# TALLER

# EDWIN JOAN GARCIA RODRIGUEZ

# CARLOS LONDOÑO

# CORPORACIÒN DE ESTUDIOS TECNOLOGICOS DEL NORTE DEL VALLE

# INTELIGENCIA ARTIFICIAL

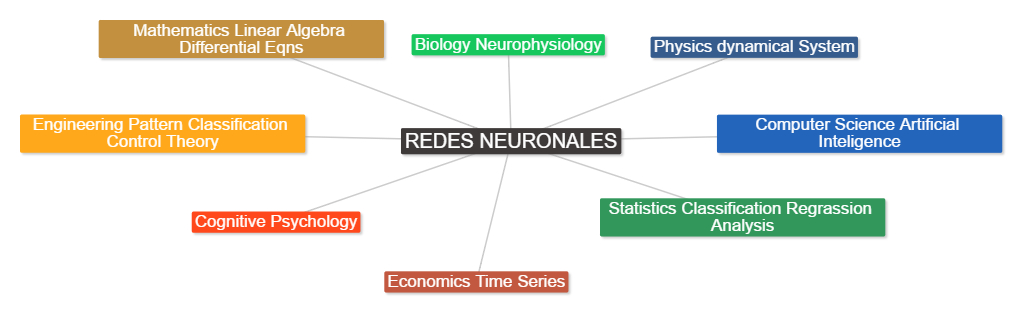
# CARTAGO VALLE

# 2016

# HISTORIA DE LAS REDES NEURONALES

Los primeros modelos de redes neuronales datan de 1943 por los neurólogos Warren McCulloch y Walter Pitts. Años más tarde, en 1949, Donald Hebb desarrolló sus ideas sobre el aprendizaje neuronal, quedando reflejado en la "regla de Hebb". En 1958, Rosenblatt desarrolló el perceptrón simple, y en 1960, Widrow y Hoff desarrollaron el ADALINE, que fue la primera aplicación industrial real.

En los años siguientes, se redujo la investigación, debido a la falta de modelos de aprendizaje y el estudio de Minsky y Papert sobre las limitaciones de la jochia. Sin embargo, en los años 80, volvieron a resurgir las RNA gracias al desarrollo de la red de Hopfield, y en especial, al algoritmo de aprendizaje de retro propagación (BackPropagation) ideado por Rumelhart y McClelland en 1986 que fue aplicado en el desarrollo de los perceptrones multicapa.2



# 2. Ventajas y Desventajas de las Redes Neuronales

**Ventajas**

Aprendizaje**:** Las RNA tienen la habilidad de aprender mediante una etapa que se llama etapa de aprendizaje. Esta consiste en proporcionar a la RNA datos como entrada a su vez que se le indica cuál es la salida (respuesta) esperada.

Auto organización**:** Una RNA crea su propia representación de la información en su interior, descargando al usuario de esto.

Tolerancia a fallos**:** Debido a que una RNA almacena la información de forma redundante, ésta puede seguir respondiendo de manera aceptable aun si se daña parcialmente.

Flexibilidad**:** Una RNA puede manejar cambios no importantes en la información de entrada, como señales con ruido u otros cambios en la entrada (ej. si la información de entrada es la imagen de un objeto, la respuesta correspondiente no sufre cambios si la imagen cambia un poco su brillo o el objeto cambia ligeramente)

Tiempo real**:** La estructura de una RNA es paralela, por lo cual, si esto es implementado con computadoras o en dispositivos electrónicos especiales, se pueden obtener respuestas en tiempo real.

Las redes neuronales**:** pueden sintetizar algoritmos a través de un proceso de aprendizaje

**Desventajas**

• Complejidad de aprendizaje para grandes tareas, cuantas más cosas se necesiten que aprenda una red, más complicado será enseñarle.

• Tiempo de aprendizaje elevado. Esto depende de dos factores: primero si se incrementa la cantidad de patrones a identificar o clasificar y segundo si se requiere mayor flexibilidad o capacidad de adaptación de la red neuronal para reconocer patrones que sean sumamente parecidos, se deberá invertir más tiempo en lograr que la red converja a valores de pesos que representen lo que se quiera enseñar.

• No permite interpretar lo que se ha aprendido, la red por si sola proporciona una salida, un número, que no puede ser interpretado por ella misma, sino que se requiere de la intervención del programador y de la aplicación en si para encontrarle un significado a la salida proporcionada.

• Elevada cantidad de datos para el entrenamiento, cuanto más flexible se requiera que sea la red neuronal, más información tendrá que enseñarle para que realice de forma adecuada la identificación.

. Otros problemas con las redes neuronales son la falta de reglas definitorias que ayuden a realizar una red para un problema dado.

