

# Nuevas Estrategias para el entrenamiento de Redes Neuronales que propagan Números Difusos

Edwin Villarreal López<sup>1</sup>, Oscar Duarte<sup>2</sup>, Daniel Alejandro Arango<sup>3</sup>

<sup>1</sup> Universidad Nacional de Colombia

<sup>2</sup> Universidad Manuela Beltrán  
edvillal@gmail.com

## Resumen

Se presenta la arquitectura básica de una red neuronal feedforward con la capacidad de propagar números difusos. Se exponen brevemente las principales tendencias en el entrenamiento de este tipo de sistemas y con base en ellas se proponen nuevas estrategias. La primera de ellas se basa en la retropropagación del error cuadrático medio en todos los  $\pi$ -cortes para pesos crisp. La segunda hace uso de un algoritmo genético con codificación real para redes con pesos crisp. La tercera consiste en la retropropagación del error en el valor promedio y la ambigüedad en todos los  $\pi$ -cortes para pesos difusos, y por último se tiene una basada en la retropropagación de una medida difusa del error para redes con pesos difusos. Luego se describen algunos experimentos realizados permitiendo identificar para qué conjuntos de datos particulares resulta útil cada una de las estrategias.

## Palabras clave:

red neuronal, algoritmos, retropropagación

## Abstract

A novel architecture of a feedforward neural network for fuzzy numbers is presented. We shortly analyze the main trends in the training of this kind of systems and from that point, we propose new strategies. The first of them is based on the backpropagation of the mean square error in all the  $\pi$ -cuts for crisp weights. The second strategy uses a real codification genetic algorithm for crisp weights networks. The third is based in the backpropagation of the mean error value and the ambiguity of all the  $\pi$ -cuts for fuzzy weights, and the last one uses the backpropagation of a fuzzy error measure for a fuzzy weighted network. A comparison analysis is then presented and some conclusions are given.

## Keywords:

neural network algorithms, backpropagation.

---

## Introducción

La mayor parte de los sistemas para el manejo y tratamiento de la información que existen en la actualidad se basan en una arquitectura de procesamiento digital, esquema que, aunque ha demostrado ser de gran utilidad, se encuentra limitado por su incapacidad de representar de manera eficaz la información procedente del mundo real en una forma legible para las máquinas, información que, por lo general, se encuentra contaminada con imprecisiones y distorsiones.

La lógica difusa y, en general, la teoría de los conjuntos difusos (Zadeh, 1975) es un área de la inteligencia artificial que se ha enfocado en desarrollar herramientas que permitan representar y realizar operaciones con cantidades inexactas e imprecisas.

Uno de los principales conceptos manejados dentro de esta teoría es el número difuso que facilita la tarea de modelar la imprecisión del mundo real, lo que permite a los sistemas operar a partir de mediciones y percepciones no muy exactas del medio. Con el objetivo de aprovechar esta cualidad y combinarla con las ventajas de otros tipos de sistemas de información, se han desarrollado múltiples técnicas híbridas, y entre éstas se destacan las redes neuronales difusas.

Una red neuronal difusa de este tipo puede verse como la generalización de una red neuronal feedforward convencional, en la que, tanto las cantidades manipuladas (entradas, salidas y pesos de las conexiones), como las operaciones necesarias para realizar la propagación (adición, multiplicación, función sigmoide) son extendidas al dominio de los números difusos mediante el principio de extensión formulado en (Zadeh, 1975), el cual ha sido reformulado de distintas formas (Klimke, 2006), entre otras, resulta sencillo llevar estas operaciones a los números difusos. Sin embargo, dicha extensión no puede realizarse a los métodos de entrenamiento.

Diversos grupos de investigadores han venido desarrollando estrategias de entrenamiento para estas redes, las cuales, en su mayoría, se simplifican las formas de las funciones de pertenencia de los números difusos propagados por la red, o se desarrollan algoritmos aplicables únicamente a ciertas topologías.

En este trabajo, se presentan nuevas estrategias de entrenamiento más generales con respecto a la geometría de los pesos difusos y la arquitectura de la red. Se utiliza la notación barra  $\bar{A}$  para denotar un número difuso. Además, se define un  $\alpha$ -corte de un número difuso  $\bar{A}$  como el conjunto de todos los  $x$  que pertenecen al conjunto difuso  $\bar{A}$  con, al menos, un grado de pertenencia  $\alpha$

$$\bar{A}_{[\alpha]} = \{x \mid \mu_{\bar{A}}(x) \geq \alpha\}$$

## Conclusiones

El uso de pesos crisp es una alternativa que debe ser tenida en cuenta a la hora de modelar la relación presente en un conjunto de datos difusos. El desempeño de esta estrategia se destacó en el problema del sistema de evaluación difusa del impacto ambiental en vertederos.

La totalidad de las estrategias de entrenamiento planteadas en este proyecto son válidas para redes con cualquier número de capas ocultas.

