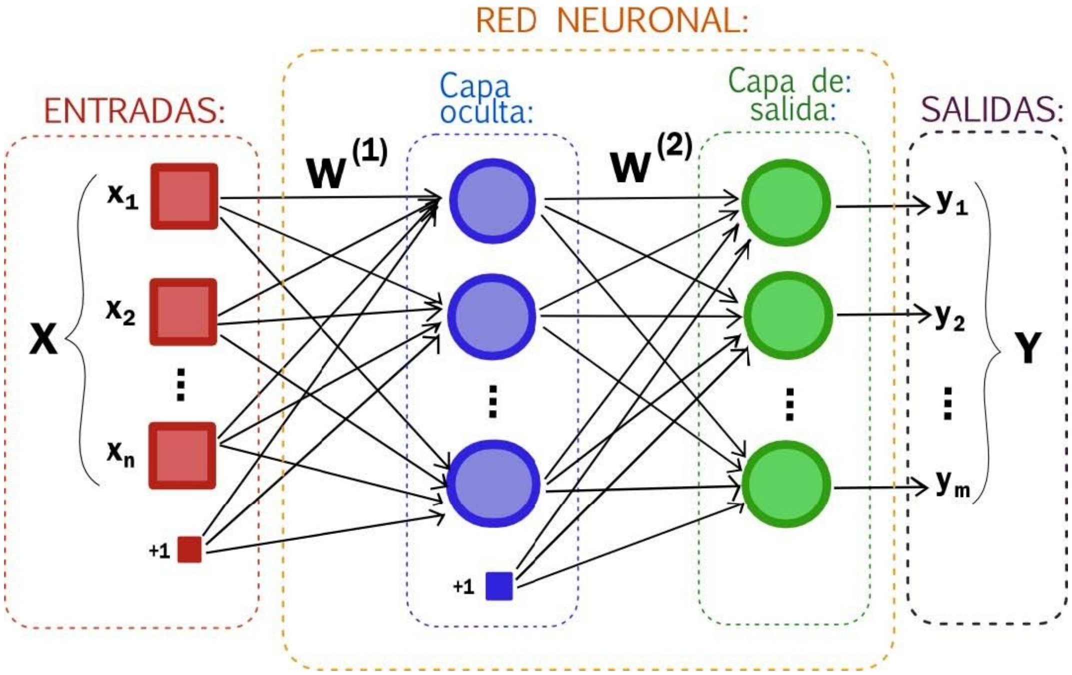
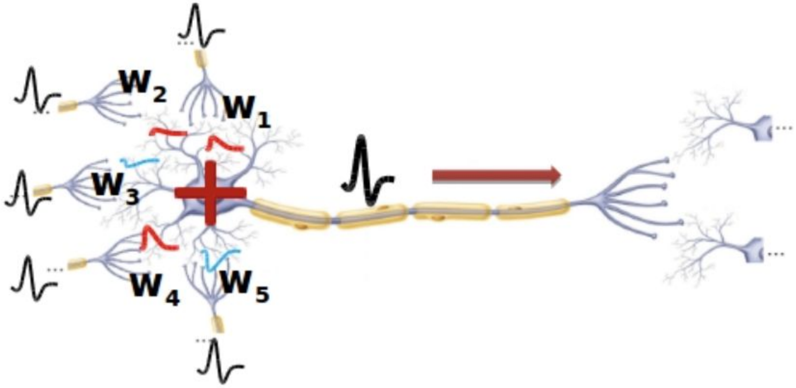
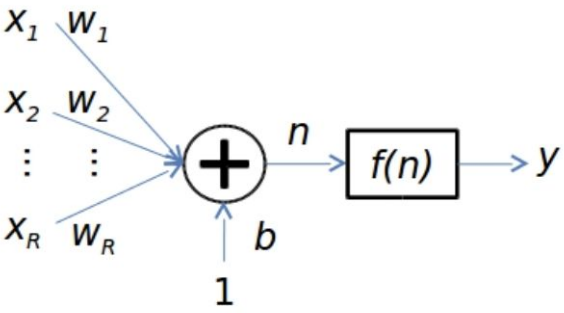
Redes neuronales



¿Qué es un modelo neuronal?

Una red neuronal artificial (ANN) es un modelo matemático que sirve para la toma de decisiones de manera probabilística, supuestamente basado en el comportamiento de las neuronas cerebrales humanas.





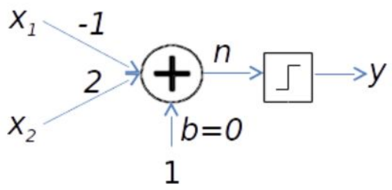
$$y = f(\boldsymbol{w}^T \boldsymbol{x} + b)$$

$\boldsymbol{x}$	vector de entrada
$\boldsymbol{w}$	vector de pesos sinápticos
b	polarización
$f(\cdot)$	función de activación
y	sálida

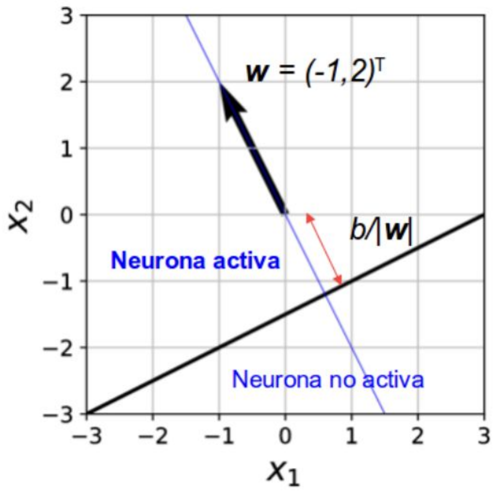
$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_R \end{bmatrix}$$

$$\boldsymbol{w} = \begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_R \end{bmatrix}$$

Interpretación Geométrica - Un ejemplo en 2D

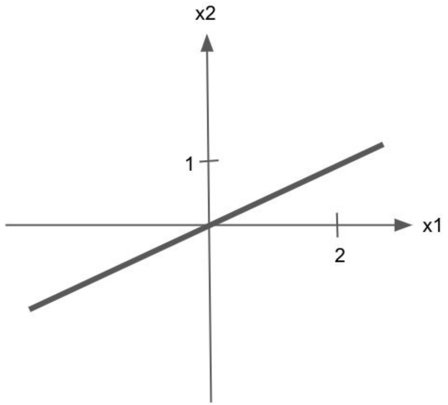


$$n = -x_1 + 2x_2 \qquad f(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$



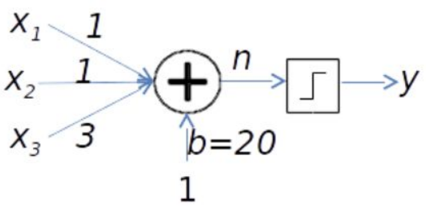
¿Qué figura geométrica forman todos los pares de valores (x1 y x2) para n = 0?

$$n = -x_1 + 2x_2 = 0$$ Es una línea recta



Conclusión: cada neurona perceptrón separa el espacio de entrada en 2 regiones (así es como clasifica).

