

Pontificia Universidad Católica de Valparaíso
Facultad de Ingeniería
Escuela de Ingeniería Informática

**REDES NEURONALES ARTIFICIALES CON
ALGORITMO DE APRENDIZAJE CUCKOO SEARCH
PARA EL PRONÓSTICO DEL VALOR CUOTA DEL
FONDO A DE AFP CUPRUM**

JUAN CARLOS ARTURO RAMOS PERONI

INFORME FINAL DEL PROYECTO
PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO CIVIL EN INFORMÁTICA

JUNIO 2015

Pontificia Universidad Católica de Valparaíso
Facultad de Ingeniería
Escuela de Ingeniería Informática

**REDES NEURONALES ARTIFICIALES CON
ALGORITMO DE APRENDIZAJE CUCKOO SEARCH
PARA EL PRONÓSTICO DEL VALOR CUOTA DEL
FONDO A DE AFP CUPRUM**

JUAN CARLOS ARTURO RAMOS PERONI

Profesor Guía: **Nibaldo Rodríguez Agurto**

Profesor Co-referente: **Wenceslao Palma Muñoz**

Carrera: **Ingeniería Civil Informática**

JUNIO 2015

Dedicatoria

*Dedicado a mi familia, por su cariño y paciencia, y a mi polola,
por su amor y apoyo incondicional*

Agradecimientos

A todas las personas que confiaron en mí. A mi familia, amigos, polola y profesor guía

Índice

Índice.....	iii
Lista de Figuras.....	v
Lista de Tablas.....	vi
Lista de Abreviaturas o Siglas.....	vii
Resumen.....	viii
Abstract.....	viii
1. Introducción.....	1
2. Objetivos.....	4
2.1. Objetivo general.....	4
2.2. Objetivos específicos.....	4
2.3. Organización del texto.....	4
3. Redes neuronales.....	5
3.1. Redes neuronales para pronóstico.....	6
3.2. Redes de base radial.....	11
3.2.1. Arquitectura de la red de base radial.....	11
4. Aprendizaje de la red.....	13
4.1. Aprendizaje supervisado.....	13
4.2. Técnicas metaheurísticas de optimización.....	13
4.2.1. Algoritmo Cuckoo Search.....	14
4.2.2. Algoritmo Modified Cuckoo Search.....	16
5. Estrategias de pronóstico.....	17
5.1. Estrategia recursiva o iterativa.....	17
5.2. Estrategia DirRec.....	17
6. Modelo propuesto.....	18
6.1. Función de activación.....	18
6.2. Métricas.....	19
7. Experimentos.....	20
7.1. Datos de entrada.....	20
7.2. Resultados.....	20
7.2.1. Estrategia recursiva.....	20
7.2.2. Estrategia DirRec.....	25

7.3. Discusión de resultados.....	27
8. Conclusión	28
9. Referencias.....	30

Lista de Figuras

Figura 1.1 Evolución de los fondos de pensión	1
Figura 1.2 Diferenciación de los multifondos.....	2
Figura 1.3 Retorno real de los fondos de pensión.....	3
Figura 3.1 Neurona artificial.....	5
Figura 4.1 Pseudocódigo Cuckoo Search.....	15
Figura 6.1 Modelo propuesto	18
Figura 7.1 Gráfico de determinación	21
Figura 7.2 Gráfico de valor observado v/s valor estimado	21
Figura 7.3 Gráfico de error relativo	21
Figura 7.4 Gráfico de determinación	22
Figura 7.5 Gráfico de valor observado v/s valor estimado	22
Figura 7.6 Gráfico de error relativo	22
Figura 7.7 Gráfico de determinación	23
Figura 7.8 Gráfico de valor observado v/s valor estimado	23
Figura 7.9 Gráfico de determinación	24
Figura 7.10 Gráfico de valor observado v/s valor estimado	24
Figura 7.11 Gráfico de error relativo	24
Figura 7.12 Gráfico de determinación	25
Figura 7.13 Gráfico de valor observado v/s valor estimado	25
Figura 7.14 Gráfico de determinación	26
Figura 7.15 Gráfico de valor observado v/s valor estimado	26
Figura 7.16 Gráfico de valor observado v/s estimado mejor modelo	27

