



# *Inteligencia Artificial*

## *Redes Neuronales Artificiales*

Prof. Wílmer Pereira

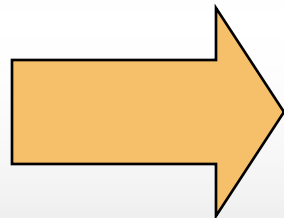


# Redes Neural Natural

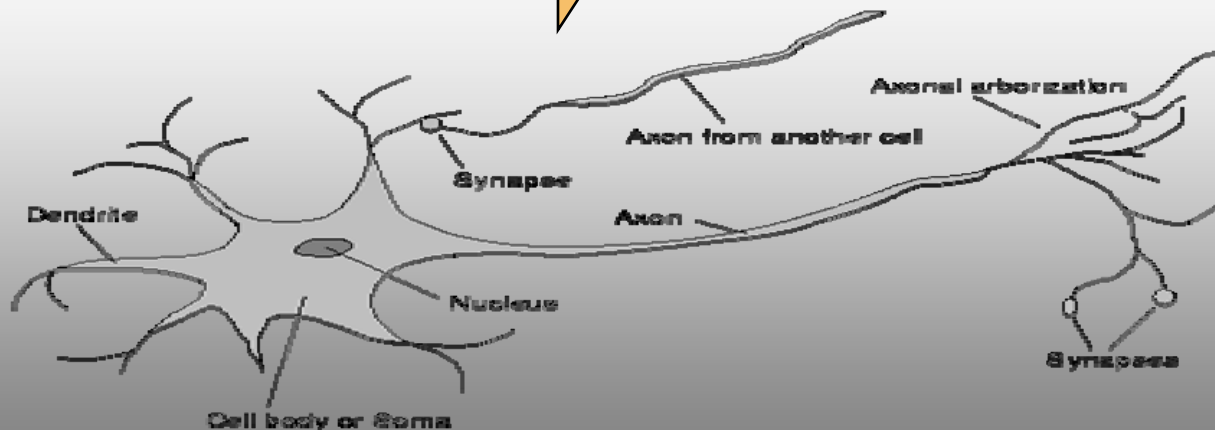
Estructura celular del cerebro donde residen las capacidades intelectuales del hombre

<u>Neurona:</u>	Célula nerviosa
<u>Soma:</u>	Núcleo celular
<u>Dendritas:</u>	Ramificaciones entre neuronas
<u>Sinapsis:</u>	Punto de unión entre dendritas

Reacciones  
Electroquímicas



Impulsos Inhibidores o  
Impulsos Excitatorios





# *Propiedades de la red Neuronal Natural*



Plasticidad: Nexos entre neuronas que se fortalecen con los patrones de estímulo



Elasticidad: Capacidad de crecer para agregar propiedades intelectuales



Todo el procesamiento tiene lugar en la corteza y por zonas  
Afasia (Pierre Broca) permitió precisar las funcionalidades de lenguaje por área



No hay una teoría definitiva sobre la memoria  
Memoria a largo plazo  
Memoria a corto plazo



Mecanicismo vs Misticismo



# Cerebro vs Computador

- Almacenamiento: Más neuronas que bits aunque la evolución computacional es vertiginosa (mucho mayor que la evolución de cerebro)
- Velocidad. Computador orden de los  $\eta$ seg  
Cerebro del orden de los  $\mu$ seg  
*pero ... el cerebro es masivamente paralelo y en definitiva el cerebro es  $10^{10}$  veces más rápido*
- Tolerancia a fallas: Una neurona natural dañada afecta de manera marginal el comportamiento del cerebro  
Cualquier mínimo error altera todo el procesamiento a nivel del computador
- Complejidad de ejecución: El cerebro realiza tareas mucho más complejas que cualquier computador



# Redes Neuronales Artificiales

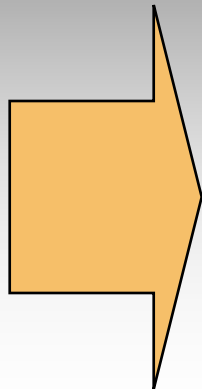
Unidades enlazadas a través de conexiones cargadas por pesos numéricos

- El aprendizaje se basa en la actualización de esos pesos que se Inician en la fase de entrenamiento de la red
- Está formada por unidades de entrada y unidades de salida (neuronas de entrada y neuronas de salida)
- El nivel de activación de la neurona artificial (equivalente al impulso excitatorio) es un cálculo individual en cada neurona, sin control global

