

2020

PERCEPTRÓN Y LÓGICA DIFUSA: Computación Blanda

JULIAN GIRALDO CARDONA
JUAN DAVID ALVAREZ
OCTUBRE DE 2020



1 CONTENIDO

1	CONTENIDO	1
2	PRESENTACIÓN	2
3	EL PERCEPTRÓN	4
4	LÓGICA DIFUSA	7
5	CONCLUSIONES	8
6	BIBLIOGRAFÍA	9

2 PRESENTACIÓN

La presente monografía está orientada a la descripción de los elementos básicos de las neuronas artificiales, en particular el perceptrón, y la teoría fundamental de la lógica difusa.

En el documento se analizan los diferentes elementos que componen ambas tecnologías, mostrando las relaciones matemáticas que dan soporte a las funcionalidades tanto del perceptrón como a los factores de incertidumbre que dan sentido a la lógica difusa.

A grandes rasgos, las redes neuronales se basan en los modelos que subyacen a las redes neuronales biológicas. El siguiente diagrama adelante algunos elementos presentes en esta tecnología.

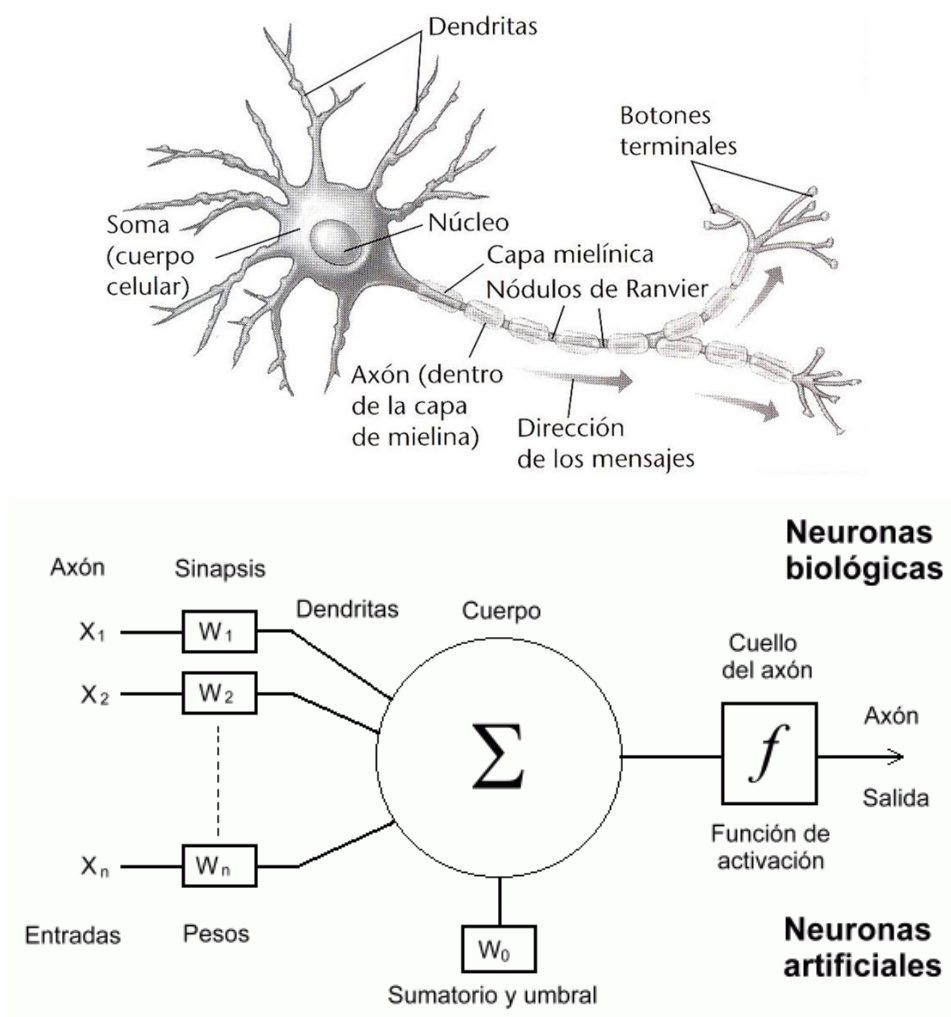


Fig. 1. Similitud entre neuronas biológicas y neuronas artificiales

En la figura 1 se puede observar la similitud que tiene una neurona biológica con una artificial y esto demuestra que lo que estamos desarrollando es una abstracción de cómo funciona una neurona biológica. En la neurona artificial tenemos unas entradas llamadas Axón, y luego a esto se le da un peso que significa la importancia de cada valor de entrada, dicho peso es representado por la variable W_1 hasta W_n y esto va dirigido a un núcleo principal del procesamiento donde se recolectan todos los valores de entrada, luego se establece un umbral que determina la importancia de cada una de la entrada mediante la función de activación y por último llegamos a un resultado gracias a la función. Este proceso por más simple que parezca, es el que realizamos en nuestras neuronas en un millón de veces más, miles de procesos trabajando simultáneamente para procesar información y así tomar decisiones.

La lógica difusa se basa en la concepción de que la verdad (y la falsedad) no son absolutas. Por este motivo, todos los conceptos que concibe el ser humano tienen cierto grado de certeza, el cual se expresa fácilmente si recurrimos a un esquema como el que se ve a continuación.

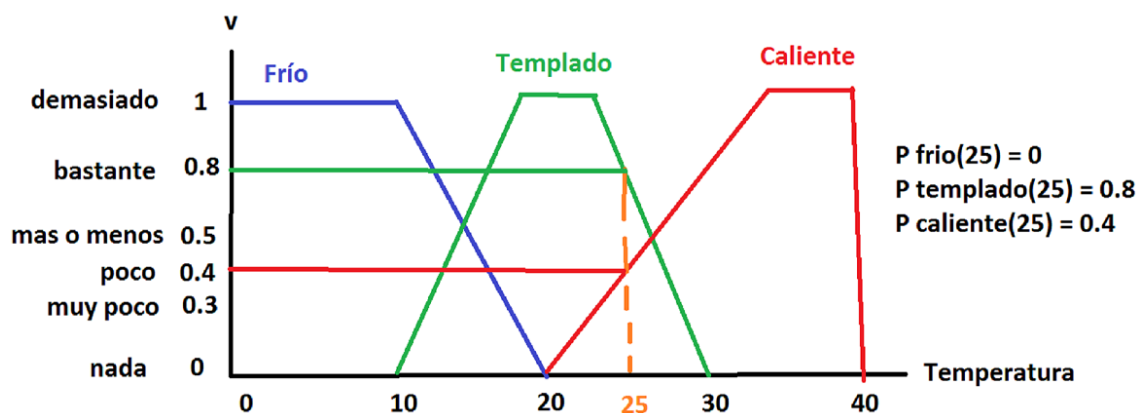


Fig. 2 Gráfico de lógica difusa temperatura

En este esquema se afirma que el Frío, la sensación de Templado, y algo que es Caliente, son curvas que varían de acuerdo con la temperatura, según se ve. En el caso particular de tener una temperatura ambiente de 25 grados, dicha temperatura tendrá un valor de verdad respecto de “Caliente” de sólo 0.4. En cambio, los 25 grados representarán, en la curva de “Templado”, un valor de verdad de 0.8. Se aprecia, además, que dichos valores se relacionan, de manera bastante cercana, con frases y/o palabras que utiliza el ser humano para describir situaciones de la vida real.

