LINA MARIA MUÑOZ OSPINA

INTELIGENCIA ARTIFICAL (IA)

Redes Neuronales

CARLOS ALBERTO LONDOÑO LOAIZA

Ing. Sistemas y Computación

Septiembre 05 DE 2016

INGENIERIA EN SISTEMAS X

CORPORACION DE ESTUDIOS TECNOLOGICOS DEL NORTE DEL VALLE

1. **HISTORIA REDES NEURONALES**

[text2mindmap.com/Q74vmMR](http://text2mindmap.com/Q74vmMR)

1. **VENTAJAS DE LAS REDES NEURONALES**

* **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.
* **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 (por ejemplo 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).
* **Generalización**: facultad de las redes neuronales de responder apropiadamente cuando se les presentan datos o situaciones a las que no había sido expuesta anteriormente. El sistema puede generalizar laentradapara obtener una respuesta. Esta característica es muy importante cuando se tiene que solucionar problemas en los cuales la información de entrada no es muy clara.
* **Auto organización:**

 Las redes neuronales usan su capacidad de aprendizaje adaptativo para organizar la información que reciben durante el aprendizaje y/o la operación. Una RNA puede crear su propia organización o representación de la información que recibe mediante una etapa de aprendizaje, esta auto organización provoca la facultad de las redes neuronales de responder apropiadamente cuando se les presentan datos o situaciones a los que no habían sido expuestas anteriormente.

* **Fácil inserción dentro de la tecnología existente:**

 Debido a que una red puede ser rápidamente entrenada, comprobada, verificada y trasladada a una implementación hardware de bajo costo, es fácil insertar RNA para aplicaciones específicas dentro de sistemas existentes (chips, por ejemplo)

* **Dinamismo:**

Su dinamismo hace que sean capaces de estar constantemente cambiando para adaptarse a las nuevas condiciones.

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

**DESVENTAJAS DE LAS REDES NEURONALES**

* Complejidad de aprendizaje para grandes tareas, cuanto 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 mas flexible se requiera que sea la red neuronal, mas información tendrá que enseñarle para que realice de forma adecuada la identificación.

**3  APLICACIONES DE REDES NEURONALES**

1. Obtención de modelos de la retina. (biología)
2. Evaluación de probabilidad de formaciones geológicas y petrolíferas.(Medio ambiente)
3. Previsión del tiempo.(Medio Ambiente
4. Previsión de la evolución de los precios.(Finanzas)
5. Identificación de falsificaciones(Finanzas)
6. Interpretación de firmas.(Finanzas)
7. Valoración del [riesgo](http://www.monografias.com/trabajos13/ripa/ripa.shtml) de los [créditos](http://www.monografias.com/trabajos15/financiamiento/financiamiento.shtml).
8. Medicina Analizadores del habla para la ayuda de audición de sordos profundos.
9. Inspección de [calidad](http://www.monografias.com/trabajos11/conge/conge.shtml).
10. Manufacturación Robots automatizados y [sistemas de control](http://www.monografias.com/trabajos6/sicox/sicox.shtml) (visión artificial y [sensores](http://www.monografias.com/trabajos10/humed/humed.shtml) de [presión](http://www.monografias.com/trabajos11/presi/presi.shtml), [temperatura](http://www.monografias.com/trabajos/termodinamica/termodinamica.shtml), [gas](http://www.monografias.com/trabajos10/gase/gase.shtml), etc.
11. **FUNCION DE ACTIVACIÓN:**

La función de activación combina el potencial postsinaptico, que nos proporciona la función de propagación con el estado actual de la neurona para conseguir el estado futuro de activación de la neurona. Esta función es representada como  un nodo que define la salida de un nodo dada una entrada o un conjunto de entradas. Se podría decir que un circuito estándar de computador se comporta como una red digital de funciones de activación al activarse como "ON" (1) u "OFF" (0), dependiendo de la entrada.