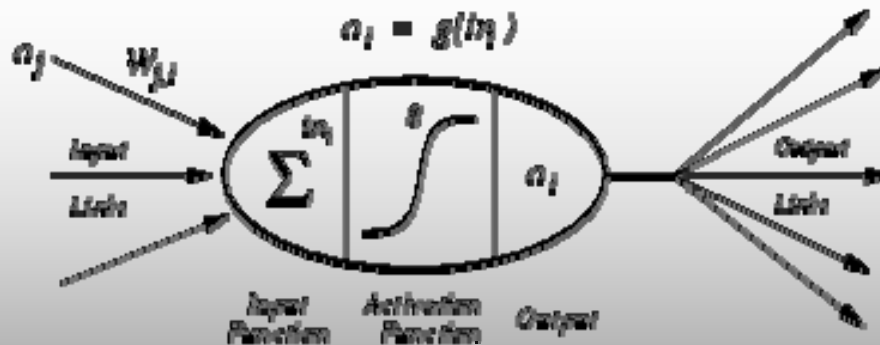




# Consideraciones en una Red Neural Artificial



- ¿ Cuantas unidades o neuronas artificiales ?
- ¿ Tipo de neurona ?
- ¿ Topología de la red ?
- ¿ Inicialización de los pesos ?
- ¿ Número de ejemplos para el entrenamiento ?
- ¿ Cómo codificar los datos de entrada y salida ?



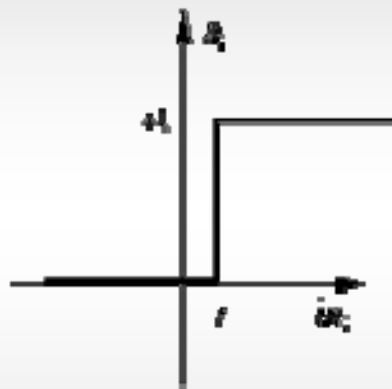
$$in_i = \sum W_{j,i} a_j$$

$$a_i = g(in_i)$$

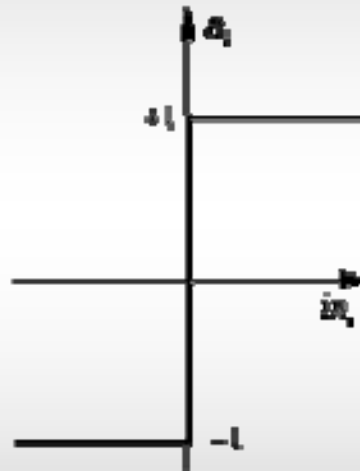


# Funciones de Activación (g)

Función común a todas las neuronas artificiales que determina su activación dado los estímulos de entrada



(a) Step function



(b) Sign function



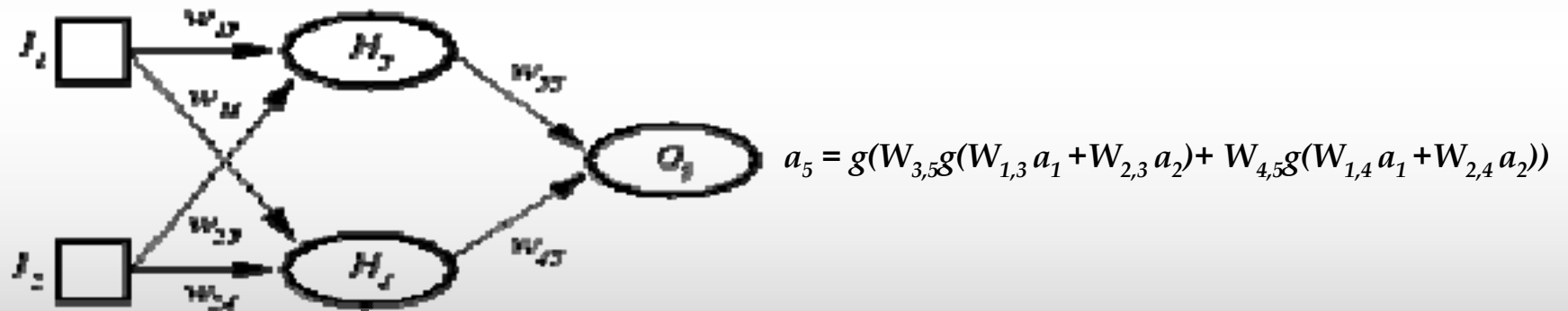
(c) Sigmoid function



# Tipos de Redes Neuronales Artificiales

## Lineales (*feed forward*):

- Unidireccionales
- Sólo conexión entre neuronas de capas adyacentes
- Sencillas y se adecuan bien a modelar reflejos
- Por no ser retroalimentadas, no tienen capacidades de memoria (el cerebro no es lineal)



## Recurrentes

- Conexiones libres
- Pueden tornarse inestables y oscilatorias por lo que requieren de avanzados métodos matemáticos





