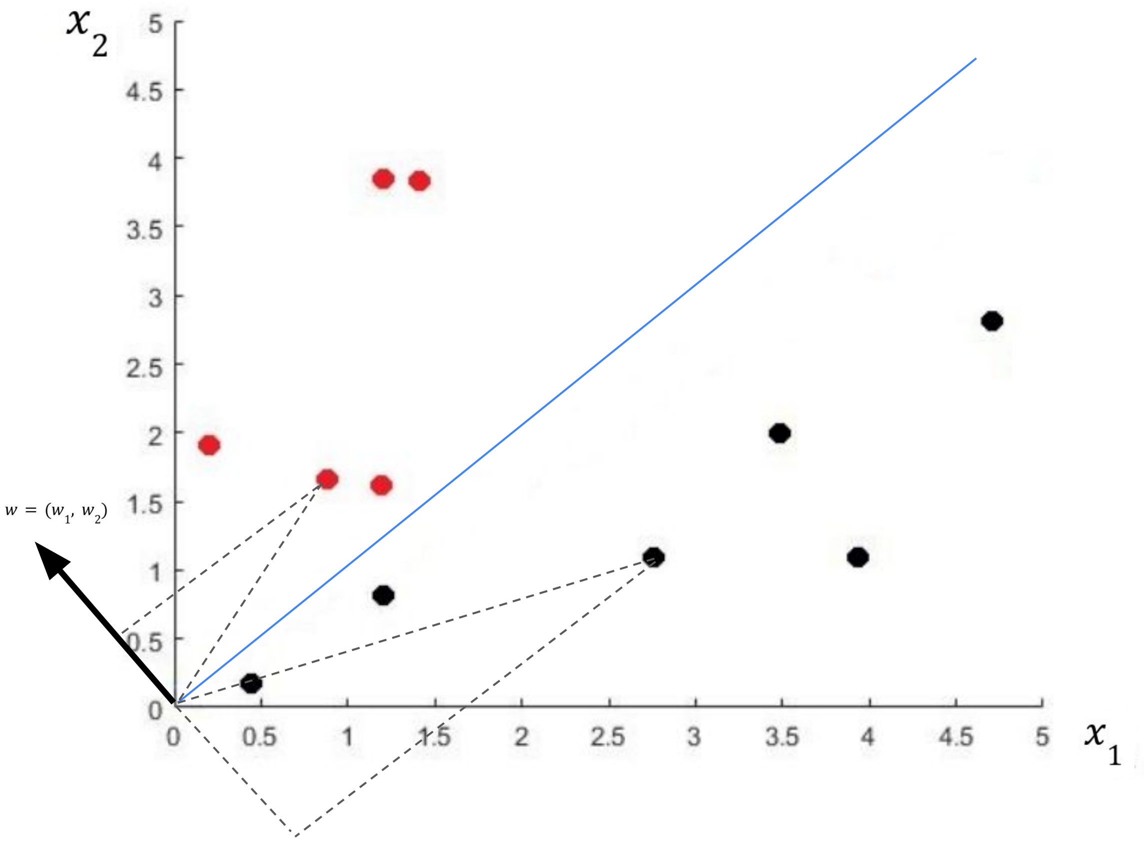


Para el vector ***w*** que se corresponde a la recta **que pasa por el origen**:

$$proyección = \sum_{i=1}^{2} x_i . w_i$$

Si la proyección es negativa, la clase del dato será **(-1)**, mientras que si es positiva la clase será **(1)**

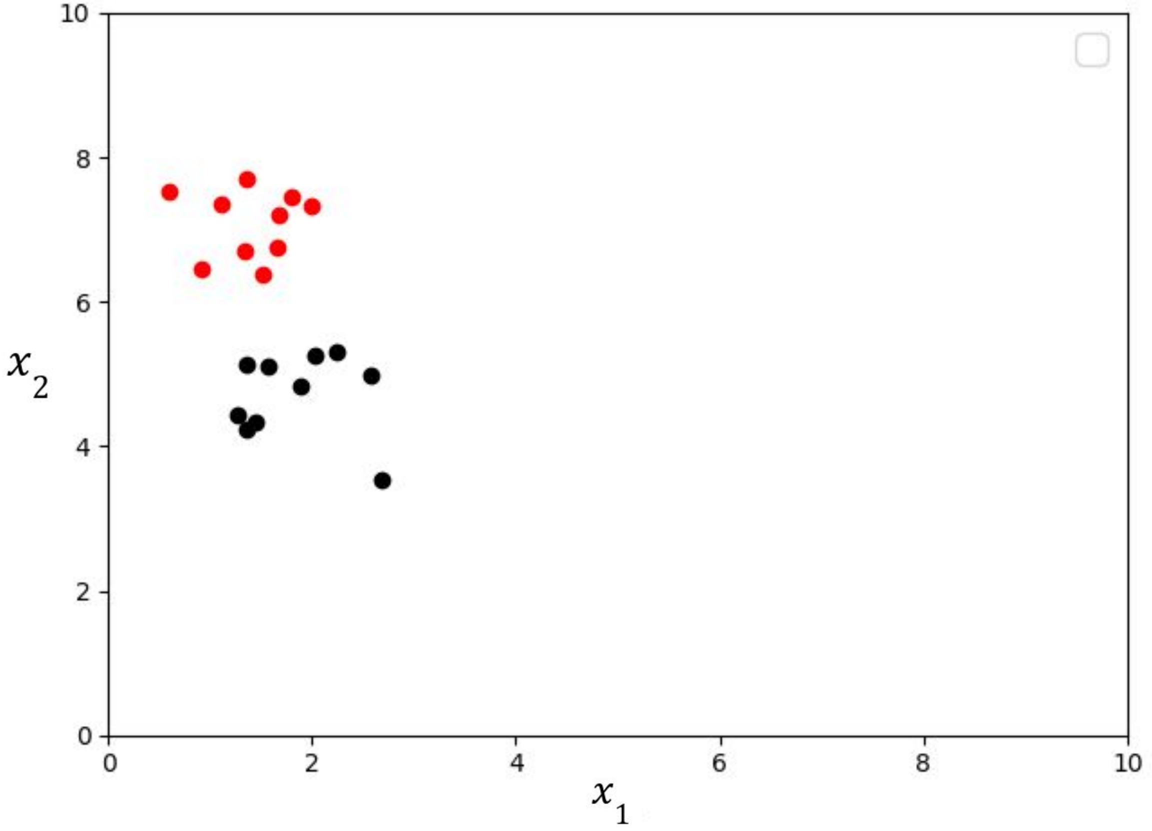
PROYECCIÓN DEL DATO SOBRE EL VECTOR ***W***