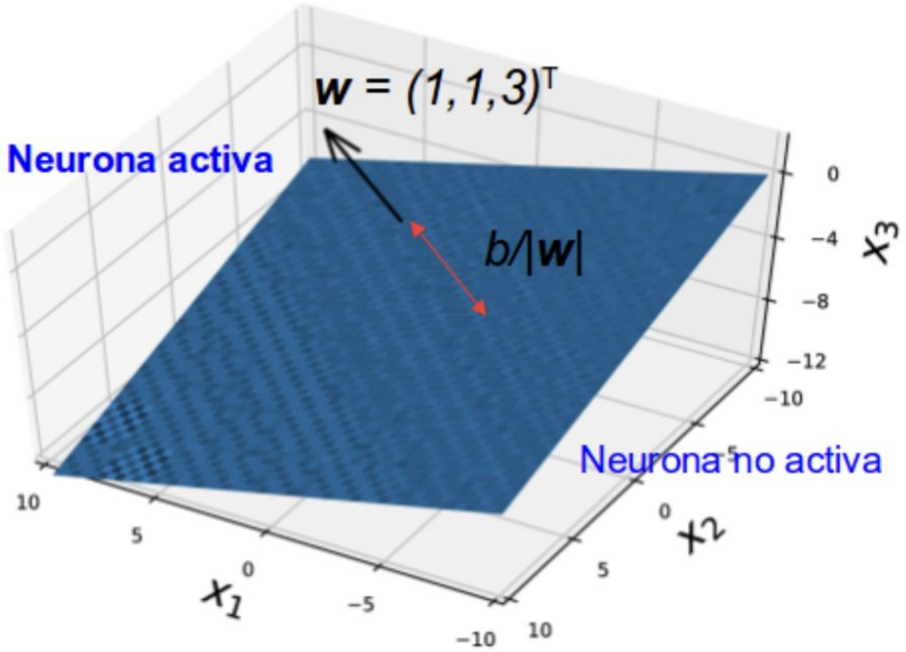
Interpretación Geométrica - Un ejemplo en 3D



¿Qué figura forman las tripletas de valores (x1,x2,x3) que hacen que n = 0?

$$n = x_1 + x_2 + 3x_3 + 20 = 0$$

Es un plano en el espacio 3D



Interpretación Geométrica - Caso general

En general, para una neurona perceptrón de R entradas
¿Qué figura forman las R-pletas (x1,x2,...,xR) que hacen que n = 0 ?

$$n = \boldsymbol{w}^T \boldsymbol{x} + b = 0$$

Un hiperplano ... que separa el espacio de entrada de R dimensiones en 2 regiones.

w y b son llamados parámetros de aprendizaje

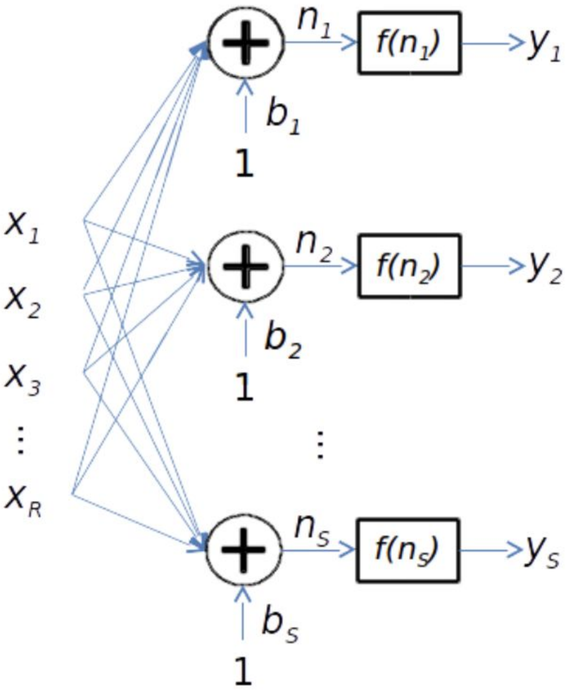
Las operaciones Prod. punto + Real es llamada Transformación Afín

Conclusión: una neurona perceptron puede separar (clasificar datos) el espacio de entrada en dos regiones, sin importar el número de entradas que tenga.

UNIVERSIDAD Panamericana

Si quieres clasificar más de dos clases ¿Qué harías?

Capa perceptron



$$y_1 = f(\boldsymbol{w}_1^T \boldsymbol{x} + b_1)$$

$$y_2 = f(\boldsymbol{w}_2^T \boldsymbol{x} + b_2)$$

$$\vdots$$

$$y_S = f(\boldsymbol{w}_S^T \boldsymbol{x} + b_S)$$

$\boldsymbol{y}$ $\boldsymbol{W}$ $\boldsymbol{b}$

$$\boldsymbol{y} = f(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b})$$

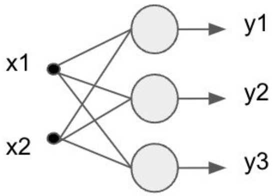
¿Cuántos hiperplanos hay?

S hiperplanos

¿Cuántos parámetros de aprendizaje hay?

R*S pesos sinápticos + S polarizaciones

Interpretación Geométrica - Un ejemplo de capa perceptrón con 3 neuronas y 2 entradas



