



Pontificia Universidad  
**JAVERIANA**  
Bogotá

## Educación **Continua**

Generamos experiencias  
educativas

# **Modelos Analíticos Predictivos: Redes neuronales**

**Basado en Material de Profesor  
Jhon Parra.**

# Redes neuronales



Pontificia Universidad  
**JAVERIANA**  
Bogotá

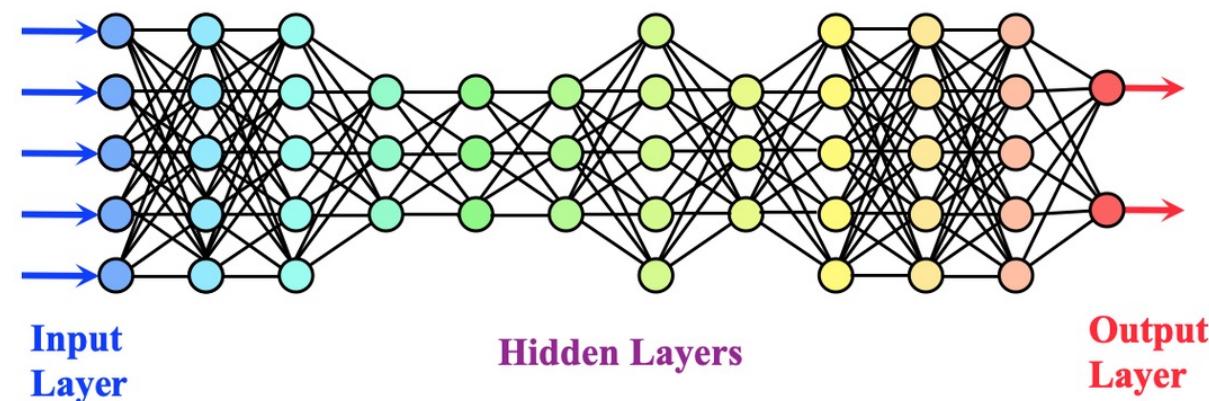
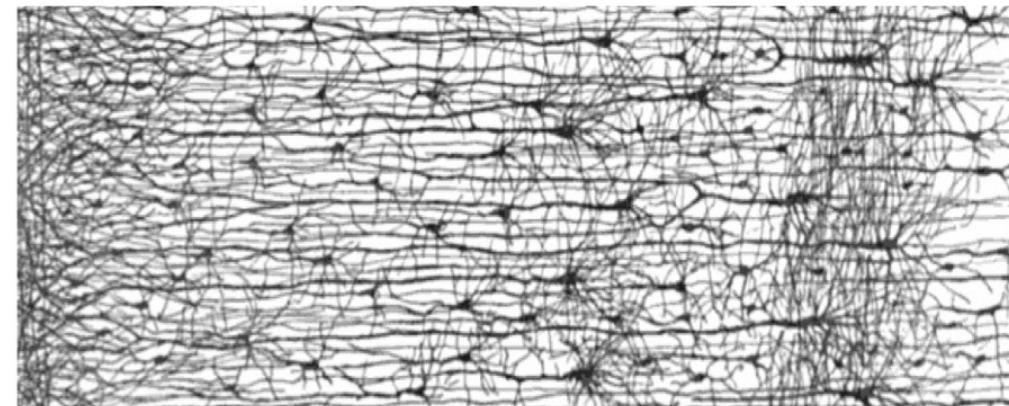


Educación **Continua**  
Generamos experiencias educativas

# Redes Neuronales Artificiales (RNA)

- Las redes neuronales artificiales (RNA) se inspiraron en las redes de neuronas biológicas del cerebro.
- A lo largo del tiempo, las RNA se han distanciado de sus contrapartes biológicas, y algunos argumentan que deberíamos abandonar completamente la analogía biológica.
- Las RNA son fundamentales en el aprendizaje profundo y se caracterizan por ser versátiles, potentes y escalables.
- Son ideales para abordar tareas complejas, como la clasificación de imágenes masivas, el reconocimiento de voz, procesamiento del lenguaje natural y juegos de estrategia.

