

# Redes neuronales

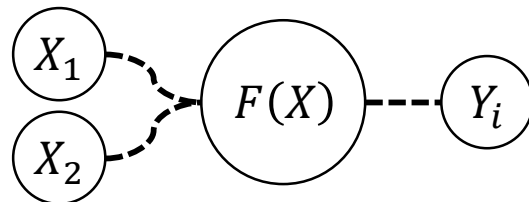
Parte 1

# Redes neuronales

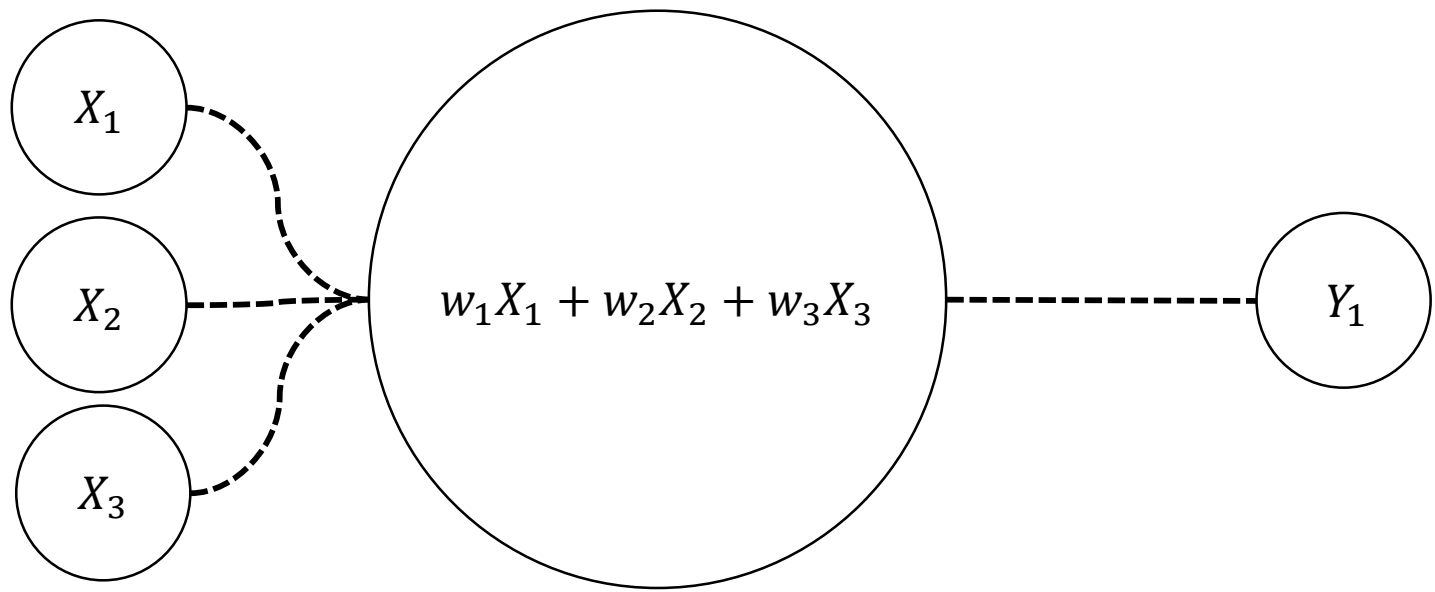
- Reconocimiento de caracteres, texto, voz, imágenes.
- Predicción bursátil
- Generación de texto
- Traducción de idiomas
- Prevención de fraude
- Conducción autónoma
- Pronóstico de enfermedades

# Redes neuronales

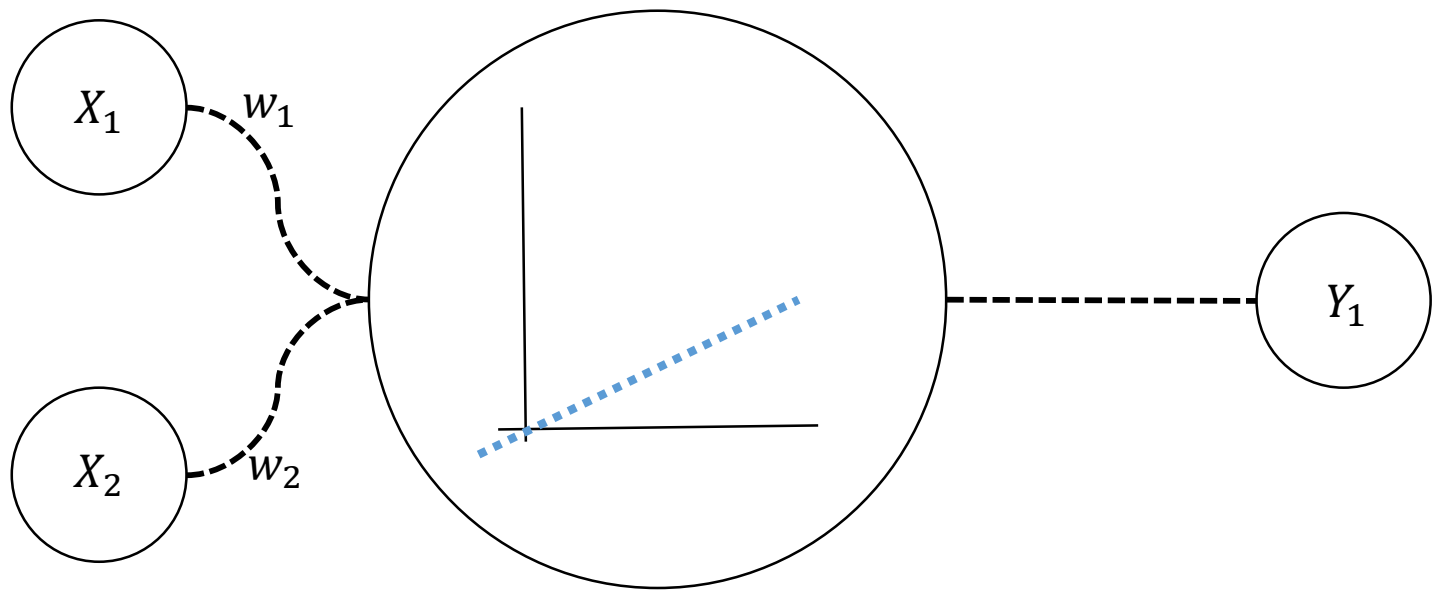
- Funcionan mediante la interacción de diferentes partes simples, las neuronas.
- Una neurona es la unidad básica de procesamiento en la red neuronal.
- Cada neurona:
  - Tiene conexiones de entrada para recibir estímulos externos (valores de entrada).
  - Realiza cálculos internos.
  - Genera un valor de salida.