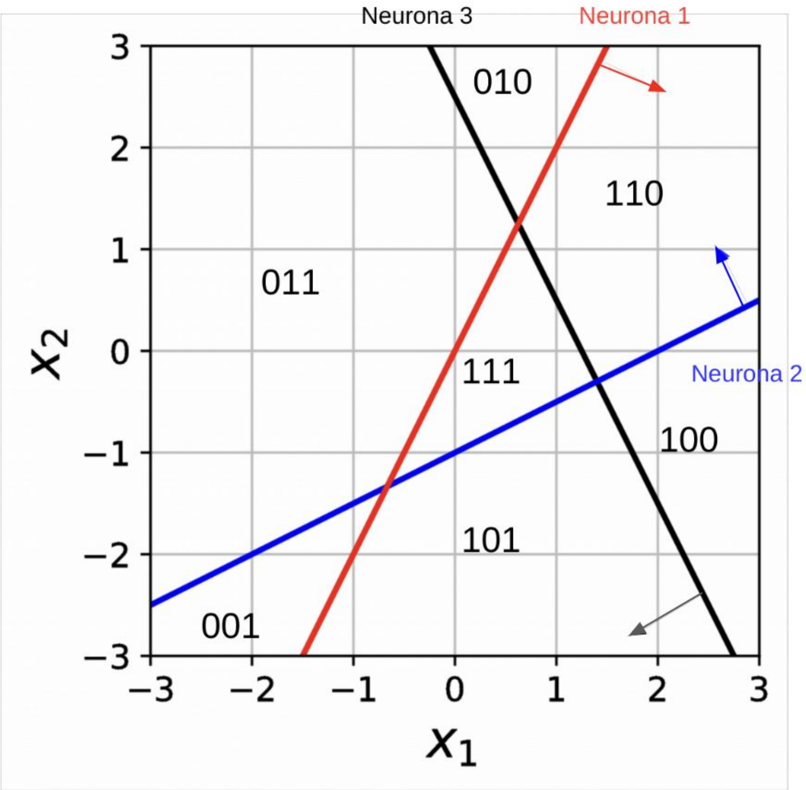
¿Cuántas clases podría separar?

Hasta 8 clases, pero debido a que hay solo 2 entradas, solamente puede 7 clases.

¿Cuántas clases podría separar una capa perceptrón con S neuronas?

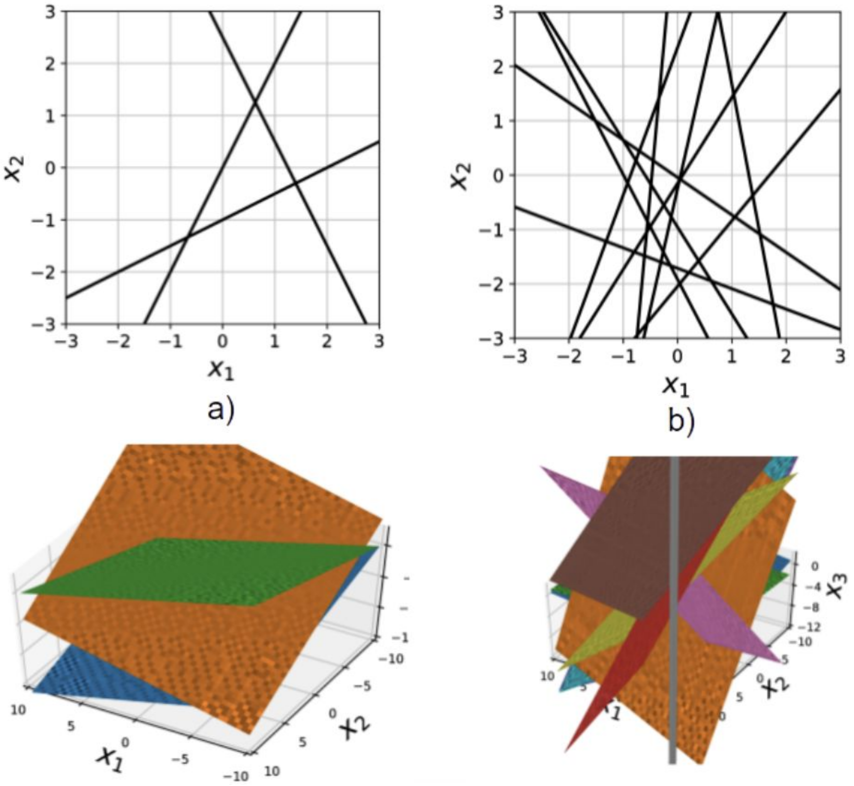
$$regiones \leq 2^S$$

Conclusión: Una capa perceptron tiene más poder para segmentar el espacio de entrada (en hasta 2^S clases)

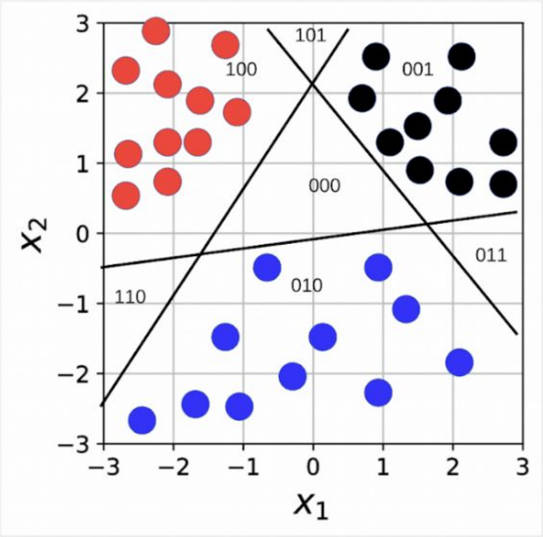


Interpretación Geométrica - Ejemplos de capa perceptrón en 2D y 3D

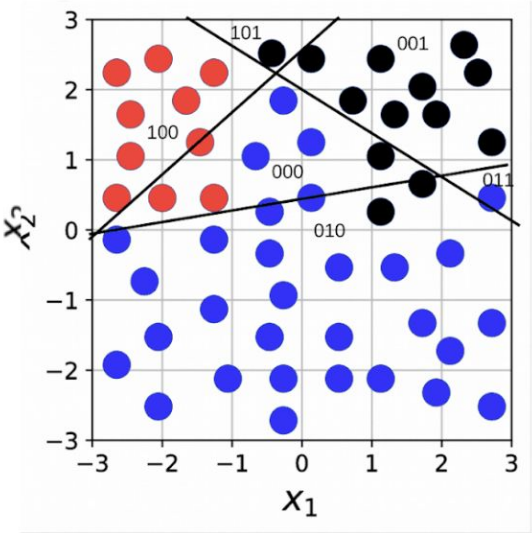
$$\boldsymbol{x} \rightarrow \boxed{\boldsymbol{y} = f(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b})} \rightarrow \boldsymbol{y}$$



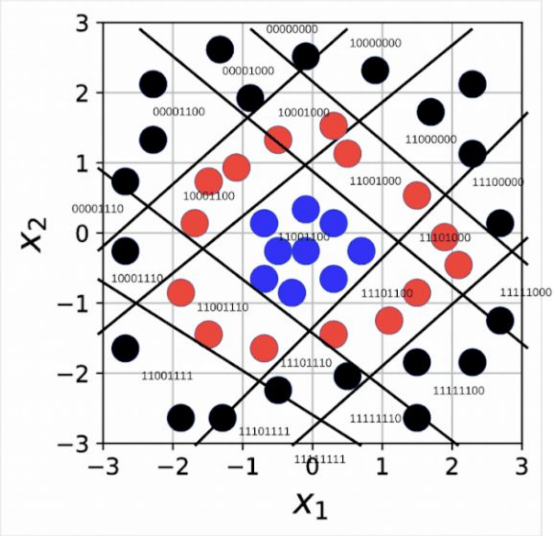
Y si quieres clasificar datos que no puede separarse con hiperplanos, ¿Qué harías?