En las próximas secciones se verán estas tecnologías con un mayor grado de detalle.



AUTORES: JULIAN GIRALDO CARDONA 1004752912

julian.giraldo2@utp.edu.co

<https://github.com/JulianGiraldoC>

JUAN DAVID ALVAREZ 1112791148

Juan.alarez2@utp.edu.co

<https://github.com/DavidAlvarez1998>

3 EL PERCEPTRÓN

A continuación, se presenta el algoritmo básico para la conversión numérica basada en divisiones sucesivas.

Las divisiones sucesivas es un método para convertir un numero decimal a binario, este algoritmo consiste en tomar el numero decimal (dividendo) y dividirlo entre 2 cuantas veces sea necesario hasta que el resultado sea 1. Siempre nos interesa la parte entera del resultado, y precisamente esa es la que tenemos en cuenta para seguir dividiendo. Por último, tomamos el último resultado y los cocientes de abajo hacia arriba.



Fig. 3 Ejemplo división sucesiva.

El perceptrón es básicamente una red neuronal artificial en forma de discriminador lineal, este tiene una gran similitud con una red neuronal biológica y fácilmente se puede hacer una comparación entra las dos neuronas, así como la neurona biológica tiene unos canales de entrada llamados dendritas y un canal de salida llamado axón. El artificial tiene lo mismo, pero en este caso es de forma matemática en la que opera, tiene sus variables de entrada y tiene una salida.

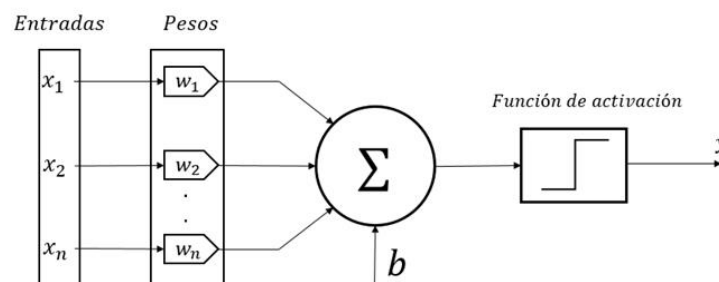


Fig 4 Neurona artificial

En la fig 4. Se puede observar en la función de activación una figura geométrica, esta figura representa el proceso de identificación de los datos para así hacer las diferentes operaciones necesarias hasta entregar un resultado final. En esa función se define un umbral, el umbral es un componente fundamental en la red neuronal artificial ya que desde este valor se eligen los mejores candidatos para saber qué tanta calidad tienen por así decirlo. Por ejemplo, tenemos una serie de valores de entrada con sus respectivos pesos y un umbral definido.

$$\text{UMBRAL} = U = 0.5$$

X1	X2	W1	W2	$X1*W1+X2*W2 > U$	Salida
0	0	0.3	0.3	$0*0.3 + 0*0.3 = 0.0$ NO	0
0	1	0.3	0.3	$0*0.3 + 1*0.3 = 0.3$ NO	0
1	0	0.3	0.3	$1*0.3 + 0*0.3 = 0.3$ NO	0
1	1	0.3	0.3	$1*0.3 + 1*0.3 = 0.6$ SI	1

Fig. 5 Ejemplo tabla AND con umbral

Tenemos diferentes valores en X1 y X2, cada valor cuenta con W1 y W2 que significa el peso que tienen, entonces se hace una sumatoria y se guarda el valor, este resultado se compara con el umbral, si supera el umbral la salida será el número binario 1, y si es menor que el umbral la salida será 0. Esto es igual que el comportamiento de la compuerta lógica AND ya que el único caso en que la salida es 0, es cuando X1 vale 1 y X2 vale 1.

$$\text{UMBRAL} = U = 0.5$$

X1	X2	W1	W2	$X1*W1+X2*W2 > U$	Salida
0	0	0.6	0.6	$0*0.6 + 0*0.6 = 0.0$ NO	0
0	1	0.6	0.6	$0*0.6 + 1*0.6 = 0.6$ SI	1
1	0	0.6	0.6	$1*0.6 + 0*0.6 = 0.6$ SI	1
1	1	0.6	0.6	$1*0.6 + 1*0.6 = 0.6$ SI	1

Fig. 5 Ejemplo tabla OR con umbral

En este caso, tenemos la operación OR representada por un umbral como se observa en la figura 5. Lo único que cambio fue el peso de cada valor de entrada y con solo hacer esto ocurrió un resultado completamente distinto, de este modo, se puede afirmar que el peso para cada valor de entrada es una parte fundamental para llegar a un valor de salida.

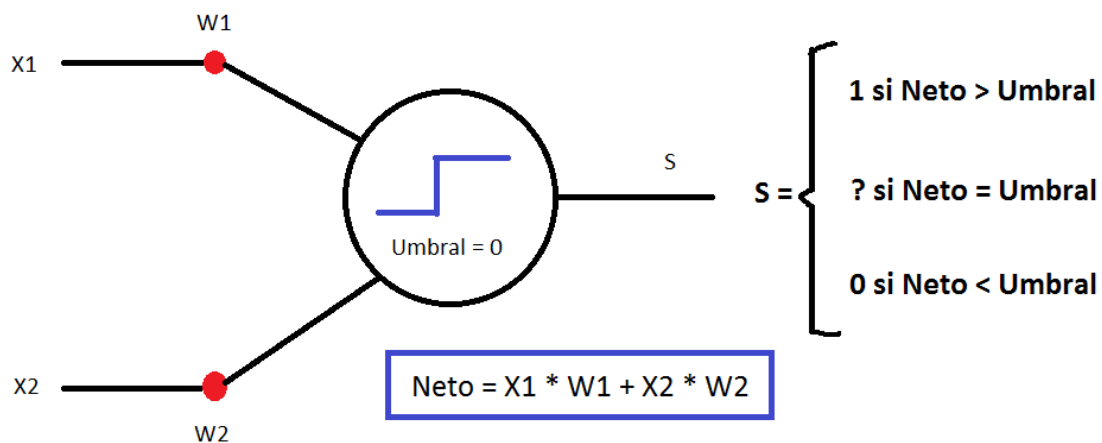


Fig. 6 Representación de peso neto y umbral para tomar decisiones

En esta figura se muestra como se vería desglosada la tabla que vimos anteriormente, $X1$ y $X2$ son valores de entrada y los puntos rojos son el peso de cada entrada, en la operación del umbral se guarda cada valor y en el canal s se hace la operación lógica. El neto es el peso en total, es decir, la sumatoria.