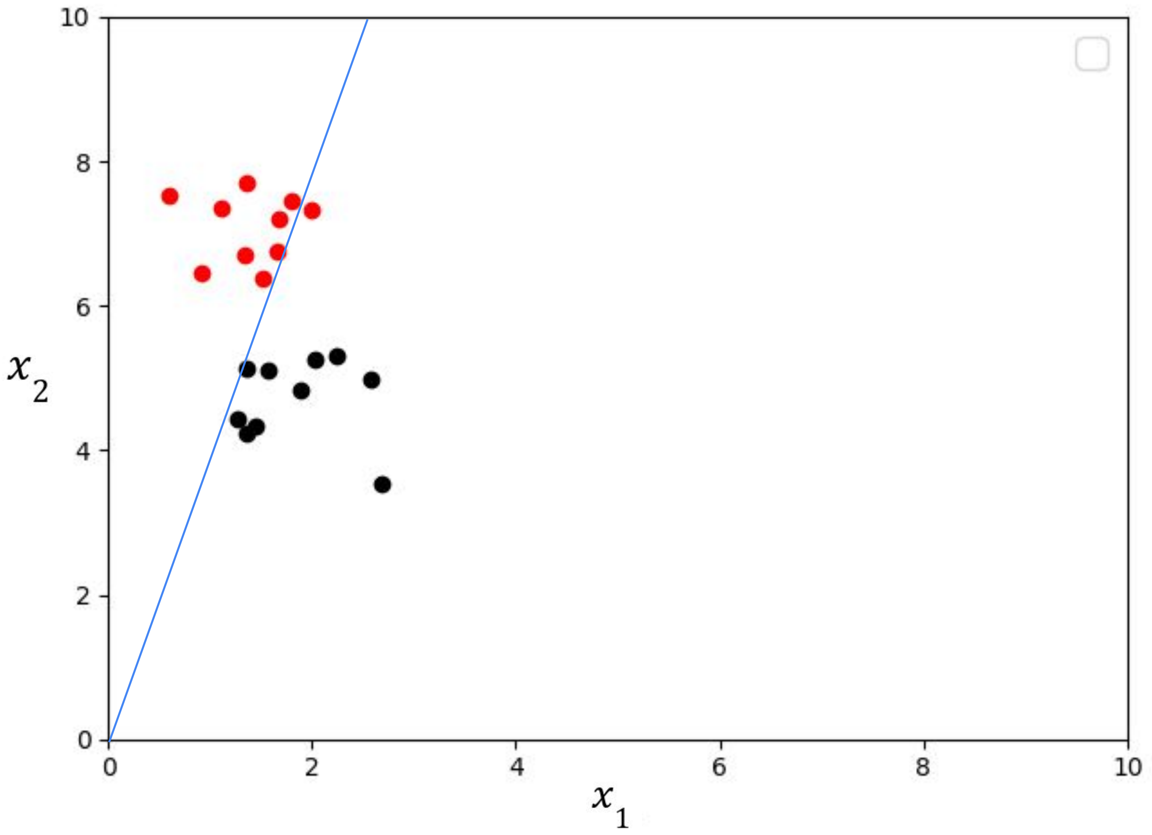
$$clase = \theta(\sum_{i=1}^{n} x_i . w_i) \qquad n = 2$$

$$\theta(x) = \begin{cases} 1 & x \geq 0 \\ -1 & en\ otro\ caso \end{cases}$$

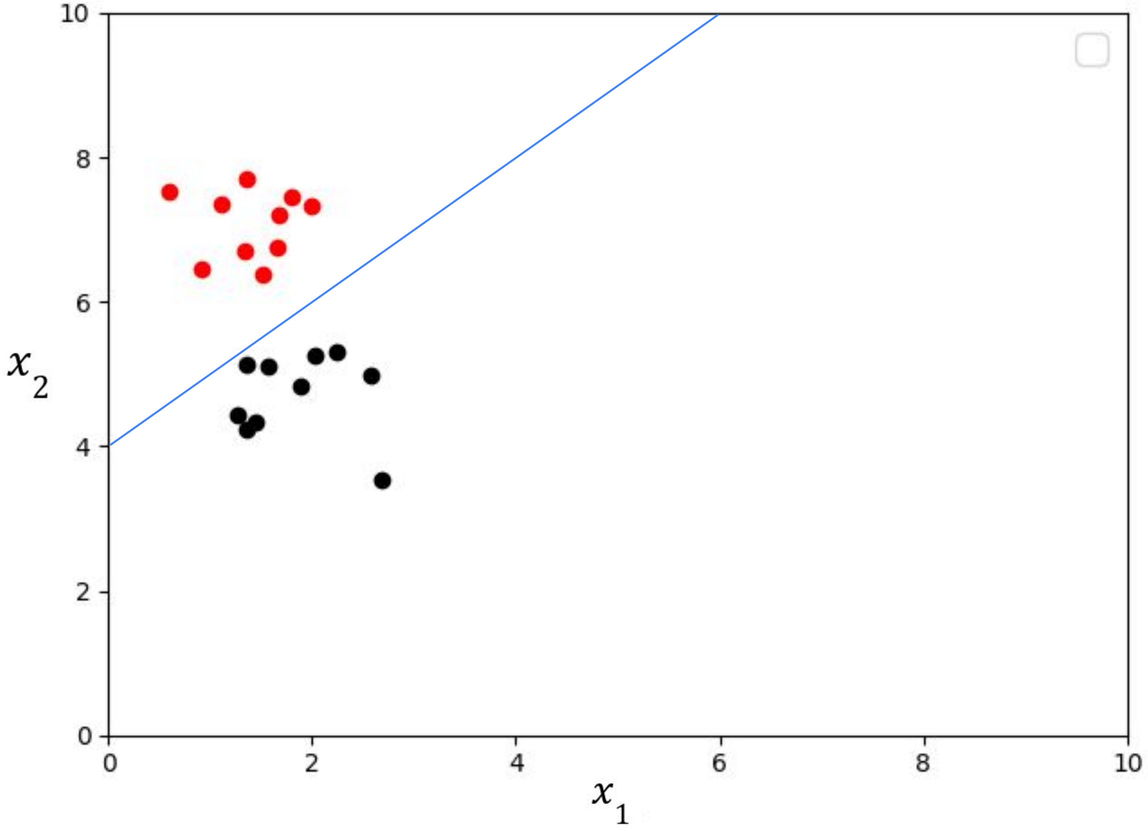
¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?



¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?



¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?

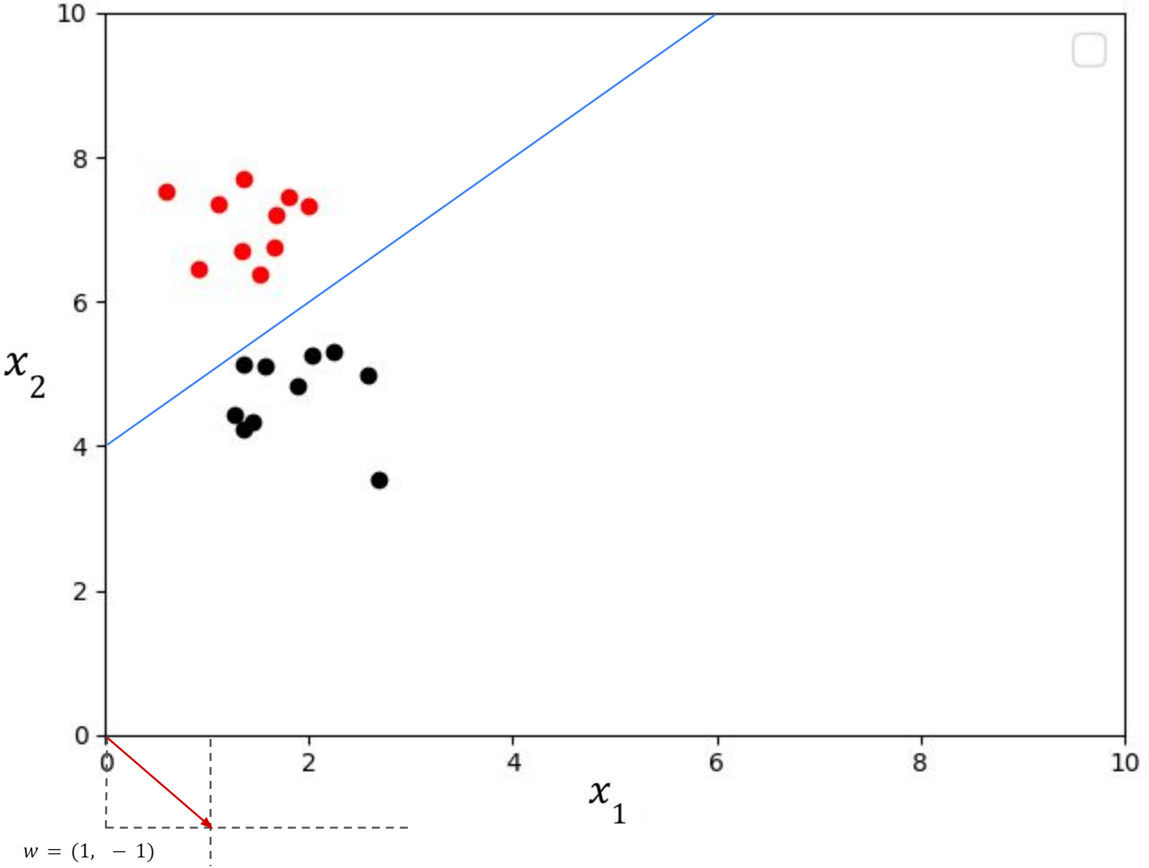


$$ax + by + c = 0$$

$$w_1x_1 + w_2x_2 - w_0 = 0$$

$$\sum_{i=1}^{n} x_i . w_i - w_0 = 0$$

¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?