a)



b)



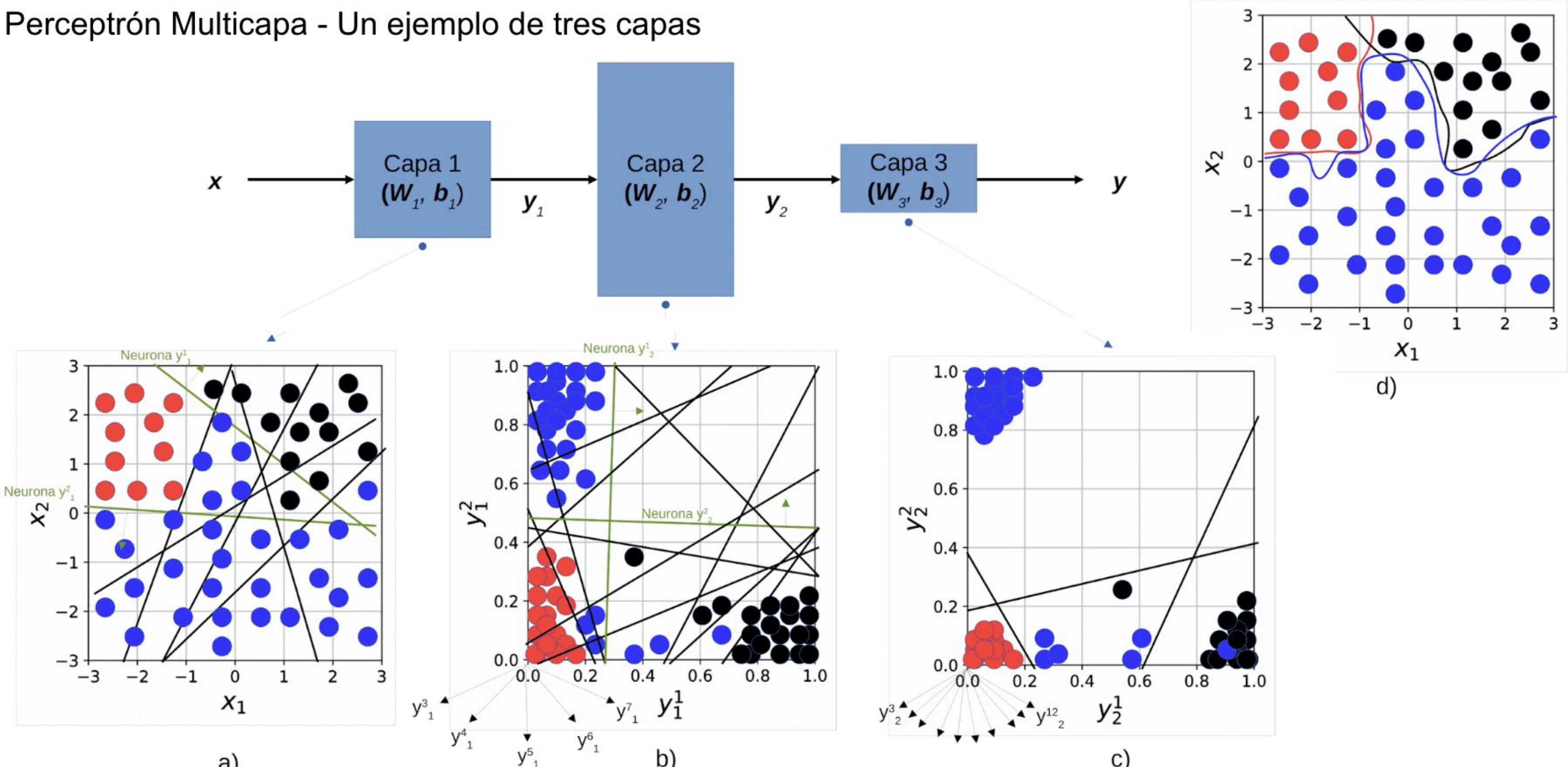
c)

Observación: Una capa perceptrón está limitada a resolver problemas con separabilidad lineal.

Solución: usar una primer capa para segmentar en subclases y usar un segunda capa para REAGRUPAR estas subclases en las tres clases.

Conclusión: uniendo dos o más capas en serie, se puede resolver problemas sin separabilidad lineal.

Perceptrón Multicapa - Un ejemplo de tres capas



a)

Espacio de entrada de la capa 1
tiene 2 dimensiones y 7 hiperplanos

b)

