

# Entrenamiento perceptrón

## Perceptron training

Autor 1: Héctor Julián Cardona Montoya  
Universidad Tecnológica de Pereira, Pereira, Colombia  
Correo-e: julian.cardona@utp.edu.co

**Resumen**— El perceptrón es la forma más simple de una red neuronal usada para la clasificación de un tipo especial de patrones, los linealmente separables, es decir patrones que se encuentran a ambos lados de un hiperplano, básicamente consiste de una neurona con pesos sinápticos y umbrales ajustables.

**Palabras clave** — Aprendizaje, Artificial, Entrenamiento, Inteligencia, Inteligencia artificial, Neurona, Perceptrón, Red.

**Abstract**— Perceptron is the simplest form of a neural network used for the classification of a special type of pattern, the linearly separable, i.e. patterns found on both sides of a hyperplane, basically consisting of a neuron with synaptic weights and adjustable thresholds.

**Key Word** — Artificial, Artificial Intelligence, Intelligence, Learning, Neuron, Network, Perceptron, Training.

### I. INTRODUCCIÓN

El concepto de redes neuronales suele ser explicado más fácilmente cuando se relaciona con el aprendizaje humano.

Tomemos como ejemplo un niño: cuando un niño comienza a aprender los colores, primeramente, debe visualizar el color, y luego con indicaciones aprende demás características sobre el color, esto le permite que en un futuro cuando vuelva a ver el mismo color lo identifique sin tener que recibir nuevamente las indicaciones.

Lo mismo sucede con las redes neuronales, cuando han sido entrenadas previamente, pueden tomar decisiones respecto a una entrada.

En otras palabras, una red neuronal es la representación computacional de las neuronas de un cerebro humano.

Para que entendamos mejor el funcionamiento de una red neuronal, vayámonos hasta la base, lo más pequeño, un perceptrón, el perceptrón es el modelo más simple de una red neuronal.

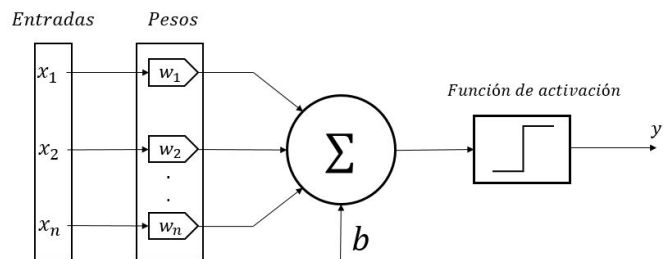
### II. CONTENIDO

**1) ¿Qué es un perceptrón?** El perceptrón es el modelo más simple de una red neuronal.

La razón de su auge en los años 60's es que tiene capacidad de aprendizaje y reconocimiento de patrones linealmente separables. El perceptrón puede ser considerado como una neurona artificial, unidad de cálculo que intenta modelar el comportamiento de una neurona natural como las que constituyen el cerebro humano.

Un perceptrón se compone por:

- Entradas: Es la información que recibe el perceptrón.
- Pesos: Son valores numéricos que se encargan de establecer la influencia de la entrada en la salida.
- Bías: Es un parámetro que tienen algunos modelos de redes neuronales el cual permite encontrar fácilmente la separación entre posibilidades de salida de una red neuronal.
- Función de activación: Es una función matemática que se encarga de determinar un valor de salida una vez se han procesado cada una de las entradas



**2) Entrenamiento del perceptrón.** Primeramente, debemos entender que para que la serie de conocimientos o información que la red neuronal podrá identificar puedan ser identificados se debe llevar a cabo el proceso de entrenamiento o aprendizaje.

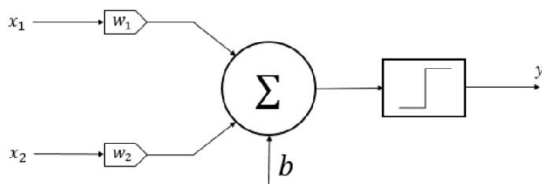
Para entender mejor, creo que más fácil dar un ejemplo.

Daremos el ejemplo de el circuito lógico AND.

Realizaremos el entrenamiento del perceptrón con diferentes pesos y bias a los que allí se indican. Esto con la finalidad de mostrar qué sucede cuando el perceptrón no se entrena en la primera iteración.