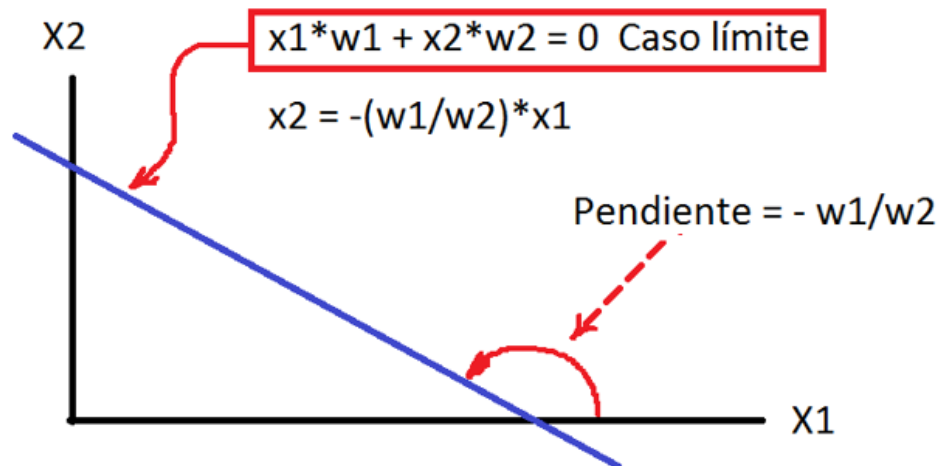
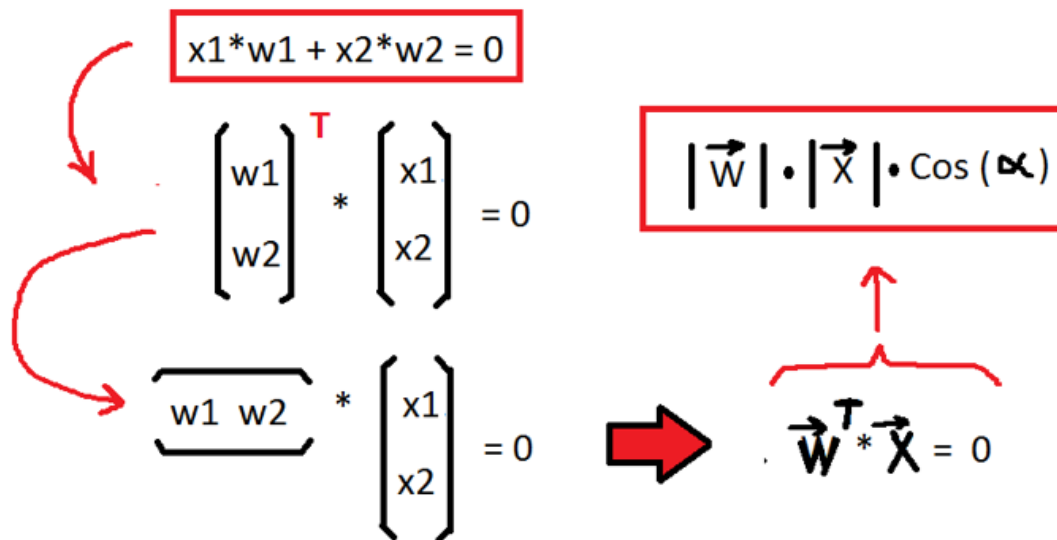


Fig. 7 Recta frontera

La recta frontera es la recta que pasa por los dos ejes X_1 y x_2 cuando el caso es limite, es decir, cuando la suma del producto de cada peso es igual a 0, la pendiente se puede hallar mediante $-w_1/w_2$. Dicha pendiente ayuda a determinar la orientación de la recta.

Análisis vectorial



$$x_1 * w_1 + x_2 * w_2 = 0$$

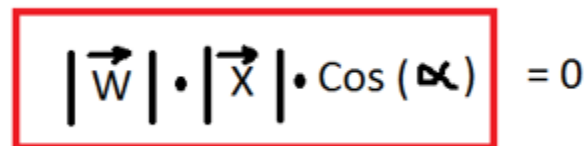
$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix}^T * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$\begin{bmatrix} w_1 & w_2 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0 \Rightarrow \vec{W}^T * \vec{X} = 0$$

$$|\vec{W}| * |\vec{X}| * \cos(\alpha)$$

Fig. 8 Análisis vectorial

En el manejo de los datos para el procesamiento nos encontramos que estos se encuentran en arreglos (arrays), entonces se pueden aplicar distintas operaciones vectoriales a estos datos para así optimizar el procesamiento, por ejemplo, aquí tenemos el producto entre el peso de cada variable X_1 y X_2 y se dice que la suma del producto de cada una es igual a 0 y esto se evidencia en el campo vectorial ya que, aunque hagamos una transversal, el resultado sigue siendo 0. Y esto se puede demostrar gracias a $W * X = 0$



$$|\vec{W}| * |\vec{X}| * \cos(\alpha) = 0$$

Fig. 9 Perpendicularidad

Como se puede apreciar en la Fig. 9, se dice que el producto de sus magnitudes por el coseno del ángulo entre dos vectores es igual a 0, esto varía depende del valor del ángulo del coseno. Para que el resultado sea igual a 0 el ángulo α debe ser igual a 90. Cabe resaltar que el vector W debe ser perpendicular al vector X para que se cumpla este caso.

$$|\vec{W}| \cdot |\vec{X}| \cdot \cos(\alpha) = 0$$

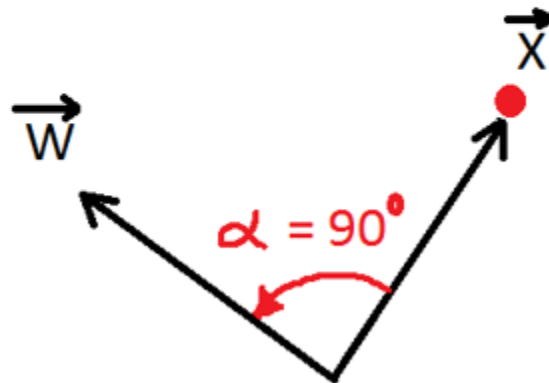


Fig. 10 Condición de frontera

Si el vector W es perpendicular a todos los puntos de una cierta recta, entonces, dicha recta cumple con la restricción de dividir el espacio en dos zonas, una de las cuales la salida es uno y en la otra es cero. Según se vio en el perceptrón, gracias a esto es que podemos realizar las operaciones and, or, xor.