. Es un argumento erróneo sostener que el tiempo empleado en el desarrollo de modelos basados en una red neuronal sea más corto que el tiempo empleado para desarrollar

# 3. Aplicaciones de las redes neuronales.

## . En la medicina

La capacidad de aprender con ejemplos y clasificar patrones, son cualidades de las redes neuronales multicapa que se han explotado en medicina como se ilustra a continuación. Análisis de imágenes En la práctica los médicos tienen que evaluar información de imágenes obtenidas con ultrasonido, resonancia magnética, medicina nuclear y radiología. Normalmente se hace un análisis cualitativo por inspección visual; sin embargo, un examen cuantitativo (2) presenta las siguientes ventajas: (i) los diagnósticos de distintos laboratorios usando los mismos criterios se pueden verificar, (ii) los datos para un sujeto se pueden comparar con una base de datos de personas normales para decidir automáticamente si existe la anormalidad, (iii) los hallazgos para un sujeto se pueden comparar con una base de datos con distintas enfermedades y detectar el tipo de anormalidad, (iv) los resultados de una serie de exámenes del mismo paciente se pueden comparar para determinar la evolución de la enfermedad y analizar la respuesta al tratamiento.

## . Medicina Nuclear

Medicina nuclear El análisis de imágenes con redes neuronales en medicina nuclear incluye tomografía por emisión de positrones (PET) y tomo grafía computarizada por emisión de un fotón (SPECT). Diagnóstico de la enfermedad de Alzheimer: se obtuvieron imágenes PET de pacientes normales y pacientes con Alzheimer. Adicionalmente, para cada sujeto se midieron ocho parámetros que representan el metabolismo de la glucosa en los ocho lóbulos del cerebro (izquierdo y derecho): frontal, parietal, temporal, y occipital. Se entrenó una red neuronal para clasificar los sujetos en las categorías normal y con enfermedad de Alzheimer, en pruebas de generalización la red clasificó correctamente el 92% de los casos. La red neuronal superó a los mí- todos estadísticos estándar como el análisis discriminante.

## . Radiología

Se han utilizado redes neuronales para analizar angiografías y mamografías.

## . Segmentación de imágenes del cerebro

la segmentación de imágenes médicas obtenidas con resonancia magnética es muy importante para la visualización de tejidos suaves en el cuerpo humano. Se entrenó una red neuronal para clasificar los siguientes seis tipos de tejido: fondo, fluido cerebroespinal, materia blanca, materia gris, cráneo y grasa. Los resultados soportan el uso de redes neuronales como método para clasificar imágenes médicas.

## . Diagnóstico de la enfermedad de Alzheimer

se obtuvieron imágenes PET de pacientes normales y pacientes con Alzheimer. Adicionalmente, para cada sujeto se midieron ocho parámetros que representan el metabolismo de la glucosa en los ocho lóbulos del cerebro (izquierdo y derecho): frontal, parietal, temporal, y occipital. Se entrenó una red neuronal para clasificar los sujetos en las categorías normal y con enfermedad de Alzheimer, en pruebas de generalización la red clasificó correctamente el 92% de los casos. La red neuronal superó a los mí- todos estadísticos estándar como el análisis discriminante.

## Enfermedades oculares

Los sistemas de diagnóstico basados en reglas se han usado exitosamente en un número de áreas en medicina; sin embargo, algunos problemas de los sistemas basados en reglas son: (i) requieren formular reglas de diagnóstico explícitas' mucho del conocimiento médico permanece implícito; (ii) formular el conocimiento implícito en reglas explí- 1. Delgado, A. Inteligencia artificial y minirobots. Bogotá: ECOE Ediciones; 1998. 2. Pattichis, C. S. and Constantinides, A. G. Medical imaging with neural networks. Proceedings of the IEEE Workshop on Neural Networks for Signal Processing 1994; 431-440. 3. Akay, Y. M., Akay, M., Welkowitz, W. citas produce distorsión y perdida de información. Además, en medicina la mayoría de las decisiones clínicas se basan en experiencia, inferencia compleja y conocimiento extensivo. En otras palabras, construir un sistema experto para diagnóstico usando reglas es una tarea lenta y costosa.

## . Caracterizacion de la dinámica en la variabilidad cardiaca

La regulación del ritmo cardiaco se lleva por un sistema dinamico operando bajo un régimen caótico

## . Detección de tumores cancerígenos

Una red neuronal entrenada localiza y clasifica en imágenes médicas la posible existencia de tumores cancerígenos

## . Predicción del riesgo de intoxicación por digoxina

En esta aplicación la tarea de la red neuronal es predecir el posible riesgo de intoxicación por digoxina que es un fármaco usado en problemas del corazón

## Perdición de enfermedades degenerativas cardiacas

Pacientes que han surgido un infarto recientemente presentan un factor de riesgo de sufrir otro se puede usar una red para modelizar el comportamiento de las arterias coronarias

## Diagnóstico de cardiopatías

este tipo de aplicación nos clasifica el electrocardiograma en diferentes tipos o clases.

# 4. Que son funciones de activación, cuales existen y para cuales redes neuronales se aplican

## FUNCIONES DE ACTIVACIÓN Y SALIDA

Además de la regla de propagación es necesario poseer expresiones para las funciones de activación (calculan la activación en función de la entrada total) y funciones de salida (calculan la salida en función de la activación).

La función de activación calcula la activación de la unidad en función de la entrada total y la activación previa, aunque en la mayor parte de los casos es simplemente una función no decreciente de la entrada total. Los tipos de función más empleados son: la función escalón, función lineal y la función sigmoidal.

La función de salida empleada usualmente es la función identidad y así la salida de la unidad de procesado es idéntica a su nivel de activación.

