

IMPLEMENTACIÓN DE ALGORITMOS DE INTELIGENCIA ARTIFICIAL PARA EL ENTRENAMIENTO DE REDES NEURONALES DE SEGUNDA GENERACIÓN

Sergio Ernesto Tostado Sánchez (1), Manuel Ornelas Rodríguez (2), Andrés Espinal Jiménez, Héctor J. Puga Soberanes

1 Ingeniería en Sistemas Computacionales, Tecnológico Nacional de México – Instituto Tecnológico de León | Dirección de correo electrónico: sergioernestotostadosanchez@gmail.com

2 División de Estudios de Posgrado e Investigación, León, Tecnológico Nacional de México – Instituto Tecnológico de León | Dirección de correo electrónico: mornelas67@yahoo.com.mx

Resumen

Las *Redes Neuronales Artificiales (RNA)* son sistemas de procesamiento de información inspiradas en las redes neuronales biológicas, cuyas aplicaciones han encontrado gran aceptación en diversas áreas y problemas, como el *Reconocimiento de Patrones*. Previamente a su operación, necesitan calibrar elementos de su arquitectura a través de la fase de *Entrenamiento*, siendo seleccionado el algoritmo de *Retropropagación* como método clásico para ello. En la práctica, se han identificado deficiencias del algoritmo de Retropropagación al entrenar RNA, relacionadas a su fundamento matemático y a la dependencia hacia expertos en el problema en cuestión, por lo que se han explorado métodos alternativos para apoyar el adiestramiento de estos sistemas, siendo empleadas las *Metaheurísticas* en diversas investigaciones. En este trabajo se comparan 3 algoritmos metaheurísticos y la Retropropagación para la calibración de los pesos sinápticos de RNA en la solución de problemas de Clasificación de Patrones. Las Metaheurísticas utilizadas fueron *Optimización por Cúmulo de Partículas*, *Evolución Diferencial* y *Estrategia Evolutiva*. Se muestran como resultado los porcentajes de clasificación de 3 instancias, las cuales son *Iris Plant*, *Ionosphere* y *Wine*.

Abstract

Artificial Neural Networks (ANN) are information processing systems that are inspired on biological neural networks, whose applications have found great approval in many areas and problems, like *Pattern Recognition*. Before ANN being used, they need to adjust their architecture elements through a *Training Phase*, where *Backpropagation* is a classical method to do it. In practice, some deficiencies of Backpropagation have been found in the ANN training, which are related to its mathematical base and dependency to an expert in the concerning problem. To solve these problems, it has been proposed the use of *Metaheuristic algorithms* to support the training phase. This work shows the comparison between Backpropagation and the *metaheuristics Particle Swarm Optimization (PSO)*, *Differential Evolution (DE)* and *Evolutionary Strategy (ES)*, taking like main criteria the Pattern Classification performance with 3 classical benchmarks: *Iris Plant*, *Ionosphere* and *Wine*.

Palabras Clave

Redes Neuronales Artificiales; Retropropagación; Metaheurísticas; Reconocimiento de Patrones

INTRODUCCIÓN

Reconocimiento de Patrones y Clasificación

El *Reconocimiento de Patrones* es un área de la Inteligencia Artificial, que se define como el estudio de los métodos y las técnicas para la distinción de objetos, a partir de sus características más sobresalientes, incluyendo la identificación de tendencias en conjuntos de datos, empleando algoritmos computacionales fundamentados en modelos matemáticos, emulando a escala habilidades innatas de los seres vivos, como la visión [1][2][3].

Se entiende por *patrón* como el objeto que es inspeccionado en el proceso del *reconocimiento*, la descripción de aquello a lo cual deseamos reconocer. Cuando el reconocimiento de patrones se orienta a la determinación de la clase a la que pertenece un patrón, se le denomina Clasificación [1][4].

