TALLER #2

INTELIGENCIA ARTIFICIAL

TITULO:

REDES NEURONALES ARTIFICIALES

PRESENTADO POR:

FAUSTO ANTONIO ANDRADE VERA

DOCENTE:

CARLOS ALBERTO LONDOÑO LOAIZA

CORPORACIÓN DE ESTUDIOS TECNOLÓGICOS DEL NORTE DEL VALLE

CARTAGO VALLE

SEPTIEMBRE/05/2016

Contenido

[Taller No. 2 4](#_Toc460861404)

[1. Consultar la historia de las redes neuronales artificiales, y haciendo uso de la herramienta text2mindmap, crear un mapa conceptual, que permita evidenciar los casos importantes. 4](#_Toc460861405)

[2. Nombre 6 o más ventajas y desventajas que tiene el uso de las redes neuronales artificiales para la solución de problemas. 5](#_Toc460861406)

[Ventajas del uso de RNA: 5](#_Toc460861407)

[Desventajas del uso de RNA: 6](#_Toc460861408)

[3. Nombre 10 aplicaciones de las redes neuronales. 6](#_Toc460861409)

[4. ¿Qué son funciones de activación, cuales existen y para cuales redes neuronales aplican? 7](#_Toc460861410)

[5. Perceptron: 8](#_Toc460861411)

[Historia 8](#_Toc460861412)

[Fórmula matemática, explique sus términos 9](#_Toc460861413)

[Dibuje la estructura 9](#_Toc460861414)

[¿Para qué se usa? 9](#_Toc460861415)

[¿Cuál es su función de activación? 10](#_Toc460861416)

[¿Cómo se entrena un perceptron, indique el nombre del algoritmo y sus pasos? 10](#_Toc460861417)

[Nombre 5 ejemplos donde se evidencie el uso del perceptron. 10](#_Toc460861418)

[6. Adaline 11](#_Toc460861419)

[Historia 11](#_Toc460861420)

[Fórmula matemática, explique sus términos 12](#_Toc460861421)

[Dibuje la estructura 13](#_Toc460861422)

[¿Cuál es su función de activación? 13](#_Toc460861423)

[¿Cómo se entrena una red adaline, indique el nombre del algoritmo y sus pasos? 13](#_Toc460861424)

[Algoritmo de Aprendizaje Off –Line con supervisión LMS en Adaline 14](#_Toc460861425)

[Nombre 5 ejemplos donde se evidencie el uso de adaline. 14](#_Toc460861426)

[7. ¿Qué es y para que se usa las redes de retropropagacion (Backpropagation)? 14](#_Toc460861427)

[Definición: 14](#_Toc460861428)

[Emplea: 15](#_Toc460861429)

[8. Explique el algoritmo backpropagation, usando imágenes. 15](#_Toc460861430)

