

Un modelo transformacional para la Predicción del índice bursátil S&P 500 utilizando redes neurodifusas y algoritmos genéticos

Diego Javier-Quispe¹, Geraldine Valdez-Yana¹, José Vargas-Huamán¹ y José Sulla-Torres, Magister¹

¹Escuela Profesional de Ingeniería de Sistemas, Universidad Nacional de San Agustín, Arequipa, Perú
dijaq08@gmail.com, geraldinetvy@gmail.com, josemvargh@gmail.com, jsulla@unsa.edu.pe

Resume - El objetivo del presente artículo es la elaboración de un modelo transformacional para la predicción de la variación diaria del índice bursátil S&P 500 utilizando algoritmos genéticos y redes neurodifusas. El modelo consta de 2 fases la primera consiste en la elaboración de las funciones y reglas difusas a través del algoritmo TSK-IRL-R de KEEL, la segunda etapa es el entrenamiento de la red neurodifusa usando ANFIS de MATLAB. Como conjunto de datos se tomaron las variaciones proporcionadas por Yahoo Finance, obteniendo como resultado el valor mínimo de 7.5843 de error en el entrenamiento.

Palabras clave- S&P 500, redes neurodifusas, algoritmos genéticos, predicción, modelo transformacional.

Abstract- The objective of this paper is the elaboration of a transformational model to forecasting the diary variance of S&P 500 by using genetic algorithms and fuzzy neural networks. The model consists of two phases, the first is the elaboration of fuzzy functions and rules through TSK-IRL-R in KEEL, and the second is the training of fuzzy neural network by using ANFIS in MATLAB. The data set was composed the diary variances from Yahoo Finance. It was obtained 7.5843 of training error.

Keywords- S&P 500, fuzzy neural networks, genetic algorithms, prediction, transformational model.

I. INTRODUCCIÓN

Los índices bursátiles son ampliamente utilizados para la toma de decisiones de inversión en el mercado, de manera tal que cada bolsa de valores cuenta con al menos un índice que indique su situación actual, para el caso concreto de la bolsa de New York, existe el S&P 500 [1] que está basado en el precio de las acciones de 500 empresas de distintos rubros a diferencia del Dow Jones y Nasdaq Composite.

Sin embargo, la variación de un índice bursátil en general depende de factores coyunturales lo cual dificulta el establecimiento de un modelo válido para cualquier periodo de tiempo, por lo que predecir un índice de la bolsa de valores representa un gran reto y en las últimas dos décadas ello ha sido objeto de muchos estudios dadas las aplicaciones comerciales que tiene [2]. Numerosos métodos han sido propuestos para brindar predicciones más exactas a los inversores los cuales pueden ser divididos en aplicaciones de modelos estadísticos (ARIMA) [3] y técnicas de inteligencia artificial [4].

Digital Object Identifier (DOI): <http://dx.doi.org/10.18687/LACCEI2017.1.1.233>
ISBN: 978-0-9993443-0-9
ISSN: 2414-6390

15th LACCEI International Multi-Conference for Engineering, Education, and Technology: "Global Partnerships for Development and Engineering Education", 19-21 July 2017, Boca Raton FL, United States.

Como objetivo de éste artículo, se ha planteado un modelo transformacional que utiliza técnicas de Soft Computing para la predicción de la variación diaria del S&P 500. Las técnicas consideradas son redes neuronales artificiales, lógica difusa y algoritmos evolutivos (Algoritmos genéticos).

El modelo consta de dos etapas bien diferenciadas la primera de ella es la generación del FIS aplicando algoritmos genéticos y la segunda consiste en el entrenamiento y evaluación de una red neurodifusa mediante la herramienta Matlab. Como base de conocimiento se tomó en cuenta las variaciones del S&P 500 desde el 03 de enero de 1950 hasta el 05 de febrero de 2016.

A continuación, se da a conocer el estado del arte de predicción de índices bursátiles y otros trabajos tomados en cuenta en la realización de este artículo. En segundo lugar, se detalla la metodología, así como las técnicas y herramientas utilizadas para la realización de los modelos y su evaluación. Finalmente se muestran los resultados, las conclusiones a las que se llegaron y los trabajos futuros de la investigación.

II. TRABAJOS RELACIONADOS

Se presenta algunos trabajos en la misma rama que anteceden este artículo:

A. Entrenamiento de redes neuronales basado en algoritmos evolutivos.

Luis Federico Bertona [5] estudia un método alternativo para el entrenamiento de redes neuronales con conexión hacia delante. Una vez determinada la topología de la red neuronal se utiliza un algoritmo genético para ajustar los pesos de la red neuronal y se evalúan diferentes variantes de los operadores genéticos para el entrenamiento de las redes neuronales. Los resultados obtenidos por el algoritmo genético son contrastados con los resultados obtenidos por el algoritmo de retropropagación de errores, como se aprecia en la Tabla I y Fig. 1.