Las RNA

Son sistemas de procesamiento de información, que a su vez son reconocidas como un paradigma matemático de computación. Son ampliamente utilizadas en diversos ambientes teóricos y prácticos, siendo del interés del presente trabajo aquellas utilizadas en la Clasificación de Patrones. En el sentido estructural, las RNA son un conjunto de unidades llamadas neuronas artificiales conectadas, cuya estructura se muestra en las referencias [5], [6] y [11].

Al igual que su homólogo biológico, las RNA requieren un proceso de aprendizaje para su uso en la resolución de un problema en específico, a lo cual llamamos *fase de entrenamiento*, cuyo resultado es la calibración de los *pesos sinápticos* entre las conexiones de las neuronas componentes de las RNA para su preparación acorde al problema o ámbito de aplicación. En

este artículo se implementaron RNA de 2° generación [3], tipo feed – forward [7].

El Algoritmo de Retropropagación

Generalmente, éste es el método seleccionado para el entrenamiento de las RNA, basado en el *Método del Gradiente* [8]; el entrenamiento de RNA con este algoritmo ha demostrado las siguientes deficiencias: *Estancamiento de soluciones en Mínimos Locales* y *Alta Dependencia de expertos en el problema* [6].

Metaheurísticas

Las Metaheurísticas son métodos estocásticos para la solución de problemas de optimización que organizan una interacción entre procedimientos y estrategias de más alto nivel para ser capaces de realizar exploraciones rigurosas dentro de los espacios de búsqueda [9][10]. Sus características permiten ser una alternativa factible para resolver el problema de la determinación de los pesos sinápticos de las RNA, principalmente por su alta adaptabilidad y flexibilidad a la resolución de diversos problemas [12].

Las Metaheurísticas consideradas en este proyecto son basadas en *población*, las cuales inician con una población de soluciones, la cual es iterativamente alterada para dar origen a nuevas generaciones de dicha población (Imagen 1) [10].

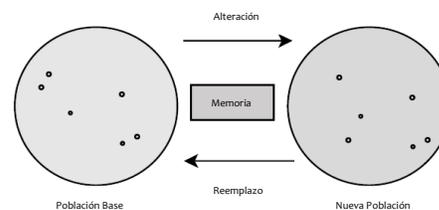


Imagen 1: Esquema genérico de las Metaheurísticas Poblacionales

Ante las deficiencias de la Retropropagación, los trabajos en el estado del arte han presentado tendencia al uso de Metaheurísticas, para ampliar la gama de métodos posibles para entrenar a las

RNA. El presente proyecto muestra la comparación del desempeño de RNA entrenadas con Retropropagación, Optimización por Cúmulo de Partículas (versión canónica) [14], Evolución Diferencial [15] y Estrategia Evolutiva [16][17] (PSO, DE y ES por sus siglas en inglés respectivamente), comprobando si las Metaheurísticas tienen un mejor desempeño que Retropropagación, así como crear una referencia útil para ayudar a la selección de uno u otro enfoque (clásico o metaheurístico) para la resolución de las instancias ya mencionadas.

MATERIALES Y MÉTODOS

A continuación se mencionan todos los parámetros utilizados para la experimentación:

Cada instancia posee un conjunto de datos correspondientes a los patrones que se utilizaron para entrenar y probar el desempeño de las RNA; cada patrón tiene un número de atributos (características distintivas de la instancia), así como la clase a la que pertenece cada patrón de la instancia.

Para el desarrollo de Entrenamientos y Pruebas, se utilizó la Validación Cruzada por K-Folds [13], utilizando el valor $k=10$ pliegues (subconjuntos de patrones de las instancias), con 9 de los pliegues para entrenamiento y el restante para la prueba de desempeño, por cada instancia para cada algoritmo implementado. La arquitectura de las RNA es la siguiente (sea $n=No. de atributos de la instancia$ y $c=No. de clases de la instancia$):

(Entrada : n , Oculta : $2n$, Salida : c)

La función objetivo considerada es la del error cuadrático (función que denota el desempeño en el entrenamiento y parte del criterio de paro de los algoritmos) [7]:

$$\sum_{p=1}^M \sum_{k=1}^m (t_k^{(p)} - y_k^{(p)})^2$$

donde M = número de patrones, m = número de salidas de la red neuronal, t = salida esperada (objetivo) e y = salida de la red neuronal.