Compuerta AND		
Entradas		Salidas
$x_1$	$x_2$	$y = x_1 * x_2$
0	0	0
0	1	0
1	0	0
1	1	1

Compuerta AND			
Entradas		Salidas	$n$
$x_1$	$x_2$	$y = x_1 * x_2$	
0	0	0	1
0	1	0	2
1	0	0	3
1	1	1	4



Paso 1: Inicializar los pesos y el bias

Cada entrada del perceptrón debe tener un peso.

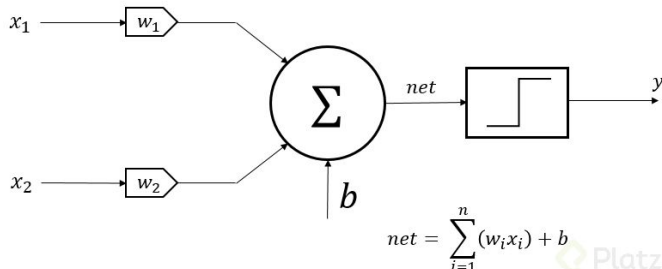
Estos valores pueden ser aleatorios, sin embargo, es mejor empezar con valores pequeños.

$$W_1 = 10 \quad W_2 = 10 \quad b = -8$$

Paso 2 Calcular las salidas (net) con los pesos y el bias

En este paso se calcula cada salida que teniendo en cuenta los posibles valores pueden tomar las entradas del perceptrón. Para este caso se realiza 4 veces debido a que la compuerta AND solo tiene ese número de posibilidades.

Es importante resaltar que la salida que se obtiene en este paso es la que se da después de la sumatoria. Por convención se nombrará en adelante como net así como se ve en la imagen.



Como se mencionó tenemos 4 posibilidades de salida.

Luego obtenemos la salida utilizando la función de activación y calculamos el valor del error.

El error lo podemos calcular de la siguiente manera:

$$e = y - y'$$

Donde  $e \rightarrow$  Valor del error

$y' \rightarrow$  Salida deseada

$y \rightarrow$  Salida obtenida

Y ahora podemos enfrentarnos a 3 posibles valores de error.

**Caso 1:** Si  $e_n = 0$ , repita el paso 2 con las siguientes entradas.

**Caso 2:** Si  $e_n \neq 0$ , continúe con el paso 4.

**Caso 3:** Si todos los valores calculados del error son iguales a 0 para todas las entradas. el entrenamiento finaliza

Conociendo esto se pueden mediante algoritmos corregir los pesos para llegar al resultado deseado. Una ecuación muy usada es:

$$W_i(k+1) = w(k) + e * X_i$$

Donde:

$W_n(K+1) \rightarrow$  Nuevo peso

$w(k) \rightarrow$  Peso actual

$e \rightarrow$  Error

$X_i \rightarrow$  Entrada.

Debemos tener en cuenta que mientras el error no sea 0, siempre debemos corregir los pesos.

Haciendo esto constantemente podremos llegar a los pesos necesarios para que nuestro perceptrón pueda realizar sus funciones de manera exacta.

## 1) CONCLUSIONES

Cuando todos los errores son iguales a 0 podemos dar el entrenamiento del perceptrón como finalizado.

El perceptrón es la neurona artificial o unidad básica de inferencia en forma de discriminador lineal, a partir de lo cual se desarrolla un algoritmo capaz de generar un criterio para seleccionar un sub-grupo a partir de un grupo de componentes más grande.

## REFERENCIAS

### Referencias de publicaciones periódicas:

- [1] <https://disi.unal.edu.co/~lctorress/RedNeu/LiRna004.pdf>
- [2] <https://platzi.com/tutoriales/1157-ia-2017/2619-entrenamiento-del-perceptron/>
- [3] <https://koldopina.com/como-entrenar-a-tu-perceptron/>
- [4] <https://www.youtube.com/watch?v=Woe8fXttC6E>
- [5] [https://www.ibm.com/support/knowledgecenter/es/SSLVMB\\_sub/statistics\\_mainhelp\\_ddita/spss/neural\\_network/idd\\_idd\\_mlp\\_training.html](https://www.ibm.com/support/knowledgecenter/es/SSLVMB_sub/statistics_mainhelp_ddita/spss/neural_network/idd_idd_mlp_training.html)
- [6] <https://elvex.ugr.es/decsai/deep-learning/slides/NN2%20Perceptron.pdf>