# Redes neuronales

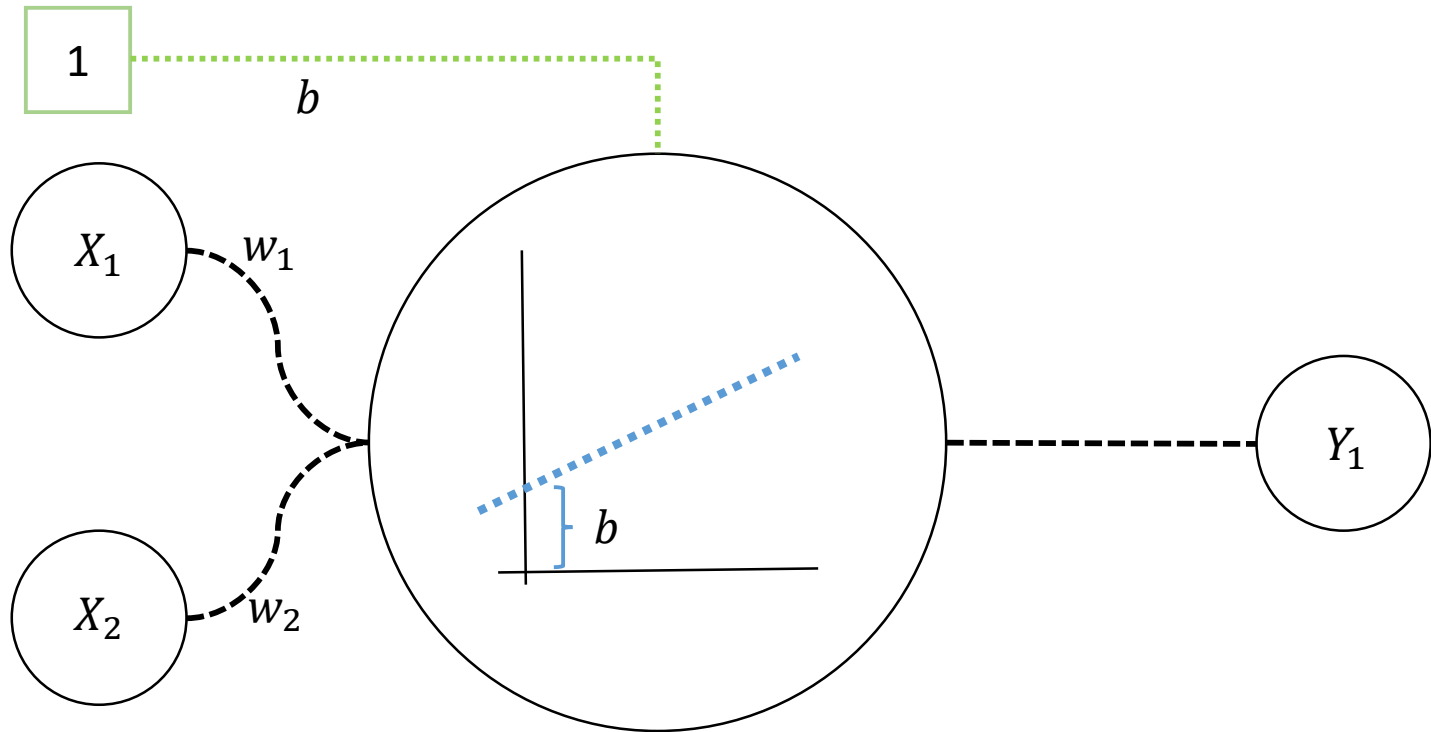


# Redes neuronales



$$y = w_1X_1 + w_2X_2$$

# Redes neuronales



$$y = w_1X_1 + w_2X_2 + b$$

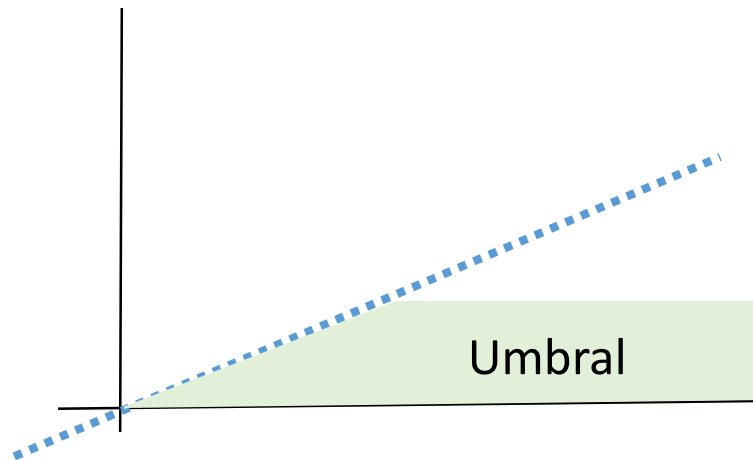
Donde  $b$  es conocido como sesgo (bias, en inglés)

# Ejemplo

- Supongan que estamos haciendo planes para hoy en la noche.
- Tenemos dos variables binarias que representan las entradas a nuestro modelo:
  - $X_1$  = tener bastante dinero (si o no)
  - $X_2$  = tener una buena compañía (si o no)
- También tendremos una variable binaria de salida que representa:
  - $Y_1$  = la situación es perfecta (si o no)

# Ejemplo

- Nuestra neurona se parece a un modelo de aproximación lineal, el cual es continuo, pero nuestro problema es binario.
- Se recurre a un umbral para determinar cuando se asigna un valor de 0 o de 1





# Ejemplo

- Nuestra neurona se parece a un modelo de aproximación lineal, el cual es continuo, pero nuestro problema es binario.
- Se recurre a un umbral para determinar cuando se asigna un valor de 0 o de 1

$$w_1X_1 + w_2X_2 + b \leq \textit{Umbral} \Rightarrow Y = 0$$

$$w_1X_1 + w_2X_2 + b > \textit{Umbral} \Rightarrow Y = 1$$

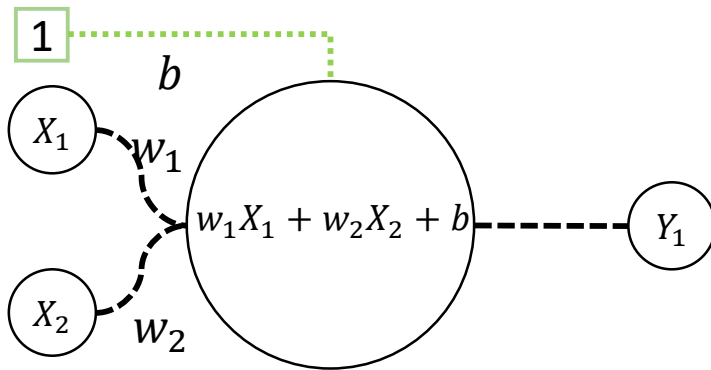
# Ejemplo

- Se puede “fusionar” el umbral en el término independiente y obtener un modelo más sencillo.
- En este caso la variable de salida sólo depende de obtener algo positivo o negativo.

$$w_1X_1 + w_2X_2 + b \leq 0 \Rightarrow Y = 0$$

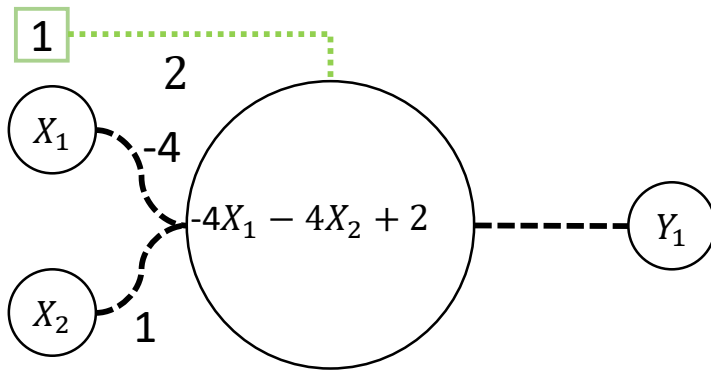
$$w_1X_1 + w_2X_2 + b > 0 \Rightarrow Y = 1$$

# Ejemplo



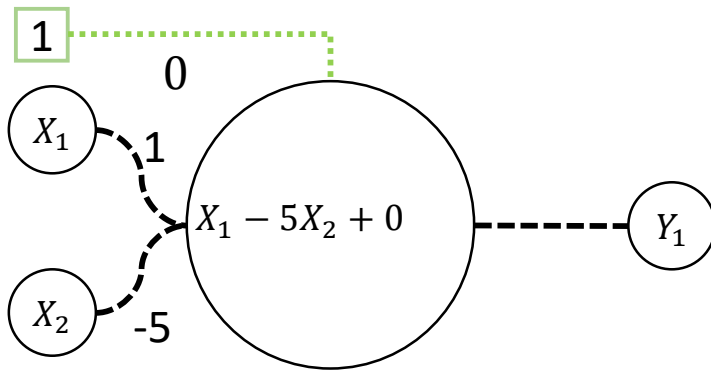
X1	X2	Objetivo	Y
0	0	0	
1	0	0	
0	1	0	
1	1	1	

# Ejemplo



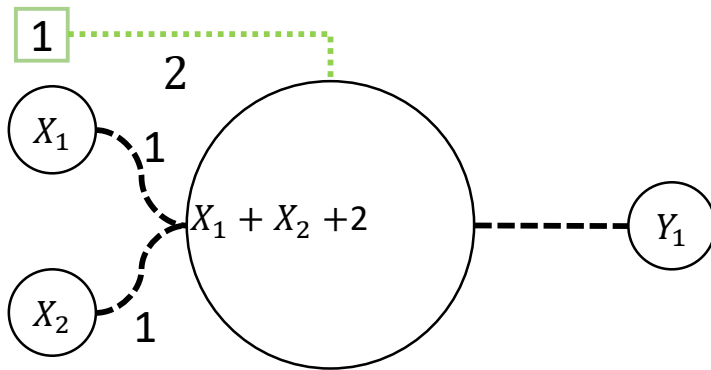
X1	X2	Objetivo	Y
0	0	0	2
1	0	0	-2
0	1	0	3
1	1	1	-2

# Ejemplo



X1	X2	Objetivo	Y
0	0	0	0
1	0	0	1
0	1	0	-5
1	1	1	-4

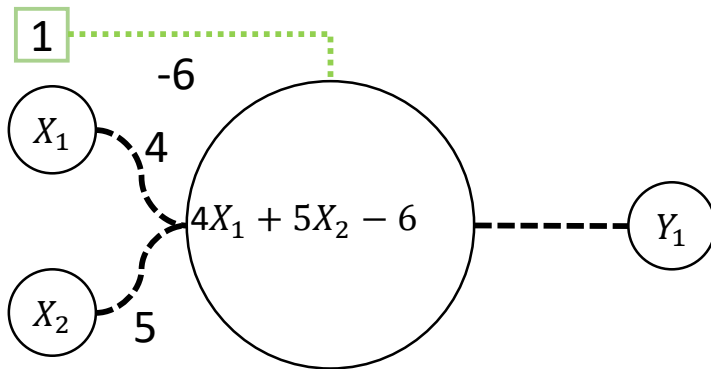
# Ejemplo



X1	X2	Objetivo	Y
0	0	0	2
1	0	0	3
0	1	0	3
1	1	1	4

# Ejemplo

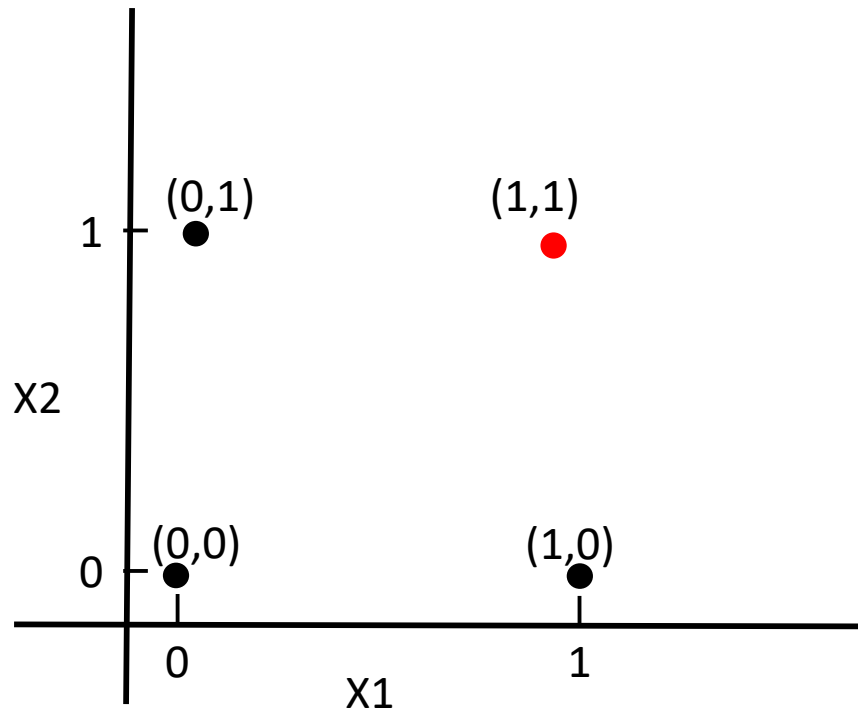
Por fin una combinación de valores que representa nuestra realidad.