La función de activación para cada una de las neuronas de las RNA es la función sigmoidea [7].

La configuración del algoritmo de Retropropagación consistió en 30,000 iteraciones y tasa de aprendizaje = 0.1.

La implementación de los algoritmos metaheurísticos se realizó de acuerdo al artículo [18] (Ver imagen 4). El espacio de búsqueda de los pesos sinápticos explorado por los algoritmos fue el intervalo $[-1000, 1000]$. La configuración de cada algoritmo metaheurístico se describe a continuación, con un total de 30,000 llamadas a la función fitness y valor mínimo de error cuadrático de 0.01 para detención del algoritmo:

PSO: 1,000 iteraciones, 30 partículas, $\phi_1 = \phi_2 = 2.05$.

DE: 1,000 iteraciones, 30 individuos, probabilidad de cruce = 0.8, factor de esquema de muta = 0.9, esquema de muta = DE / Rand1.

ES: 1,000 iteraciones, 30 padres, 30 hijos, operador/variante = $(\mu + \lambda)$, $\sigma \approx 3$.

RESULTADOS Y DISCUSIÓN

A continuación, en las tablas 1 y 2 se muestra los resultados estadísticos de los procesos de

Tabla 1: Comparación del valor mínimo, máximo, promedio y desviación estándar (D.E.) de valor final de error cuadrático en la validación cruzada K-Folds $k = 10$

DATASET	ALGORITMO				
	Retropropagación	PSO	DE	ES	
Iris Plant	Mín	0.0099	1.9333	1.9833	19
	Max	2.0065	11.25	15.9999	103
	Media	1.48126	6.51763	7.41511	70.95
	D.E.	0.806031036	3.286549609	4.39829778	27.1051963
Ionósfera	Mín	0.0099	64.0021	68.5714	48.5714
	Max	2.0034	110.0008	91.4285	80
	Media	0.60765	88.05517	77.48408	64.11902
	D.E.	0.962469025	15.06135558	7.81206234	11.0474787
Wine	Mín	109.0255	26.054	83	118
	Max	111.3051	91.8943	94	135
	Media	110.18122	74.07299	89.78137	129.05883
	D.E.	0.767012891	18.08250706	3.8232691	5.82763779

Tabla 2: Comparación del valor mínimo, máximo, media y desviación estándar (D.E.) de los porcentajes de clasificación de las RNA entrenadas

DATASET	ALGORITMO				
	Retropropagación	PSO	DE	ES	
Iris Plant	Mín	80	86.6	86.6	20
	Max	100	100	100	73.3
	Media	95.6	95.98	95.31	52.63
	D.E.	6.459824559	5.649936086	5.51592644	17.8884227
Ionósfera	Mín	82.85	57.14	68.57	48.57
	Max	94.28	82.85	91.42	80
	Media	88.32	67.82	77.48	64.116
	D.E.	4.534252358	7.505963555	7.80992104	11.04829117
Wine	Mín	0	22.22	33.33	11.11
	Max	0	77.77	61.11	44.44
	Media	0	54.603	46.663	23.492
	D.E.	0	18.5297239	9.25305241	12.09914855