Espacio de entrada de la capa 2
tiene 7 dimensiones y 12 hiperplanos

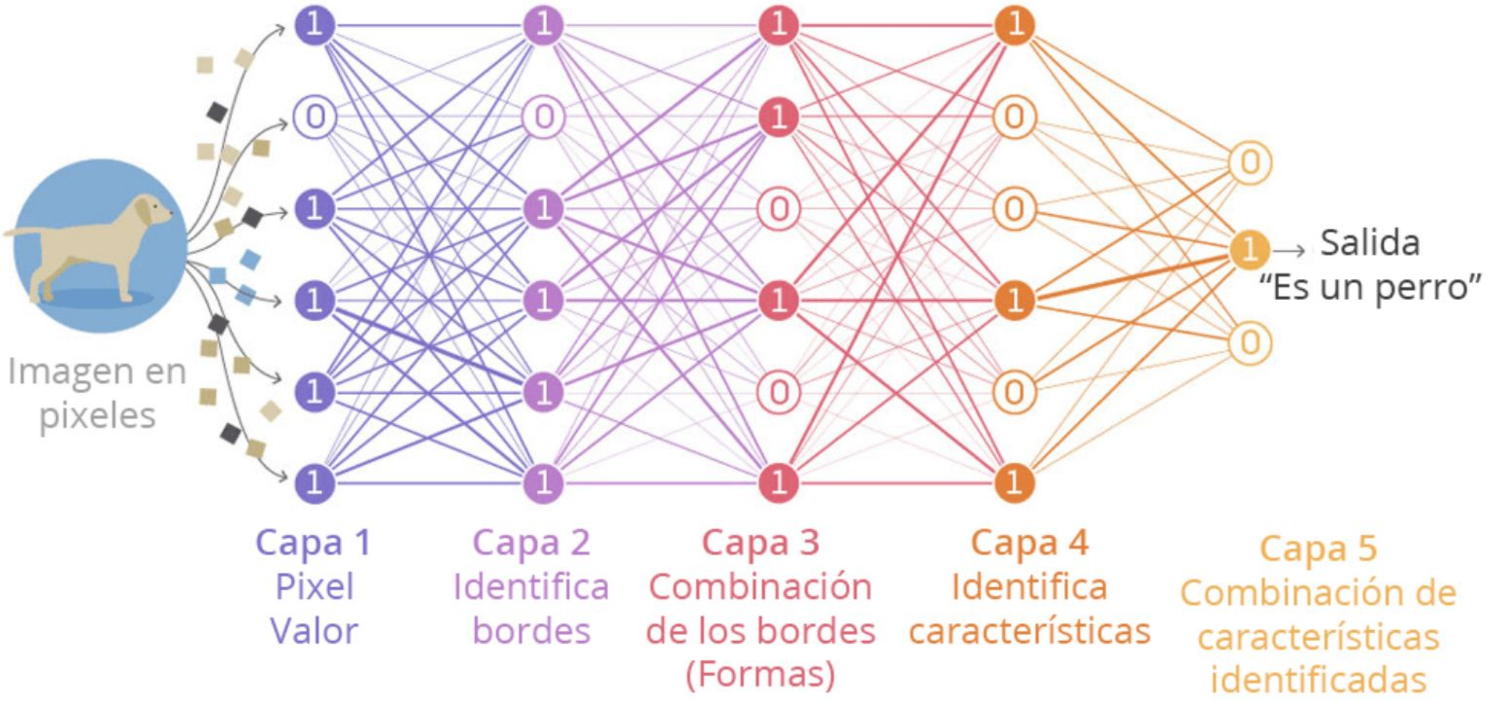
c)

Espacio entrada de la capa 3
tiene 12 dimensiones y 3 hiperplanos

d)

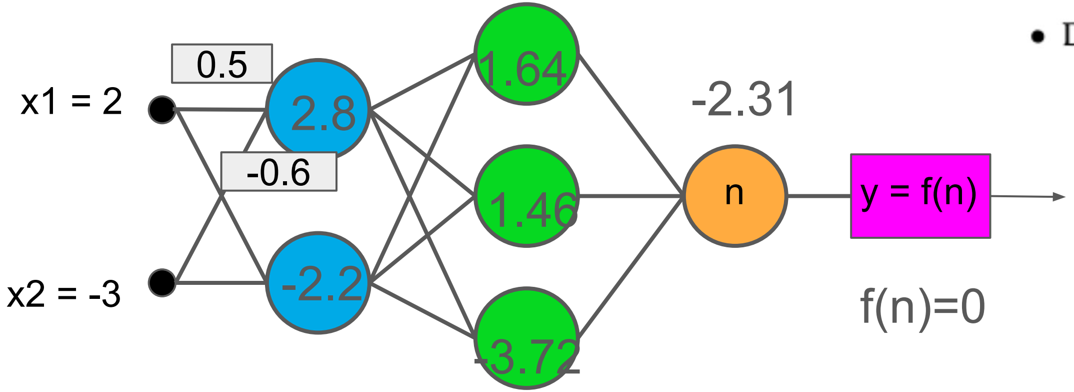
Entonces… ¿Qué es Deep Learning?

Es un modelo neuronal artificial que contiene **varias capas de profundidad** dedicadas a funciones específicas o a extracción de características a diferentes niveles cognitivos.



Ejercicio 1:

Obtener la salida “n” y la salida “y” del siguiente modelo neuronal:



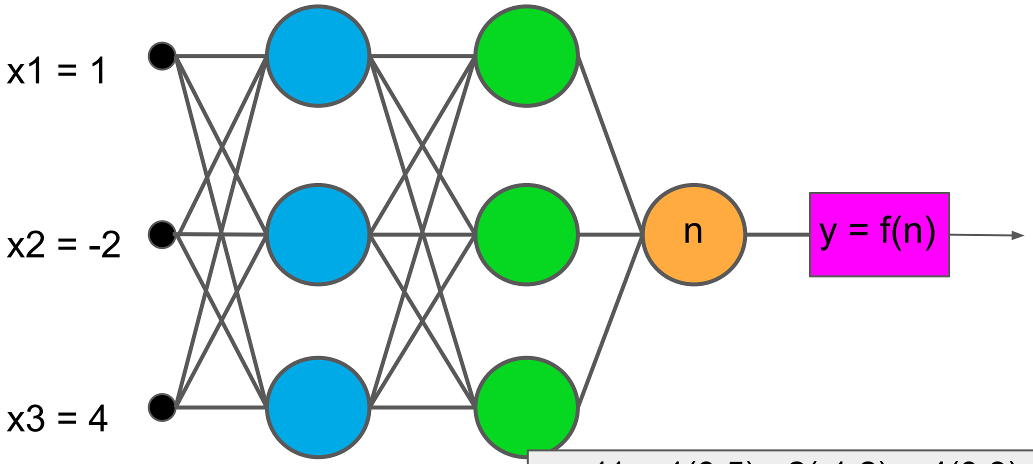
Asumamos los siguientes pesos para la red:

- Desde la entrada a la primera capa oculta:
  - Neurona 1: $w_{11} = 0.5, \quad w_{12} = -0.6$
  - Neurona 2: $w_{21} = 0.1, \quad w_{22} = 0.8$
- Desde la primera capa oculta a la segunda capa oculta:
  - Neurona 1: $u_{11} = 0.4, \quad u_{12} = -0.2$
  - Neurona 2: $u_{21} = 0.6, \quad u_{22} = 0.1$
  - Neurona 3: $u_{31} = -0.7, \quad u_{32} = 0.8$
- Desde la segunda capa oculta a la neurona de salida:
  - $v_1 = 0.3, \quad v_2 = -0.9, \quad v_3 = 0.4$

$$f(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$

Ejercicio 2:

Obtener la salida "n" y la salida "y" del siguiente modelo neuronal:
(Asumir que b en todas las neuronas es igual a 0)



Asumamos los siguientes pesos para la red:

- Desde la entrada a la primera capa oculta:
  - Neurona 1: $w_{111} = 0.5, w_{112} = -1.2, w_{113} = 0.3$
  - Neurona 2: $w_{121} = 0.8, w_{122} = 0.1, w_{123} = -0.4$
  - Neurona 3: $w_{131} = -0.5, w_{132} = 0.7, w_{133} = 1.0$
- Desde la primera capa oculta a la segunda capa oculta:
  - Neurona 1: $w_{211} = 0.6, w_{212} = -0.3, w_{213} = 0.2$
  - Neurona 2: $w_{221} = -0.7, w_{222} = 0.4, w_{223} = 0.8$
  - Neurona 3: $w_{231} = 0.1, w_{232} = 0.5, w_{233} = -0.6$
- Desde la segunda capa oculta a la neurona de salida:
  - $w_{31} = 0.9, w_{32} = -0.2, w_{33} = 0.5$

$$f(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$

n11 = 1(0.5) - 2(-1.2) + 4(0.3) =4.1
n12 = 1(0.8) -2(0.1) + 4(-0.4) = -1
n13 = 1(-0.5) -2(0.7) + 4(1) = 2.1

n21 = 4.1(0.6) -1(-0.3) + 2.1(0.2)

UNIVERSIDAD Panamericana

¿Qué es la función de activación?