TABLA I
COMPARACIÓN DE LA CANTIDAD DE ACIERTOS [5]

Tipo	Acierto Prom.	Desvío Aciertos	Acierto Prom. Eval	Desvío Acierto Eval.	Acierto Prom. Total%	Desvío Acierto Total%
Algoritmo Entrena.	94.09%	0.92%	82.70%	2.40%	91.81	1.04
Algoritmo Genético	94.32%	0.83%	83.29%	2.03%	92.11	0.88
Red Neuronal						

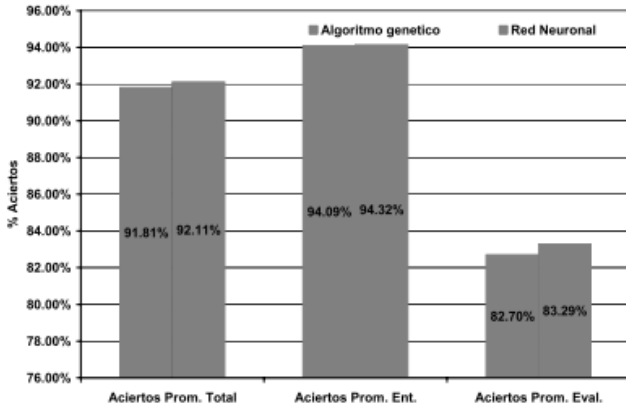


Fig. 1. Comparación de la cantidad de aciertos [5]

B. Modelos de algoritmos genéticos y redes neuronales en la predicción del signo de variación del IPSA

En el estudio de Antonio Parisi F et al. [6] se analiza la capacidad de los modelos multivariados dinámicos recursivos construidos a través de algoritmos genéticos y de redes neuronales recursivas para predecir el signo de las variaciones semanales del IPSA. Los datos corresponden al periodo comprendido entre el 14 de julio de 1997 y el 9 de diciembre de 2002. Los modelos analizados fueron evaluados en 60 series generadas por un proceso block-bootstrap. Los resultados señalan que la red Ward tendría mayor capacidad que el modelo de algoritmos genéticos y el modelo Naive para predecir el signo de las variaciones del IPSA, que esta capacidad predictiva sería significativa, y que una estrategia de trading basada en las señales de compraventa dadas por este modelo permitirían obtener retornos relativamente más altos. Se destaca que la red Ward y el modelo de algoritmos genéticos superaron, en promedio, a la estrategia buy and hold, aun cuando se consideró un costo por transacción equivalente al 0,1 % del monto transado.

C. Métodos para predecir índices bursátiles

Martha G., Aura J., Luis G. y Jorge L. [7] presentan una revisión bibliográfica acerca de los métodos que se han utilizado en las últimas dos décadas para predecir índices bursátiles. Los métodos estudiados van desde aquellos que logran capturar las características lineales presentes en los índices de bolsa, pasando por los que se enfocan en las características no lineales y finalmente métodos híbridos que son más robustos, pues capturan características lineales y no lineales. Además, se incluyen aquellos métodos que utilizan

variables macroeconómicas para predecir los índices de diferentes Bolsas de Valores en el mundo.

D. Un sistema de soporte de decisiones inteligente para la compra de acciones mediante la integración de algoritmos genéticos basados en redes neurodifusas y redes neuronales artificiales.

En este estudio realizado por R.J. Kuo, C.H. Chen y Y.C. Hwang [8] se investiga la eficacia de un enfoque híbrido basada las redes neuronales artificiales para las propiedades en las series de tiempo, tales como las redes neuronales de retardo de tiempo de adaptación (ATNNs) y las redes neuronales de retardo de tiempo (TDNNs), con algoritmos genéticos (AG) para la predicción de la bolsa de valores.

Para estimar los diversos aspectos del diseño y ATNN TDNN, se propone un método general sobre la base de ensayo y error, pero no se garantiza la mejora del rendimiento. Por lo que se usan AG para apoyar la optimización de la cantidad de tiempo de retrasos y factores de arquitectura de la red.

E. Un enfoque híbrido basado en redes neuronales y algoritmos genéticos para la detección de patrones temporales en la bolsa de valores.

Hyun-jung Kim y Kyung-shik Shin [9] desarrollan una red neuronal difusa basada en algoritmos genéticos, para formular la base del conocimiento de las reglas de inferencia difusas que pueden medir aspectos cualitativos en la bolsa de valores.