Las redes neuronales están formadas por un conjunto de neuronas artificiales interconectadas. Las neuronas de la red se encuentran distribuidas en diferentes capas de neuronas, de manera que las neuronas de una capa están conectadas con las neuronas de la capa siguiente, a las que pueden enviar información.

La arquitectura más usada en la actualidad de una red neuronal, se presente en la Fig. 1.8, la cual consiste en:

1. Una primera capa de entradas, que recibe información del exterior.
2. Una serie de capas ocultas (intermedias), encargadas de realizar el trabajo de la red.
3. Una capa de salidas, que proporciona el resultado del trabajo de la red al exterior.

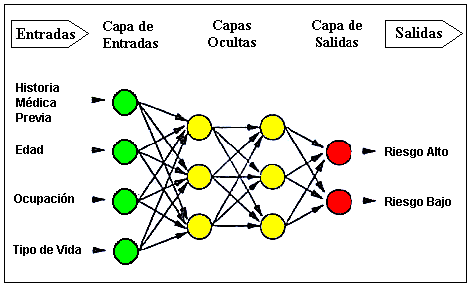


Figura 1.8 Esquema de una red neuronal antes del entrenamiento. Los círculos representan neuronas, mientras las flechas representan conexiones entre las neuronas.

El número de capas intermedias y el número de neuronas de cada capa dependerá del tipo de aplicación al que se vaya a destinar la red neuronal.

## FUNCIONES DE ACTIVACIÓN Y SALIDA

Un modelo más académico que facilita el estudio de una neurona, puede visualizarse en la Fig. 1.9.

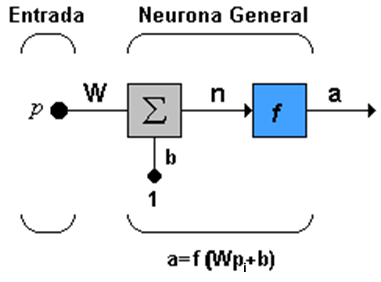


Figura 1.9 Neurona de una sola entrada.

Donde:

1. *nk: es la salida lineal de la neurona.*
2. *Wij: es la matriz de pesos.*
3. *Pi:es el patrón de entrenamiento.*
4. *bk:es el umbral de activación de la neurona.*

Las entradas a la red serán ahora presentadas en el vector **p**, que para el caso de una sola neurona contiene solo un elemento, **w** sigue representando los pesos y la nueva entrada **b** es una ganancia que refuerza la salida del sumador **n**, la cual es la salida neta de la red; la salida total está determinada por la función de transferencia , la cual puede ser una función lineal o no lineal de **n**, y que es escogida dependiendo de las especificaciones del problema que la neurona tenga que resolver; aunque las RNA se inspiren en modelos biológicos no existe ninguna limitación para realizar modificaciones en las funciones de salida, así que se encontrarán modelos artificiales que nada tienen que ver con las características del sistema biológico.

Función Escalón (Hardlim)

La Fig. 1.10, muestra como esta función de transferencia acerca la salida de la red a cero, si el argumento de la función es menor que cero y la lleva a uno si este argumento es mayor que uno. Esta función crea neuronas que clasifican las entradas en dos categorías diferentes, característica que le permite ser empleada en la red tipo Perceptrón.

http://www.hugo-inc.com/RNA/Unidad%201/imagen/ec1.3.jpg

(1.3)

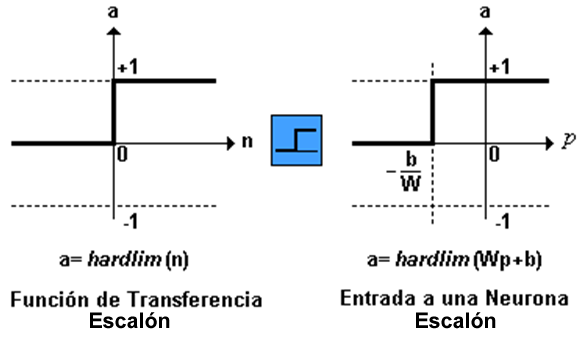


Figura 1.10 Función de Transferencia Escalón (Hardlim).

El ícono para la función escalón (Hardlim) reemplazará a la letra f en la expresión general, cuando se utilice la función Hardlim.

Una modificación de esta función puede verse en la Fig. 1.11, la que representa la función de transferencia Escalón Simétrica (Hardlims) que restringe el espacio de salida a valores entre 1 y –1.

http://www.hugo-inc.com/RNA/Unidad%201/imagen/image044.png

(1.4)

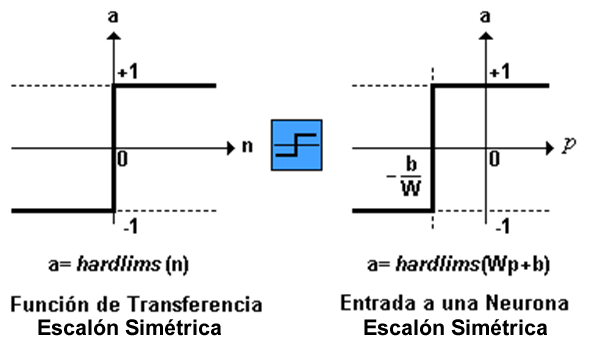


Figura 1.11 Función de Transferencia de Escalón Simétrica.