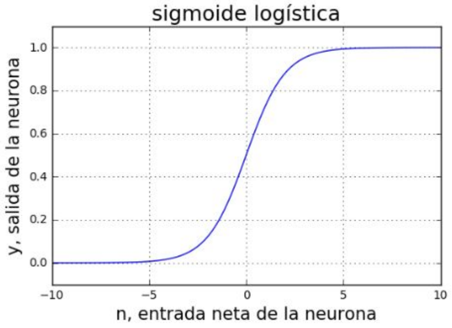
Función matemática que permite obtener un resultado con base en la entrada a partir de una ecuación específica. Existen las siguientes familias:

1. Familia de funciones sigmoidales

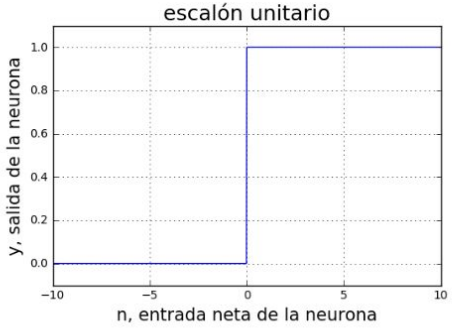
1. Familia de funciones de rectificación

1. Funciones especiales

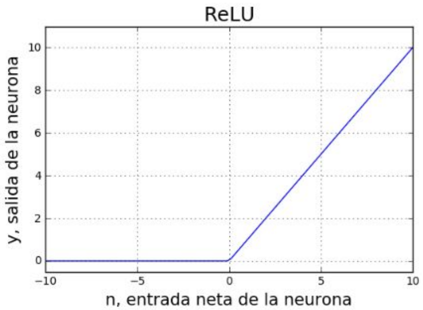
1. Capas de activación



$$f(n) = \frac{1}{1 + \exp(-n)}$$

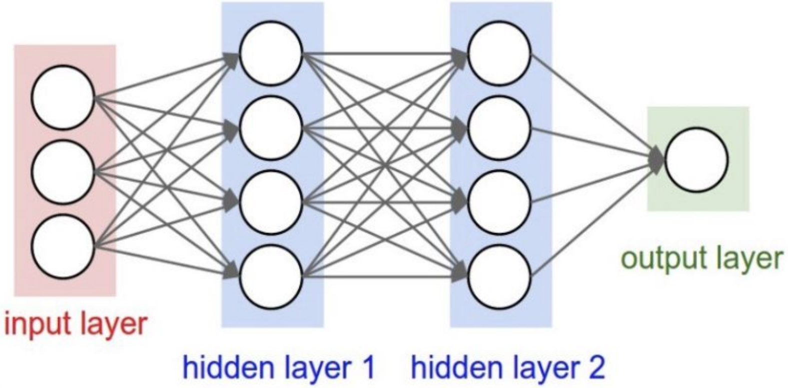


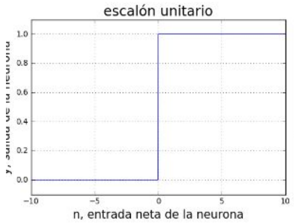
$$f(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$



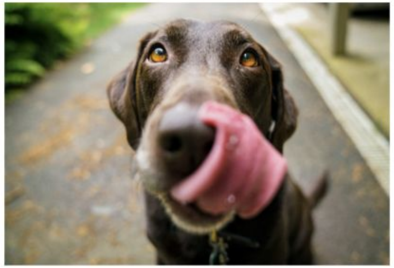
$$f(n) = \max(0, n)$$

Diferencia entre activación Biclase y Multiclase

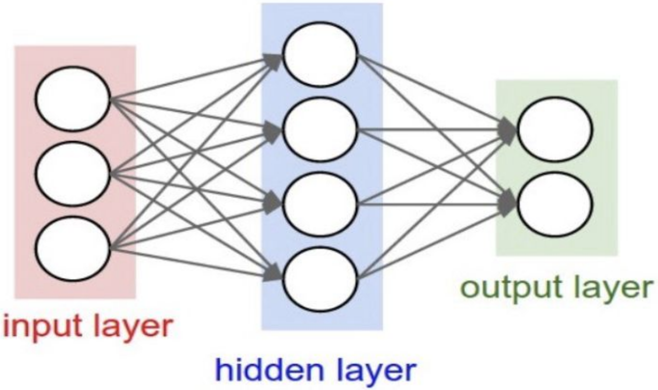


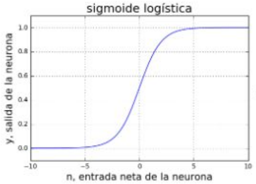


$$f(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$

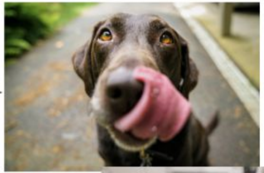


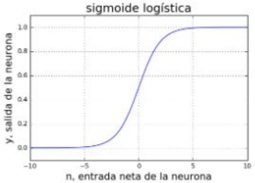
¿Es perro? - Sí / No





¿Qué tanto es perro?