Se realiza el experimento basado en la bolsa de valores de Taiwan. Los resultados muestran que es mejor usar la red neuronal que considera los factores cuantitativos y cualitativos. Los resultados presentados demuestran que el método propuesto tiene la capacidad de entrenar redes neuronales con conexión hacia delante con las mismas características que los métodos tradicionales de entrenamiento. También demuestran que el método propuesto puede incluso mejorar la performance del entrenamiento en aquellos aspectos en que los métodos tradicionales suelen tener dificultades.

F. KEEL: Una herramienta software para el análisis de sistemas difusos evolutivos.

En este trabajo Joaquín Derrac del Dept. de Ciencias de la Computación e Inteligencia Artificial, CITIC-UGR, Universidad de Granada [10] presenta KEEL, una herramienta software no comercial, capaz de abordar una amplia gama de problemas de minería de datos (regresión, clasificación, asociación, agrupamiento, ...) mediante la generación de experimentos estándar y educativos. Dispone de una amplia librería de algoritmos de Sistemas Difusos Evolutivos (SDE) basados en diferentes esquemas: Michigan, Pittsburgh, IRL y GCCL. También dispone de módulos de tratamiento de datos y análisis estadísticos. A modo de ejemplo en el artículo se muestra un caso de estudio sobre el análisis de dos propuestas diferentes de aprendizaje de Sistemas Difusos, para ilustrar el

proceso de generación de experimentos en KEEL, incluyendo el empleo de test estadísticos.

El artículo finaliza recalcando que KEEL es una herramienta de software capaz de preparar adecuadamente algoritmos evolutivos para problemas de minería de datos, incluyendo algoritmos de SDEs, esta herramienta libera a los investigadores de la mayoría del trabajo técnico y les permite centrarse en el análisis de sus nuevos algoritmos en comparación con los ya existentes, permitiendo que incluso investigadores con conocimientos básicos en Lógica Difusa y Computación Evolutiva puedan emplear los SDEs en su trabajo.

III. METODOLOGÍA

A. Origen de los datos

El conjunto de datos fue obtenido de [11] considerando únicamente los precios de cierre del S&P 500 desde 03 de enero de 1950 hasta 05 de febrero de 2016. El número total de observaciones es de 16 633, como se muestra en la Fig. 2.

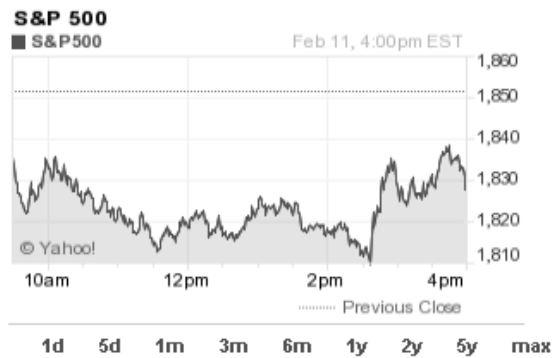


Fig. 2. Valor del cierre del S&P 500 [11]

Se calcularon las variaciones diarias (V) a partir de los valores de cierre diarios (C) mediante la Ec. (1).

$$V_i = C_i - C_{i-1} \quad (1)$$

Las variaciones obtenidas fueron fuzzificadas mediante la herramienta KEEL y posteriormente sirvieron de entrada para el entrenamiento y verificación de la red ANFIS siguiendo el modelo detallado en la Fig. 3.