Imagen 2: Diagrama de Caja del valor final del error cuadrático

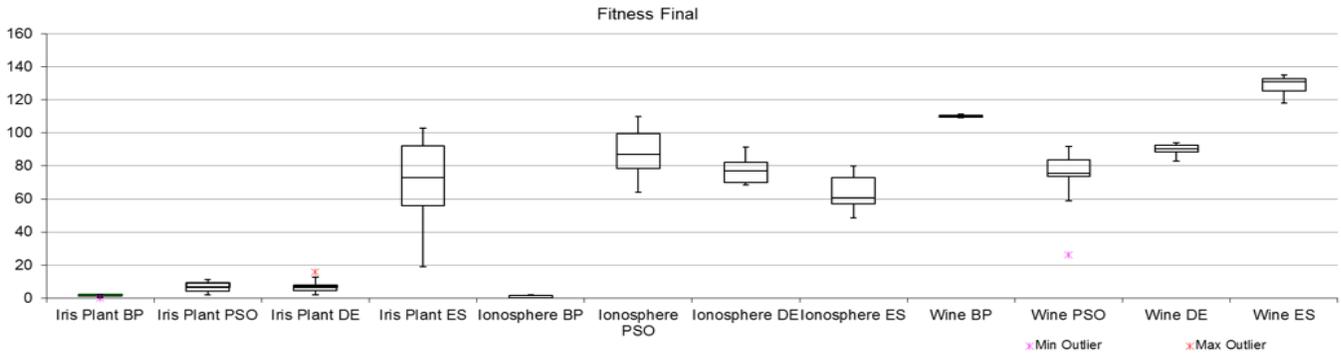


Imagen 3: Diagrama de Caja de los porcentajes del desempeño de las RNA

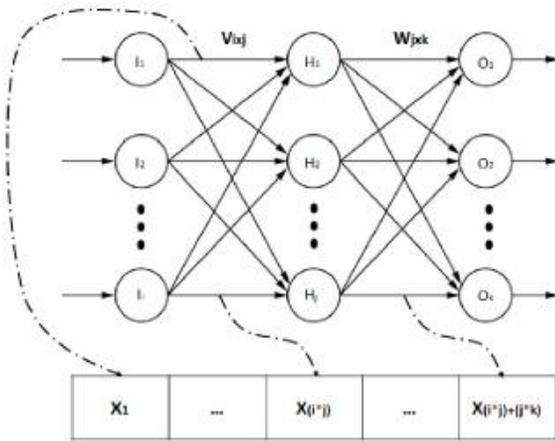
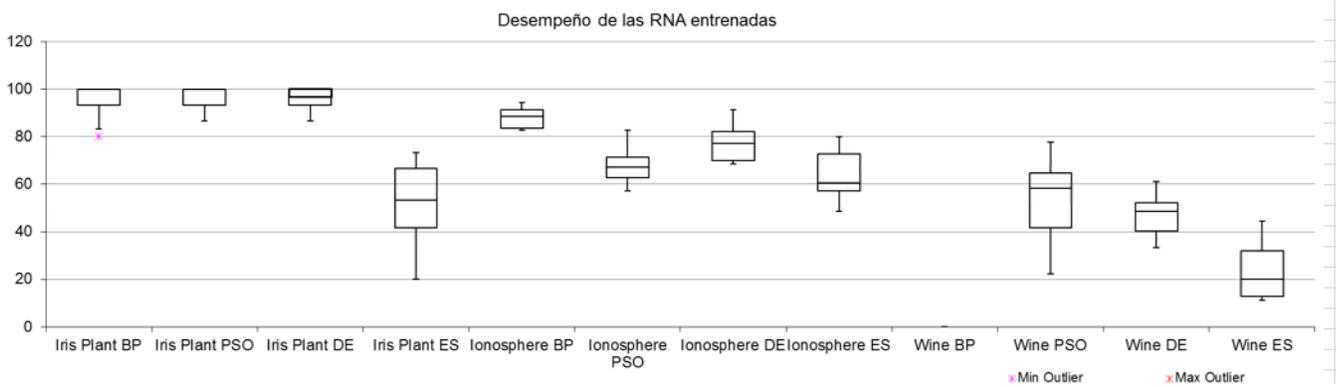


Imagen 4: Diagrama de Configuración de pesos sinápticos de una RNA

entrenamiento y prueba respectivamente, de las 10 pruebas resultantes de la Validación Cruzada para cada instancia. Las imágenes 2 y 3 muestran gráficamente los resultados de las tablas anteriores.

CONCLUSIONES

En este trabajo se ha mostrado la comparación de 4 algoritmos para entrenar RNA de segunda generación aplicadas a la Clasificación de Patrones de 3 instancias usadas frecuentemente en el estado del arte.