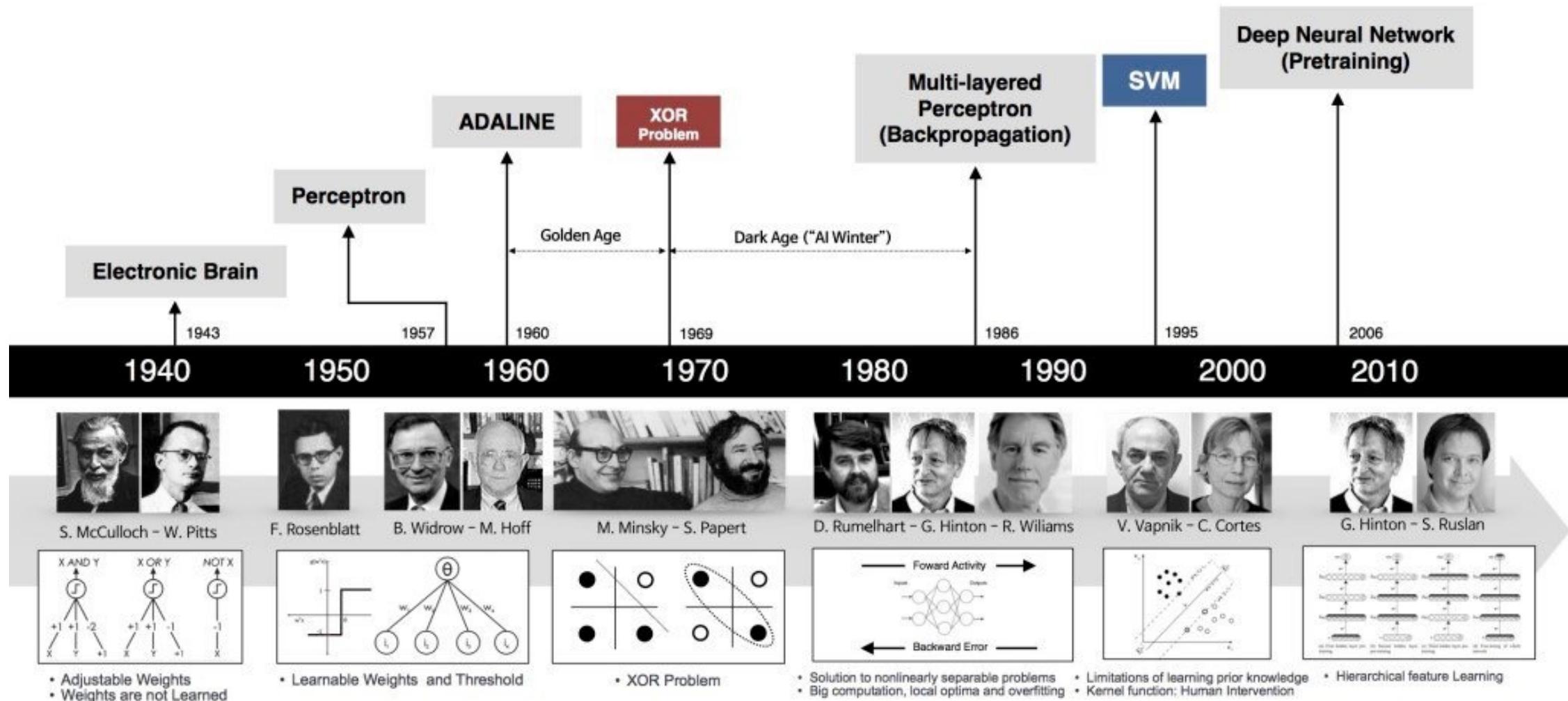
Redes Neuronales Biológicas



Redes Neuronales Artificiales



# Evolución del Concepto de Redes Neuronales



# El Auge de las Redes Neuronales

- La creciente disponibilidad de una gran cantidad de datos (Big Data) ha permitido entrenar redes neuronales con éxito en problemas complejos.
- El aumento significativo en la potencia computacional, impulsado por la Ley de Moore y la industria de videojuegos, ha hecho posible entrenar redes neuronales grandes en tiempos razonables.
- Se han realizado ajustes en los algoritmos de entrenamiento, lo que ha tenido un impacto positivo en el rendimiento de las redes neuronales.
- Limitaciones teóricas que se pensaba que afectarían a las redes neuronales en la práctica han resultado ser menos problemáticas de lo esperado.
- El interés y financiación crecientes han generado un círculo virtuoso de avances en las redes neuronales, impulsando productos innovadores y atractivos en diversos campos.



# Como trabajan las Redes Neuronales



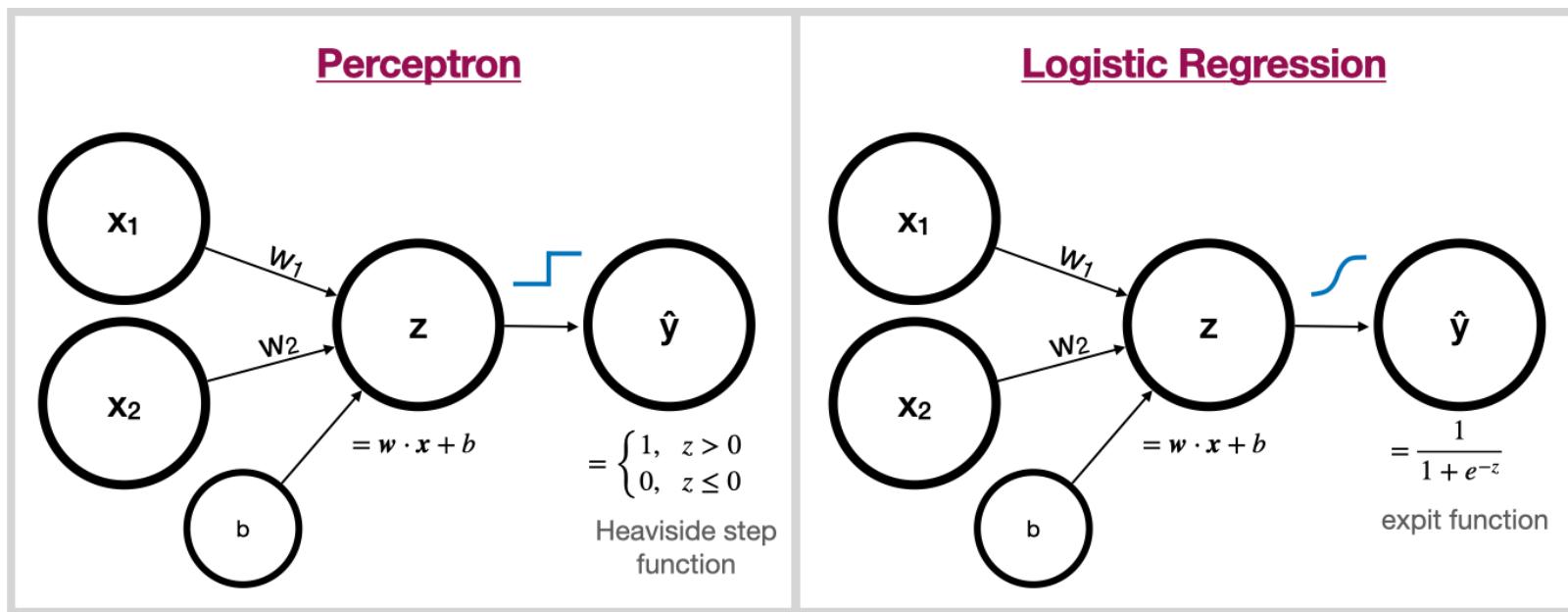
Pontificia Universidad  
**JAVERIANA**  
Bogotá