**Valor o Estado de Activación**: El estado de activación en un tiempo t es un valor numérico que se representa por un vector Aj(t) y puede ser continuo o discreto, limitado o ilimitado. Lo mas simple es considerado “1” y “0” como on/off o activo /pasivo

**Función de Activación (f):** Se utiliza para determinar el nuevo estado de activación Aj (t+1) de la neurona j-esima, teniendo en cuenta la entrada total calculada Netj y el estado de activación Aj(t)

**Tipos de Funciones de Activación:**

**Lineal**:  
Algunas redes neuronales usan esta función de activación como el Adeline por su eficiencia y facilidad.

**Escalón:**  
Esta función es la más usada para redes neuronales binarias ya que no es lineal y es muy simple. Algunas redes que usan esta función son el Perceptrón y Hopfield.

**Hiperbólicas o tangenciales:**Las redes con salidas continuas, como el Perceptron multicapa con retro propagación, usan esta función ya que su algoritmo de aprendizaje necesita una función derivable

1. **PERCEPTRON**

**a). HISTORIA**

En 1943, Warren McCulloch y Walter Pitts introdujeron una de las primeras neuronas artificiales. La característica principal de su modelo de neurona es que una suma ponderada de las señales de entrada se compara con un umbral para determinar la neurona de salida. Cuando la suma es mayor o igual al umbral, la salida es 1. Cuando la suma es menor que el umbral, la salida es 0. A finales de 1950 Frank Rosenblatt y otros investigadores desarrollaron una clase de redes neuronales llamadas perceptrones. Las neuronas de estas redes eran similares a las de McCulloch y Pitts.

La contribución clave de Rosenblatt fue la introducción de una regla de aprendizaje para la formación de redes perceptrón para resolver problemas de reconocimiento de patrones. Demostró que su regla de aprendizaje siempre convergirá a los pesos correctos de la red, si existen pesos que solucionan el problema. El Perceptrón pudo incluso aprender cuando se inicializaba con valores aleatorios de sus pesos y bias.

Ellos demostraron que las redes perceptrón eran incapaces de implementar ciertas funciones elementales. No fue sino hasta la década de los 8O’s que estas limitaciones fueron superadas con las redes perceptrón mejoradas (multicapa) asociadas con reglas de aprendizaje.

**b)**  **Formula Matemática**

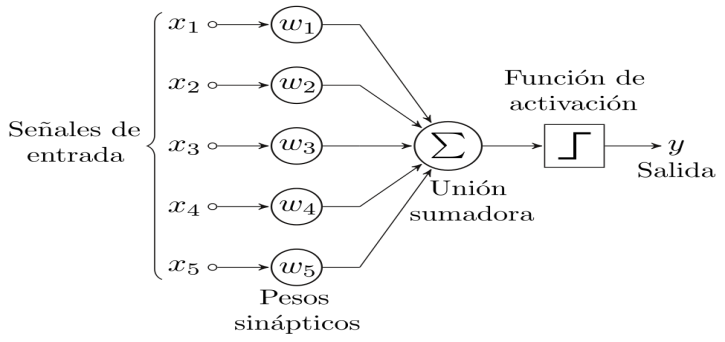
**f**= puede ser la función hardlim (función de transferencia)

**Wi** = Pesos Sinápticos; pesos con lios que se ponderan los valores de entrada

**Xi**= Valores de Entrada

**Bias**=

**c) Estructura del Perceptrón**

****

**d) Uso del Perceptrón**

Se llega a la teoría de que el uso del Perceptrón consistía en la separabilidad estadística que utilizaba para caracterizar las propiedades más visibles de las redes de interconexión ligeramente aleatorias.

**e) Función de Activación del Perceptrón**

La función que usa el perceptrón es la de Escalón

**f) Algoritmo de entrenamiento del Perceptrón**

**Paso 0: Inicialización**

Inicializar los pesos sinápticos con números aleatorios del intervalo [-1,1].Ir al paso 1 con k=1

**Paso 1: (k-ésima iteración)**

Calcular

**Paso 2:** **Corrección de los pesos sinápticos**

Si z(k)≠y(k) modificar los pesos sinápticos según la expresión:

w*j*( ) k +1 = w*j*(k )+η [z(k )− y(k )] xj (k ), j = 1,2,...,n +1

**Paso 3:** **Parada**

Si no se han modificado los pesos en las últimas p iteraciones, es decir,

W*f* ( r)= W*f* (k), j=1,2,…*n+*1 r= k+1,…, k+*p*,

parar. La red se ha estabilizado. En otro caso, ir al Paso 1 con k=k+1.