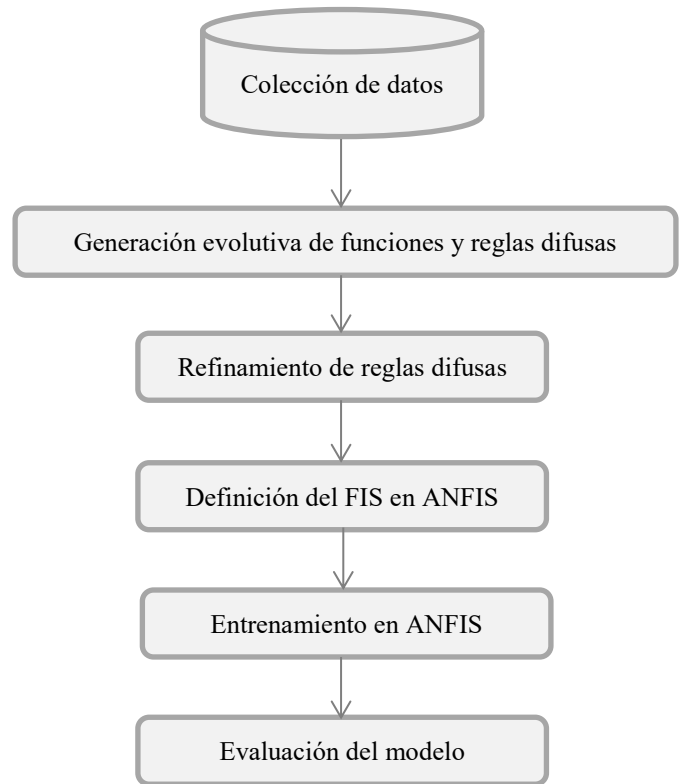


Fig. 3. Modelo propuesto para la predicción del S&P 500

B. Técnicas y Herramientas

1) Lógica Difusa

La lógica difusa permite el tratamiento de información poco definida e imprecisa, de tal manera que se realiza el análisis de la información de manera similar a lo realizado por un humano con el uso de etiquetas lingüísticas. Al contrario de la lógica convencional que trata solo con valores discretos, la lógica difusa establece grados de pertenencia a más de un conjunto difuso, es decir un determinado dato puede pertenecer a más de un conjunto y no solo a uno como lo establece la lógica convencional.

Lofti Zadeh en 1965 propuso la idea de Conjuntos Difusos [12], los cuales son una colección de valores que van asociándose al conjunto a través de un proceso conocido como fuzzificación. A partir de los datos fuzzificados se obtiene una salida la cual puede ser difusa o desfuzzificada para obtener un valor discreto CRISP [13].

El grado de pertenencia está determinado por una función de pertenencia mediante la Ec.(2)

$$A = \{x, \mu_A(x) | x \in U\} \quad (2)$$

$\mu_A: x \rightarrow [0, 1]$ función de pertenencia
 $\mu_A(x)$ grado de pertenencia de la variable x
 U dominio de la aplicación

En este artículo, se utilizan cinco etiquetas lingüísticas para las variaciones diarias: muy bajo la media (MBM), bajo la media (BM), en la media (EM), sobre la media (SM) y muy sobre la media (MSM).

La Función de pertenencia indica el grado en el que un elemento pertenece a un conjunto dado, para el caso de lógica difusa toma los valores entre 0 y 1. Las funciones que se utilizan en su mayoría son: Triangular, Trapezoidal y Gaussiana.

El tipo de función que se utilizó fue triangular, que se muestra en la Fig 4., siendo cinco funciones miembros por cada entrada, la función triangular está definida mediante tres parámetros los cuales determinan las coordenadas de los tres vértices del triángulo, mediante la Ec. (3).

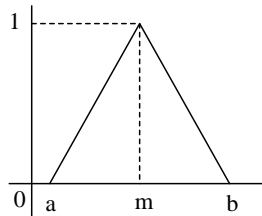


Fig. 4. Función triangular

$$\mu(x) = \begin{cases} 0 & ; \quad x \leq a \\ \frac{x-a}{m-a} & ; \quad a < x \leq m \\ \frac{b-x}{b-m} & ; \quad m < x \leq b \\ 0 & ; \quad x > b \end{cases} \quad (3)$$

2) KEEL

KEEL (Knowledge Extraction based on Evolutionary Learning) es una herramienta de código abierto que puede ser usada para varias tareas de descubrimiento de información. KEEL provee una GUI basada en el flujo de datos para el diseño de experimentos con diferentes conjuntos de datos y algoritmos de inteligencia computacional. Contiene una amplia variedad de algoritmos de extracción de conocimiento, técnicas de preprocesamiento, inteligencia computacional basada en algoritmos de aprendizaje, modelos híbridos, metodologías estadísticas, etc. [14].

3) TSK-IRL-R

La Inferencia TSK, fue propuesto por Takagi, Sugeno y Kang, también conocido como Sugeno, es un método de inferencia para la obtención de un valor de salida dado un valor de entrada.

A diferencia del modelo de inferencia Mandani, no es necesario un método extra para la desfuzzificación, por ende las reglas de inferencias TSK son de la forma de la forma:

$$p: \text{SI } x \text{ es } A \text{ y } Y \text{ es } B \text{ ENTONCES } z \text{ es } f(x, y) \quad (4)$$

En la estructura de TSK-IRL-R, muestra que el algoritmo está compuesto por 2 etapas: un proceso de generación evolutiva y un proceso de refinamiento evolutivo.

El primer proceso está basado en un algoritmo iterativo que estudia la existencia de los datos en los diferentes sub-espacios difusos de entrada. En cada iteración, los datos son localizados en alguno de ellos aplicando un método de aprendizaje de las reglas del consecuente TSK para determinar la relación de la entrada y la salida.