# Redes de Hopfield

- Bidireccionales y simétricas en sus pesos  $W_{i,j} = W_{j,i}$
- La función de activación es la función signo
- Sólo tiene neuronas de entrada y neuronas de salida
- Se conocen como de memoria asociativa

*Reproducen ejemplos de salida a partir de con sólo un trozo del ejemplo de entrada*



Red de Hopfield: lineal, sin capas intermedias con dos neuronas de entrada, una neurona de salida y dos neuronas ocultas ( $I_1$  y  $I_2$ )

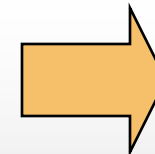


# Redes de Boltzman

- Tiene también pesos simétricos
- La función de activación es estocástica
- Tiene capas intermedias (red multicapas)

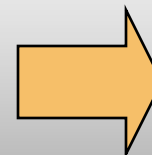
## Tamaño de la Red Neural

- Muy pequeñas no aprenden bien
- Muy grandes se aprende los ejemplos de memoria y no son capaces de extrapolar



**Sobrecompensación**

Para solucionar la sobrecompensación se debe hacer un buen tanteo, comenzando con una red neural pequeña y agrandarla o una red grande y reducirla

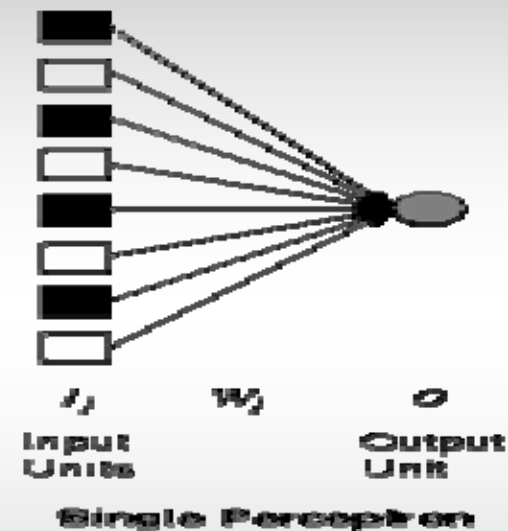
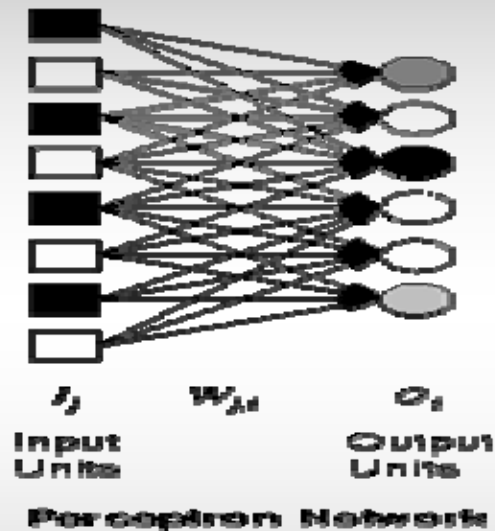


**¿ Cuántos ejemplos serán necesarios para el entrenamiento de la red neural ?**



# Perceptrón

Red Neural lineal a dos capas  
(sólo neuronas de entrada y salida)



El perceptrón aprende comenzando con pesos aleatorios ajustandolos mientras se entrena (sencillo pues las neuronas de entrada van conectadas directamente con las de salida)

$$Err = T - O$$

$$W_j = W_j + \alpha \cdot I_j \cdot Err$$

Si  $Err > 0$  aumentar  $O$ . Si  $Err < 0$  disminuir  $O$

donde  $O$  : ejemplo predicho

$T$  : ejemplo correcto

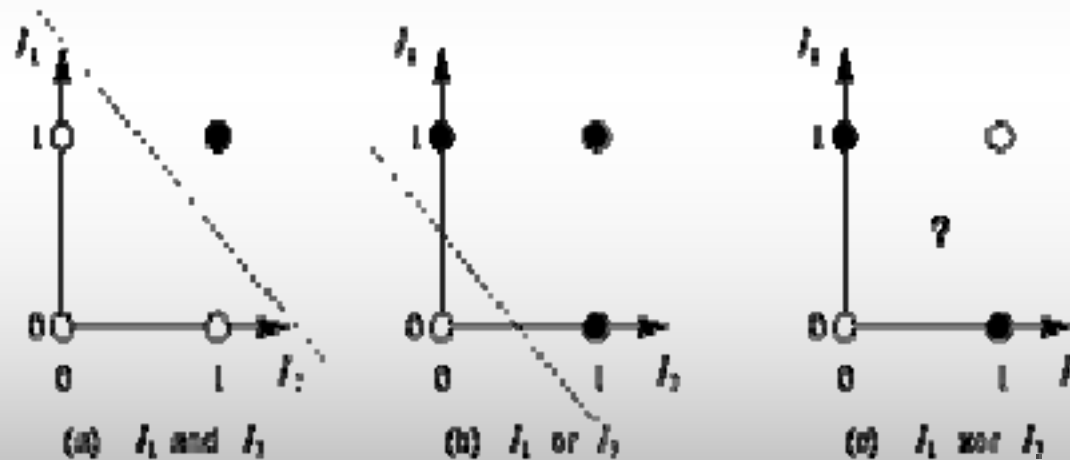
$\alpha$  : velocidad de aprendizaje



# Problemas del Perceptrón

Minsky y Papert publicaron en 1969, un artículo donde mostraron las limitaciones de los perceptrones