X1	X2	Objetivo	Y
0	0	0	-6
1	0	0	-2
0	1	0	-1
1	1	1	3

# Ejemplo

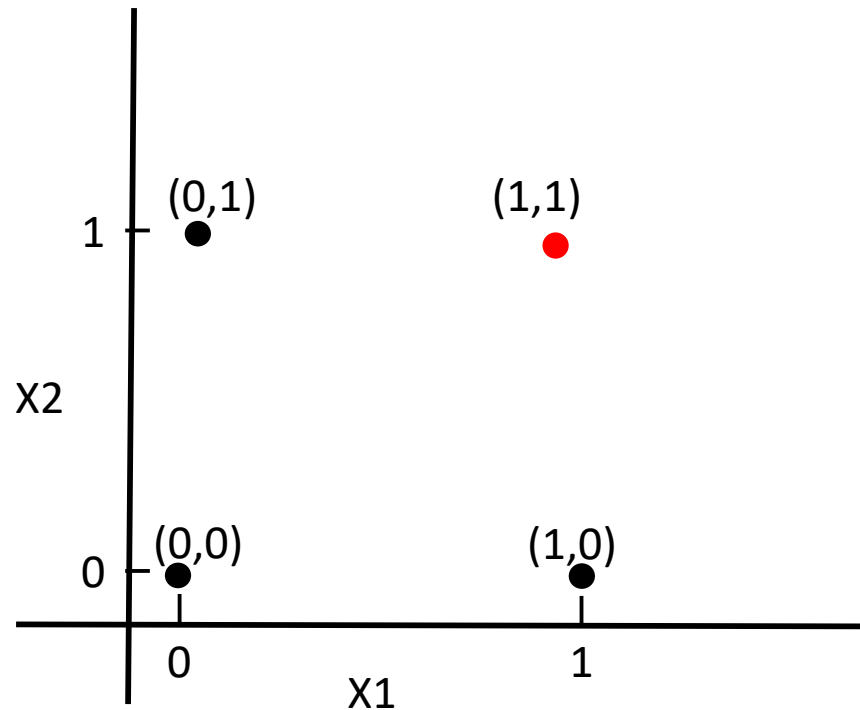
- Visualicemos la tabla





# Ejemplo

- Veamos como una recta el modelo obtenido

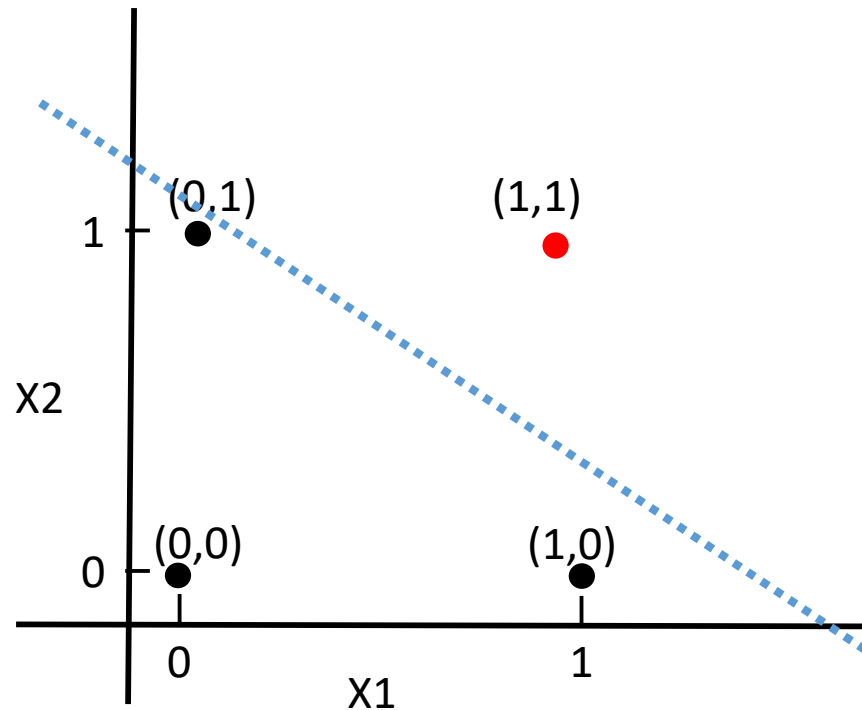


$$4X_1 + 5X_2 - 6$$

$$X_2 = \frac{-2}{3}X_1 + \frac{6}{5}$$

# Ejemplo

- La recta obtenida divide los valores en dos grupos

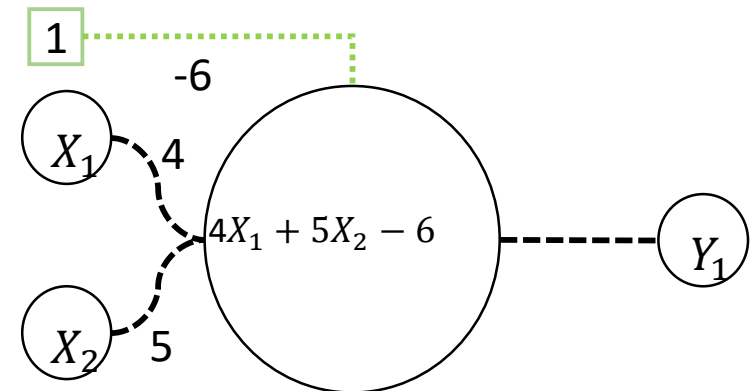
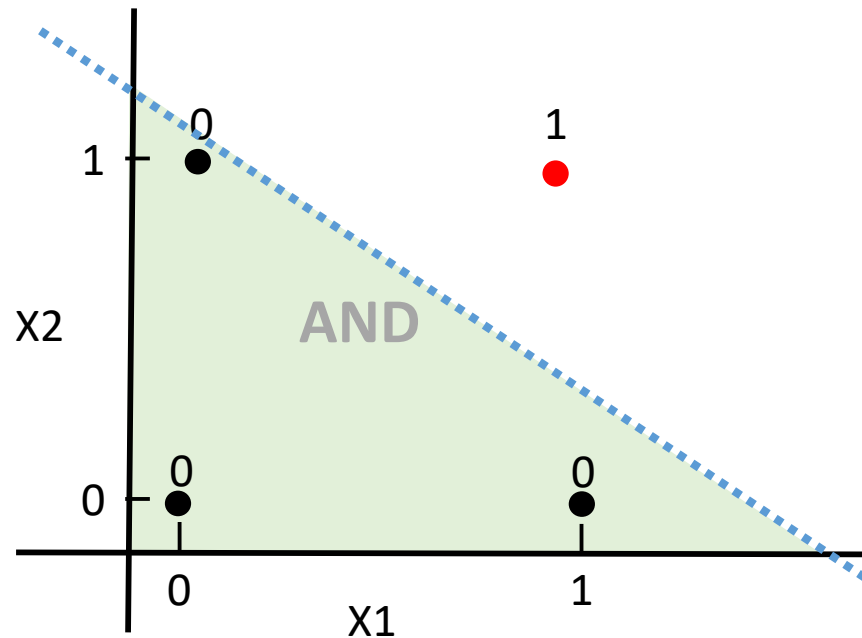


$$4X_1 + 5X_2 - 6$$

$$X_2 = \frac{-2}{3}X_1 + \frac{6}{5}$$

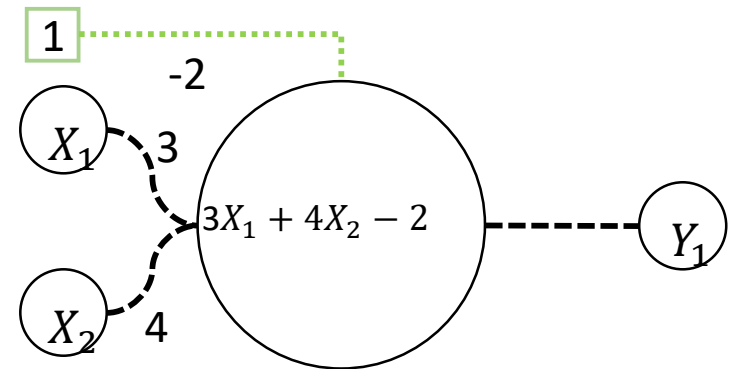
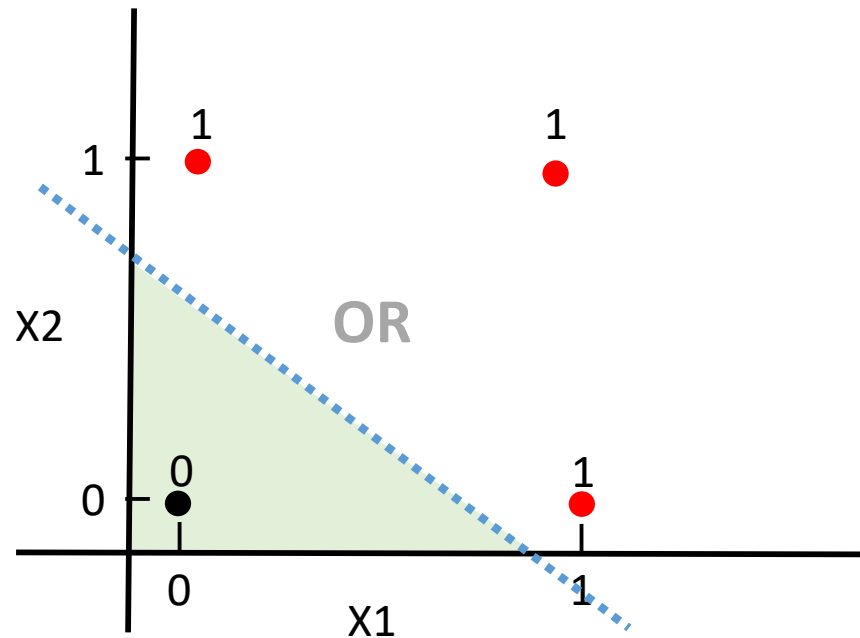
# Ejemplo