Función de Transferencia lineal (purelin)

La salida de una función de transferencia lineal es igual a su entrada, la cual se representa en la figura 1.12.

http://www.hugo-inc.com/RNA/Unidad%201/imagen/image047.png

(1.5)

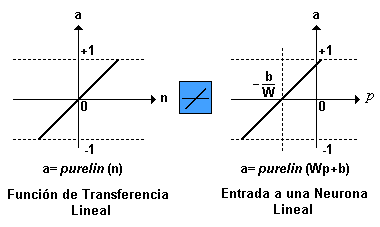


Figura 1.12 Función de Transferencia Lineal (purelin).

En la gráfica del lado derecho de la figura 1.12, puede verse la característica de la salida a de la red, comparada con la entrada p, más un valor de ganancia b, neuronas que emplean esta función de transferencia son utilizadas en la red tipo Adaline.

Función de Transferencia sigmoidal (logsig)

Esta función toma los valores de entrada, los cuales pueden oscilar entre más y menos infinito, y restringe la salida a valores entre cero y uno, de acuerdo a la expresión:

http://www.hugo-inc.com/RNA/Unidad%201/imagen/image050.png

(1.6)

Esta función es comúnmente usada en redes multicapa, como la Backpropagation, en parte porque la función logsig es diferenciable, como se muestra en la Fig. 1.13.

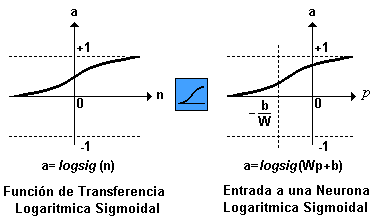


Figura 1.13 Función de Transferencia Sigmoidal.

La Fig. 1.14, hace una relación de las principales funciones de transferencia empleadas en el entrenamiento de redes neuronales.

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Relación Entrada /Salida** | **icono** | **Función en Matlab** |
| **Limitador Fuerte** | harlim | ic1 | hardlim |
| **Limitador Fuerte Simétrico** | harlims | ic2 | hardlims |
| **Lineal Positiva** | poslin | ic3 | poslin |
| **Lineal** | purelin | ic4 | purelin |
| **Lineal Saturado** | satlin | ic5 | satlin |
| **Lineal Saturado Simétrico** | satlins | ic6 | satlins |
| **Sigmoidal Logarítmico** | logsig | ic7 | logsig |
| **Tangente Sigmoidal Hiperbólica** | tangsig | ic8 | tansig |
| **Competitiva** | **a=1 Neurona con n max**  **a=0 El resto de las neuronas** | ic9 | compet |

# perceptron

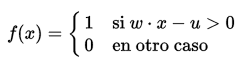
## HISTORIA

El modelo biológico más simple que un perceptrón es una neurona y viceversa. Es decir, el modelo matemático más simple de una neurona es un perceptrón. La [neurona](https://es.wikipedia.org/wiki/Neurona) es una célula especializada y caracterizada por poseer una cantidad indefinida de canales de entrada llamados [dendritas](https://es.wikipedia.org/wiki/Dendrita) y un canal de salida llamado [axón](https://es.wikipedia.org/wiki/Axon). Las dendritas operan como sensores que recogen información de la región donde se hallan y la derivan hacia el cuerpo de la neurona que reacciona mediante una [sinapsis](https://es.wikipedia.org/wiki/Sinapsis) que envía una respuesta hacia el cerebro, esto en el caso de los seres vivos.

Una neurona sola y aislada carece de razón de ser. Su labor especializada se torna valiosa en la medida en que se asocia a otras neuronas, formando una red. Normalmente, el[axón](https://es.wikipedia.org/wiki/Axon) de una neurona entrega su información como "señal de entrada" a una dendrita de otra neurona y así sucesivamente. El perceptrón que capta la señal en adelante se entiende formando una red de neuronas, sean éstas biológicas o de sustrato semiconductor (compuertas lógicas).

El perceptrón usa una [matriz](https://es.wikipedia.org/wiki/Matriz_(matem%C3%A1tica)) para representar las redes neuronales y es un discriminador terciario que traza su entrada {\displaystyle x} (un [vector](https://es.wikipedia.org/wiki/Vector) [binario](https://es.wikipedia.org/wiki/Binario)) a un único valor de salida {\displaystyle f(x)} (un solo valor binario) a través de dicha matriz.