La segunda fase consiste en el ajuste del consecuente y el antecedente de las reglas difusas obtenidas en el primer proceso. El ajuste o refinamiento se realiza a través de un algoritmo genético el cual codifica la base de conocimiento obtenida en cromosomas y la función de ajuste es el error cuadrático medio [15].

Se aplicó este algoritmo a través de la herramienta KEEL y se diseñó el experimento mostrado en la Fig. 5.



Fig. 5. Experimento diseñado en KEEL

Los parámetros especificados para el algoritmo TSK-IRL-R son especificados en la Tabla II.

TABLA II
PARÁMETROS DEL ALGORITMO TSK-IRL-R

Parámetros	Valor
Iteraciones Estrategia Evolutiva	500
Número de padres para la estrategia evolutiva	15
Número de descendientes	100
Tamaño de la desviación estándar	2
Operador de recombinación	3
Operador de recombinación para la desviación de la secuencia	2
Número de padres para recombinar la secuencia solución	15
Número de padres para recombinar la desviación de la secuencia	1
Número de etiquetas	5

La ejecución del experimento genera varios archivos entre ellos la generación de los conjuntos difusos y las reglas difusas. Se definieron 5 funciones miembro por cada entrada, como se muestra en la Fig. 6, y se obtuvieron 92 reglas difusas que se aprecian en la Fig. 7.

Initial Data Base:

```

Variable de estado 1:
Label 1: (-159.595,-106.85,-54.105)
Label 2: (-106.85,-54.105,-1.3599999999999994)
Label 3: (-54.105,-1.3599999999999994,51.38499999999999)
Label 4: (-1.3599999999999994,51.38499999999999,104.13)
Label 5: (51.38499999999999,104.13,156.87499999999997)

Variable de estado 2:
Label 1: (-159.595,-106.85,-54.105)
Label 2: (-106.85,-54.105,-1.3599999999999994)
Label 3: (-54.105,-1.3599999999999994,51.38499999999999)
Label 4: (-1.3599999999999994,51.38499999999999,104.13)
Label 5: (51.38499999999999,104.13,156.87499999999997)

Variable de estado 3:
Label 1: (-159.595,-106.85,-54.105)
Label 2: (-106.85,-54.105,-1.3599999999999994)
Label 3: (-54.105,-1.3599999999999994,51.38499999999999)
Label 4: (-1.3599999999999994,51.38499999999999,104.13)
Label 5: (51.38499999999999,104.13,156.87499999999997)

```

Fig. 6. Definición de los conjuntos difusos generado por KEEL

Numero de reglas: 92

```

-159.595 -106.85 -54.105
-106.85 -54.105 -1.3599999999999994
-1.3599999999999994 51.38499999999999 104.13
-0.014577644782935597 0.0039106688733581444
0.0054274313024145066 45.72206193447351

-159.595 -106.85 -54.105
-106.85 -54.105 -1.3599999999999994
51.38499999999999 104.13 156.87499999999997
0.0037551955265726668 -2.755209740949475 -0.36655022990565145
3.6298459803195846

-159.595 -106.85 -54.105
-54.105 -1.3599999999999994 51.38499999999999
-159.595 -106.85 -54.105
0.0018299913094992383 0.003986552964615619 -0.001257920554151624
50.07268489539697

-159.595 -106.85 -54.105
-54.105 -1.3599999999999994 51.38499999999999
-106.85 -54.105 -1.3599999999999994
0.5058123260101296
1.4678236525968675 -0.646646642619596 -1.8153720065278212

```

Fig. 7. Reglas difusas generadas por KEEL

Las tres variables de entrada consideradas son las variaciones en el periodo t , $t-1$ y $t-2$ mientras que el valor de salida es la variación del S&P 500 en el momento $t+1$.

Las etiquetas lingüísticas *Label 1*, *Label 2*, *Label 3*, etc. corresponden a los conjuntos difusos muy bajo la media (MBM), bajo la media (BM), en la media (EM), sobre la media (SM) y muy sobre la media (MSM) respectivamente.

Con las reglas difusas y los conjuntos generados se procedió a la creación de un modelo Sugeno en Matlab, según la Fig 8, y las reglas ingresadas en la Fig 9.