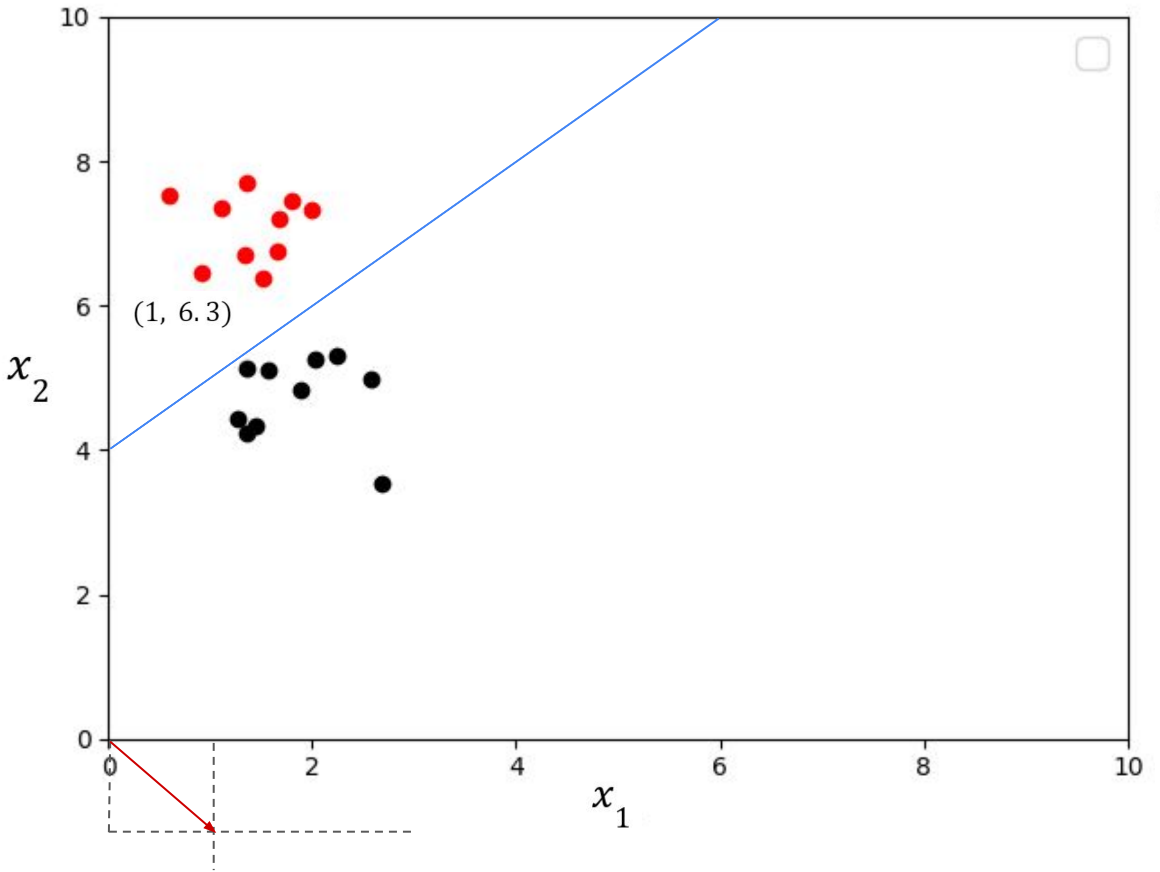


$$y = x + 4$$

$$(1)x + (-1)y - (-4) = 0$$

$$w = (1, \ -1) \qquad w_0 = -4$$

¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?



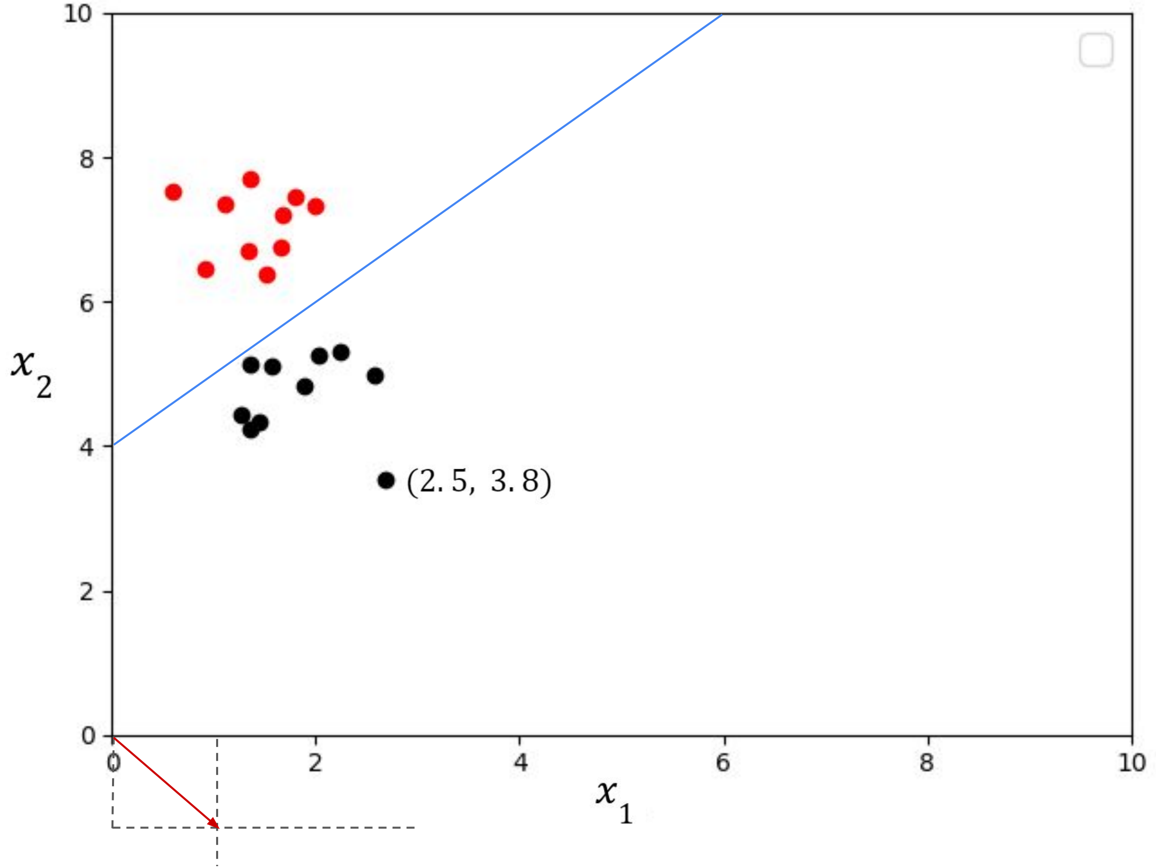
$$y = x + 4$$

$$(1)x + (-1)y - (-4) = 0$$

$$w = (1,\ -1) \quad w_0 = -4$$

$$\sum_{i=1}^{n} x_i . w_i - w_0 = -1.3$$

¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?