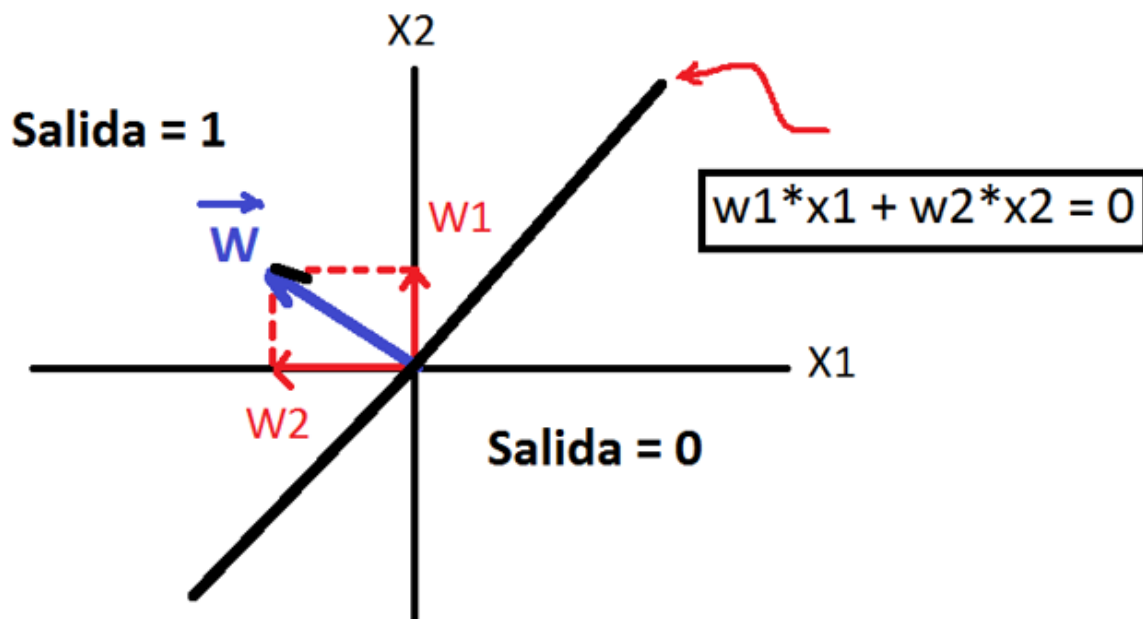


Fig. 11 Separación espacial

Como se mencionaba anteriormente de dividir el espacio en dos zonas, una en 1 y otra en 0. Para llegar a la zona de valor 1 basta con hallar la resultante cuando w_1 es positivo y w_2 negativo en los ejes x_1 y x_2 .

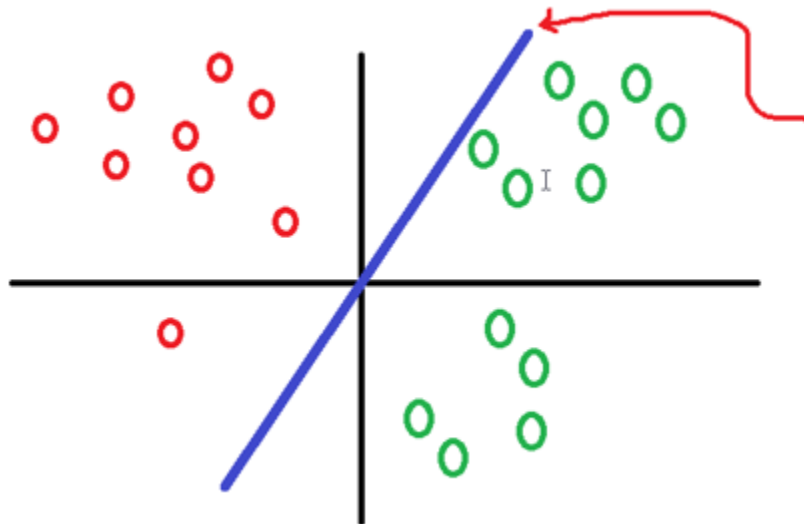


Fig. 12 Solución probable

La recta de color azul es una solución probable, aquí nos encontramos con una incógnita. ¿Cómo se puede calcular automáticamente esta solución probable de modo que divida el espacio en dos áreas de clasificación? Lo que tenemos que hacer es modificar los pesos de manera iterativa (n veces), hasta que dichos pesos sean un vector perpendicular a la recta solución.

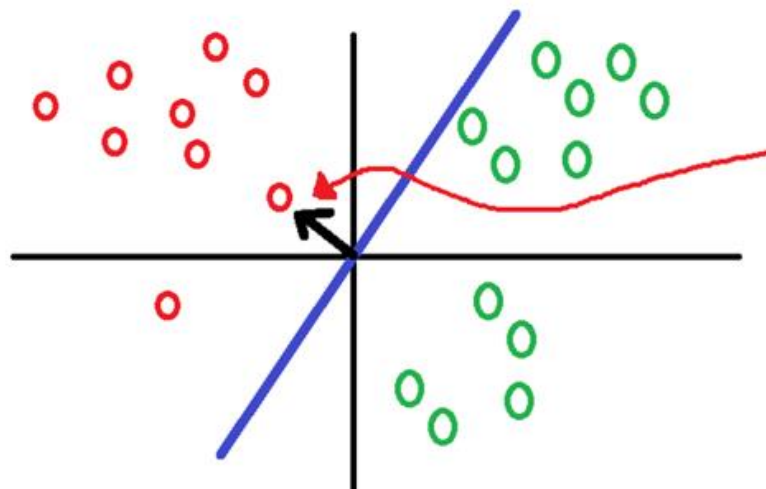


Fig. 13 Solución iterativa

El vector W de la figura 13, está representado por la recta de color negro en el segundo cuadrante, y este debe ser encontrado de manera automática en el sistema, como se mencionaba anteriormente, es perpendicular a la recta de color azul.

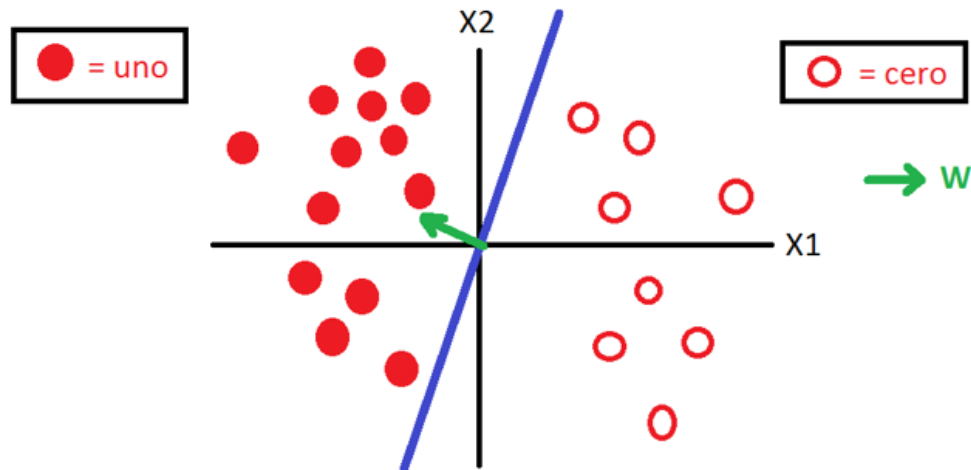


Fig. 14 Solución intuitiva, W apunta hacia los "1"

Luego de entrenar una serie de datos y dividirlos entre "0" y "1" podemos desarrollar una solución intuitiva, esto se logra por medio del vector W que tenemos que hallar, podemos darnos cuenta que estamos en el espacio de los "1" cuando el vector haga intersección con uno de los puntos.