Educación **Continua**  
Generamos experiencias educativas

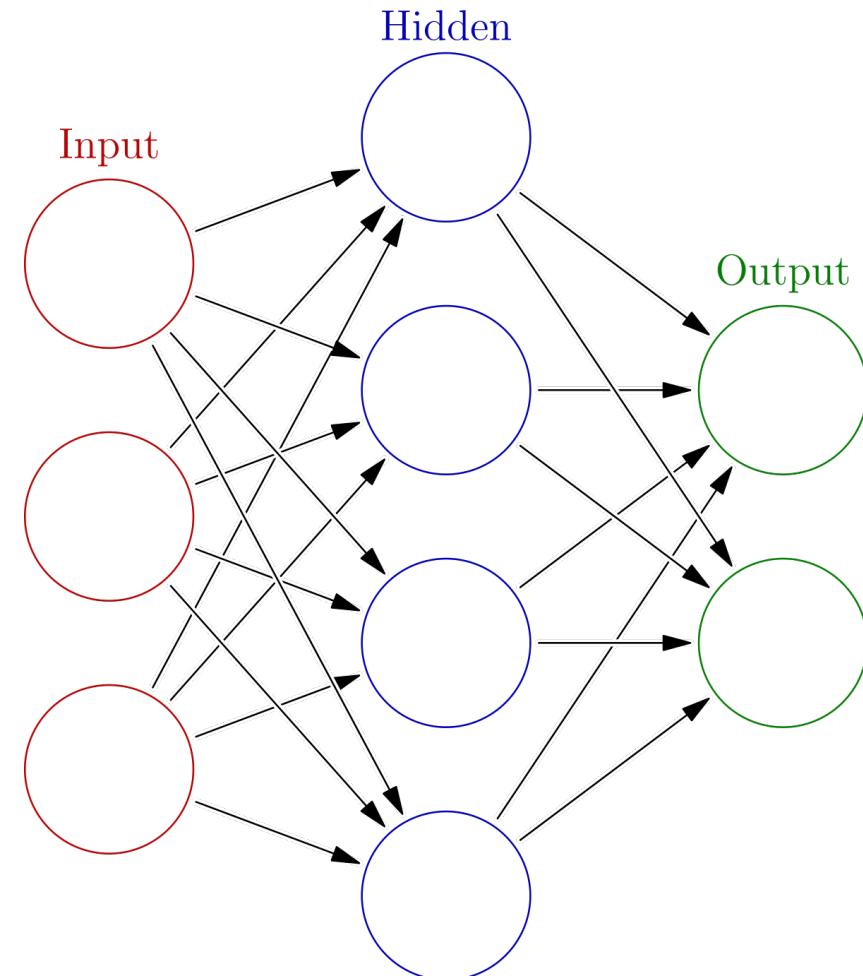
# Algoritmo Perceptrón

- Es un algoritmo de aprendizaje supervisado que se utiliza para la clasificación binaria.
- Utiliza una función de activación escalón que produce salidas binarias (0 o 1) basadas en una combinación lineal de las características de entrada.
- Aunque el Perceptrón en su forma original tiene limitaciones importantes, sentó las bases para el desarrollo posterior de redes neuronales más complejas y potentes.

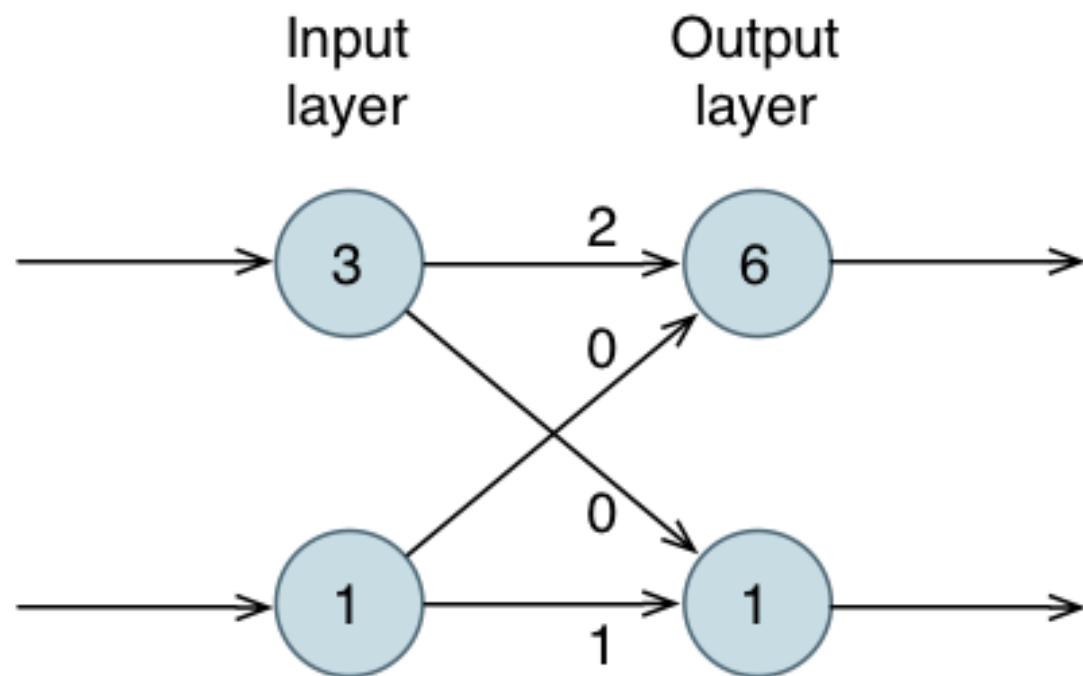


# Un modelo más complejo

- Hasta ahora hemos visto modelos que asumen (mayoritariamente) **comportamientos lineales** o modelos que **dividen el espacio en varios espacios**.
- Pero, ¿qué pasa si tenemos datos con comportamientos no lineales y no separables mediante divisiones del espacio?
- Para esos casos son especialmente útiles las **redes neuronales**.
- En el gráfico de la derecha podemos ver una caracterización más precisa de la **arquitectura básica de una red neuronal**.