## FORMULA MATEMATICA, EXPLIQUE SUS TERMINOS

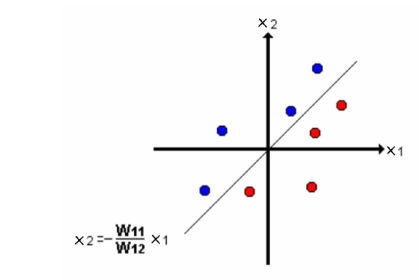
****

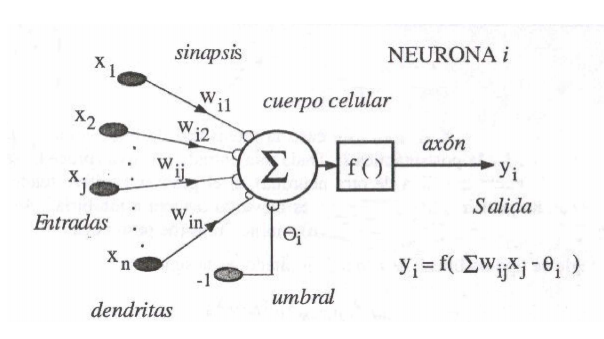
{\displaystyle f(x)={\begin{cases}1&{\text{si }}w\cdot x-u>0\\0&{\text{en otro caso}}\end{cases}}}Donde {\displaystyle w} es un vector de pesos reales y {\displaystyle w\cdot x}es el [producto escalar](https://es.wikipedia.org/wiki/Producto_escalar) (que computa una [suma ponderada](https://es.wikipedia.org/w/index.php?title=Suma_ponderada&action=edit&redlink=1)). {\displaystyle u}es el 'umbral', el cual representa el grado de inhibición de la neurona, es un término constante que no depende del valor que tome la entrada.

El valor de {\displaystyle f(x)} (0 o 1) se usa para clasificar {\displaystyle x}como un caso positivo o un caso negativo, en el caso de un problema de clasificación binario. El umbral puede pensarse de como compensar la función de activación, o dando un nivel bajo de actividad a la neurona del rendimiento. La suma ponderada de las entradas debe producir un valor mayor que{\displaystyle u} para cambiar la neurona de estado 0 a 1.

## ESTRUCTURA DE LA RED

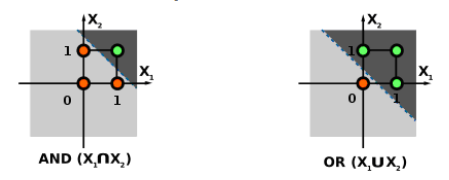
Definición La única neurona de salida del Perceptrón realiza la suma ponderada de las entradas, resta el umbral y pasa el resultado a una función de transferencia de tipo escalón. La regla e decisión es responder +1 si el patrón presentado pertenece a la clase A, o –1 si el patrón pertenece a la clase B la salida depende de la entrada neta (n = suma de las entradas xi ponderadas). · · · a1 a2 an wiai Ǉ Condición Umbral Se puede simplificar la función de activación incorporando el valor umbral en el sumatorio. Basta añar a la red una unidad extra tal que: § Siempre tiene la salida al valor 1 § Se conecta a todas las unidades de la red § El peso de la conexión es el umbral de la neurona a la que se conecta y = g (å -q = i n i i w x 1) = g (å= n i i i w x 0)



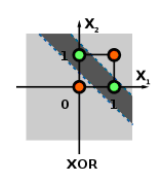


**PARA QUE SE UTILIZA**

Sólo puede discriminar patrones que pertenezcan a clases linealmente separables



Una función no separable no se puede aproximar con un solo plano de separación



## Cuál es su función de activación

En las redes neurales inspiradas sobre la biología, la función de activación es usualmente una abstracción representando una tasa de *potencial de activación* gatillándose en la celda. En su forma simplificada, esta función es binaria, esto es, se activa la neurona o no. La función se ve como {\displaystyle \phi (v\_{i})=U(v\_{i})}, donde {\displaystyle U}es la función escalón. En este caso, un gran número de neuronas deben ser usadas en computación más allá de la separación lineal de las categorías.

Una función rampa también puede ser usada para reflejar el incremento del potencial de activación que ocurre cuando la entrada se incrementa. La función podría ser de la forma {\displaystyle \phi (v\_{i})=\mu v\_{i}}, donde {\displaystyle \mu }es la pendiente. Esta función de activación es lineal, y por consiguiente tiene los mismos problemas que la función binaria. En adición, las redes neurales construidas usando este modelo tienen convergencia inestable porque a la larga, las entradas a la neurona tienden a incrementarse sin límite, esta función no es normalizable.

## EJEMPLOS DEL PERCEPTON

**EJEMPLO**

**X**

**1**

**X**

**2**

**X**

**1**

**X**

**2**

000

011

101111

1.

Pesos elegidos aleatoriamente: w1=0.5, w2=1.5, w0 = 1.5,

2.

Tomar uno a uno los cuatro patrones de entrada y se aplica el método explicado.

**Patrón de entrada: 00**

◦

Entradas: x1=0, x2=0, x0=1

◦

Pesos: w0 = 1.5, w1=0.5, w2=1.5

◦

Neti: 0(0.5) + 0(1.5) + 1(1.5) = 1.5

◦

Salida producida por f: 1 ( Neti >=0)

◦

Salida deseada: 0

◦

Error: 0

 –

1 = -1

◦

Pesos modificados:



w0(t + 1) = 1.5 + (-1)1 = 0.5



w1(t + 1) = 0.5 + (-1)0 = 0.5



w2(t + 1) = 1.5 + (-1)0 = 1.5

**Valores deseados**

de la función OR

**Conjunto de patrones**

{ 00, 01, 10, 11 }

**= 1**

**PRIMERA CORRIDA**

*Ing IvanMejiaCabrera*

**PRIMERA CORRIDA**

**Patrón de entrada: 01**

◦

Entradas: x1=0, x2=1, x0=1

◦

Pesos: w1 = 0.5, w2=1.5,w0=0.5

◦

Neti: 0(0.5) + 1(1.5) + 1(0.5) =2

◦

Salida producida por f: 1 ( Neti>=0)

◦

Salida deseada: 1

◦

Error: 1

 –

1 = 0

◦

Los pesos no se modifican:

◦

wi (t + 1) = wi ( t )

**Patrón de entrada: 10**

◦

La salida es igual a la deseada, por lo que no varían los pesos.