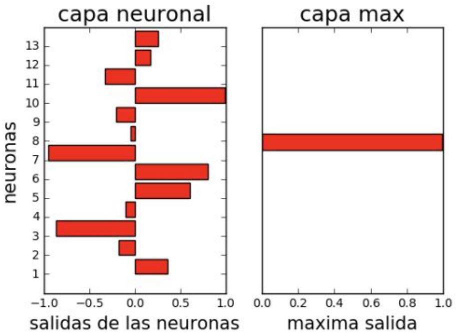




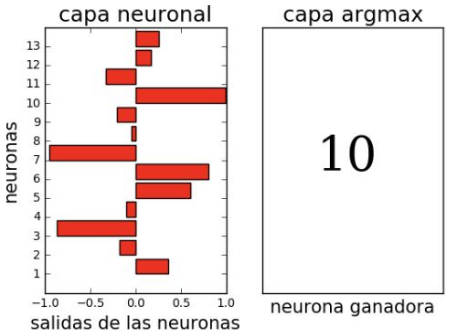
¿Qué tanto es gato?

$$f(n) = \frac{1}{1 + \exp(-n)}$$

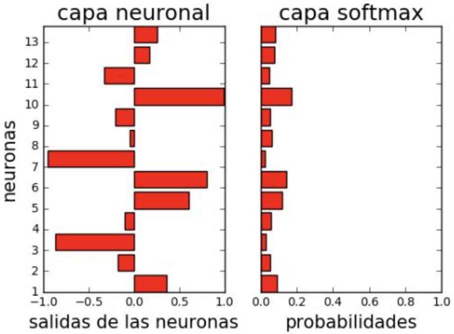
Capas de activación neuronales (Multiclase)



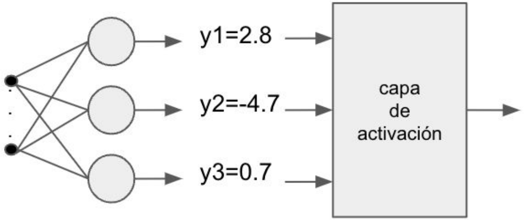
$$f(\boldsymbol{y}) = \max_i(y_i)$$



$$f(\boldsymbol{y}) = \mathrm{argmax}_i(y_i)$$



$$f_i(\boldsymbol{y}) = \frac{\exp(y_i)}{\sum_{k=0}^{S} \exp(y_k)}$$



Si es max -> y = max(2.8,-4.7,0.7) = 2.8

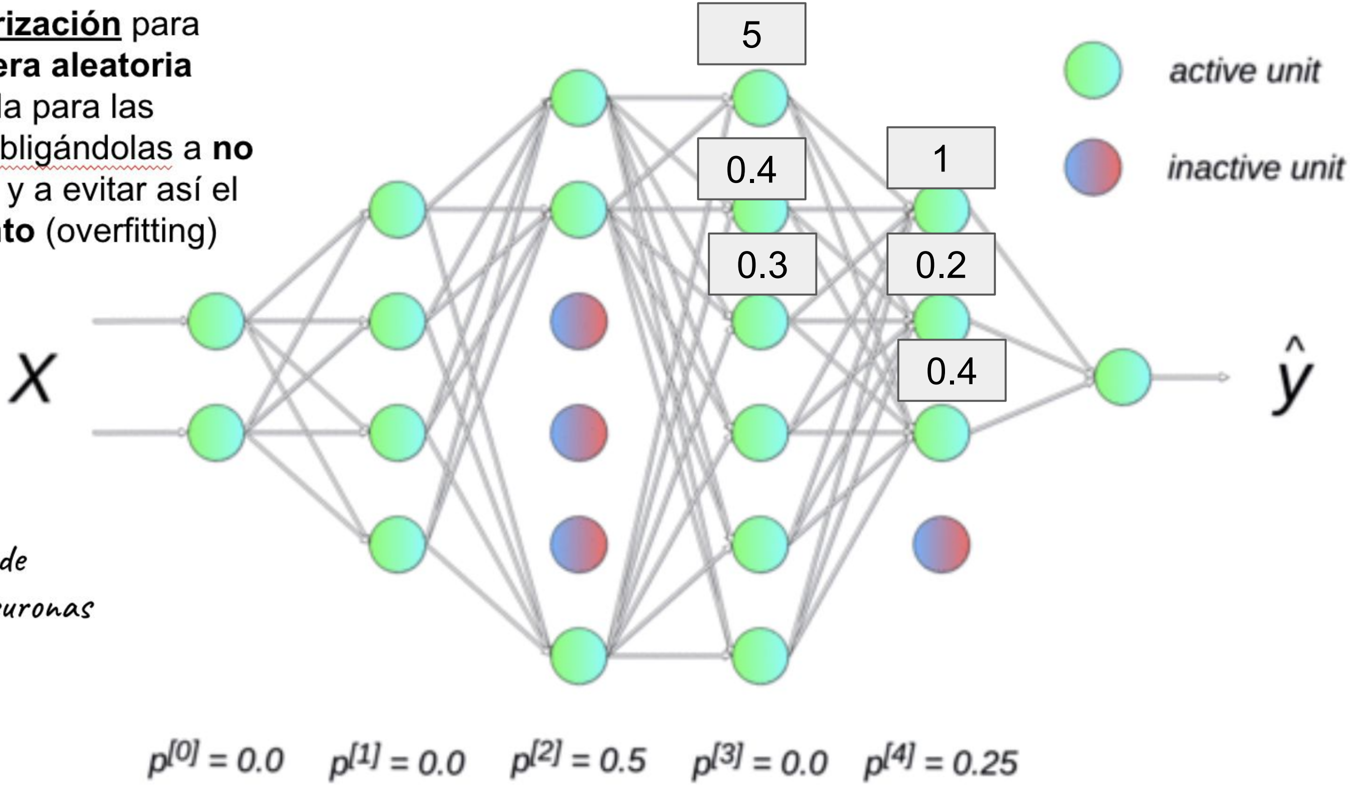
Si es argmax -> y = argmax(2.8,-4.7,0.7) = 1

Si es softmax -> **y** = [0.89; 0.00; 0.11]

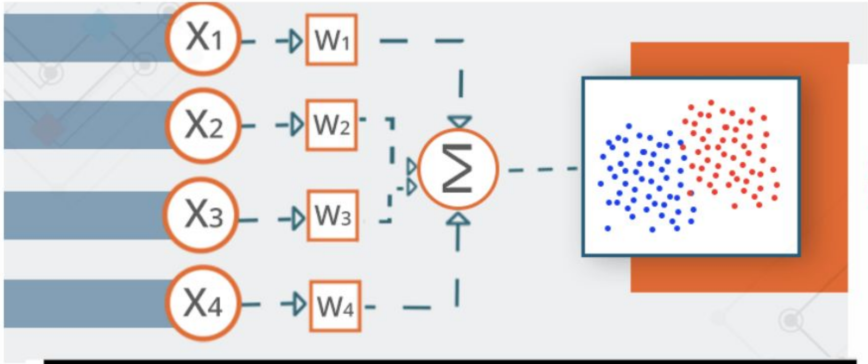
¿Qué es Dropout?