- Este ejemplo puede verse como los resultados obtenidos al usar el operador AND



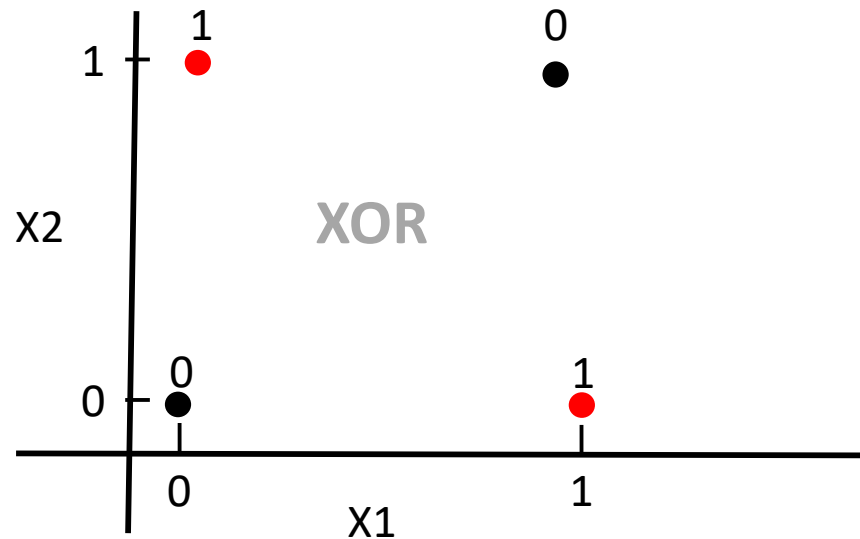
# Otro ejemplo

- Veamos que ocurre con el operador OR



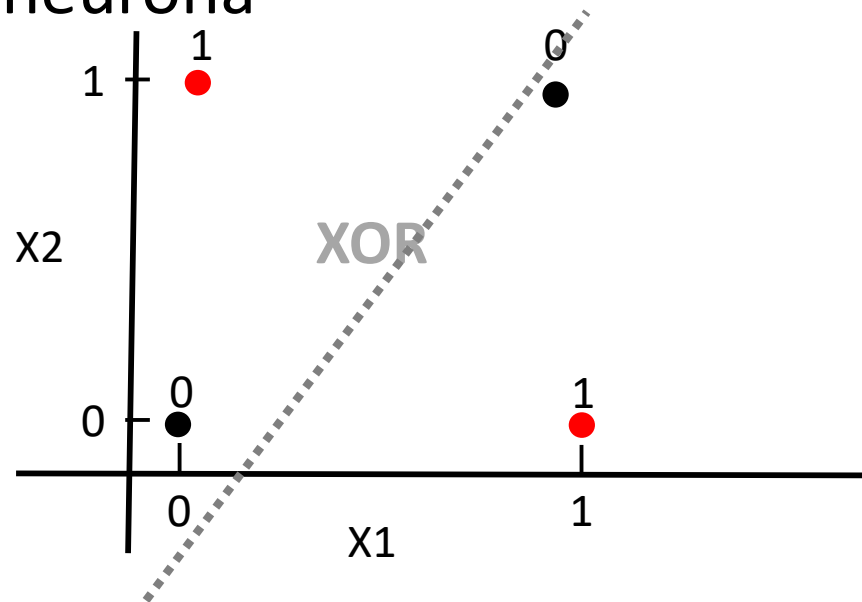
# Otro ejemplo

- Veamos que ocurre con el operador XOR
- En este caso no es posible separarlos con una sola neurona



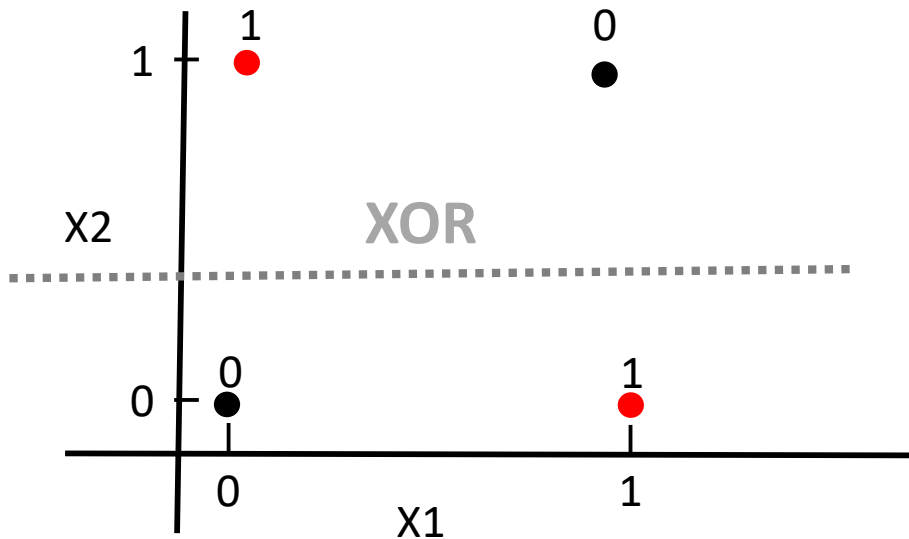
# Otro ejemplo

- Veamos que ocurre con el operador XOR
- En este caso no es posible separarlos con una sola neurona



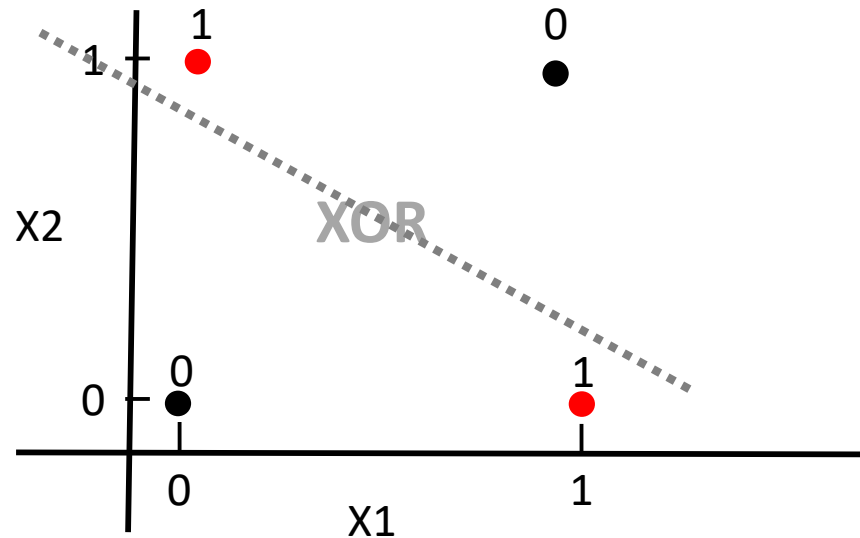
# Otro ejemplo

- Veamos que ocurre con el operador XOR
- En este caso no es posible separarlos con una sola neurona



# Otro ejemplo

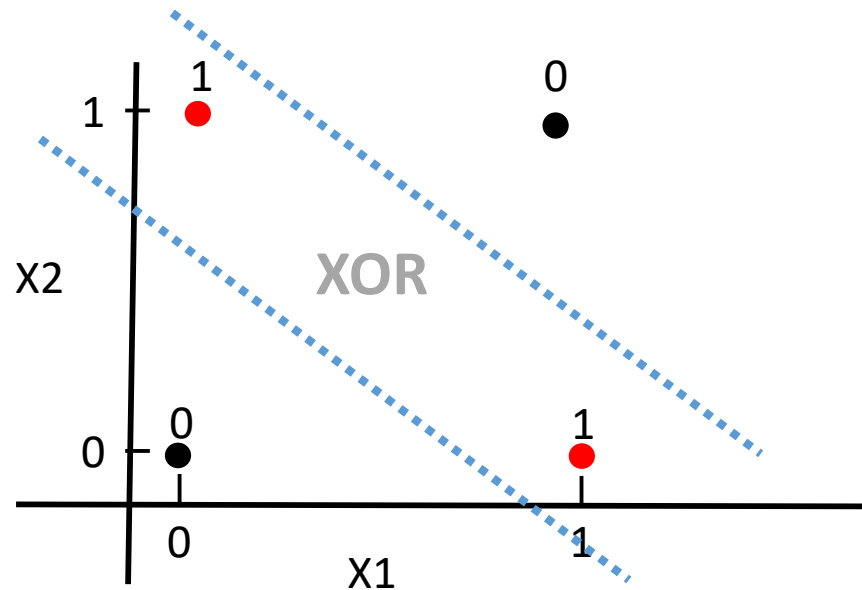
- Veamos que ocurre con el operador XOR
- En este caso no es posible separarlos con una sola neurona