Para las instancias Iris Plant y Ionosphere las Metaheurísticas PSO y DE obtuvieron porcentajes de clasificación similares a Retropropagación, no obstante que sus valores de error cuadrático fueron mayores.

Para la instancia Wine, las Metaheurísticas demostraron tener desempeño superior a Retropropagación.

Como trabajo a futuro se realizarán pruebas con un mayor número de llamadas a función, otras instancias de prueba, así como variantes en los parámetros de los algoritmos, con el objetivo de obtener diferencias más claras entre

Retropropagación y los algoritmos metaheurísticos.

AGRADECIMIENTOS

Los autores agradecen al Tecnológico Nacional de México – Instituto Tecnológico de León por el apoyo brindado para la realización del presente trabajo.

REFERENCIAS

- G. Dreyfus, *Neural Networks Methodology and Applications*, Berlin, Alemania: Springer, 2005.
- D. Montana and L. Davis, "Training Feedforward Neural Networks Using Genetic Algorithms," in 11th International Joint Conference on Artificial Intelligence, San Francisco, Estados Unidos, 1989.
- W. Mass, "Networks of Spiking Neurons: The third Generation of Neural Network Models," in *Neural Networks 10*, Graz, Austria, 1997.
- M. Kantardzic, *DATA MINING Concepts, Models, Methods, and Algorithms*, Segunda ed., Hoboken, Estado Unidos: John Wiley & Sons Inc., 2011.
- Y.H. Hu, J.N. Hwang, *Handbook of NEURAL NETWORK SIGNAL PROCESSING*, Estado Unidos: CRC Press, 2002.
- M. Friedman and A. Kandel, *Introduction to Pattern Recognition Statistical, Structural, Neural and Fuzzy Logic approaches*, Londres, Reino Unido: World Scientific, 2000, pp. 1-6, 298-299.
- E. N. Sánchez Camperos y A. Y. Alanís García, *Redes neuronales Conceptos fundamentales y aplicaciones a control automático*, Madrid, España: Pearson Prentice Hall, 2006.
- M. Ginsberg, *Essentials of Artificial Intelligence*, San Francisco, Estados Unidos: Morgan Kaufmann Publishers, 1993.
- F. Glover and G. Kochenberger, *HANDBOOK OF METAHEURISTICS*, Dordrecht, Alemania: Kluwer Academic, 2003.
- E. G. Talbi, *Metaheuristics: From Design to Implementation*, Hoboken, Estado Unidos: John Wiley & Sons, Inc, 2009.
- S. Judd, *Neural Network Design and Complexity of Learning*, Londres, Inglaterra: The MIT Press, 1990, pp. 15-16.
- S. Ding, H. Li, C. Su, J. Yu and F. Jin, *Evolutionary artificial neural networks: a review*, 2011.
- Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. *International joint conference on artificial intelligence (IJCAI)*, 1995.
- J. Kennedy y R.C. Eberhat, "Particle Swarm optimization", *IEEE Int. Conf. Neural Netw.*, vol. 4pp. 1942 – 1948, 1995.
- R. Storn y K. Price, "Differential Evolution – a simple an efficient metaheuristic for global Optimization over continuous spaces", *J. of Global Optimization*, vol. 11, pp. 341 – 359, December 1997, [Online]. Disponible: <http://portal.acm.org/citation.cfm?id=596061.596146>

Rechenber, L, "Evolutionstrategie: optimierung technischer systeme nach prinzipien der biologischen evolution". Fromman – Holzboo, 1973.

Schwefel, H.P. , "Numerische optimierung von computer-modellen mittels der evolutionstrategies", 1997. [Online]. Disponible: <http://dx.doi.org/10.1007/978-3-0348-5927-1>.

A. Espinal, M. Sotelo Figueroa, J. Soria Alcaraz, M. Ornelas, H. Puga, M. Carpio, R. Baltazar and J. Rico, "Comparison of PSO and DE for training neural networks," in 10th Mexican International Conference on Artificial Intelligence, León, México, 2011.