# Ejemplificación del cálculo en una red neuronal



- Cada nodo es una combinación lineal de la capa anterior.
- Entre cada capa, hay una **función de activación**, que es un "umbral" para que esa neurona se active y, cuando se combina en capas, esto permite que nuestra red desarrolle sensibilidades no lineales.
- Cuando hay varias capas, podemos seguir agregando más combinaciones lineales, lo que puede conducir a **modelos no lineales más complejos**.



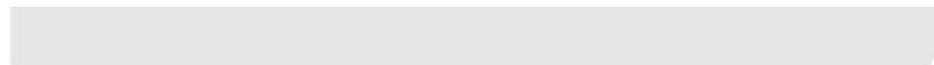
# Resumiendo el cálculo en nodos

En resumen el valor de cada nodo se puede definir como:

$$z = b + \sum_i w_i X_i$$

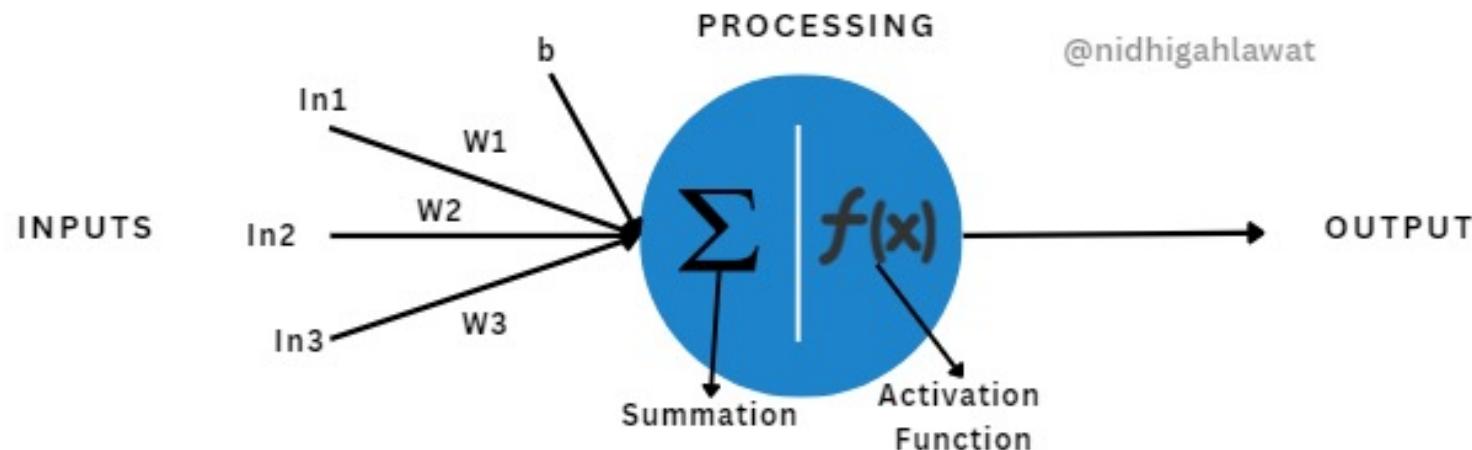
- Donde,  $w$  es el peso asignado a cada conexión.
- $X$  es el valor de cada nodo precedente.
- $b$  es el valor base o intercepto de cada nodo.

Sin embargo, si pensamos que las redes neuronales emulan el comportamiento de nuestro cerebro y las conexiones entre nuestras neuronas, hay que considerar que no todos los estímulos que recibimos tienen una respuesta de nuestra parte. En las redes neuronales, esto se llama **funciones de activación**.



# Funciones de activación

- Básicamente, las funciones de activación permiten **calcular la salida final** de un nodo.
- En otras palabras, la **función de activación transforma el resultado** del cálculo matemático realizado en cada neurona en un valor que va a ser entrada de otros nodos en la red.
- Algunas de las funciones de activación permiten mapear los resultados de cada nodo en diferentes escalas y manejar datos predicciones no lineales.

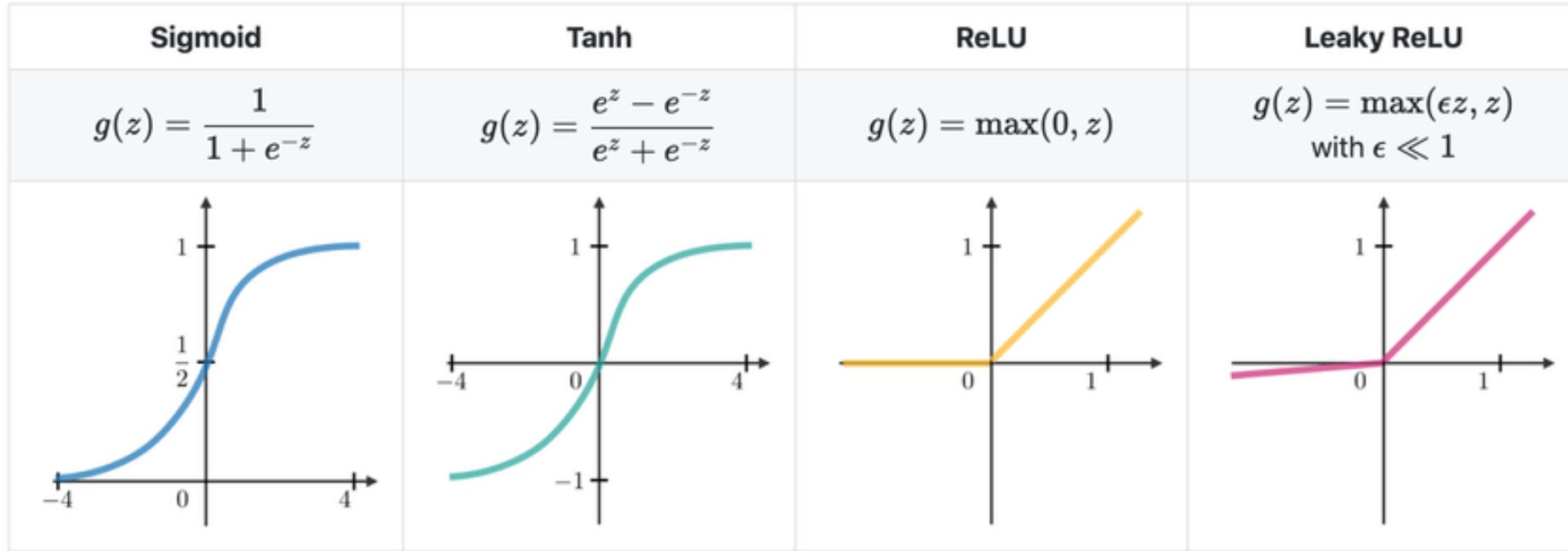


@nidhigahlawat

Fuente: [Medium](#)