*El problema está en que el perceptrón sólo puede representar funciones linealmente separables ya que el perceptrón es una función lineal de las neuronas de entrada*



*Las funciones linealmente separables son muy escasas y además, según Minsky y Papert, aún las redes neurales multicapas no resuelven el problema pues son una extensión del perceptrón*



# Redes Neurales Multicapas

Bryson y Ho publicaron también en 1969, un artículo sobre la retropropagación (*back propagation*) que valorizaba el uso de las redes neurales multicapas

*No obstante sus trabajos no fueron tomados en cuenta y no fueron considerados sino hasta 1980 con el resurgimiento de las Redes Neurales*

El problema radicaba en como ajustar los pesos de las neuronas intermedias mientras se estaba en fase de entrenamiento



Evaluar las consecuencias de un error y dividirlo entre los pesos de las neuronas intermedias contribuyentes



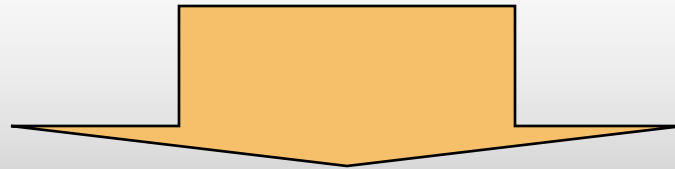
# Retropropagación

## Back Propagation

La idea es que la neurona oculta  $j$  es responsable de alguna fracción proporcional del error  $\Delta_i$

$$W_{j,i} = W_{j,i} + \alpha \cdot a_j \cdot \text{Err}_i \cdot g'(in_i) \quad \text{donde } \Delta_i = \text{Err}_i \cdot g'(in_i)$$

La fórmula propaga hacia atrás, capa por capa,  
hasta las neuronas de entrada



Este método también tiene sus limitaciones pues está demostrado que es una tarea intratable (NP-completo)



## Limitaciones Generales de las Redes Neuronales

- ¿ Cuántas capas y neuronas se deben considerar en un diseño ?



Se hace empíricamente lo cual es muy cuestionable desde el punto de vista científico

- El tiempo de aprendizaje crece exponencialmente
- La transparencia genera cajas negras que impiden saber con certeza como trabaja la red neural una vez entrenada



No tienen capacidad de explicación

- Conocimiento a priori no puede ser bien aprovechado



# *Ejemplos de Redes Neuronales*

## NETtalk (lector de textos escritos):

Aprende a pronunciar textos escritos

Recibe como entrada caracteres alfanuméricos y signo de puntuación

Entrega en las neuronas de salida fonemas

Muestra un 95% de precisión después del entrenamiento

## Lector de códigos de área en el sistema postal:

Utiliza cámaras que buscan el código de área sobre los sobres

Una vez con el código, toma una foto, la filtra e intenta reconocer

Aunque tiene buen rendimiento, rechaza muchos casos

## Robótica:

Algoritmos de navegación



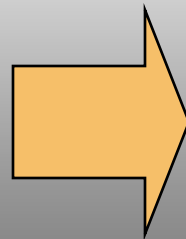


# Navegación con aprendizaje

*Diseño y construcción de un robot autónomo con el módulo RCX 1.0 para ambientes no predeterminados*

- Desplazamientos en un espacio con obstáculos desconocidos para el robot
- Sensores ultrasónicos para la entrada de información y dos motores para salida o efectores del sistema
- Programación con aprendizaje mediante una red neural ART para aprender los patrones de desplazamiento
- Construcción de un mapa del espacio indicando los obstáculos en una interfaz visible en el computador gracias a la torre infraroja

Software:



Firmware LegOs (lenguaje C)  
Protocolo LNP (Infrarojo)  
Red Neural ART



# *Conclusiones de Navegación con aprendizaje*

- Robots con poca capacidad de memoria (32K de RAM) para adaptar comportamiento (módulo de ajuste de posición tuvo que ser abandonado)
- Patrones de aprendizaje limitados y no explotaron las funcionalidades de la red neural
- Excelente desempeño de los sensores ultrasónicos para problemas de navegación robótica