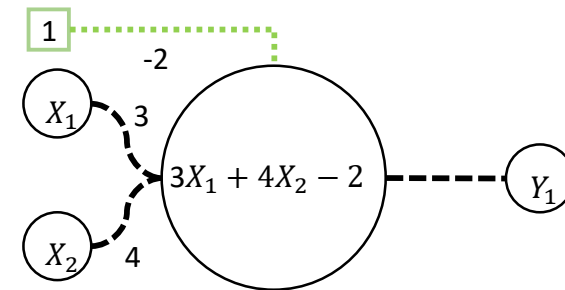
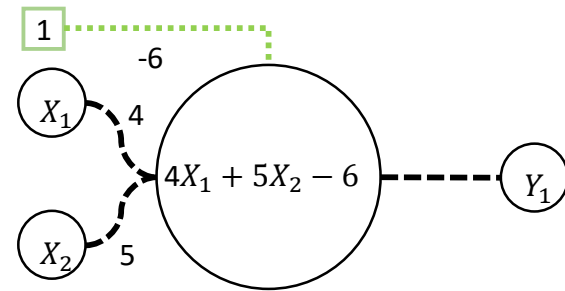
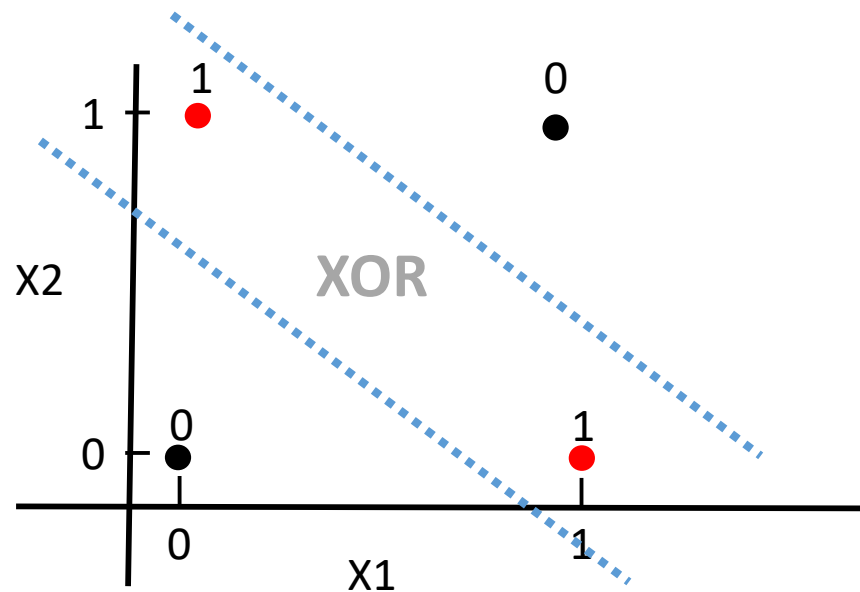
# Otro ejemplo

- Este problema es conocido desde 1969 y sugiere la necesidad de emplear varias neuronas.

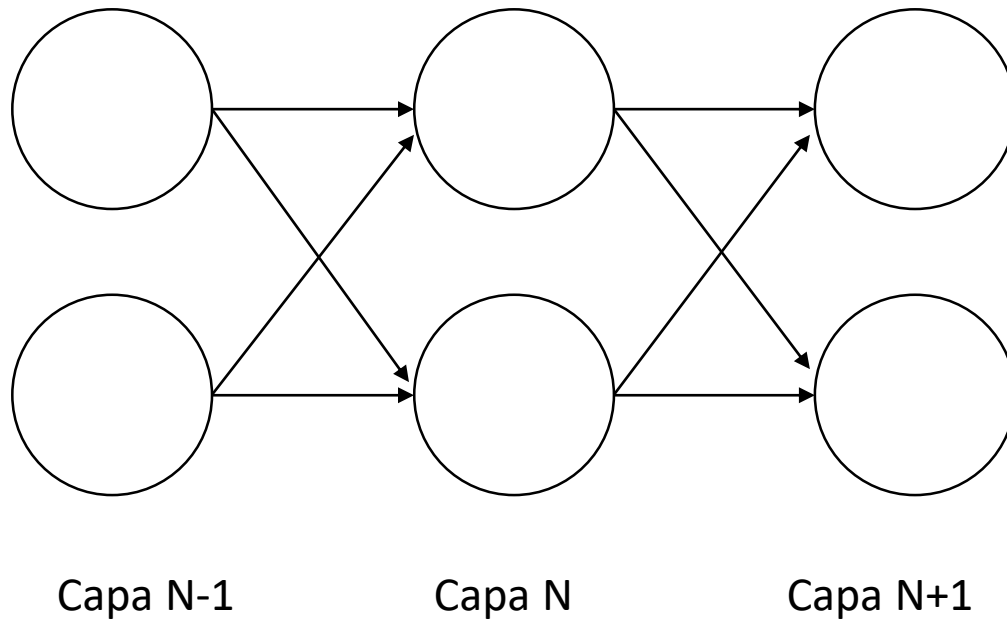


# Otro ejemplo

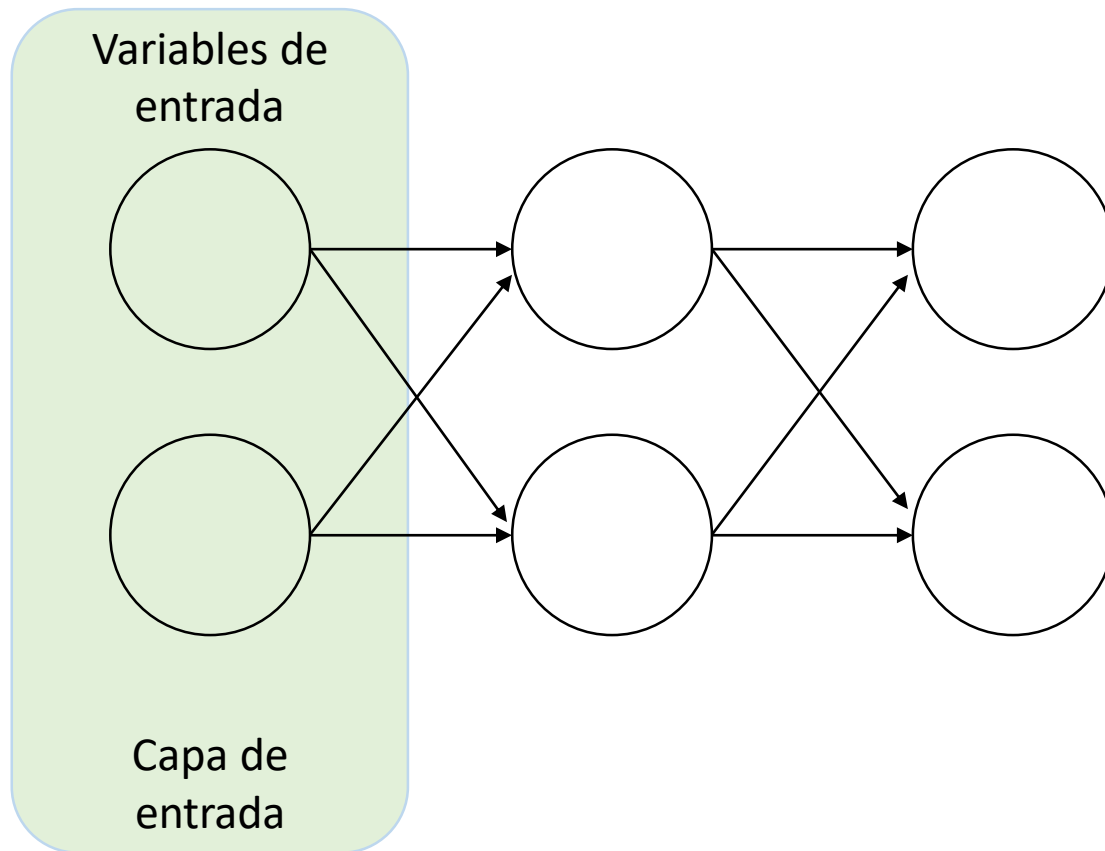
- Este problema es conocido desde 1969 e impulsa la necesidad de emplear varias neuronas.