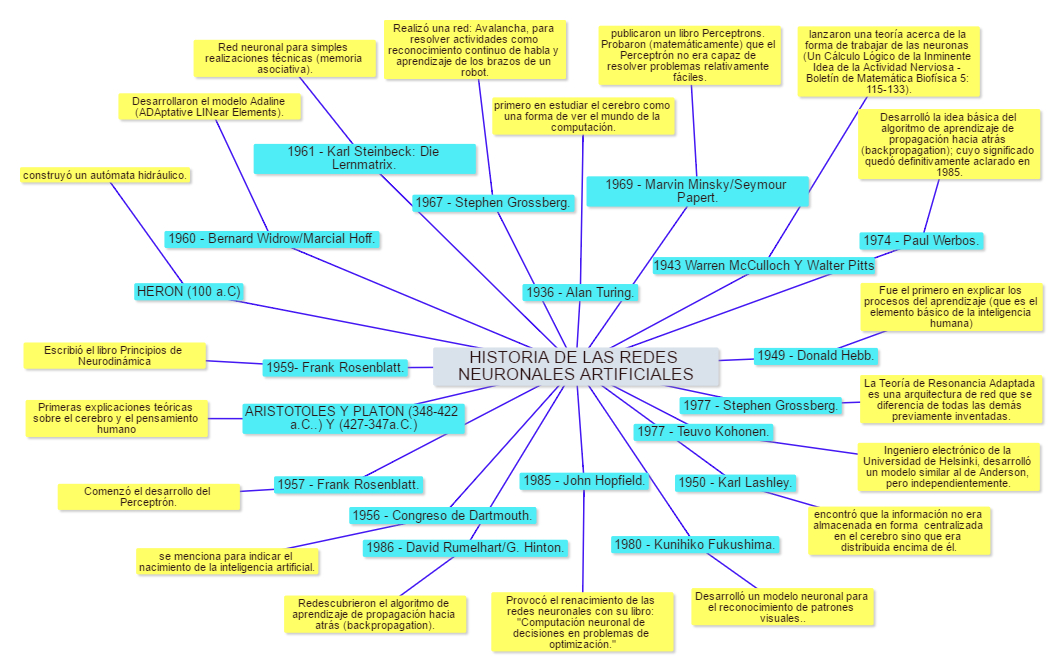
[DESCRIPCIÓN DEL PROCESO DEL ALGORITMO BACKPROPAGATION 15](#_Toc460861431)

[10. ¿Qué es el teorema de Kolmogorov, explicar? 18](#_Toc460861432)

[Bibliografía 19](#_Toc460861433)

Taller No. 2

1. Consultar la historia de las redes neuronales artificiales, y haciendo uso de la herramienta text2mindmap, crear un mapa conceptual, que permita evidenciar los casos importantes.



2. Nombre 6 o más ventajas y desventajas que tiene el uso de las redes neuronales artificiales para la solución de problemas.

Ventajas del uso de RNA:

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.

Clasificadores:Son excelentes como clasificadores/reconocedores de patrones – y pueden ser usadas donde las técnicas tradicionales no funcionan.

Entrada de datos: Pueden manejar excepciones y entradas de datos anormales, muy importante para sistemas que manejan un amplio rango de datos (sistemas de radar y sonar, por ejemplo).

Muchas redes neuronales son biológicamente plausibles, lo que significa que pueden proveer pistas de cómo trabaja el cerebro según progresen.

Desventajas del uso de RNA:

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 más flexible se requiera que sea la red neuronal, más información tendrá que enseñarle para que realice de forma adecuada la identificación.

Falta de reglas definitorias que ayuden a realizar una red para un problema dado.

La llamada caja negra el problema es que cuando modelamos estadísticamente somos capaces de ver que variables forman parte del modelo o cuales de las que finalmente se utilizaron para modelar fueron seleccionadas por los algoritmos para predecir o clasificar, podemos ver sus pesos y la ecuación final, cosa que no es posible en las redes neuronales.

1. Nombre 10 aplicaciones de las redes neuronales.
2. Conversión Texto a Voz
3. Procesado Natural del Lenguaje
4. Compresión de Imágenes
5. Reconocimiento de Caracteres
6. Reconocimiento de Patrones en Imágenes
7. Problemas de Combinatoria
8. Procesado de la Señal
9. Predicción
10. Modelado de Sistemas
11. Filtro de Ruido
12. Modelos Económicos y Financieros
13. Servo Control
14. ¿Qué son funciones de activación, cuales existen y para cuales redes neuronales aplican?

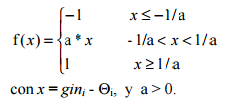
Función de activación **(*activation function*):**

Una neurona biológica puede estar activa (excitada) o inactiva (no excitada); es decir, que tiene un “*estado de activación*”. Las neuronas artificiales también tienen diferentes estados de activación; algunas de ellas solamente dos, al igual que las biológicas, pero otras pueden tomar cualquier valor dentro de un conjunto determinado.