# Funciones de activación



Fuente:  
[ResearchGate](#)

Existen muchas otras funciones de activación, si quieren conocer más sobre estas pueden consultar este [artículo](#).



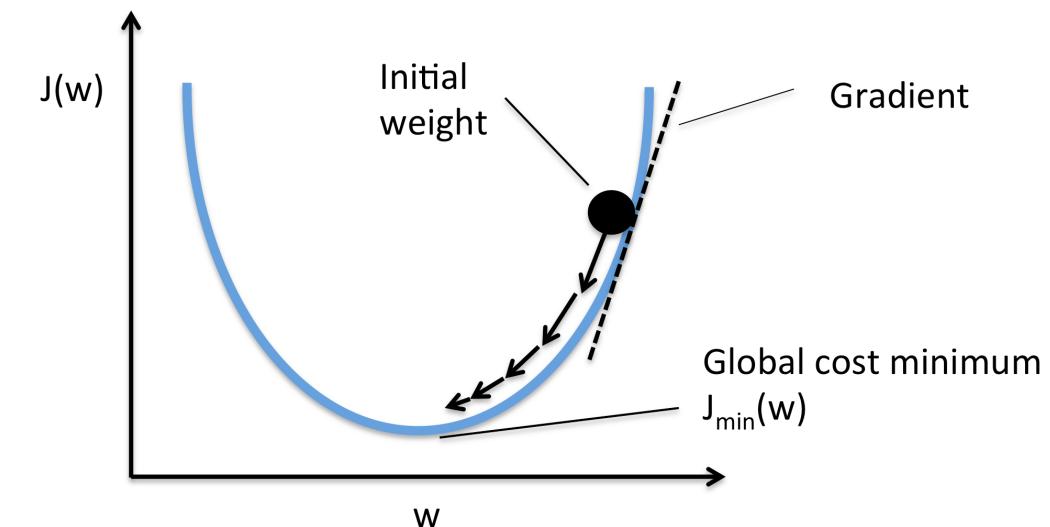
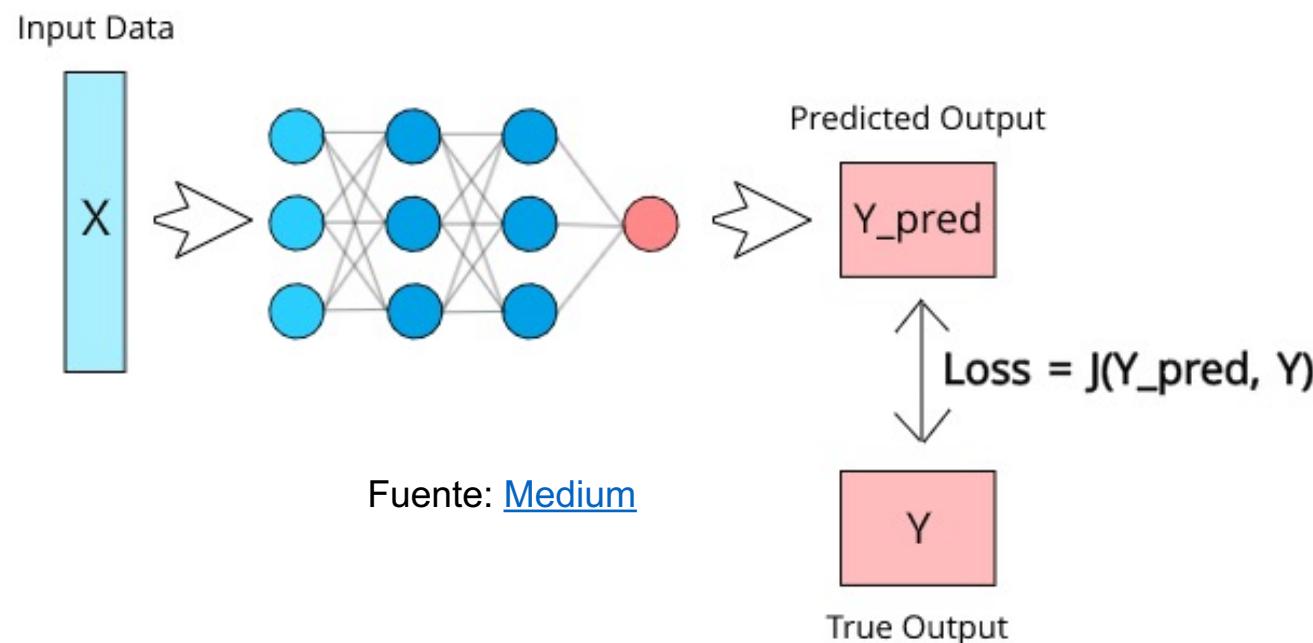
Pontificia Universidad  
**JAVERIANA**  
Bogotá