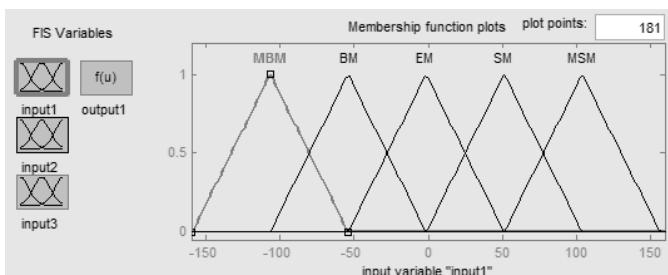


Fig. 8. Definición del FIS en Matlab

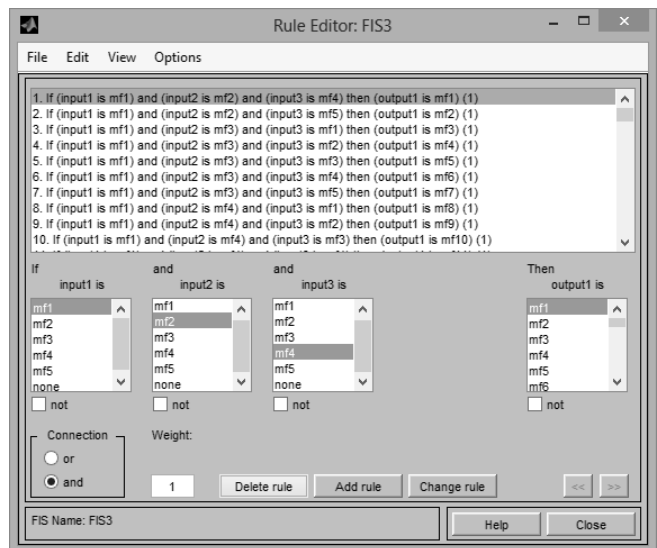


Fig. 9. Reglas ingresadas en MATLAB

4) ANFIS

El Sistema Adaptativo de Inferencia Neurodifuso (ANFIS por sus siglas en inglés) es una red adaptativa que representa la implementación del modelo de inferencia TSK [16].

Las redes adaptativas son estructuras compuestas por diversas capas de nodos conectados a través de enlaces. Cada nodo representa una unidad de procesamiento que va a desempeñar una función sobre la señal recibida para la generación de otra salida. Cada enlace puede depender además de un conjunto de parámetros asociados o no en caso de nodos no adaptativos que solo dependen de las entradas. La Arquitectura del modelo ANFIS se muestra en la Fig. 10.

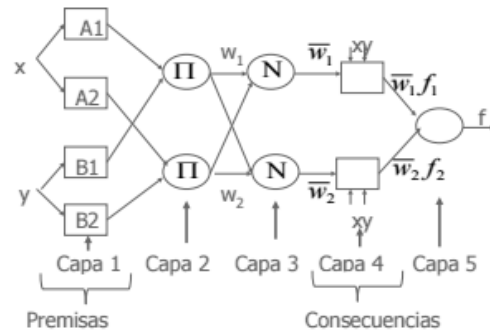


Fig. 10. Arquitectura modelo ANFIS [12]

Se dividió los datos para el entrenamiento y verificación de la red neurodifusa siendo 16 462 observaciones para entrenar la red y 166 restantes para la verificación. Los datos de entrenamiento en MatLab se muestran en la Fig. 11.

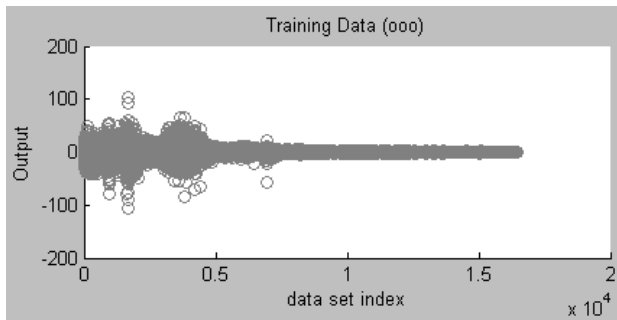


Fig. 11. Datos de entrenamiento en Matlab.

Para obtener un modelo más confiable se realizó pruebas variando las iteraciones consideradas para el entrenamiento de la red neurodifusa. (Ver Tabla III)

TABLA III

COMPARACIÓN DE RESULTADOS POR CADA NÚMERO DE ITERACIÓN

Iteraciones	Error Entrenamiento	Error Promedio Entrenamiento	Error Verificación
5	7.6687	7.6668	24.1721
10	7.6598	7.6582	24.2860
15	7.6521	7.6506	24.3941
25	7.6389	7.6377	24.6035
50	7.6148	7.6140	25.0711
75	7.5978	7.5972	25.4320
100	7.5843	7.5838	25.7070

De los resultados obtenidos se considera mejor el modelo generado con 50 iteraciones por ser en promedio el que obtiene un menor error tanto en entrenamiento como en verificación, como se muestra en la Fig. 12.