La *función activación* calcula el estado de actividad de una neurona; transformando la entrada global (menos el umbral, Θi) en un valor (estado) de activación, cuyo rango normalmente va de (0 a 1) o de (–1 a 1). Esto es así, porque una neurona puede estar totalmente inactiva (0 o –1) o activa (1). La función activación, es una función de la entrada global (*gini*) menos el umbral (Θi). Las funciones de activación más comúnmente utilizadas se detallan a continuación:

Funciones de Activación:

1. Función Lineal

****

1. Función Sigmoidea

****

1. Función Tangente Hiperbólica

****

1. Función Gausiana

****

Para cuales redes neuronales aplican?

Aplican para las redes neuronales de Perceptron y Adaline

1. Perceptron:

Historia

En 1969, Minsky y Papert, demuestran que el perceptrón simple y [ADALINE](https://es.wikipedia.org/wiki/ADALINE) no puede resolver problemas no lineales (por ejemplo, [XOR](https://es.wikipedia.org/wiki/Disyunci%C3%B3n_exclusiva)). La combinación de varios perceptrones simples podría resolver ciertos problemas no lineales pero no existía un mecanismo automático para adaptar los pesos de la capa oculta. Rumelhart y otros autores, en 1986, presentan la "Regla Delta Generalizada" para adaptar los pesos propagando los errores hacia atrás, es decir, propagar los errores hacia las capas ocultas inferiores. De esta forma se consigue trabajar con múltiples capas y con funciones de activación no lineales. Se demuestra que el perceptrón multicapa es un aproximador universal. Un perceptrón multicapa puede aproximar relaciones no lineales entre los datos de entrada y salida. Esta red se ha convertido en una de las arquitecturas más utilizadas en el momento.

Fórmula matemática, explique sus términos

**Wi** = Es un peso modificable asociado con la señal de entrada Xi.

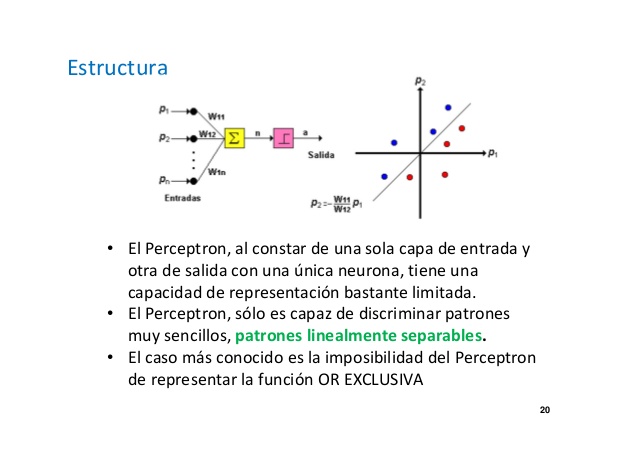
**Xi =** Patrones de entrada representados por el vector X.

**b =** Valor umbral = W0 puede verse como un peso entre la unidad de entrada y una señal ficticia de valor X0 = 1

**Hardlims =** sgn(x): la función signo

Step (x): la función paso

Dibuje la estructura



¿Para qué se usa?

Para clasificar problemas linealmente separables, cosa que ya se podía hacer mediante métodos estadísticos, y de una forma mucho más eficiente.

¿Cuál es su función de activación?

Función de activación de las neuronas de la capa de salida es de tipo escalón, dando de esta manera sólo salidas binarias.

¿Cómo se entrena un perceptron, indique el nombre del algoritmo y sus pasos?

El entrenamiento del Perceptron consiste en presentar a la red todos los elementos del conjunto de entrenamiento constituido por parejas de vectores (entrada y salida deseada) de forma secuencial.

Algoritmo de entrenamiento del Perceptron:

1. Se inicializa la matriz de pesos y el valor de la ganancia, por lo general se asignan valores aleatorios a cada uno de los pesos Wi y al valor b.
2. Se presenta el primer patrón a la red, junto con la salida esperada en forma de pares entrada/salida {p1, d1}, {p2, d2}, {p3, d3}…
3. Se calcula la salida de la red por medio de:

Donde f puede ser la función hardlim o hardlims.