Lista de Tablas

Tabla 3.1 Resultados obtenidos [4].....	6
Tabla 3.2 Resultados obtenidos [12].....	7
Tabla 3.3 Resultados obtenidos [8].....	9
Tabla 3.4 Resultados obtenidos [7].....	10
Tabla 7.1 Resultados obtenidos testing Recursivo Cuckoo Search	21
Tabla 7.2 Resultados obtenidos extrapolación Recursivo Cuckoo Search	23
Tabla 7.3 Resultados obtenidos testing Recursivo Modified Cuckoo Search	23
Tabla 7.4 Resultados obtenidos extrapolación Recursivo Modified Cuckoo Search	25
Tabla 7.5 Resultados obtenidos testing DirRec Cuckoo Search	26
Tabla 7.6 Resultados obtenidos extrapolación DirRec Cuckoo Search	26
Tabla 7.7 Comparación resultados obtenidos por los modelos propuestos	27

Lista de Abreviaturas o Siglas

AFP	: Administradoras de Fondos de Pensión.
CS	: Cuckoo Search.
MCS	: Modified Cuckoo Search.
PSO	: Particle Swarm Optimization.
GA	: Genetic Algorithm.
APV	: Ahorro Previsional Voluntario.
MAE	: Error Absoluto Medio.
MAPE	: Error Medio Absoluto Porcentual.
MSE	: Error Cuadrático Medio.
RMSE	: Raíz del Error Cuadrático Medio.
GCV	: Validación Cruzada Generalizada.
LLRBFN	: Red Neuronal con Función de Base Radial Local Lineal.
LLWNN	: Red Neuronal con Función Wavelet Local Lineal.
ARIMA	: Modelo Autoregresivo Integrado de Media Móvil.
WDBP	: Wavelet Denoising-based Back Propagation.
AWGN	: Ruido Aditivo Blanco Gaussiano.

Resumen

El problema de pronóstico de series de tiempo, consiste en predecir valores futuros que no están medidos, utilizando como variable de análisis los valores históricos de la serie. En este documento, se plantean distintos modelos para el problema del pronóstico de la rentabilidad del fondo A de AFP Cuprum, utilizando como variable de análisis los datos históricos del valor cuota.

Para resolver este problema, se plantea el uso de Redes Neuronales Artificiales de base radial, con función de activación Radial Multicuadrática. Como algoritmo de entrenamiento, se proponen dos metaheurísticas, Cuckoo Search (CS) y Modified Cuckoo Search (MCS), que se comparan ente sí utilizando la estrategia de pronóstico iterativa, que consiste en utilizar los valores pronosticados como datos conocidos para predecir los próximos.

Finalmente, con la metaheurística que obtuvo mejores resultados, se realiza un nuevo pronóstico utilizando la estrategia DirRec, que combina las arquitecturas y los principios subyacentes de las estrategias directa e iterativa. Posteriormente, se comparan los resultados obtenidos por todos los modelos propuestos, siendo el mejor de ellos el Recursivo Cuckoo Search, quien logró un 74,34% en la predicción del sentido para un horizonte de siete días.

Palabras claves: *Pronóstico, Rentabilidad, Valor cuota, Redes Neuronales Artificiales, Cuckoo Search, Modified Cuckoo Search, Estrategia Iterativa, Estrategia DirRec.*

Abstract

The problem of time series forecasting is to predict future values are not measured, using variable analyzes historical values of the series. In this paper, different models for the problem of forecasting the profitability of A Fund of AFP Cuprum, as variable analysis using historical data of share value.

To resolve this problem, the use of artificial neural networks radial basis with activation function Multiquadratic Radial. As training algorithm, two Metaheuristics, Cuckoo Search (CS) and Modified Cuckoo Search (MCS), which compares entity itself using iterative prognosis strategy, which consists in using known values predicted as data to predict the next values.

Finally, with the metaheuristic best performer, a new forecast is made using the DirRec strategy, that combines architectures and underlying principles of direct and iterative strategies. Subsequently, the results obtained for all the proposed models are compared, being the best of them Cuckoo Search Recursive, who achieved 74.34% in predicting sense for a seven-day horizon.

Keywords: *forecast, profitability, share value, Artificial Neural Networks, Cuckoo Search, Modified Cuckoo Search, Iterative Strategy, DirRec Strat*

1. Introducción

Las entidades encargadas de administrar los fondos de pensión y ahorros previsionales son las Administradoras de Fondos de Pensiones, más conocidas como AFP. Éstas instituciones son de carácter privado y su finalidad única es la administración de los fondos de pensión, así como el otorgamiento de las prestaciones por vejez, invalidez y sobrevivencia a sus afiliados. Todo trabajador que posea un contrato de trabajo debe estar afiliado a una AFP, depositando mensualmente el 10% de su remuneración y renta imponible para acumular recursos destinados a su pensión, a esta acción se le denomina cotización.