A continuación, se presentan otros casos donde se entrenan los "1" y "0", pero estos se encuentran en una zona diferente.

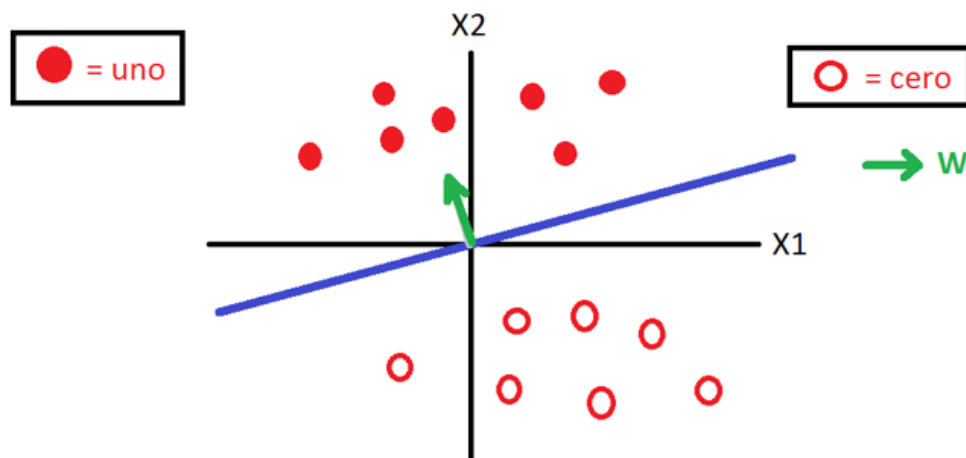


Fig. 15 Solución "Intuitiva" caso 2 W apunta hacia los "1"

Al cambiar la dirección de la recta que hace la división entonces el vector W también cambia su orientación ya que siempre tiene que estar perpendicular.

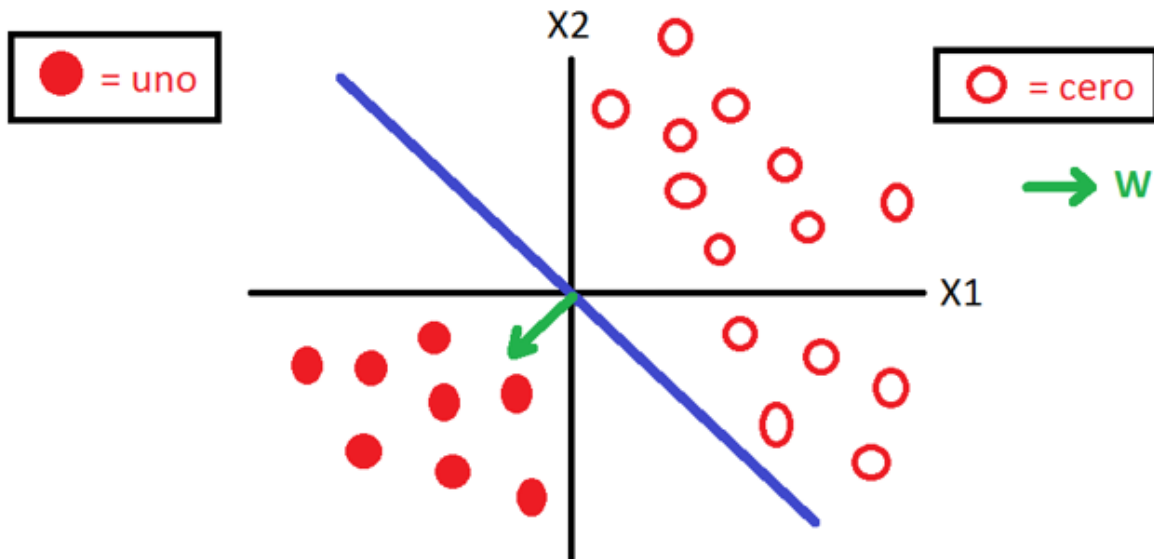


Fig. 16 Solución "Intuitiva" caso 3 W apunta hacia los "1"

Nuevamente, al cambiar la orientación de la recta divisora, el vector W también lo hace y así, de forma perpendicular, podemos encontrar los "1".

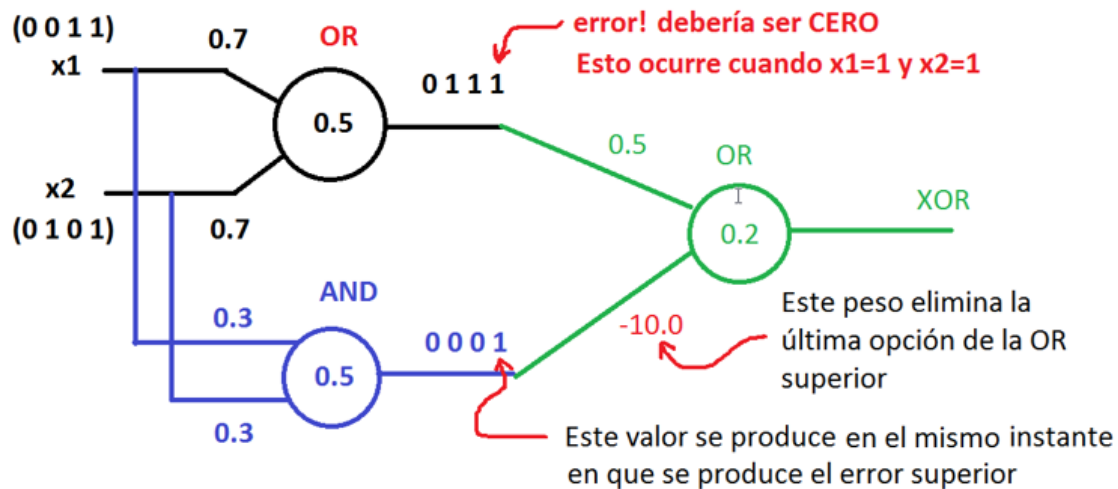


Fig. 17 Solución Compuerta XOR

Como se puede observar en la fig. 17, podemos crear una compuerta XOR por medio de las entradas x_1 y x_2 , para cada compuerta que entran, primero pasan por un peso que este se le agrega y luego se procesa por el umbral para saber cuál será el resultado. Como se demuestra en la primera compuerta, x_1 toma los valores de $[0, 0, 1, 1]$ y x_2 $[0, 1, 0, 1]$ entonces al pasar por el peso 0.7 con el umbral 0.5 se hace la siguiente operación: $0 \cdot 0.7 + 0 \cdot 0.7$ y como no supera el umbral 0.5 entonces el resultado es 0. El siguiente resultado sería: $0 \cdot 0.7 + 1 \cdot 0.7$ y como el resultado supera el umbral 0.5 entonces la salida es 1, y así sucesivamente por cada umbral establecido. Por la parte final se agrega un peso negativo de gran tamaño para así anular el peso de arriba y obligar al sistema a que la salida sea menor del umbral.