Nombre 5 ejemplos donde se evidencie el uso del perceptron.

Se conocen cinco aplicaciones tecnológicas extendidas:

1. Reconocimiento de textos manuscritos
2. Reconocimiento del habla
3. Simulación de centrales de producción de energía
4. Detección de explosivos
5. Identificación de blancos de radares
6. 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.

Fórmula matemática, explique sus términos

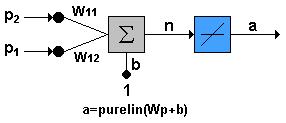
Generalmente se compone de una sola capa de *n* neuronas (por tanto *n* valores de salida) con *m* entradas con las siguientes características:

* Las *m* entradas representan un vector {\displaystyle x}X de entrada que pertenece al espacio {\displaystyle R^{m}}R^m.
* Por cada neurona, existe un vector {\displaystyle w} de pesos sinápticos que indican la fuerza de conexión entre los valores de entrada y la neurona. En la práctica representan la ponderación de cada entrada sobre la neurona.
* Una constante {\displaystyle \theta }Ɵ.
* La salida {\displaystyle y}ƴ de la neurona se representa por la función de activación, que se define como {\displaystyle y=\sum \_{i=1}^{n}x\_{i}w\_{i}+\theta }

Generalmente se compone de una sola capa de *n* neuronas (por tanto *n* valores de salida) con *m* entradas con las siguientes características:

* Las *m* entradas representan un vector {\displaystyle x}x de entrada que pertenece al espacio {\displaystyle R^{m}}
* Por cada neurona, existe un vector {\displaystyle w}w de pesos sinápticos que indican la fuerza de conexión entre los valores de entrada y la neurona. En la práctica representan la ponderación de cada entrada sobre la neurona.
* Una constante {\displaystyle \theta }C

Dibuje la estructura



En donde:

*p*: Patrones de entrada

b: Umbrales de activación

*a*: Salida de la neurona

La salida de la red está dada por:

ecuacion

1. ¿Para qué se usa?

La red Adaline ha sido ampliamente utilizada en el procesamiento de Señales, resolver problemas linealmente separables.

¿Cuál es su función de activación?

Función de activación de las neuronas de la capa de salida es de tipo Lineal.

¿Cómo se entrena una red adaline, indique el nombre del algoritmo y sus pasos?

El entrenamiento de la red consiste en adaptar los pesos a medida que se vayan presentando los patrones de entrenamiento y salidas deseadas para cada uno de ellos, para cada combinación E/S se realiza un proceso automático de pequeños ajustes en los valores de los pesos hasta que se obtienen las salidas correctas.

Algoritmo de Aprendizaje Off –Line con supervisión LMS en Adaline

1. Se aplica un patrón de entrada P.
2. Se obtiene la salida del ALC y se calcula la diferencia con respecto a la deseada (error).
3. Se actualizan los pesos.
4. Se repiten pasos 1 a 3 con todos los vectores de entrada.
5. Si el Error es un valor aceptable, detenerse, si no repetir algoritmo.

Nombre 5 ejemplos donde se evidencie el uso de adaline.

Se puede evidenciar el uso de adaline en:

1. Filtros de ecuación adaptativos en módems de alta velocidad.
2. Filtrado de señales de comunicación de larga distancia y comunicaciones vía satélite.
3. eliminación del ruido materno de las grabaciones electrocardiografías (EGC) del latido de corazón del feto humano.
4. Procesamiento de señales digitales.
5. En la eliminación de ecos en circuitos electrónicos
6. ¿Qué es y para que se usa las redes de retropropagacion (Backpropagation)?

Definición:

La importancia de la red backpropagation consiste en su capacidad de auto adaptar los pesos de las neuronas de las capas intermedias para *aprender* la relación que existe ente un conjunto de patrones de entrada y sus salidas correspondientes. Es importante la capacidad de *generalización*, facilidad de dar salidas satisfactorias a entradas que el sistema no ha visto nunca en su fase de entrenamiento. La red debe encontrar una representación interna que le permita generar las salidas deseadas cuando se le dan entradas de entrenamiento, y que pueda aplicar, además, a entradas no presentadas durante la etapa de aprendizaje para clasificarlas.