El entrenamiento de una ABCWN con pesos crisp mediante algoritmos genéticos con codificación real, puede arrojar resultados similares a los encontrados con BCrisp, en cuanto a la calidad de la aproximación. Sin embargo, el elevado tiempo de cálculo, debido a la gran cantidad de parámetros a ajustar, limita la aplicación de esta estrategia a problemas relativamente pequeños. Este hecho hace dudar de la viabilidad del empleo de alguna técnica similar que considere pesos difusos, puesto que se tendría una cantidad aún mayor de parámetros a ajustar.

Ninguna de las dos estrategias para pesos difusos formuladas (BaFuzzy, BEFuzzy) presentan limitaciones en cuanto a la geometría de los pesos difusos (siempre que sean números difusos).

A pesar de que la estrategia BaFuzzy no maneja una función de error global, sino múltiples funciones de error independientes, mostró tener un comportamiento aceptable en los experimentos realizados, con excepción del problema EDIAV.

La estrategia fundamentada en la retropropagación de un error difuso (BEFuzzy) se obtuvo al extender algunos conceptos del cálculo crisp al dominio de los números difusos.

Las redes con pesos difusos mostraron ser el mecanismo más adecuado para representar la incertidumbre propia de un sistema. Los resultados de este enfoque se destacaron en especial a la hora de aproximar conjuntos de datos con entradas crisp y salidas difusas.

Las estrategias para redes con pesos crisp mostraron los mejores desempeños a la hora de aproximar conjuntos de datos provenientes de funciones extendidas a los números difusos.

## Referencias

- [1] [Bede, Rudas, & Bencsik, 2007] Bede, B., Rudas, I., & Bencsik, A. (2007). First order linear fuzzy differential equations under generalized differentiability. *Information Sciences*, 177, 1648-1662.
- [2] [Buckley, Czogala, & Hayashi, 2003] Buckley, J., Czogala, E., & Hayashi, Y. (2003). Fuzzy neural networks with fuzzy signals and fuzzy weights. *Inter. J. Intelligent Systems*, 8, 527-537.
- [3] [Buckley, Czogala, & Hayashi, 2008] Buckley, J., Czogala, E., & Hayashi, Y. (2008). Adjusting fuzzy weights in fuzzy neural nets. *Second international conference on Knowledge-based intelligent electronic systems*.
- [4] [Delgadillo, Madrid, & Velez, 2004] Delgadillo, A., Madrid, J., & Velez, J. (2004). Ampliación de UN-genético: Una Librería en C++ de Algoritmos genéticos con Codificación Híbrida. Universidad Nacional de Colombia.
- [5] [Duarte, 2005] Duarte, O. (2005). FUZZYNET 1.0 Software Para el Diseño e Implementación de Redes de Sistemas de Computación con Palabras. Universidad Nacional de Colombia.
- [6] [Dunyak & Wunsch, 1997] Dunyak, J., & Wunsch, D. (1997). A training technique for fuzzy number neural networks. *Proc. of the International Conference on Neural Networks*.
- [7] [Dunyak & Wunsch, 2000] Dunyak, J., & Wunsch, D. (2000). Fuzzy regression by fuzzy number neural networks. *Fuzzy Sets and Systems*, 112, 371-380.
- [8] [Ishibuchi & Nii, 2001] Ishibuchi, H., & Nii, M. (2001). Numerical analysis of the learning of fuzzified neural networks from fuzzy if-then rules. *Fuzzy Sets and Systems*, 120, 281-307.
- [9] [Ishibuchi, Okada, & Tanaka, 1993] Ishibuchi, H., Okada, H., & Tanaka, H. (1993). Fuzzy neural networks with fuzzy weights and fuzzy biases. *Proc. of ICNN'93, San Francisco*.
- [10] [Klimke, 2006] Klimke, A. (2006). *Uncertainty Modeling using Fuzzy Arithmetic and Sparse Grids* (PhD Tesis). Universität Stuttgart, Alemania.
- [11] [Krishnamraju, Buckley, Hayashi, & Reilly, 2004] Krishnamraju, P., Buckley, J., Hayashi, Y., & Reilly, K. (2004). Genetic learning algorithms for fuzzy neural nets. *IEEE World Congress on Computational Intelligence* (págs. 26-29).
- [12] [Lippe, Feuring, & Mischke, 2006] Lippe, W., Feuring, T., & Mischke, L. (2006). Supervised learning in fuzzy neural networks. Department of Computer Science, University of Munster.
- [13] [Riedmiller, 1994] Riedmiller, M. (1994). *Rprop-Description and Implementation Details*.
- [14] [Rumelhart, Hinton, & Willimas, 1986] Rumelhart, D., Hinton, G., & Willimas, R. (1986). Learning representations by back-propagating errors. *Nature*, 323, 533-536.



[15] [Villarreal, 2008] Villarreal, E. (2008). Estrategias de Entrenamiento para un Red Neuronal Difusa (Tesis de Maestría en Automatización Industrial). Universidad Nacional de Colombia.

[16] [Zadeh, 1975] Zadeh, L. (1975). The Concept of a Linguistic Variable and its Application to Approximate Reasoning. IEEE Trans. Systems, Man, and Cybernet.