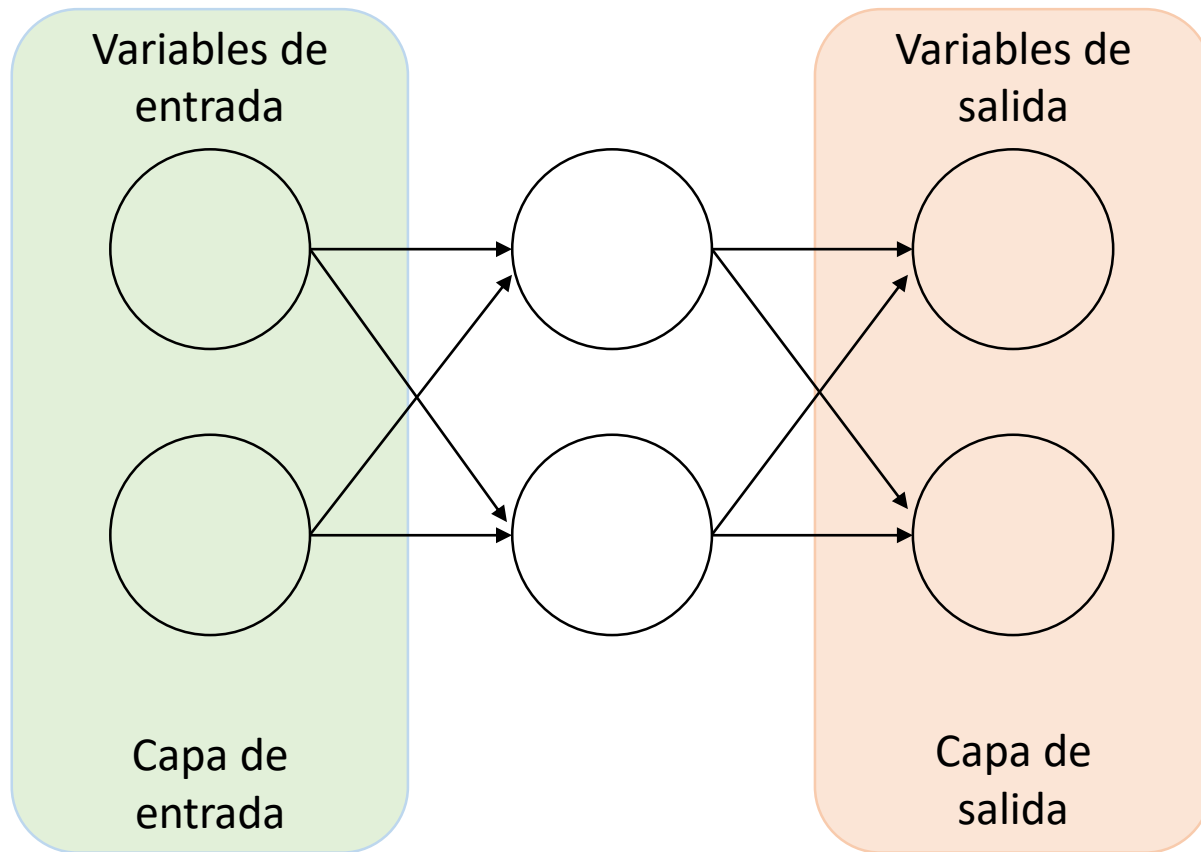
# Capas



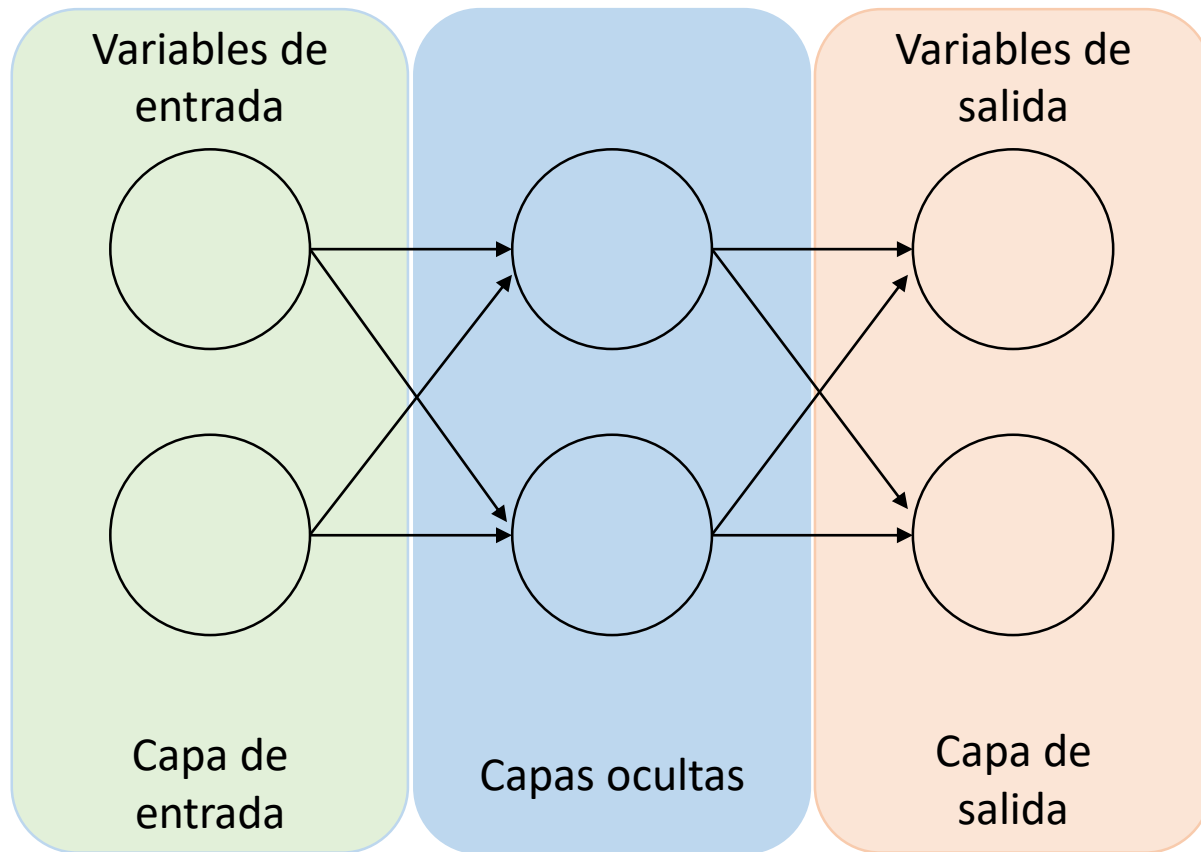
# Capas



# Capas



# Capas



# Acomodo de neuronas

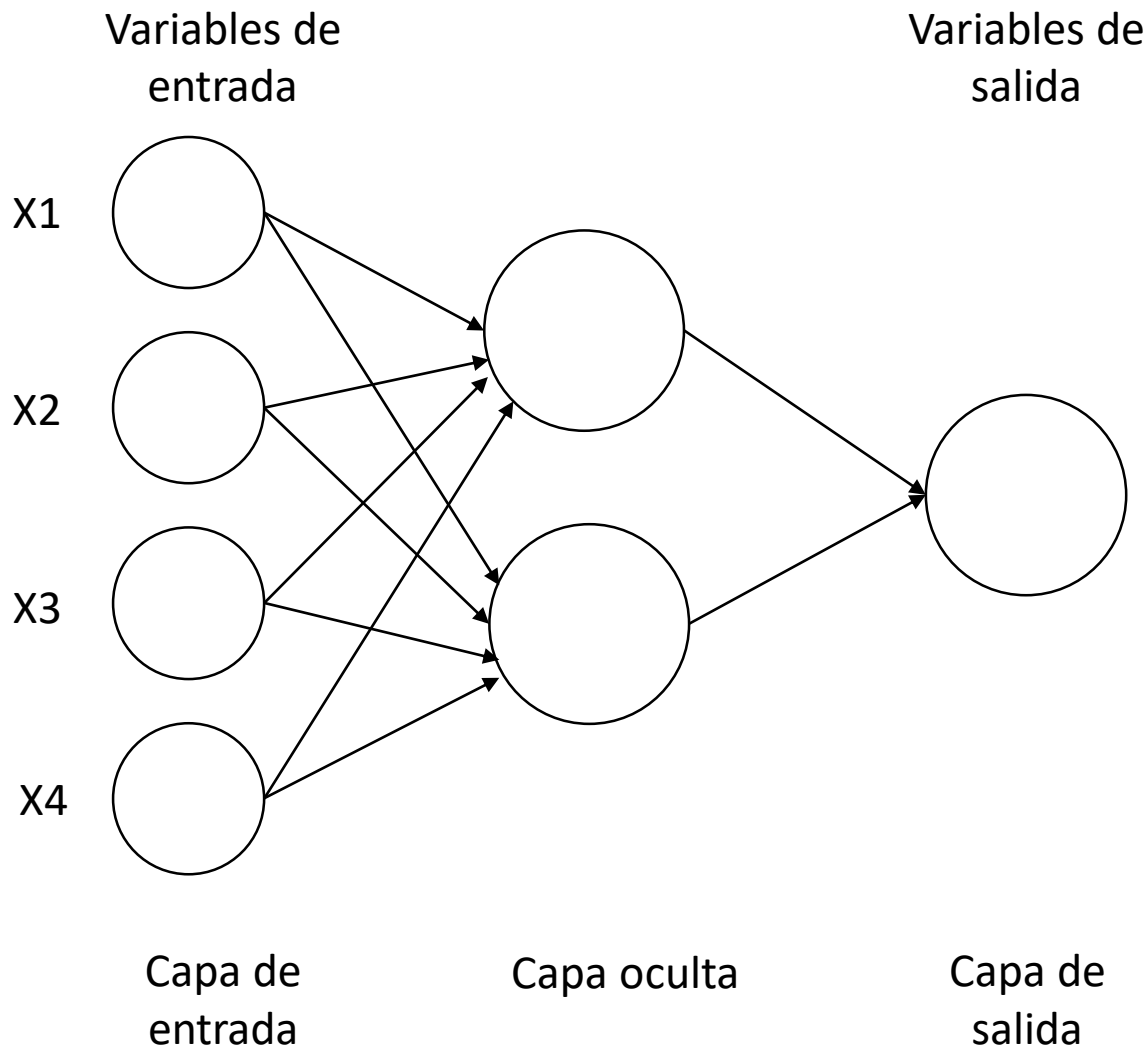
- Acomodar las neuronas en un mayor número de capas les permite tener un aprendizaje más sofisticado.
- La información es procesada y combinada múltiples veces para un mejor aprendizaje.
- De esta forma la red adquiere conocimiento jerarquizado.