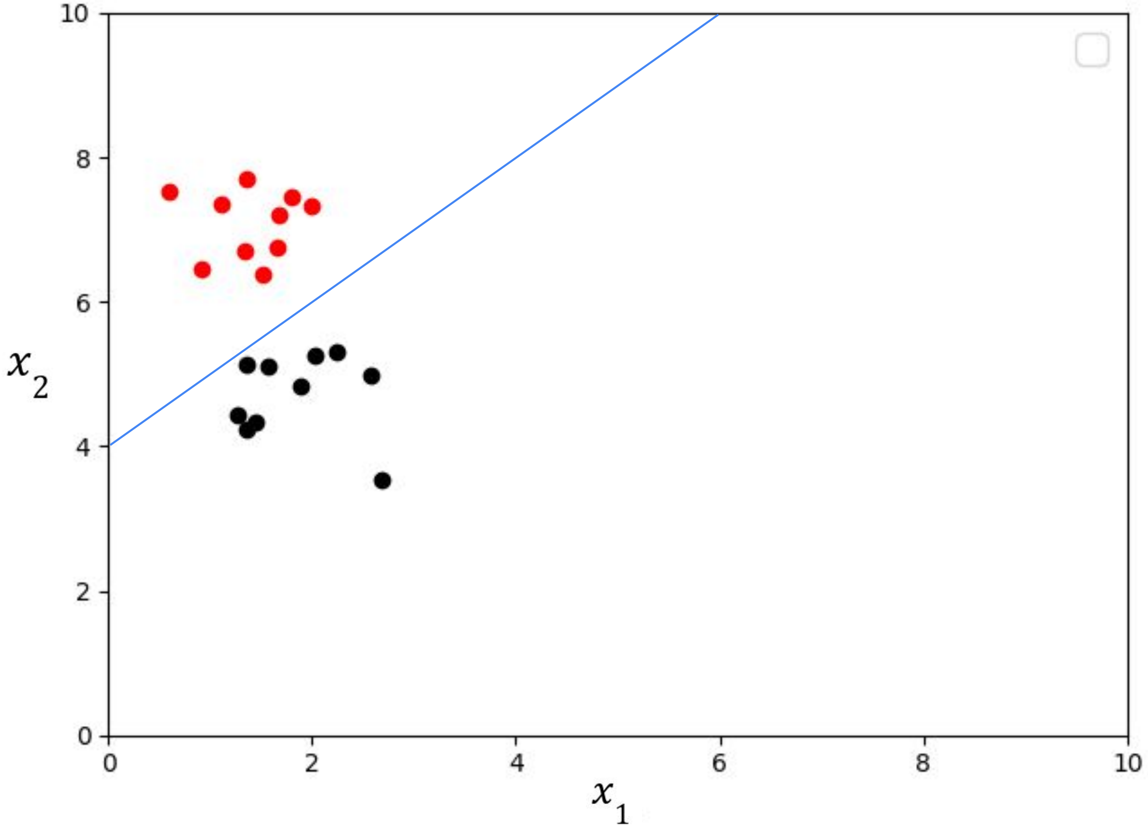
$$y = x + 4$$

$$(1)x + (-1)y - (-4) = 0$$

$$w = (1, -1) \quad w_0 = -4$$

$$\sum_{i=1}^{n} x_i . w_i - w_0 = 2.7$$

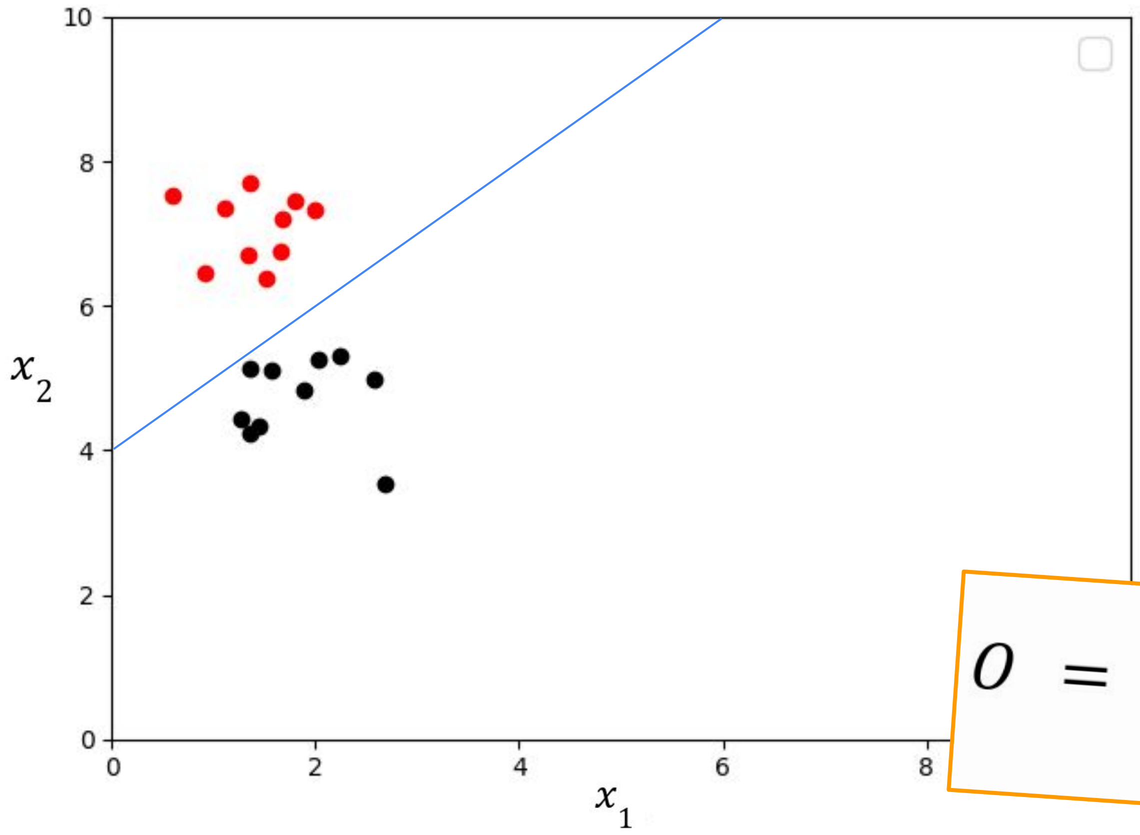
¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?



$$clase = \theta\left(\sum_{i=1}^{n} x_i . w_i - w_0\right)$$

$$\theta = \begin{cases} 1 & x \geq 0 \\ -1 & en\ otro\ caso \end{cases}$$

¿QUÉ TIPOS DE PROBLEMA PUEDE RESOLVER DICHO MODELO?