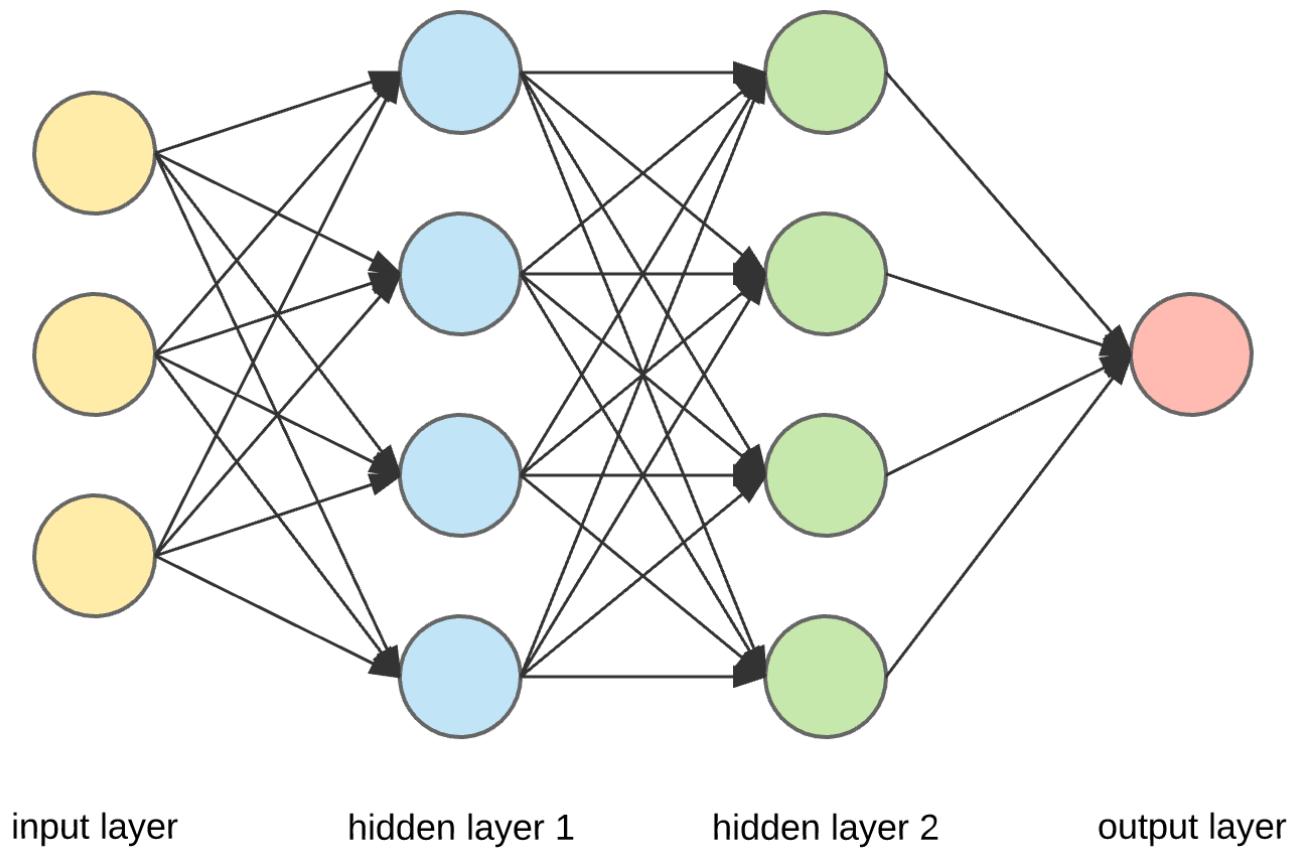
Educación **Continua**  
Generamos experiencias educativas

# ¿Buenas o malas decisiones?

¿Cómo tomamos buenas o malas decisiones dentro de la red? Comparamos los resultados de las predicciones (usando la función de pérdida) y hacemos pequeños cambios para comparar los resultados. Con mayor frecuencia, utilizamos una tasa de aprendizaje y un método de descenso de gradiente para estimar los cambios que han utilizado nuestros modelos sucesivos.



# Redes neuronales profundas



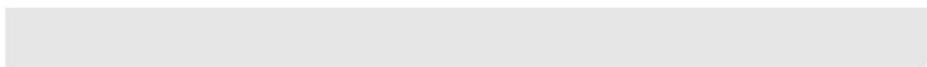
- Podemos agregar más capas ocultas en el medio para permitir que la red neuronal cree funciones más complejas.
- A esto se le llama **red neuronal profunda**.
- Las redes neuronales profundas modernas pueden tener cientos de capas.
- La cantidad de capas que agrega es parte del arte y la ciencia de construir una red neuronal adecuada.
- Cabe señalar que más capas no necesariamente es mejor.