**Patrón de entrada: 11**

◦

La salida es igual a la deseada, por lo que no varían los pesos.



¡¡ Existe un patrón de entrada, 00, para el cual el error cometido no es cero, por lo tanto, se repite el proceso a partir de 2!!

*Ing. IvanMejiaCabrera*

**SEGUNDA CORRIDA**

3. Se toman de nuevo los cuatro patrones de entrada.

**Patrón de entrada: 00**

◦

Entradas: x1=0, x2=0, x0=1

◦

Pesos: w1 = 0.5, w2=1.5, w0=0.5

◦

Neti: 0(0.5) + 0(1.5) + 1(0.5) = 0.5

◦

Salida producida por f: 1

◦

Salida deseada: 0

◦

Error: 0

 –

1 = -1

◦

Pesos modificados

:



w0(t + 1) = 0.5 + (-1)1 = -0.5



w1(t + 1) = 0.5 + (-1)0 = 0.5



w2(t + 1) = 0.5 + (-1)0 = 1.5

**Patrón de entrada: 01**

◦

Entradas: x1=0, x2=1, x0=1

◦

Pesos: w1 = 0.5, w1=1.5, w0= -0.5

◦

Neti: 0(0.5) + 1(1.5) + 1(-0.5) =1

◦

Salida producida por f: 1

◦

Salida deseada: 1

◦

Error: 1

 –

1 = 0

◦

Los pesos no se modifican:wi (t + 1) = wi ( t )

**Patrón de entrada: 10**

◦

La salida es igual a la deseada, por loque no varían los pesos.

**Patrón de entrada: 11**

◦

La salida es igual a la deseada, por lo que no varían los pesos.

*Ing. IvanMejiaCabrera*

**TERCERA CORRIDA**



Se toman de nuevo los cuatro patrones.

**Patrón de entrada: 00**

◦

Entradas: x1=0, x2=0, x0=1

◦

Pesos: w1 = 0.5, w2=1.5, w0= -0.5

◦

Neti: 0(0.5) + 0(1.5) + 1(-0.5)= -0.5

◦

Salida producida por f: 0

◦

Salida deseada: 0

◦

Error: 0

 –

0 = 0

◦

No varían los pesoswi (t + 1) = wi ( t )

**Patrón de entrada: 10**

◦

La salida es igual a la deseada, por lo que no varían los pesos

.

**Patrón de entrada: 11**

◦

La salida es igual a la deseada, por lo que no varían los pesos

.



¡¡ Con estos nuevos pesos los patrones de entrada coinciden con las salidas, ya no se comete ningún error por lo tanto la etapa de aprendizaje concluye!

# ADALINE

## HISTORIA:

Desde hace miles de años se ha estudiado el cerebro humano con el fin de modelar su funcionamiento, pero fue con el avance de la electrónica que esta idea tomó fuerza.

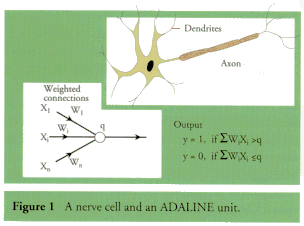
El primer paso se dio en 1943 con el neurofisiólogo Warren McCulloch y el matemático Walter Pitts quienes escribieron un documento en el cual explicaban el posible funcionamiento de las neuronas e hicieron un modelo simple de una red neuronal con circuitos eléctricos.

En 1949, Donald Hebb apoya el concepto de neurona y su funcionamiento escribiendo un libro titulado “The Organization of Behavior” en el cual comenta la actividad existente en las neuronas cada vez que son usadas.

En la década de 1950, con el avance de las computadoras, se hizo posible simular una red neuronal. Nathanial Rochester, de los laboratorios de investigación de la IBM, dio el primer paso en la simulación de la red neuronal, pero desafortunadamente fracaso. Posteriormente se hicieron otras simulaciones ya con éxito.

En 1959, Bernard Widrow y Marcian Hoff de la universidad de Stanford desarrollaron un modelo llamado “ADALINE” y “MADALINE”. ADALINE proviene de ADAptive LINear Element y MADALINE de Multiple ADAptive LINear Element (Many Adalines).

ADALINE fue desarrollado para el reconocimiento de patrones binarios, por ejemplo, predecir el siguiente bit en una línea telefónica. En la figura 1 podemos observar la representación de una ADALINE y su analogía con una Neurona.



ADALINE funciona tomando la suma de los pesos de las entradas y produce una salida con 0 o 1 dependiendo si pasa o no un umbral, esto haciendo analogía al funcionamiento de una neurona que se dispara si la actividad total procedente de las conexiones con las otras neuronas sobrepasa un nivel.

Varias ADALINE pueden ser organizadas en capas de tal manera que se obtengan grandes arquitecturas formando una red MADALINE la cual produce funciones más complicadas.

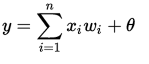
MADALINE fue la primera red neuronal aplicada a un problema real, se usó como un filtro para eliminar el eco en las líneas telefónicas.