$$clase = \theta(\sum_{i=1}^{n} x_i . w_i - w_0)$$

$$\theta = \begin{cases} 1 & x \geq 0 \\ -1 & en\ otro\ caso \end{cases}$$

Formalización del perceptrón

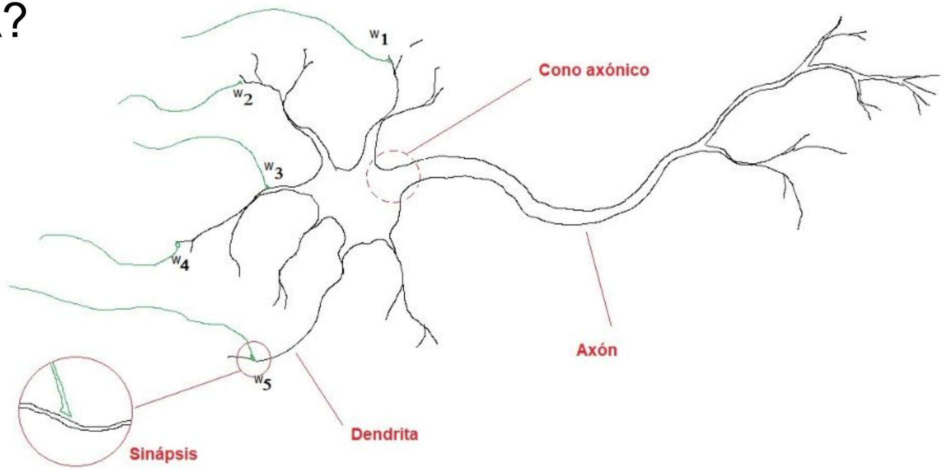
$$O = \theta(\sum_{i=1}^{n} x_i . w_i - u)$$

¿QUÉ TENEMOS HASTA AHORA?

- Modelo matemático de neurona biológica

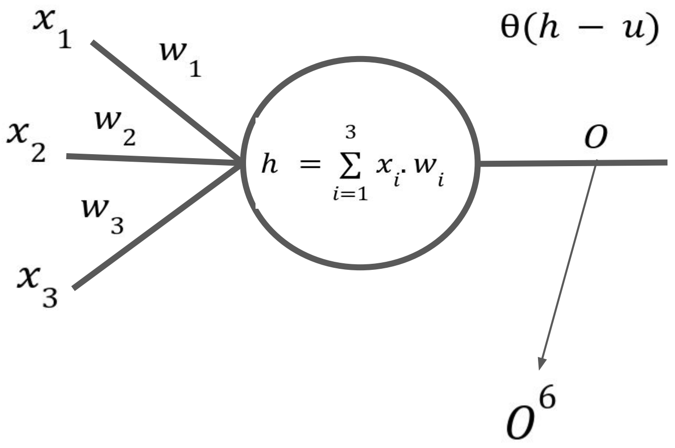
- Herramienta para resolver problemas linealmente separables en R^n

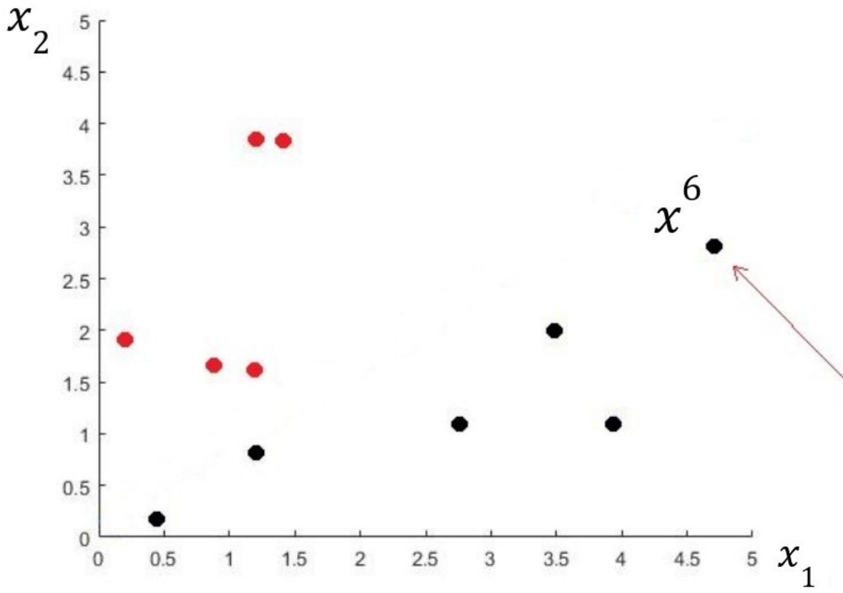
¿QUÉ NOS FALTA?



REPASO

p es la cantidad de datos. μ representa el dato (0…***p***-1)





x_1	x_2	ζ
1.1946	3.8427	1
0.8788	1.6595	1
1.1907	1.6117	1
1.4180	3.8272	1
0.2032	1.9208	1
2.7571	1.0931	-1
4.7125	2.8166	-1
3.9392	1.1032	-1
1.2072	0.8132	-1
3.4799	1.9982	-1
0.4763	0.1020	-1

ζ^6

x_1^6 x_2^6

Frank Rosenblatt 1957

Cada vez que la neurona recibe un estímulo, los pesos sinápticos pueden actualizarse (proceso iterativo):

$$w^{nuevo} = w^{anterior} + \Delta w$$

¿Cómo calculamos el delta?

$$\Delta w = 2\eta x^{\mu}\zeta^{\mu}$$

salida esperada

entrada o dato

tasa de aprendizaje
(escalar)

$$O^{\mu} \neq \zeta^{\mu}$$

salida obtenida

Paper: https://psycnet.apa.org/record/1959-09865-001

ACTUALIZACIÓN DE LOS PESOS SINÁPTICOS

¿Cómo calculamos el delta?

$$\Delta w = \begin{cases} 2\eta x^{\mu}\zeta^{\mu} & O^{\mu} \neq \zeta^{\mu} \\ 0 & en\ otro\ caso \end{cases}$$