Emplea:

Ya que esta red aprovecha la naturaleza paralela de las redes neuronales se utiliza para reducir el tiempo requerido por un procesador secuencial para determinar la correspondencia entre unos patrones dados. Además el tiempo de desarrollo de cualquier sistema que se esté tratando de analizar se puede reducir como consecuencia de que la red puede aprender el algoritmo correcto sin que alguien tenga que deducir por anticipado el algoritmo en cuestión.

1. Explique el algoritmo backpropagation, usando imágenes.

DESCRIPCIÓN DEL PROCESO DEL ALGORITMO BACKPROPAGATION

La unidad procesadora básica de la red Backpropagation se representa en la Figura (5.1). Las entradas se muestran a la izquierda, y a la derecha se encuentran unidades que reciben la salida de la unidad procesadora situada en el centro de la figura.

****

Normalmente, la Backpropagation utiliza tres o más capas de unidades procesadoras. La Figura (5.2) muestra la topología backpropagation típica de tres capas.



La Figura (5.3) muestra la fase de propagación hacia adelante.



El valor de error calculado para este tipo de unidades procesadoras se obtiene a partir de la ecuación (5.4).



La Figura (5.5) representa la obtención del valor δj para las unidades de las capas ocultas.



10. ¿Qué es el teorema de Kolmogorov, explicar?

Cualquier función continua *f*(*x1*, *x2*,…,*xn*) definida en [0 1]*n* , *n*≥2, se puede representar Mediante la expresión



Donde las funciones *gi* son funciones continuas y reales de una sola variable, elegidas adecuadamente, y las funciones φ*ij* son continuas y monótonas crecientes independientes de *f*.

Este resultado establece que cualquier función vectorial continua de *Rn* en *Rm* definida sobre un conjunto compacto se puede expresar en términos de sumas y composiciones de un número finito de funciones de una sola variable.

Sin embargo, las redes neuronales constituyen una clase de funciones más reducida que la establecida por el teorema de Kolmogorov ya que limitan las funciones φ*ij* a funciones sigmoidales. Una demostración rigurosa de que las redes neuronales son aproximadotes universales fue dada independientemente por Cybenko (1989), Funahashi (1989) y por Hornik, Stinchcombe y White (1989).

# **Bibliografía**

http://text2mindmap.com/gKG2xZH

http://redes-neuronales.wikidot.com/definicion-ventajas-desventajas

https://www.uaeh.edu.mx/docencia/P\_Presentaciones/huejutla/sistemas/redes\_neuronales/introduccion.pdf

http://www.ciberesquina.una.edu.ve:8080/2014\_2/350\_E.pdf

https://www.frro.utn.edu.ar/repositorio/catedras/quimica/5\_anio/orientadora1/monograias/matich-redesneuronales.pdf

https://es.wikipedia.org/wiki/Perceptr%C3%B3n\_multicapa

http://image.slidesharecdn.com/utpsirns4redperceptron-140919121756-phpapp02/95/utp-sirns4red-perceptron-20-638.jpg?cb=1411129425

https://es.wikipedia.org/wiki/Red\_neuronal\_artificial#Ejemplos

http://cursos.itam.mx/akuri/2002/S22002/RNS/Presentaciones/Adaline/ADALINEYMADALINE2.doc

http://www.repositoriodigital.ipn.mx/bitstream/handle/123456789/8628/Archivo%20que%20incluye%20portada,%20%C3%ADndice%20y%20texto.pdf?sequence=1

https://www.frro.utn.edu.ar/repositorio/catedras/quimica/5\_anio/orientadora1/monograias/matich-redesneuronales.pdf

http://www.lcc.uma.es/~munozp/documentos/modelos\_computacionales/temas/Tema5MC-05.pdf