# Acomodo de neuronas

- En caso de contar con mayor número de datos, podemos incluir más neuronas en la capa de entrada.
- Cada neurona en la misma capa puede especializarse en un objetivo diferente.
- Mientras más capas se incluyan, más complejo puede ser el conocimiento que se elabora.



# Capas

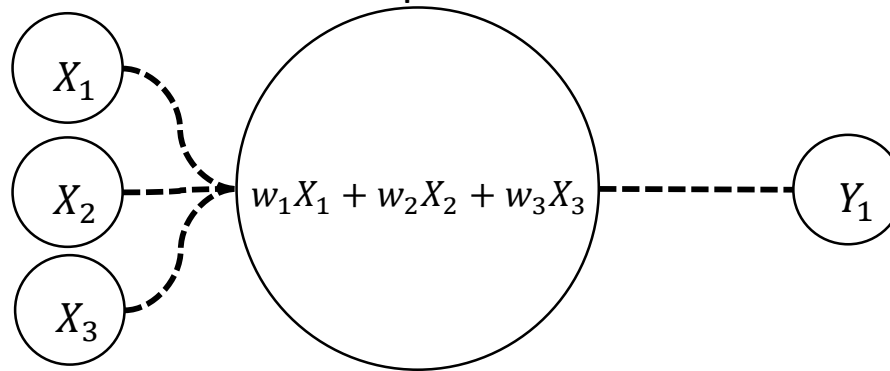


# Función de activación

- Lamentablemente, las redes construidas hasta ahora son enlaces entre transformaciones lineales.
- Cualquier suma de transformaciones lineales sigue siendo lineal.
- Por lo tanto, la red de la diapositiva anterior se puede resumir a una red con una sola neurona en la capa interior.
- Es necesario incluir perturbaciones que permitan simular procesos no lineales.

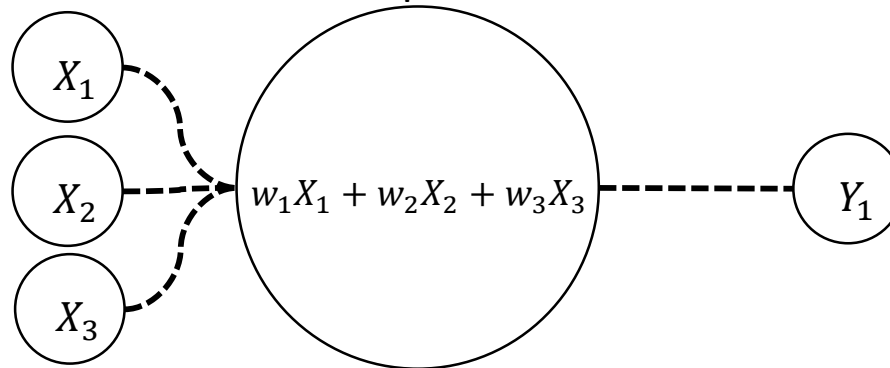
# Función de activación

Originalmente nuestras neuronas solo producen relaciones lineales

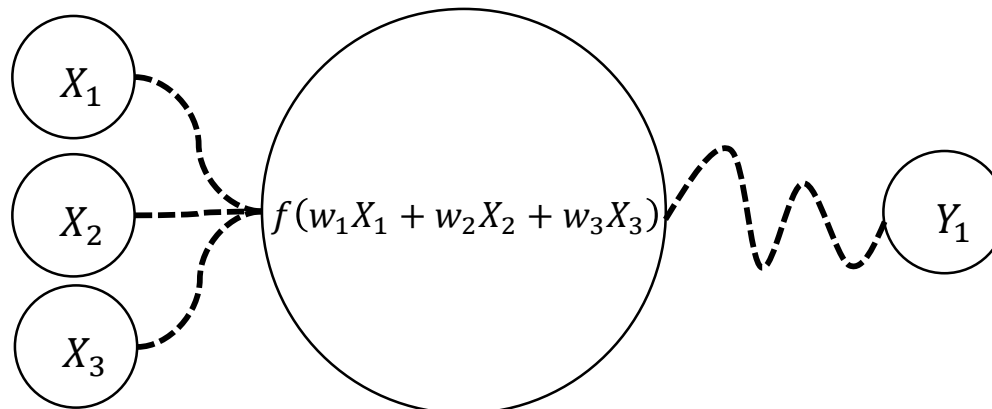


# Función de activación

Originalmente nuestras neuronas solo producen relaciones lineales

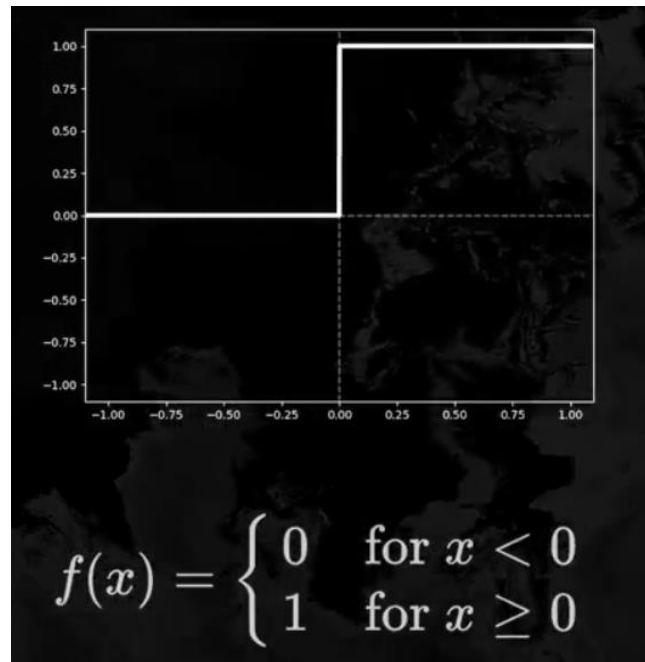


Vamos a introducir funciones que les permitan producir relaciones no lineales



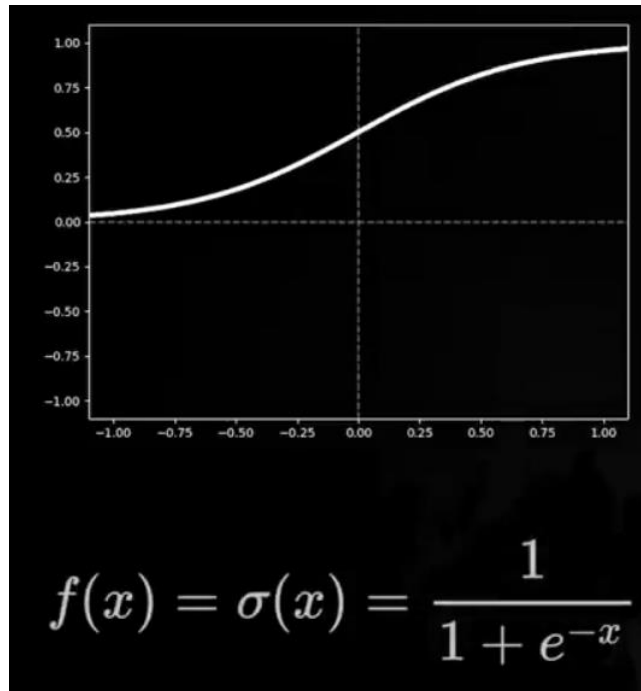
# Función de activación

- Función escalonada
- No favorece el aprendizaje, por lo cual no será usada en el futuro



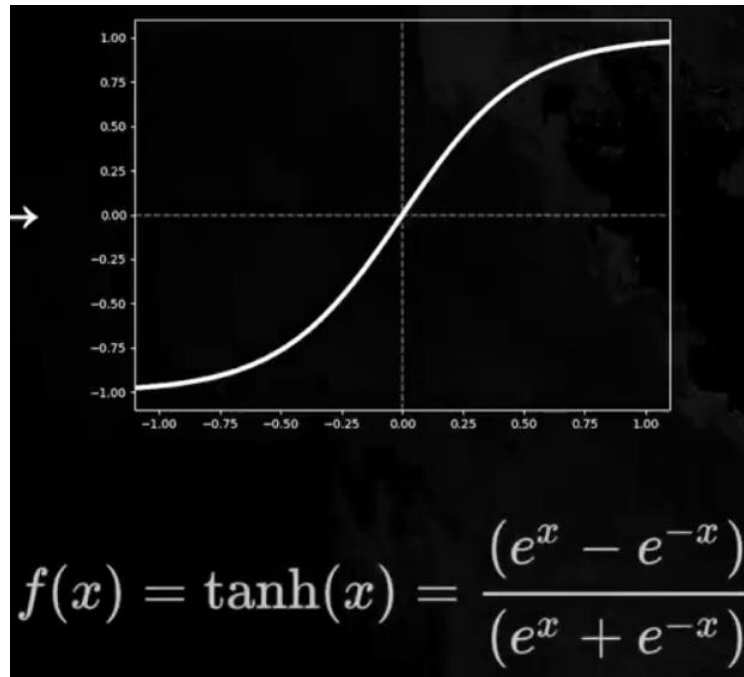
# Función de activación

- Función sigmoide
- Valores muy grandes se saturan en 1 y valores muy pequeños se saturan en 0