**g)** **Ejemplos de usos del Perceptrón**

* Predicción de cambios en el valor de los instrumentos financieros
* Diagnósticos médicos,
* NETtalk es un Perceptrón que es capaz de transformar texto en ingles en sonido individual,
* la pronunciación con la utilización de un sintetizador de voz

**6) ADALINE**

1. **Historia**

A finales de los 50 Bernard widrow empezaba a trabajar en redes neuronales, en el tiempo en el que Frank Rosenblat desarrollo la regla de aprendizaje de Perceptrón.

En 1960 Widrow y su asesorado Marcían Hoff presentaron la red Adaline (adaptative linear neuron ) y una regla de aprendizaje la cual denominaron LMS

La red Adaline es muy parecida al perceptrón excepto que su función de transferencia es linear, en vez de escalón; tanto Adaline como el perceptrón sufren la misma limitación , solo pueden resolver problemas linealmente separables.

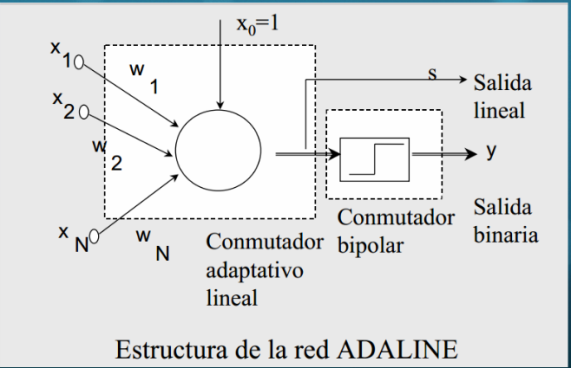
1. **Fórmula Matemática**

**Y**= Estimado de la neurona de la salida esperada

**W**= Vector de Pesos

**X**= Vector de Entrada

1. **Estructura de Adaline**



1. **Uso de Adaline**

El uso de Adaline es principalmente en el campo de procesamiento de señales

1. **Función de Activación**

Adaline usa como función de activación de Tipo Lineal

**f) Algoritmo de Entrenamiento de Adaline**

**Paso 1:** Inicializar los pesos de forma aleatoria

**Paso 2**: Presentar un patrón de entrada

**Paso 3:** Calcular la salida, compararla con la deseada y obtener la diferencia:

(d p - y p)

**Paso 4:** Para todos los pesos, multiplicar dicha diferencia por la entrada correspondiente y ponderarla por la tasa de aprendizaje

**Paso 5:** Modificar el peso sumando al valor antiguo la cantidad obtenida en el paso 4

**Paso 6:** Repetir los pasos 2, 3 y 4 para todos los patrones

**Paso 7**: Repetir los pasos 2, 3, 4,5 y 6 tantos ciclos hasta cumplir el criterio de parada

**Criterios de parada:**

**Criterio 1:** Fijar un número de ciclos máximo. Dicho número debe garantizar que el error cuadrático para los patrones de entrenamiento se haya estabilizado.

**Criterio 2:** Cuando el error cuadrático sobre los patrones de entrenamiento no cambia durante x ciclos.

**Criterio 3: C**uando el error cuadrático sobre los patrones de validación no aumenta o se mantiene estable a lo largo de x ciclos.

1. **Aplicaciones de Adaline**

* Específicamente en el diseño de filtros, estas son capaces de eliminar ruido en señales portadoras de información.
* Filtros adaptivos: Un Adaline tiene la suficiente capacidad  de predecir el valor de una señal en el instante
* Filtros de ecualización adaptativos en módems de alta velocidad.
* Cancelación del ruido materno de grabaciones ECG del latido del feto humano.
* Eliminación de ecos en circuitos telefónicos.