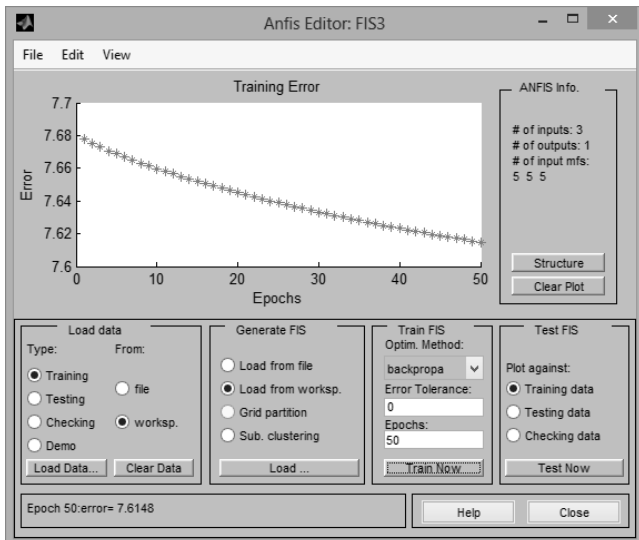


Fig. 12. Error de entrenamiento con 50 iteraciones

En las Fig. 13, Fig. 14, Fig. 15 y Fig. 16, se muestra los resultados obtenidos al poner en funcionamiento los datos en Matlab.