Si consideramos que la salida puede ser 1 o -1, podemos reformular:

$$\Delta w = \eta(\zeta^{\mu} - O^{\mu})x^{\mu}$$

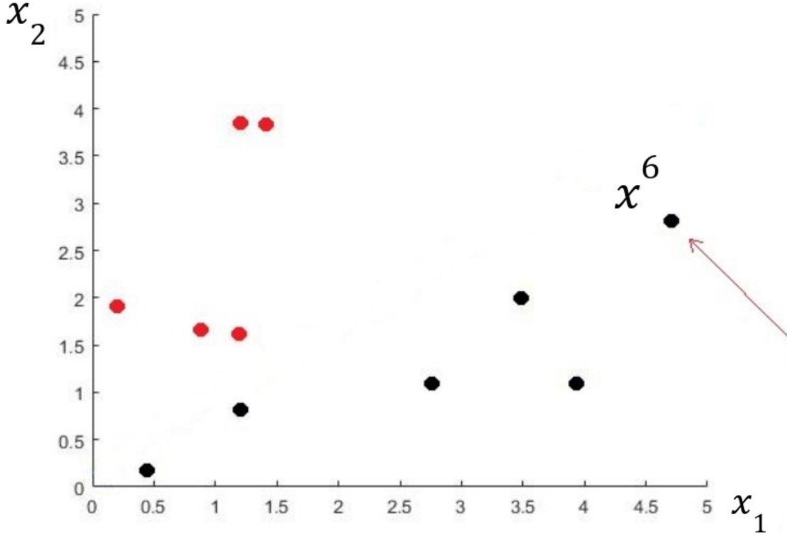
EJEMPLO NUMÉRICO

$$\boxed{\Delta w = \eta(\zeta^{\mu} - O^{\mu})x^{\mu}}$$

$$\boxed{w^{nuevo} = w^{anterior} + \Delta w}$$

$$(x_1^6, x_2^6) = (4.7125, 2.8166) \qquad \zeta^6 = (-1)$$

$$(w_1, w_2) = (3.25, 4.3) \qquad w_0 = 1.86$$



x_1	x_2	ζ
1.1946	3.8427	1
0.8788	1.6595	1
1.1907	1.6117	1
1.4180	3.8272	1
0.2032	1.9208	1
2.7571	1.0931	-1
4.7125	2.8166	-1
3.9392	1.1032	-1
1.2072	0.8132	-1
3.4799	1.9982	-1
0.4763	0.1020	-1

ζ^6

x_1^6 $\qquad$ x_2^6

EJEMPLO NUMÉRICO

$$\boxed{\Delta w = \eta(\zeta^{\mu} - O^{\mu})x^{\mu}}$$

$$\boxed{w^{nuevo} = w^{anterior} + \Delta w}$$

$$(x_1^6, x_2^6) = (4.7125, 2.8166) \qquad \zeta^6 = (-1)$$

$$(w_1, w_2) = (3.25, 4.3) \qquad w_0 = 1.86$$

$$O = \theta(4.7125 \,.\, 3.25 + 2.8166 \,.\, 4.3 - 1.86)$$

$$O = \theta(25.567005) = 1$$

$$\Delta w = \eta(\zeta^{\mu} - O^{\mu})x^{\mu}$$

$$w^{nuevo} = w^{anterior} + \Delta w$$

EJEMPLO NUMÉRICO

$$(x_1^6, x_2^6) = (4.7125, 2.8166) \qquad \zeta^6 = (-1)$$

$$(w_1, w_2) = (3.25, 4.3) \qquad w_0 = 1.86$$

$$O = \theta(4.7125 \,.\, 3.25 + 2.8166 \,.\, 4.3 - 1.86)$$

$$O = \theta(25.567005) = 1$$

$$w^{nuevo} = (1.86, 3.25, 4.3) + 0.1(-2)(1, 4.7125, 2.8166)$$

$$w^{nuevo} = (2.3075,\ 3.73668,\ 1.66)$$

$$\boxed{\Delta w = \eta(\zeta^{\mu} - O^{\mu})x^{\mu}}$$

EJEMPLO NUMÉRICO

$$\boxed{w^{nuevo} = w^{anterior} + \Delta w}$$

$$(x_1^6, x_2^6) = (4.7125, 2.8166) \qquad \zeta^6 = (-1)$$

$$(w_1, w_2) = (3.25, 4.3) \qquad w_0 = 1.86$$

$$O = \theta(4.7125 . 3.25 + 2.8166 . 4.3 - 1.86)$$

$$O = \theta(25.567005) = 1$$

$$w^{nuevo} = (1.86, 3.25, 4.3) + 0.1(-2)(1, 4.7125, 2.8166)$$

$$w^{nuevo} = (2.3075, 3.73668, 1.66)$$

$$O = \theta(4.7125 . 2.3075 + 2.8166 . 3.73668 - 1.66)$$

$$O = \theta(19.7388) = 1$$

ALGORITMO DEL PERCEPTRÓN SIMPLE

1. Inicializar los pesos sinápticos en valores aleatorios pequeños o cero $w = (w_0, w_1, ... w_n)$

2. Definir: tasa de aprendizaje, épocas. $\eta \sim 0.1$

3. Para cada elemento del conjunto de datos

   a. Calcular la salida de la neurona $O = \theta(\sum_{i=1}^{n} x_i . w_i - w_0)$

   b. Actualizar los pesos sinápticos $w^{nuevo} = w^{anterior} + \eta(\zeta^{\mu} - O^{\mu})x^{\mu}$

4. Calcular el error del perceptrón para verificar si se alcanzó convergencia
   a. Si el perceptrón alcanzó convergencia, finalizar.

5. Repetir 3 y 4 hasta alcanzar convergencia o hasta finalizar la cantidad de épocas

ALGORITMO DEL PERCEPTRÓN SIMPLE

1. Inicializar los pesos sinápticos en valores aleatorios pequeños o cero

2. Definir: tasa de aprendizaje, épocas.

3. Para cada elemento del conjunto de datos

    a. Calcular la salida de la neurona
    b. Actualizar los pesos sinápticos

1 época

4. Calcular el error del perceptrón para verificar si se alcanzó convergencia
    a. Si el perceptrón alcanzó convergencia, finalizar.

5. Repetir 3 y 4 hasta alcanzar convergencia o hasta finalizar la cantidad de épocas

ALGORITMO DEL PERCEPTRÓN SIMPLE

4. Calcular el error del perceptrón para verificar si se alcanzó convergencia
    a. Si el perceptrón alcanzó convergencia, finalizar.