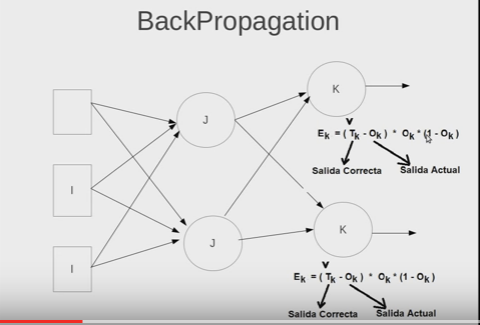
**7.   Redes de Retro propagación**

Las redes de retro propagación es un 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

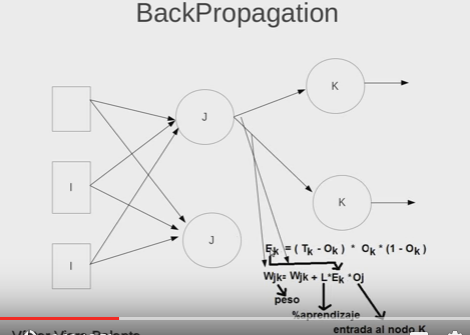
La Retro propagación consiste en propagar el error hacia atrás, es decir, de la capa de salida hacia la capa de entrada, pasando por las capas ocultas intermedias y ajustando los pesos de las conexiones con el fin de reducir dicho error

**8. Algoritmo Backpropagation**

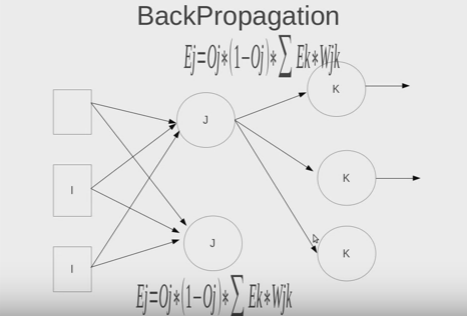
Se inicializan los pesos de la red con valores aleatorios; Se presenta un valor de entrada y se especifica la salida deseada.

****

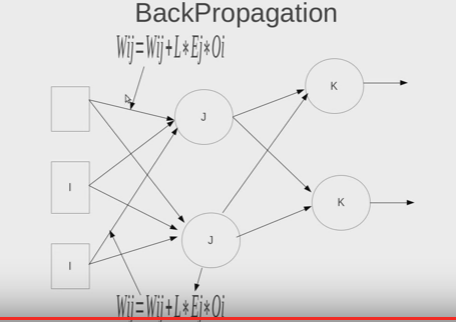
Se realiza el cálculo de los pesos



Se Calculan los términos de error para cada una de las neuronas.

****

Ajuste de Pesos en la capa de Entrada.

****

1. **La ley cero-uno de Kolmogórov**

 Es un teorema de la teoría de las [probabilidades](https://es.wikipedia.org/wiki/Probabilidad), donde se dice que la probabilidad de cierto tipo de [eventos](https://es.wikipedia.org/wiki/Evento_estad%C3%ADstico) llamados *eventos de cola* (tail event en inglés) es cero o uno. Los eventos de cola son aquellos eventos definidos por una sucesión infinita de eventos [independientes](https://es.wikipedia.org/wiki/Independencia_estad%C3%ADstica), pero que son independientes de cualquier subconjunto finito de éstos.

la probabilidad de que dos hechos debidos al azar (y que cumplen unas condiciones determinadas), pasen conjuntamente... es "pequeñísima".

BIBLIOGRAFIA

ESCOM. (22 de 09 de 2009). *http://es.slideshare.net.* Recuperado el 30 de 09 de 2016, de http://es.slideshare.net: http://es.slideshare.net/mentelibre/redes-neuronales-adaline

Malaga, A. B. (21 de 06 de 2007). *http://www.redes-neuronales.com.es.* Recuperado el 30 de 09 de 2016, de http://www.redes-neuronales.com.es: http://www.redes-neuronales.com.es/tutorial-redes-neuronales/historia-de-las-redes-neuronales.htm

muñozp. (12 de 09 de 2010). *http://www.lcc.uma.es.* Recuperado el 30 de 09 de 2016, de http://www.lcc.uma.es: http://www.lcc.uma.es/~munozp/documentos/modelos\_computacionales/temas/Tema4MC-05.pdf

SYSTEM, G. (19 de 07 de 2013). *www.youtube.com.* Recuperado el 30 de 09 de 2016, de www.youtube.com: https://www.youtube.com/watch?v=0odQ286nsIY