# Conceptos básicos

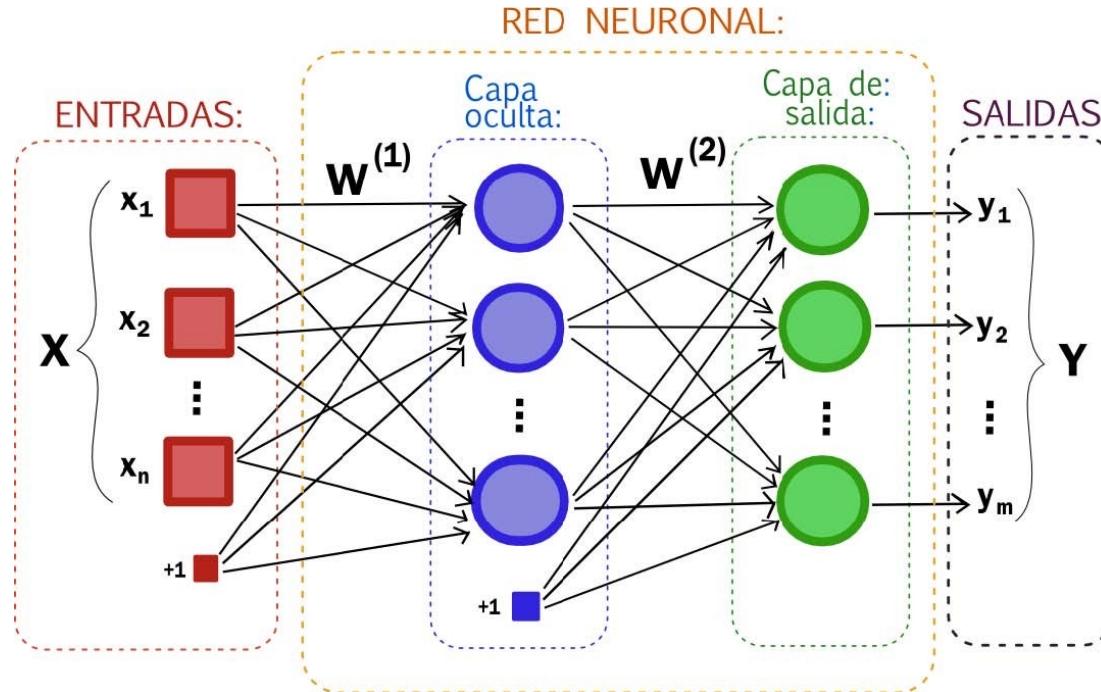


Pontificia Universidad  
**JAVERIANA**  
Bogotá



Educación **Continua**  
Generamos experiencias educativas

# La Arquitectura de una Red Neuronal



Fuente: [LinkedIn](#)

- El funcionamiento interno del algoritmo toma **entradas en el lado izquierdo** para proporcionar **salidas en el lado derecho** multiplicando cada capa por los bordes entre las neuronas para obtener la siguiente capa.
- Luego, el algoritmo utiliza un proceso conocido como **gradiente descendiente** para optimizar cada borde.
- La retropropagación es el método para **ajustar los pesos en cada capa oculta** de acuerdo con qué tan bien se desempeñó la red en comparación con las salidas reales en cada paso de iteración.



# Conceptos básicos de redes neuronales

- 1. Capa de entrada:** Esta es la capa que gobierna cómo se estructurarán las entradas de datos, es decir, cuántos nodos se refieren a cuántos parámetros diferentes está utilizando).
- 2. Capas ocultas:** Estas son las capas internas de la red neuronal que realizan cálculos ponderando los datos de entrada de varias maneras y que, en última instancia, determinan la salida. Tenemos que determinar la cantidad de nodos para estas capas y cuántos usaremos.
- 3. Capa de salida:** La capa que rige la estructura de su salida, que diferirá de una situación a otra. Por ejemplo, en la clasificación binaria, puede tener 1 nodo como salida y si devuelve  $> 0,5$ , entonces será clase A; en caso contrario, clase B.
- 4. Función de activación:** El fórmula entre capas que determina la activación de las neuronas. Para nuestros ejercicios sólo usaremos ReLu, que es extremadamente común.



# Conceptos básicos de redes neuronales

4. **Función de pérdida:** Determina cómo medimos la precisión o el rendimiento de nuestro modelo. Aquí utilizaremos el error cuadrático medio (MSE), con el que estamos familiarizados por la regresión lineal de mínimos cuadrados ordinarios.
5. **Optimizador:** Este es el algoritmo que determinará cómo minimizamos nuestra función de pérdida.
7. **Epoch:** Esta es la cantidad de veces que ejecutamos nuestro modelo a través del proceso de ajuste de los datos. No existe una ciencia exacta para elegir las épocas a primera vista.
8. **Batch Size:** Este es el número de puntos de datos que se introducen en el modelo para el entrenamiento en cualquier momento. Es un acto de equilibrio elegir un número que no sea demasiado alto (porque será demasiado lento para converger a una solución) y no demasiado bajo (porque, aunque tendrá una mayor probabilidad de converger a una solución óptima, tomará más tiempo). entrenar en general).



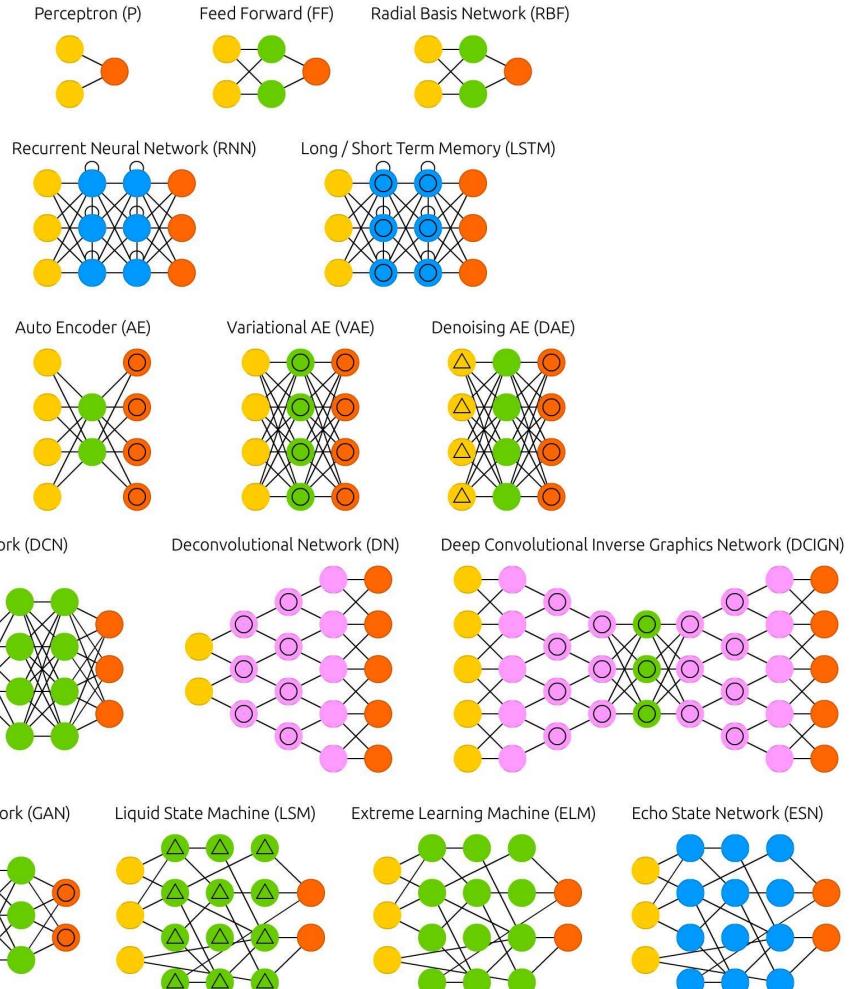
# Tipos de redes neuronales y aplicaciones



# Tipos de redes neuronales

- Perceptron.
- Multilayer Perceptron.
- Convolutional Neural Network.
- Recurrent Neural Network.
- Long Short Term Memory.
- Modular Neural Network.
- Generative Adversarial Network.
- Radial Basis Functional Neural Network.
- Autoencoder.
- LLM and Transformers.