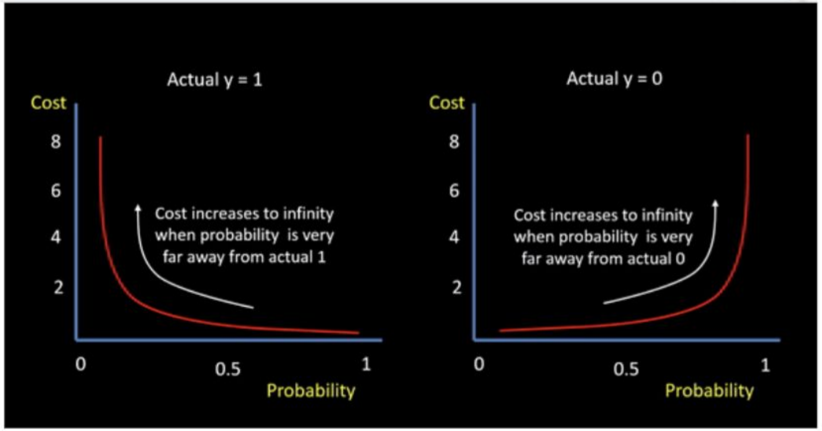
Método de **<u>Regularización</u>** para desactivar de **manera aleatoria** neuronas de entrada para las siguientes capas, obligándolas a **no depender** de ellas, y a evitar así el **sobreentrenamiento** (overfitting)

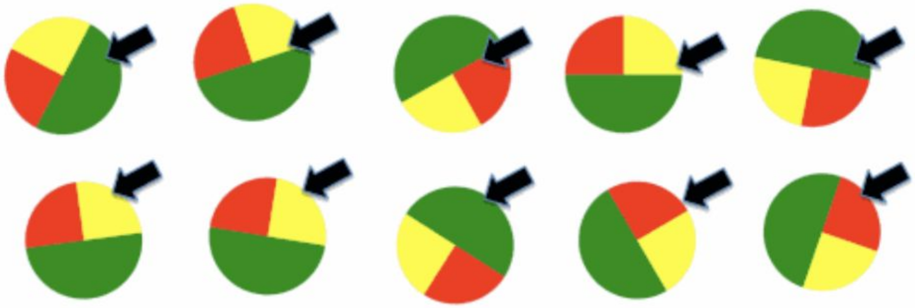
p = *probabilidad de apagado de las neuronas de entrada.*



Entropía Binaria y Categórica

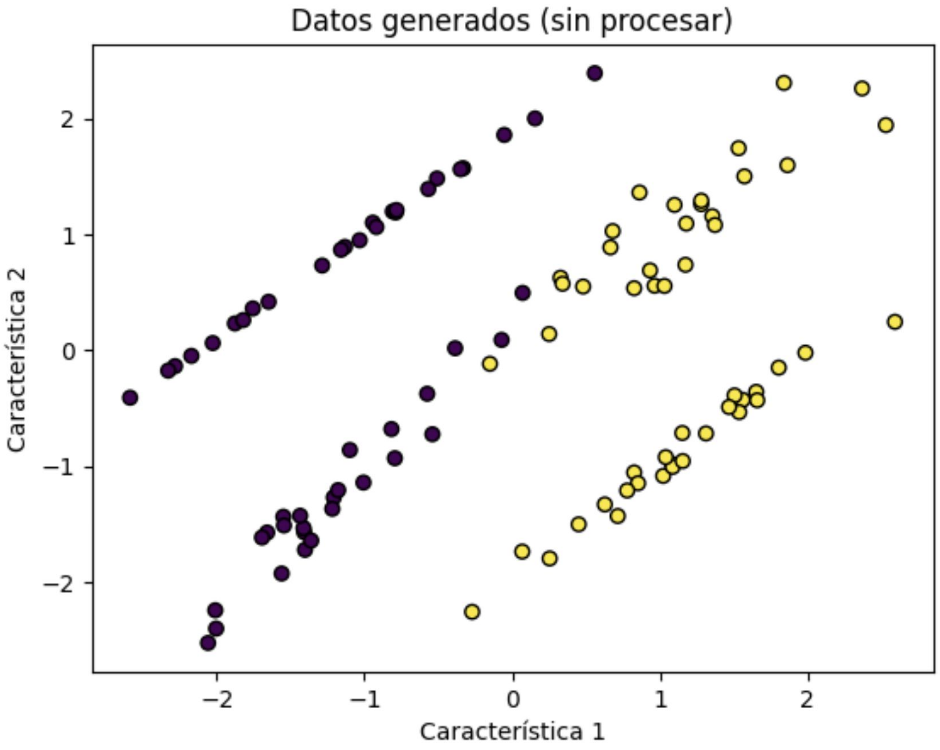






	Red	Green	Yellow
COUNT	2	4	4
PROBABILITY	2/10=.2	4/10=.4	4/10=.4
INFORMATION (LOG of PROBABILITY)	LOG(.2)=-2.3	-1.4	-1.4

UNIVERSIDAD Panamericana


Datos generados (sin procesar)
Característica 2
Característica 1
2
1
0
−1
−2
−2
−1
0
1
2

Imagina que tienes una red neuronal con una sola capa oculta. La capa de entrada recibe dos características ($x1$ y $x2$), la capa oculta tiene tres neuronas, y hay una sola neurona en la capa de salida. La función de activación en todas las neuronas es la función sigmoide, que se define como $\sigma(x) = 1 / (1 + e^{-x})$.

Las conexiones entre las neuronas tienen los siguientes pesos:

Desde la entrada a la capa oculta:
$w11 = 0.5$, $w12 = -0.6$ (pesos de la primera neurona de la capa oculta para $x1$ y $x2$ respectivamente)
$w21 = 0.1$, $w22 = 0.8$ (segunda neurona)
$w31 = -0.3$, $w32 = 0.7$ (tercera neurona)
Desde la capa oculta a la salida:
$v1 = 0.3$, $v2 = -0.9$, $v3 = 0.4$ (pesos de las neuronas de la capa oculta a la salida)
El bias para cada neurona en la capa oculta es 0, y el bias para la neurona de salida también es 0.

Tarea:
Dadas las entradas $x1 = 1$ y $x2 = 1$, calcula la salida de la red neuronal. Para hacer esto, realiza los siguientes pasos:

Calcula la entrada total a cada neurona en la capa oculta.
Aplica la función de activación sigmoide a cada salida de la capa oculta.
Calcula la entrada total a la neurona de la salida usando las salidas de la capa oculta.
Aplica la función de activación sigmoide a la salida de la neurona de salida para obtener el resultado final.

UNIVERSIDAD Panamericana