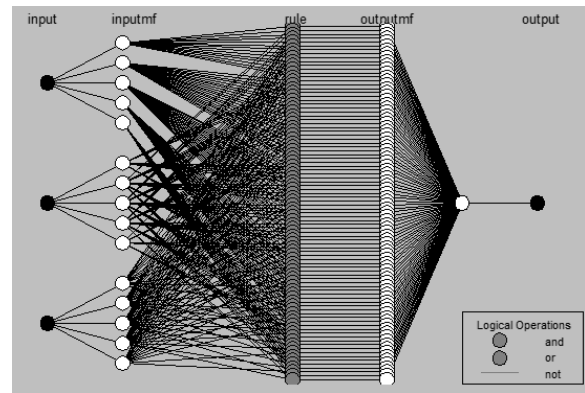


Fig. 13. Arquitectura de la red neurodifusa

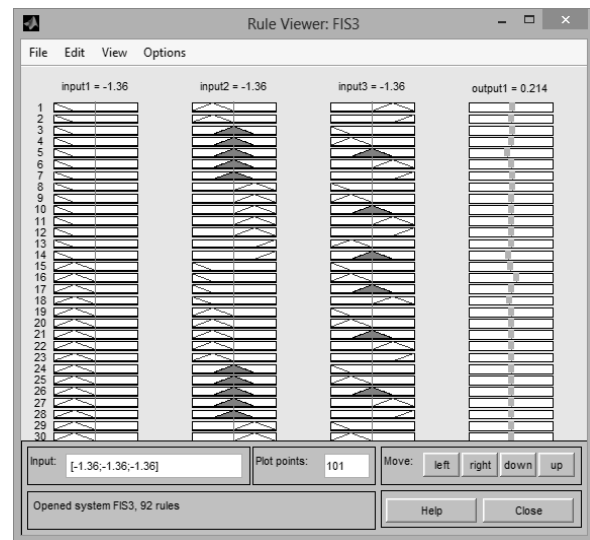


Fig. 14. Reglas difusas con 50 iteraciones

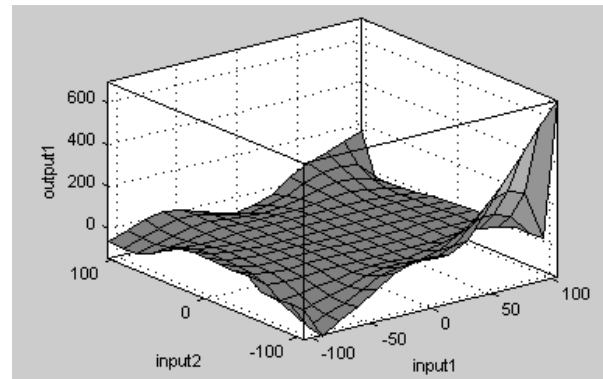


Fig. 15. Superficie de Var1, Var2 y VarP

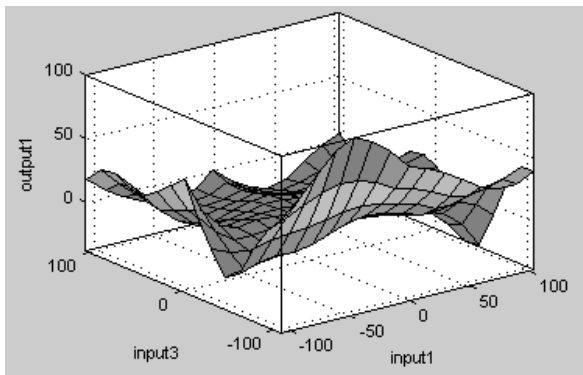


Fig. 16. Superficie de Var1, Var3 y VarP

V. CONCLUSIONES

El modelo propuesto obtiene un menor número de error de entrenamiento en comparación con trabajos anteriores en donde solo se utilizaba las redes neurodifusas. Con este modelo se alcanzó un mínimo de 7.5843 de error en el entrenamiento con 100 iteraciones siendo inferior al valor obtenido anteriormente 7.7249. La inclusión de algoritmos genéticos para la fuzzificación de los datos obtuvo mejores resultados que el algoritmo Grid Partition utilizado en Matlab.

Si bien la generación de un modelo de predicción de un índice bursátil no es del todo factible los resultados obtenidos demuestran que se obtienen valores cercanos a los reales. Se considera interesante la inclusión de algoritmos evolutivos para el entrenamiento y la estructura de la red neuronal para mejorar los resultados obtenidos en este artículo. De manera similar se podría optar por considerar mayor cantidad de entradas para la red neurodifusa.

REFERENCIAS

- [1] R. C. Anderson and D. M. Reeb, "Founding-family ownership and firm performance: Evidence from the S&P 500," *J. Finance*, vol. 58, no. 3, pp. 1301–1328, 2003.
- [2] M. Hagenau, M. Liebmann, and D. Neumann, "Automated news reading: Stock price prediction based on financial news using context-capturing features," *Decis. Support Syst.*, vol. 55, no. 3, pp. 685–697, 2013.
- [3] A. a. Ariyo, A. O. Adewumi, and C. K. Ayo, "Stock Price Prediction Using the ARIMA Model," *2014 UKSim-AMSS 16th Int. Conf. Comput. Model. Simul.*, pp. 106–112, 2014.
- [4] A. A. Adebiyi, A. O. Adewumi, and C. K. Ayo, "Comparison of ARIMA and artificial neural networks models for stock price prediction," *J. Appl. Math.*, vol. 2014, 2014.
- [5] M. I. P. Britos, "Entrenamiento de redes neuronales basado en algoritmos evolutivos," 2005.
- [6] D. Parisi, Antonino; Parisi, Franco; Diaz, "Modelos de algoritmos genéticos y Redes Neuronales en la predicción de índices bursátiles asiáticos," *Cuad. Econ.*, vol. 43, no. Noviembre, pp. 251–284, 2006.
- [7] M. C. García, A. M. Jalal, L. A. Garzón, and J. M. López,

"Métodos para predecir índices bursátiles," in *Ecos de economía*, 2013, pp. 51–82.

- [8] R. J. Kuo, C. H. Chen, and Y. C. Hwang, "An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network," *Fuzzy Sets Syst.*, vol. 118, no. 1, pp. 21–45, 2001.
- [9] K. Hyun-jung and S. Kyung-shik, "A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets," *Appl. Soft Comput.*, vol. 7, no. 2, pp. 569–576, 2007.
- [10] J. Alcalá-Fdez, M. J. del Jesus, J. M. Garrell, F. Herrera, C. Herbás, and L. Sánchez, "Proyecto KEEL: Desarrollo de una herramienta para el análisis e implementación de algoritmos de extracción de conocimiento evolutivos," *Tendencias la Minería Datos en España, Red Española Minería Datos y Aprendiz.*, pp. 413–424, 2004.
- [11] Y. Finance, "Yahoo Finance." Feb-2016.
- [12] L. a. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, no. 3, pp. 338–353, 1965.
- [13] C. González Morcillo, "Lógica Difusa, una introducción práctica," *Técnicas de Softcomputing*, p. 29, 2011.
- [14] J. Alcalá-Fdez *et al.*, "KEEL: a software tool to assess evolutionary algorithms for data mining problems," *Soft Comput.*, vol. 13, no. 3, pp. 307–318, 2009.
- [15] O. Cerdón and F. Herrera, "A two-stage evolutionary process for designing TSK fuzzy rule-based systems," *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, vol. 29, no. 6, pp. 703–715, 1999.
- [16] G. J. Correa-Henao and L. M. Montoya-Suárez, "Aplicación del Modelo ANFIS para Predicción de Series de Tiempo," *Lámpasakos*, no. 9, pp. 12–25, 2013.