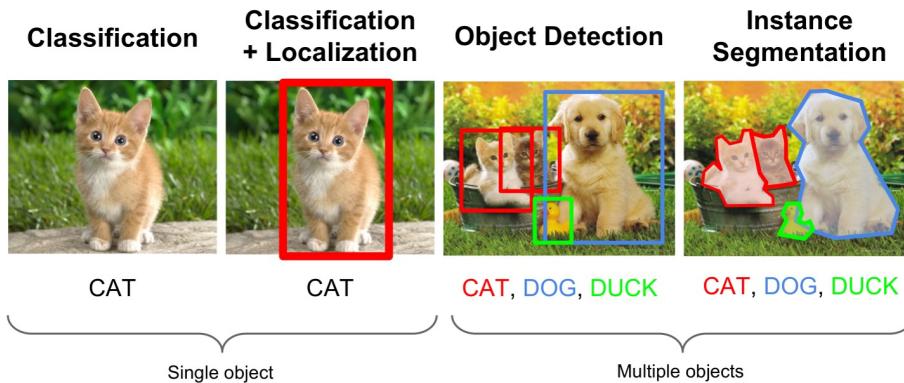
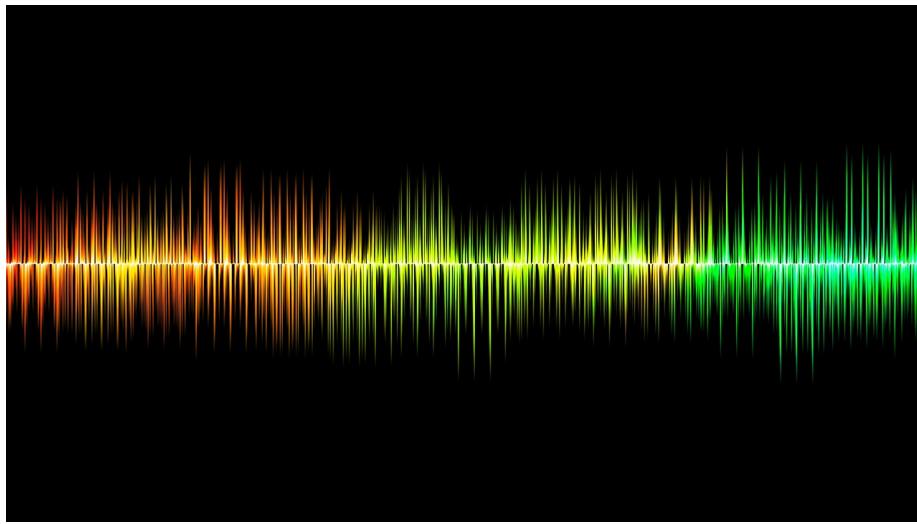
- Input Cell
- Backfed Input Cell
- △ Noisy Input Cell
- Hidden Cell
- Probabilistic Hidden Cell
- △ Spiking Hidden Cell
- Capsule Cell
- Output Cell
- Match Input Output Cell
- Recurrent Cell
- Memory Cell
- △ Gated Memory Cell
- Kernel
- Convolution or Pool



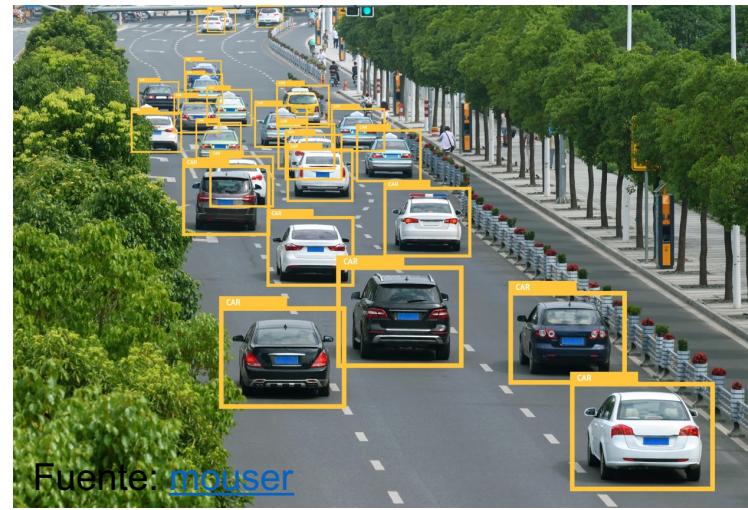
Fuente: [StackExchange](#)



# Aplicaciones



Fuente: [mocomakers](#)



Fuente: [mouser](#)



## ChatGPT

LOGOS.ADM96.COM



Pontificia Universidad  
**JAVIERIANA**  
Bogotá

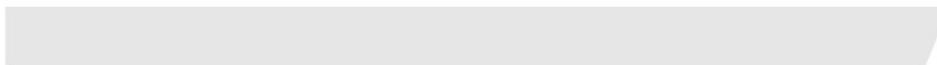


Educación **Continua**  
Generamos experiencias educativas

# ¡Gracias!



Pontificia Universidad  
**JAVERIANA**  
Bogotá



Educación **Continua**  
Generamos experiencias educativas