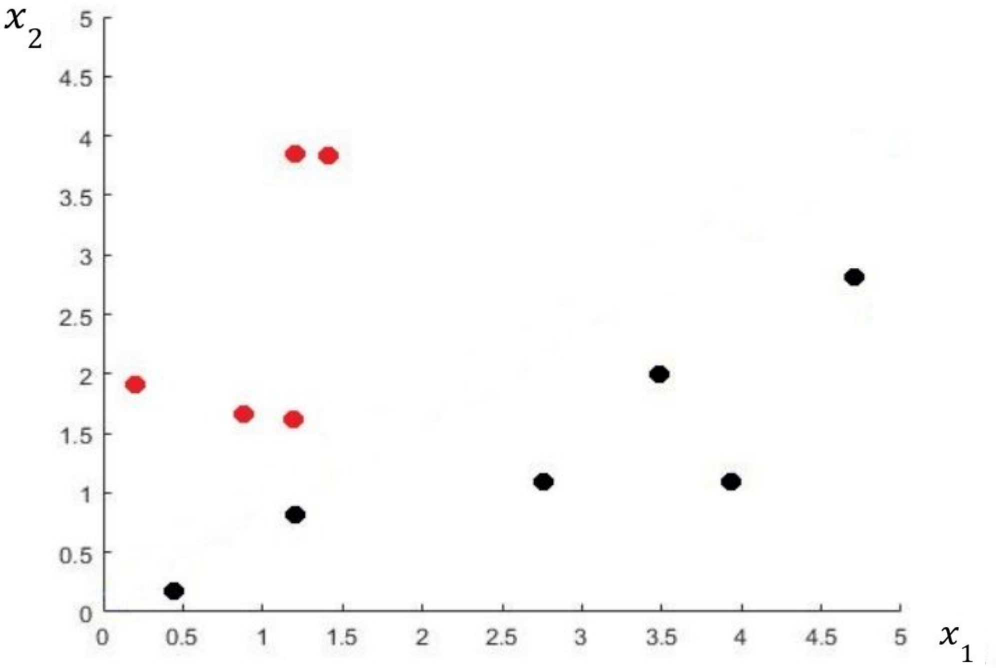
Distintas formas:

- Los pesos no se actualizaron en la última iteración
- La suma del valor absoluto de los errores devuelve cero
- Accuracy es del 100% (para problemas de clasificación)

$$Accuracy = (correct\ predictions)/(total\ predictions)$$

AGREGAR EL UMBRAL/BIAS

$$w^{nuevo} = (1.86, 3.25, 4.3) + 0.1(-2)\boxed{(1,}\,4.7125, 2.8166)$$



x_0	x_1	x_2	ζ
1	1.1946	3.8427	1
1	0.8788	1.6595	1
1	1.1907	1.6117	1
1	1.4180	3.8272	1
1	0.2032	1.9208	1
1	2.7571	1.0931	−1
1	4.7125	2.8166	−1
1	3.9392	1.1032	−1
1	1.2072	0.8132	−1
1	3.4799	1.9982	−1
1	0.4763	0.1020	−1

MÉTODOS PARA ACTUALIZAR LOS PESOS SINÁPTICOS

INCREMENTAL/ONLINE:

- Enviamos una entrada, a través del perceptrón.
- Calculamos la actualización de pesos.
- **Aplicamos la actualización**
- Repetimos hasta presentar todas las entradas.

LOTE/BATCH:

- Enviamos una entrada a través del perceptrón.
- Calculamos la actualización de pesos.
- Guardamos la información.
- Repetimos hasta presentar todas las entradas.
- **Aplicamos la actualización conjunta.**

ALGORITMO DEL PERCEPTRÓN SIMPLE

1. Inicializar los pesos sinápticos en valores aleatorios pequeños o cero

2. Definir: tasa de aprendizaje, épocas.

3. Para cada elemento del conjunto de datos

    a. Calcular la salida de la neurona
    b. Actualizar los pesos sinápticos

4. Calcular el error del perceptrón para verificar si se alcanzó convergencia
    a. Si el perceptrón alcanzó convergencia, finalizar.

5. Repetir 3 y 4 hasta alcanzar convergencia o hasta finalizar la cantidad de épocas

¿PARA QUÉ NOS SIRVE ESTA HERRAMIENTA?

<u>Aprendizaje</u>

Refiere al proceso de entrenar un **perceptrón** sobre un conjunto de datos, con el objetivo de minimizar el error o la función de costo de la red sobre las entradas del conjunto de datos.

<u>Generalización</u>

Refiere a la habilidad del **perceptrón** de desempeñarse correctamente sobre datos que no fueron alimentados durante el entrenamiento

¿ES EL PERCEPTRÓN LA ÚNICA FORMA DE RESOLVER ESTOS PROBLEMAS?

- SVM (Support Vector Machine) (1990s)
- Naive Bayes Classifier (18th century)
- Árboles de decisión (1950s/1960s)
- …

RESUMEN

- McCulloch y Pitts sientan las bases del modelo de neurona que se utiliza en el área de redes neuronales. Este modelo se denomina **Perceptrón**.

- El modelo de McCulloch y Pitts permite resolver problemas linealmente separables.

- Rosenblatt provee el mecanismo que permite obtener los pesos del perceptrón de manera iterativa

- No es lo mismo aprendizaje que generalización

TRABAJO PRÁCTICO N°3

Ejercicio 1

Implementar el algoritmo de perceptrón simple con función de activación escalón y utilizar el mismo para aprender los siguientes problemas:

Función lógica “Y” con entradas:
$x = \{\{-1, 1\}, \{1, -1\}, \{-1, -1\}, \{1, 1\}\}$

y salida esperada:
$y = \{-1, -1, -1, 1\}$

Función lógica “O exclusivo” con entradas:
$x = \{\{-1, 1\}, \{1, -1\}, \{-1, -1\}, \{1, 1\}\}$

y salida esperada:
$y = \{1, 1, -1, -1\}$

¿Qué puede decir acerca de los problemas que puede resolver el perceptrón simple escalón en relación a los problemas planteados en la consigna?

ALGUNAS VARIACIONES POSIBLES

Variable	Posibles Valores
Función de activación (Θ)	Escalón
Tasa de aprendizaje (η)	~0.1
Actualización de los pesos	Batch/Online
Error del perceptrón (función de costo)	Accuracy, suma valores absolutos, …
Épocas	

- Graficar error en función de la época
- Graficar la recta final (pueden hacer animación de todo el proceso)
- Programa con archivo de configuración