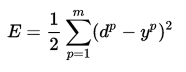
En los años siguientes el neuro-biólogo Frank Rosenblatt comenzó a trabajar con el perceptrón. Se encontró que una simple capa de perceptrones era suficiente para clasificar un conjunto de valores continuos en una de dos clases.

Desafortunadamente el perceptrón estaba limitado y fue probado por Marvin Minsky y Seymour Papert durante los años de desilusión en el libro “Perceptrons”.

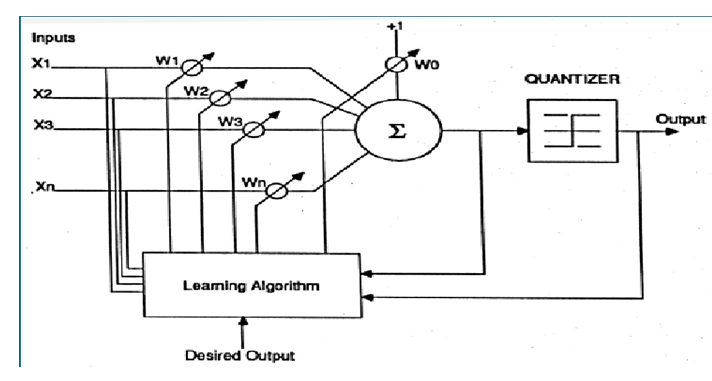
Durante los próximos 20 años se dejó atrás el estudio de las redes neuronales y fue hasta 1982 con John Hopfield que se retomó el interés en la materia. Hopfield presentó un documento a la Academia Nacional de Ciencias en el cual a través de análisis matemático mostraba como las redes neuronales funcionan y para que podrían servir.

## FORMULA MATEMATICAS:

****

****

**ESTRUCTURA:**

****

PARA QUE SE UTILIZA:

Principal mente se utiliza para el campo de procesamiento de señales concretamente en el diseño de filtros capaces de eliminar señales de ruido portadoras de información.

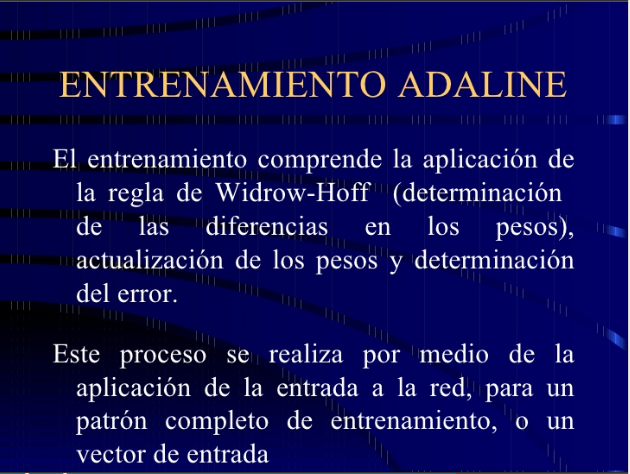
CUAL ES SU FUNCION DE ACTIVACION:

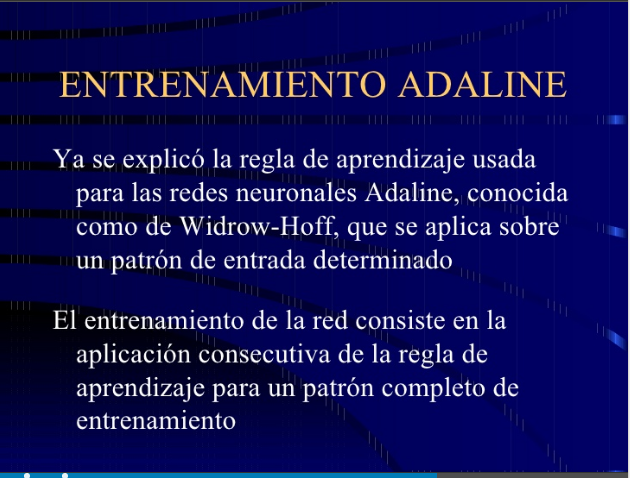
- ADALINE utiliza un aprendizaje OFF LINE con supervisión.

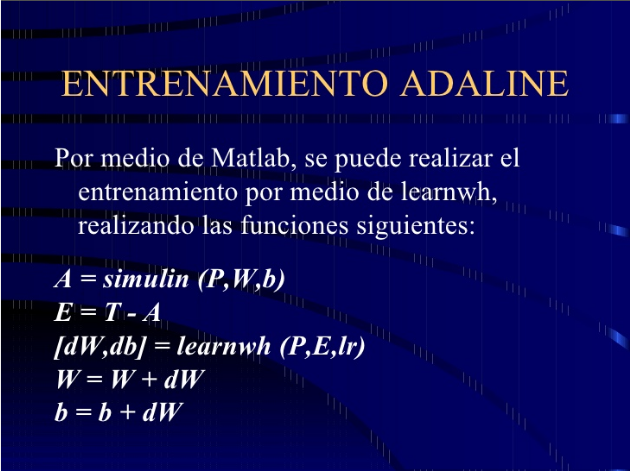
- Este aprendizaje es la llamada Regla de Widrow-Hoff (Regla Delta o Regla del Mínimo Error Cuadrático Medio LMS Least Mean Square.