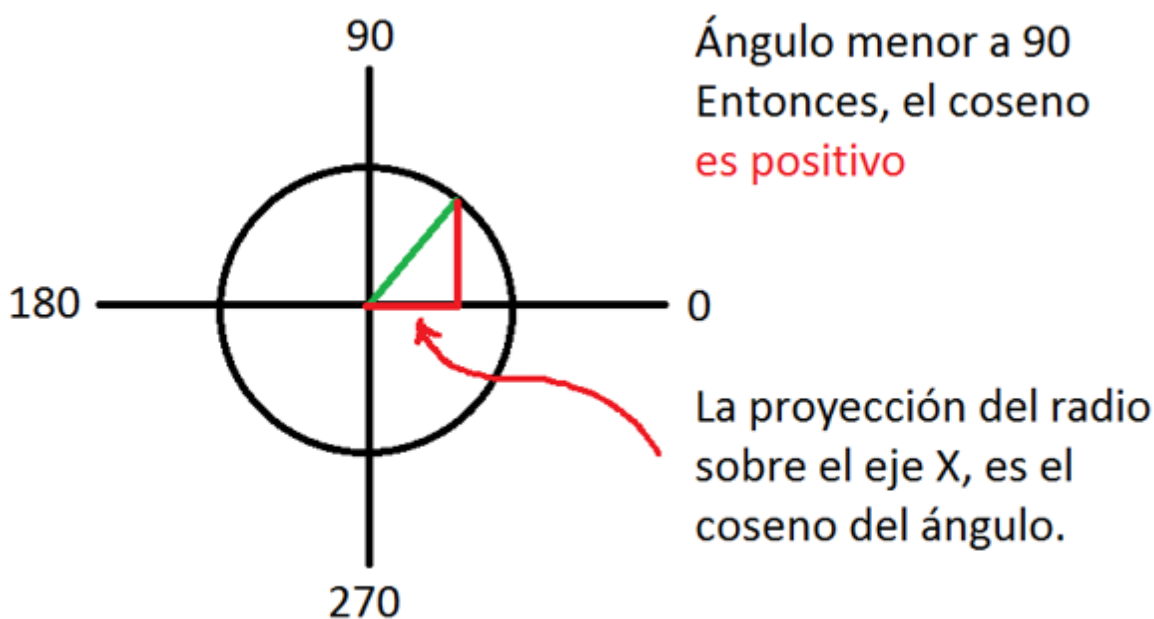


Fig. 18 Función coseno (positivo)

Establecemos un círculo, creando así un plano cartesiano indicando cada grado en cada eje, dentro del círculo, en la parte central, indicamos un radio y a ese radio marcamos el ángulo que es aproximadamente de 45 grados. Este ángulo es positivo y al hacer la proyección del radio podemos hallar el coseno del ángulo mediante funciones trigonométricas.

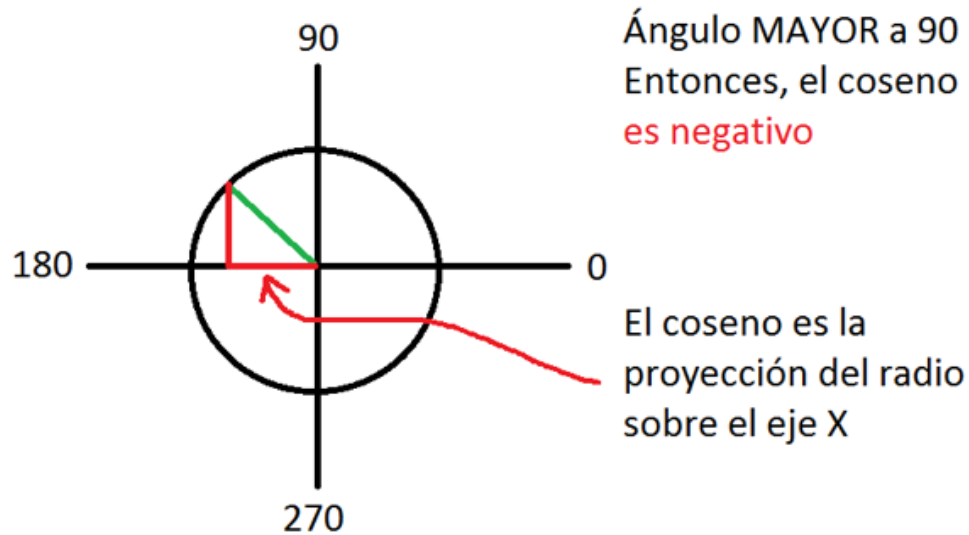
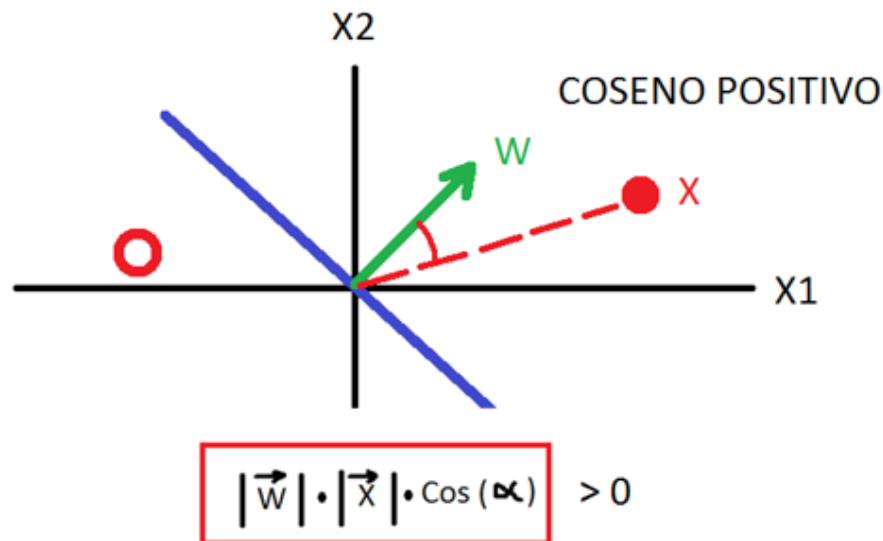


Fig. 19 Función coseno (negativo)

Al incrementar el ángulo (mayor de 90 grados), la función coseno se vuelve negativa y esto lo podemos hallar del mismo modo que la figura anterior, haciendo proyección del radio y así utilizar las funciones trigonométricas.



SALIDA = 1, el punto es ROJO

Fig. 20 Principio de entrenamiento

Mediante los ángulos el coseno podemos determinar la salida de unos datos de entrenamiento, esto se logra mediante el producto de los vectores por el coseno de alpha, entonces, si el coseno es positivo (alpha menor de 90 grados), la salida será 1. Y se representa por el punto rojo.