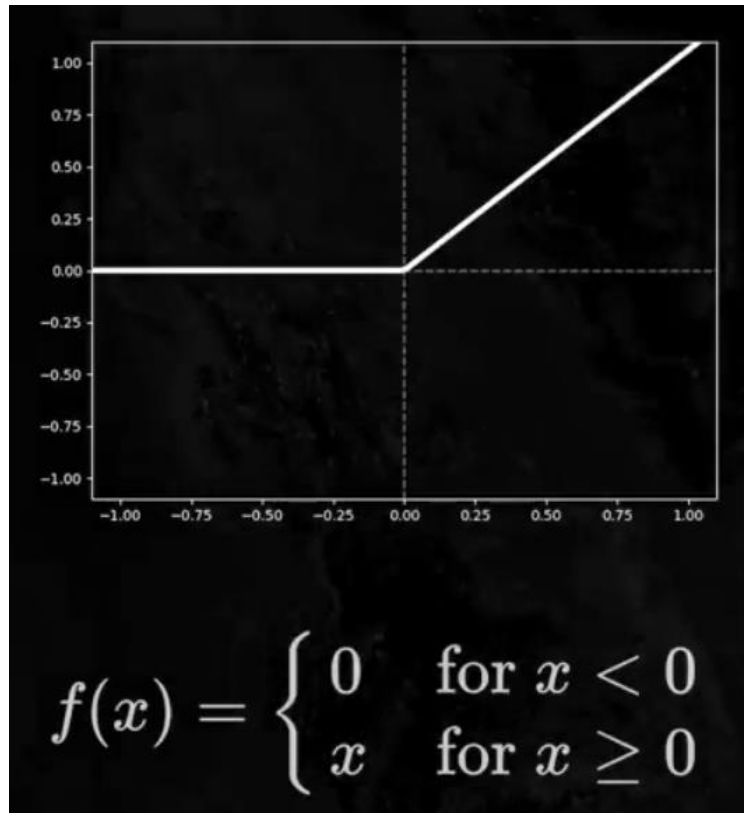
# Función de activación

- Función tangente hiperbólica
- Valores muy grandes se saturan en 1 y valores muy pequeños se saturan en -1



# Función de activación

- Función Relu (unidad rectificada lineal)



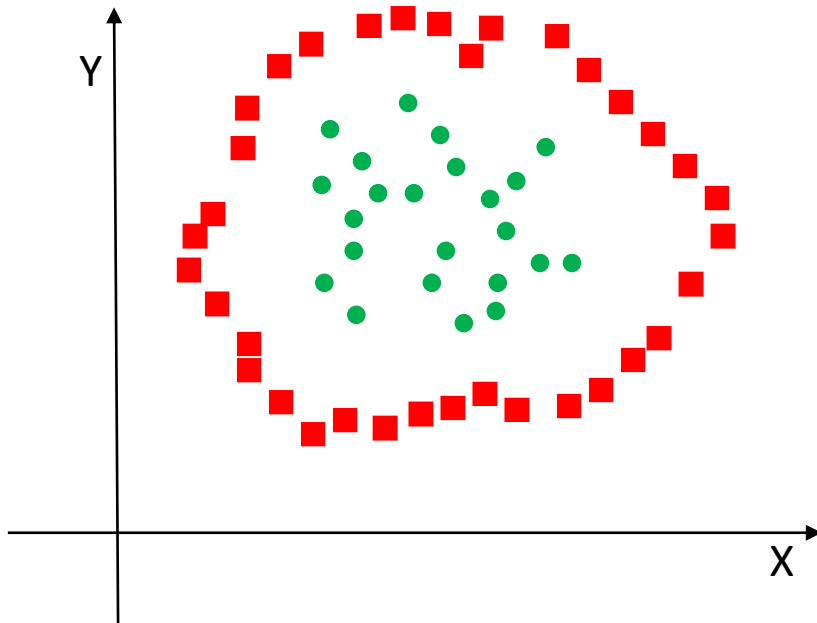


# Función de activación

- Ahora si, al encadenar varias neuronas con sus funciones de activación, podemos generar comportamientos no lineales.
- Por lo tanto, la red es más potente que el uso de una sola neurona.

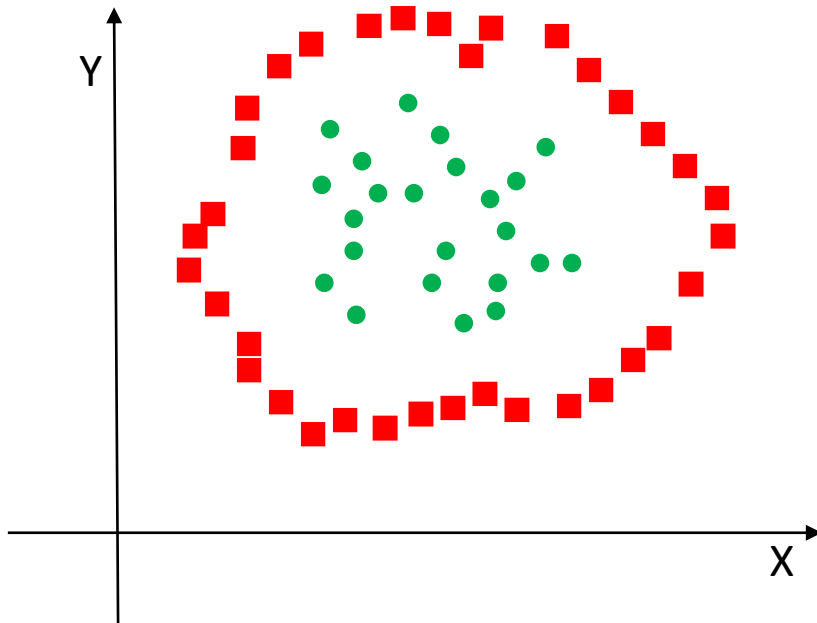
# Ejemplo

- Supongan que queremos generar una división de los puntos verdes con los rojos mediante una red neuronal



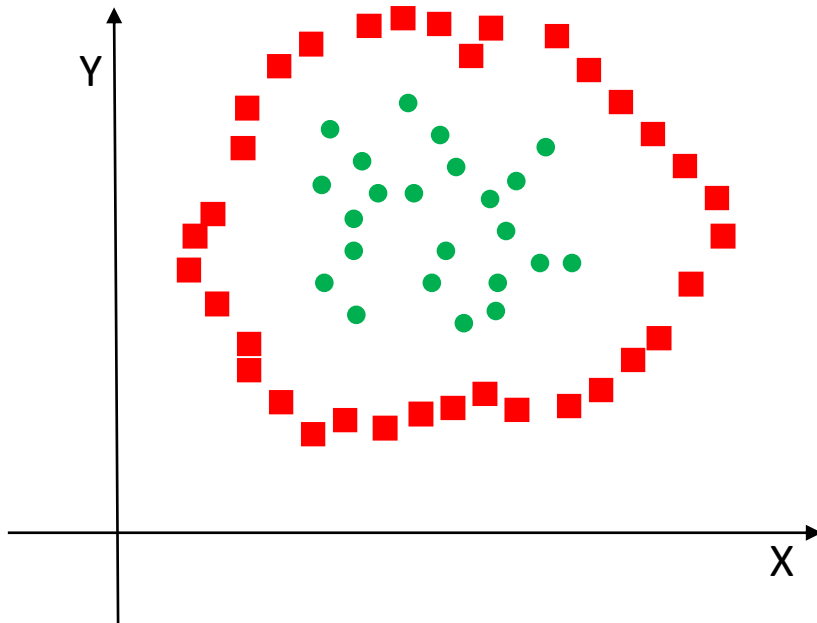
# Ejemplo

- Para este ejemplo vamos a usar una función de activación sigmoide.
- Veamos la gráfica con diferentes parámetros.



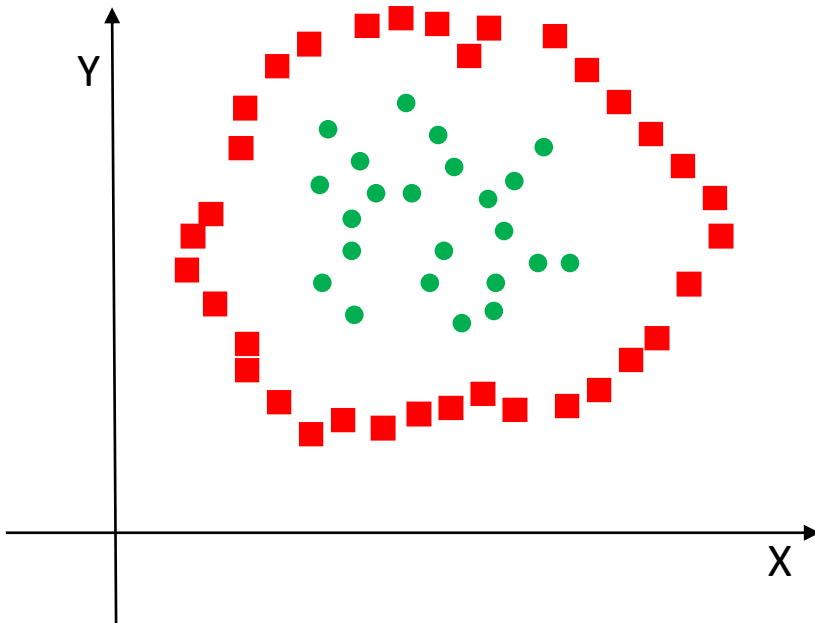
# Ejemplo

- Observa como la región de activación cambia al jugar con diferentes parámetros para  $X$  y  $Y$ .



# Ejemplo

- Ahora vamos a usar 4 neuronas, cada una con la función de activación sigmoide.



# Ejemplo

- Por último, podemos buscar los parámetros adecuados.