Se utiliza el método de gradiente decreciente para saber en qué dirección se encuentra el mínimo global de dicha superficie. Las modificaciones que se realizan a los pesos son proporcionales al gradiente decreciente de la función de error, por lo que cada nuevo punto calculado está más próximo al punto mínimo.

## COMO SE ENTRENA UNA RED ADALINE INDIQUE EL NOMBRE DEL ALGORITMO Y SUS PASOS:







## ALGORITMO

1. inicialización de peso
2. se aplica in patrón de entrada (entrada y salida deseadas)
3. se computa la salida lineal que se obtiene de la red
4. se calcula el error cometido para dicho patrón
5. se utiliza las conexiones mediante la ecuación obtenida anteriormente
6. se repiten los pasos del dos al cinco para todos los patrones de entrenamiento
7. si el error cuadrático medio es un valor reducido aceptable, termina el proceso sino se vuelve al paso dos.

APLICACIONES:

## PROCESAMIENTO DE SEÑALES

Filtros que eliminen el ruido en señales portadoras de información.

Filtros de ecuaciones adaptivos en módems de alta velocidad

Cancelación del ruido materno de grabaciones ECG del latido del feto humano

## Eliminación de ecos en circuitos telefónicos

Canceladores adaptivos del eco para el filtrado de señales en comunicaciones telefónicas de larga distancia y comunicaciones vía satélite.

# QUE ES Y PARA QUE SE UTILIZA LAS REDES DE RETROPROPAGACION (BACKPROPAGATION)

**La propagación hacia atrás de errores** o **retro propagación** (del inglés *backpropagation*) es un [algoritmo](https://es.wikipedia.org/wiki/Algoritmo) de [aprendizaje supervisado](https://es.wikipedia.org/wiki/Aprendizaje_supervisado) que se usa para entrenar [redes neuronales artificiales](https://es.wikipedia.org/wiki/Redes_neuronales_artificiales). El algoritmo emplea un ciclo propagación – adaptación de dos fases. Una vez que se ha aplicado un patrón a la entrada de la red como estímulo, este se propaga desde la primera capa a través de las capas superiores de la red, hasta generar una salida. La señal de salida se compara con la salida deseada y se calcula una señal de error para cada una de las salidas.

Las salidas de error se propagan hacia atrás, partiendo de la capa de salida, hacia todas las neuronas de la capa oculta que contribuyen directamente a la salida. Sin embargo, las neuronas de la capa ocultan solo reciben una fracción de la señal total del error, basándose aproximadamente en la contribución relativa que haya aportado cada neurona a la salida original. Este proceso se repite, capa por capa, hasta que todas las neuronas de la red hayan recibido una señal de error que describa su contribución relativa al error total.

La importancia de este proceso consiste en que, a medida que se entrena la red, las neuronas de las capas intermedias se organizan a sí mismas de tal modo que las distintas neuronas aprenden a reconocer distintas características del espacio total de entrada. Después del entrenamiento, cuando se les presente un patrón arbitrario de entrada que contenga ruido o que esté incompleto, las neuronas de la capa oculta de la red responderán con una salida activa si la nueva entrada contiene un patrón que se asemeje a aquella característica que las neuronas individuales hayan aprendido a reconocer durante su entrenamiento.

# ¿QUE ES EL TEOREMA DE KOLMOGOROV, EXPLICAR?

El **teorema de Kolmogórov–Arnold–Moser** o **teorema KAM** es un resultado de [sistemas dinámicos](https://es.wikipedia.org/wiki/Sistemas_din%C3%A1micos) sobre la persistencia de [movimientos cuasi periódicos](https://es.wikipedia.org/wiki/Movimiento_cuasiperi%C3%B3dico). Este teorema resuelve parcialmente el [problema de los divisores pequeños](https://es.wikipedia.org/w/index.php?title=Problema_de_los_divisores_peque%C3%B1os&action=edit&redlink=1) (que origina problemas de convergencia en sistemas con múltiples frecuencias). El teorema explica cómo se modifica el aspecto de las trayectorias de un sistema integrable bajo pequeñas perturbaciones.

El movimiento en un [sistema integrable](https://es.wikipedia.org/wiki/Sistema_integrable) está confinado a una [híper]superficie [toroidal](https://es.wikipedia.org/wiki/Toro_(geometr%C3%ADa)#El_toro_en_n_dimensiones). Diferentes condiciones iniciales del sistema originan diferentes toros en el [espacio fásico](https://es.wikipedia.org/wiki/Espacio_f%C3%A1sico). Que las trayectorias de un sistema integrable de dimensión *n* están confinadas a hipersuperficies de tipo {\displaystyle \mathbb {T} ^{n}} pueden deducirse del tratamiento de las [variables acción-ángulo](https://es.wikipedia.org/wiki/Sistema_hamiltoniano_integrable#Sistema_que_admite_n_variables_.C3.A1ngulo-acci.C3.B3n), al existir *n* variables "ángulo" periódicas.

El teorema KAM establece que, si un sistema está sometido a una pequeña perturbación no lineal, algunos toros serán deformados y otros destruidos. Los que sobreviven son aquellos que tienen un cociente de frecuencias suficientemente irracional. Es decir, se destruyen aquellos cuyo cociente de frecuencias se acerca más a un número racional, dados por la relación