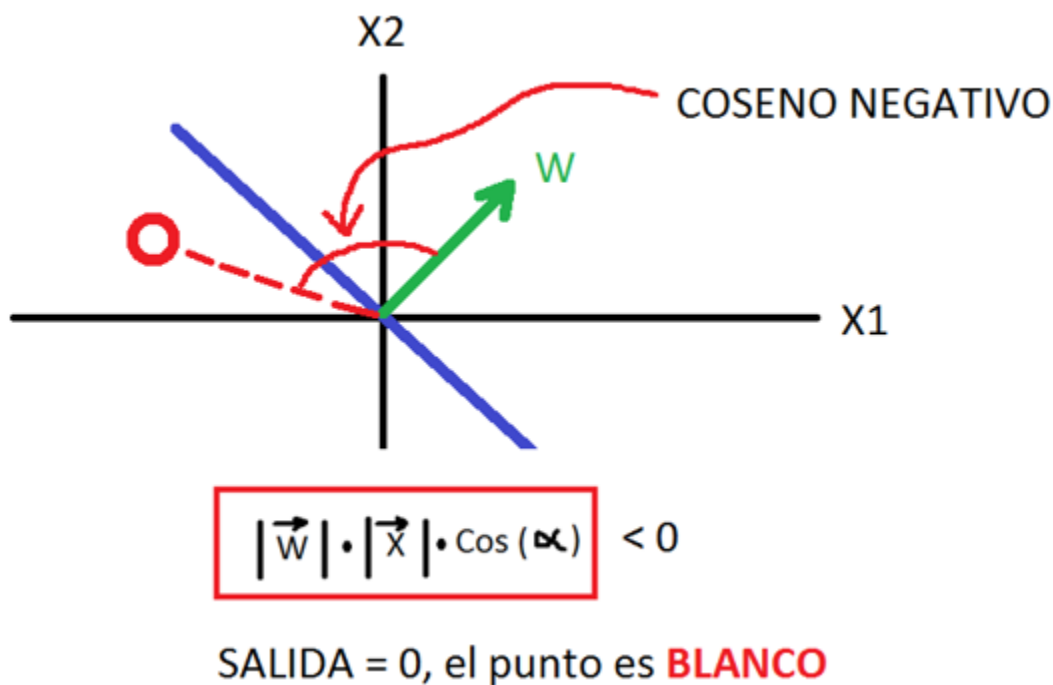


Fig. 21 Principio de entrenamiento 2

Como en el caso anterior, al modificar el ángulo del coseno entonces el resultado del producto de los vectores será diferente y si este es menor que 0 entonces la salida será 0. Y de esta forma es como se entrenan los datos divididos por una recta. Para saber qué ángulo tiene dicho punto, debe de tomar de referencia el vector W y así medir el ángulo hasta el punto.

Problema: el punto está en la zona en donde se produce la Salida = 1, pero él debe producir una Salida = 0

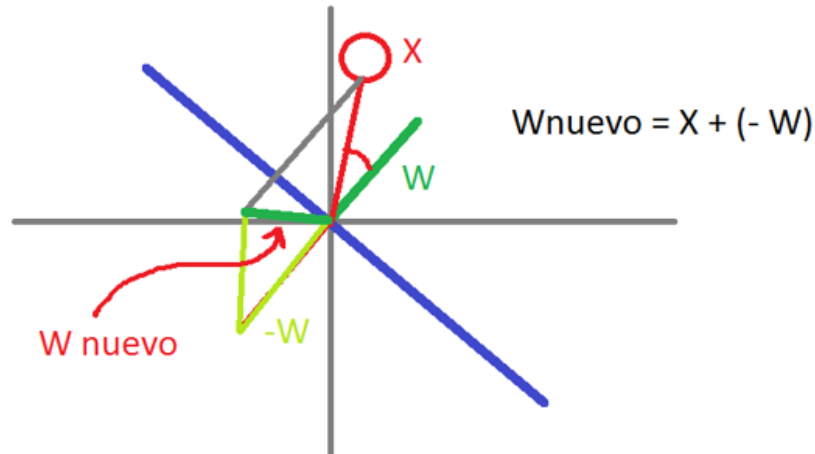


Fig. 22 Algoritmo de entrenamiento – error

En la fig. 22 encontramos un problema de entrenamiento de datos, el punto está en la zona de salida de los “1” pero en realidad su valor es “0” entonces hay que encontrar una forma de crear un nuevo vector de referencia W_{nuevo} para así añadir un nuevo ángulo y modificar la salida.

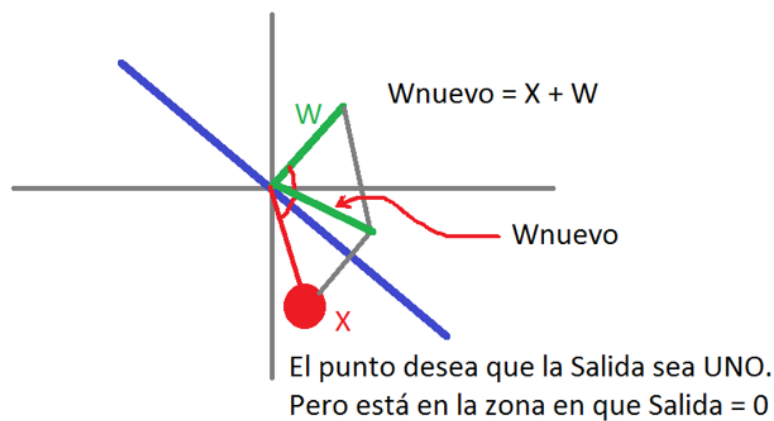


Fig. 23 Algoritmo de entrenamiento – error

En este caso, tenemos un valor de salida 1 que se encuentra en la zona de salida 0 entonces se desea modificar el ángulo y se hace lo mismo, creamos un nuevo vector llamado W_{nuevo} y este lo que hace es crear una nueva referencia para así modificar el ángulo y mover el punto a la zona que se requiere.

4 LÓGICA DIFUSA - INTRODUCCIÓN

La lógica difusa tiene su origen en el año 1965, investigada por primera vez por el ingeniero eléctrico de origen azerbaiyano y nacionalizado en estados unidos Lotfy A. Zadeh.

Zadeh nace en 1921 en Bakú, Azerbaiyán. Cuando tenía 10 años de edad se muda con su familia a Irán donde entra a la Universidad de Teherán, más tarde se mudaría a los estados unidos donde continuaría con sus estudios en el Instituto de Tecnología de Massachusetts (MIT), en la universidad de Columbia y finalmente en la Universidad de Berkeley.

Zadeh introduce el concepto de conjunto difuso, a diferencia de la teoría de conjuntos clásica en la que un elemento pertenece o no pertenece al conjunto (1,0), un conjunto difuso permite que un elemento pertenezca o no parcialmente al conjunto (0.1, 0.2, ...0.9), esto permite que la lógica difusa se adapte mucho mejor al mundo real, incluso puede comprender y funcionar con nuestras expresiones, del tipo “hace mucho calor”, “no es muy alto”, “el ritmo del corazón está un poco acelerado”, etc.

Los términos lingüísticos son mucho menos precisos que los datos numéricos, pero estos aportan una información útil para el razonamiento humano.

en un principio la lógica difusa encontró una fuerte resistencia entre la comunidad científica, sin embargo, hubieron varios investigadores que se hicieron fieles seguidores de las teorías de Zadeh y mientras él siguió ampliando y asentando los fundamentos de la teoría de conjuntos difusos estos investigadores exploraron estas nuevas teorías durante la década siguiente.

A principios de la década de 1970 se establecieron varios grupos investigadores de en Japón tanto en Tokio como en Osaka, estos grupos lograron grandes contribuciones tanto al desarrollo de la teoría de la lógica difusa como al estudio de sus aplicaciones, todo esto a pesar de encontrasen en una situación hostil de parte del resto de la comunidad científica.



La primera implantación real de un controlador del tipo difuso fue realizada en 1980 por F.L. Smidth & Co, en una planta cementera en Dinamarca, 3 años después en 1983 la empresa Fujifilm Holdings Corporation en Japón aplica la lógica difusa para el control de inyección química en plantas depuradoras de agua, por primera vez en Japón en el año 1987 la empresa Hitachi pone en marcha un controlador difuso para el control del tren-metro de Sendai.

En la década posteriormente se hicieron grandes contribuciones a la lógica difusa, en parte esto se debe al gran interés por las redes neuronales y su similitud con los sistemas difusos, esta tendencia consiste en buscar vías de relación entre las dos técnicas y los resultados son los llamados neuro-fuzzy systems.

En la década de 1990 aparecen los algoritmos genéticos, estos algoritmos, las redes neuronales y la lógica difusa pueden combinarse de diversas maneras y hay quienes las consideran complementarias, la combinación de estas técnicas permitió el desarrollo de herramientas muy potentes en los sistemas de control de los últimos años.

En un principio la intención del profesor Zadeh era la de construir un formalismo para manipular de manera más precisa la difusividad del razonamiento humano expresado de manera lingüística. Sin embargo, causó cierta sorpresa que el éxito de la lógica borrosa llegase en el campo del control automático de procesos. Esto se debió al gran impacto que obtuvo la lógica difusa en Japón en el año 1997, esto fue el resultado de una estrecha colaboración entre el gobierno, las universidades y las industrias japonesas, estableciendo dos proyectos nacionales de gran escala en donde se involucraron cerca de 50 compañías durante 6 años.

Desde entonces, han sido ininidad los productos lanzados al mercado que usan tecnología difusa, muchos de ellos utilizando la etiqueta fuzzy (difuso) como símbolo de calidad y prestaciones avanzadas. El control difuso ha sido aplicado con éxito en muy diversas ramas tecnológicas, por ejemplo, la metalurgia, robots para la fabricación, controles de maniobras de aviones, sensores de imagen y sonido etc.

Estas son algunas de las muchísimas aplicaciones de la lógica difusa, que ya están funcionando en el campo de los llamados sistemas expertos. Todos estos sistemas utilizan información, esencialmente, imprecisa con el fin de lograr sus cometidos.

Actualmente los investigadores están buscando nuevas áreas en las cuales aplicar la lógica difusa, actualmente se está investigando en áreas como el reconocimiento de patrones visuales o la identificación de segmentos de ADN.

La teoría clásica de conjuntos sólo contempla la pertenencia o no pertenencia de un elemento a un conjunto, sin embargo la teoría de conjuntos difusos contempla la pertenencia parcial de un elemento a un conjunto, es decir, cada elemento presenta un grado de pertenencia a un conjunto difuso que puede tomar cualquier valor entre 0 y 1. Un ejemplo más claro de esto es el conjunto de los hombres altos donde se establece que si se mide 1.80 m en adelante es considerado un hombre alto, sin embargo alguien que mida 1.79 m no haría parte de este conjunto, desde un punto de vista humano esto no es del todo cierto puesto que como puede ser que alguien sea considerado alto y alguien que mide tan solo 1cm menos no lo es. Es por eso que se establece un grado de pertenencia, podría decirse que alguien que mida entre 170m y 179m es 0.8 alto de esta manera pertenecerá parcialmente al conjunto de los hombres altos un poco similar a la manera en la que el ser humano razona.

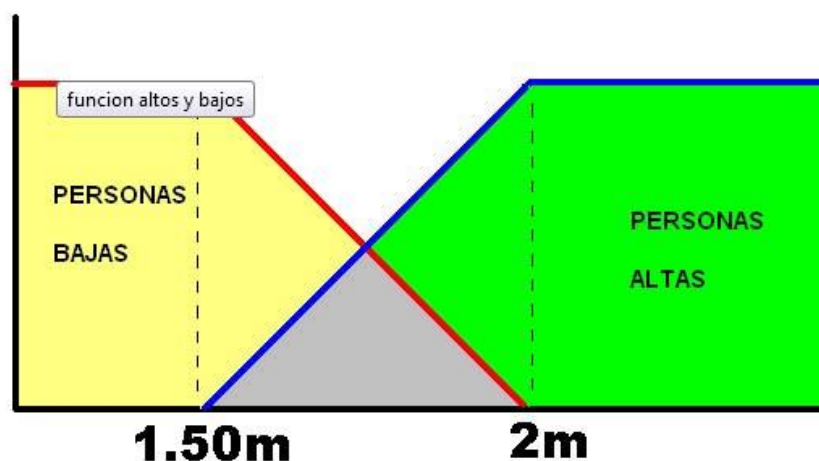


Fig. 24 Lógica difusa, altura

5 CONCLUSIONES

El desarrollo de las temáticas elaboradas en clase utilizando el lenguaje JavaScript prueba ser un mecanismo de gran valor para el aprendizaje de los conceptos básicos de la materia.

La teoría de redes neuronales artificiales, presenta grandes ventajas con respecto a los otros modelos típicos de solución de problemas de ingeniería, una de ellas es su inspiración en modelos biológicos del funcionamiento del cerebro, lo que facilita su estudio debido a las analogías que pueden introducirse para su análisis.

Los modelos matemáticos en que han sido desarrollados los algoritmos para todos los tipos de redes son modelos sencillos, que, aunque exigen cierto grado de conocimientos de cálculo diferencial, pueden ser asimilados y desarrollados en cualquier lenguaje de programación.

La red tipo perceptrón es una red que puede implementarse exitosamente para resolver problemas de clasificación de patrones que sean linealmente separables, la red responderá mejor entre más sencillos sean los patrones que debe clasificar. A pesar de que cuenta con serias limitaciones, esta red conserva su importancia ya que sirvió como inspiración para otros tipos de redes, como por ejemplo las redes multicapa.



6 BIBLIOGRAFÍA

<https://es.wikipedia.org/wiki/Perceptr%C3%B3n>

[https://es.wikipedia.org/wiki/L%C3%B3gica_difusa#:~:text=La%20l%C3%B3gica%20difusa%20\(tambi%C3%A9n%20llamada,contextualizados%20y%20referidos%20entre%20s%C3%AD.](https://es.wikipedia.org/wiki/L%C3%B3gica_difusa#:~:text=La%20l%C3%B3gica%20difusa%20(tambi%C3%A9n%20llamada,contextualizados%20y%20referidos%20entre%20s%C3%AD.)

<http://bibing.us.es/proyectos/abreproy/11084/fichero/Memoria+por+cap%C3%ADtulos+%252FCap%C3%ADtulo+4.pdf>