En 1980, se diseña el nuevo sistema de pensiones dictada por la ley N° 3500, entrando en operación al año siguiente. Al principio se conformó por un solo fondo de pensión, denominado fondo tipo 1, el cuál invertía en renta fija e inversión nacional. En 1985, se autoriza la inversión en renta variable. En 1988, se autoriza el ahorro voluntario, comúnmente llamado APV. En 1990, se autoriza la inversión extranjera, siendo en 1993 su comienzo. En el año 2000, se crea el segundo fondo, el fondo tipo 2. Ya en el 2002, comienzan a operar los multifondos, correspondiendo el fondo C al tipo 1 y el fondo E al tipo 2. Este año también se realiza una reforma al APV. Los fondos existentes por los cuales se puede optar van desde el fondo A, que cuenta con una mayor proporción de renta variable, siendo el más riesgoso y volátil, pero que entrega mayor rentabilidad en el largo plazo, hasta el fondo E, menos riesgoso y con mayor proporción en renta fija. En la figura 1.1, se ilustra la evolución en el tiempo de los fondos de pensión.

El sistema previsional está compuesto por tres pilares fundamentales, esto en pos de asegurar la protección social de cada uno de los ciudadanos del país. El pilar solidario, es para prevenir la pobreza de la población adulta mayor. El pilar obligatorio, es la contribución que

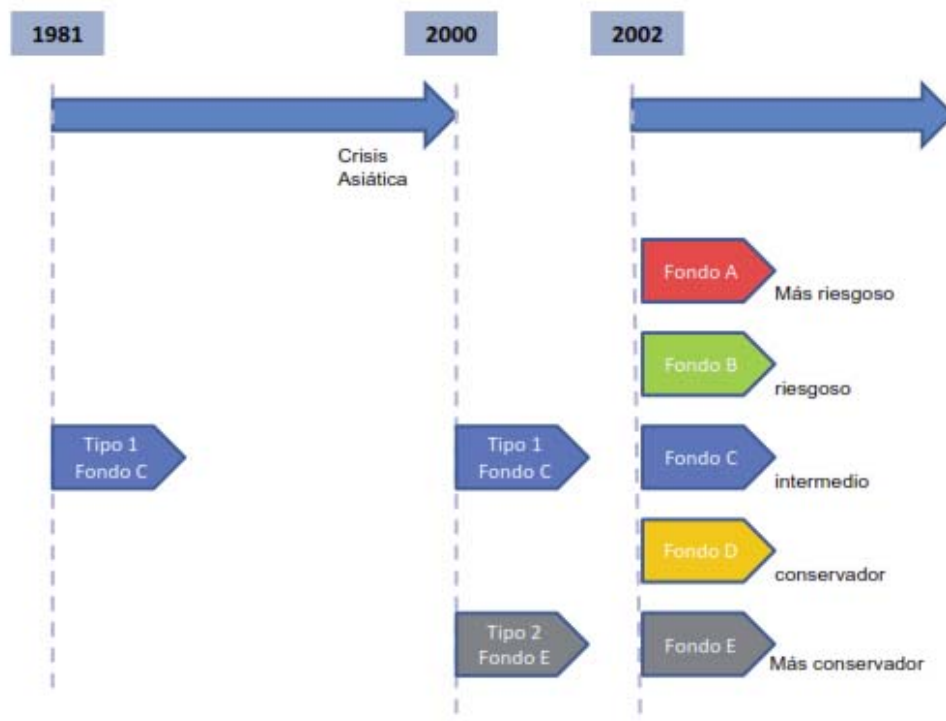


Figura 1.1 Evolución de los fondos de pensión

hace cada afiliado cotizando. Y el pilar voluntario, es una alternativa de ahorro complementario para aumentar el monto de la pensión o anticipar la edad de pensión. El sistema no había sufrido una reforma integral desde su creación sino hasta el 2008, año en el que aumenta la integración y coherencia entre estos tres pilares.

En sus inicios el sistema estuvo conformado por 12 AFPs, pasando a 6 en la actualidad, e incluyendo una ley para la creación de una AFP estatal. La cantidad de afiliados a junio de 2015 en el sistema corresponde a 9.698.105, de los cuales sólo 5.452.389 cotizan.

La cartera de inversiones de los fondos está limitada por ley, la cual define el instrumento de inversión en el que pueden participar y la cantidad a invertir de cada fondo. Para el fondo A, el máximo en renta variable es un 80% del valor del fondo previsional y en inversión extranjera un 100%. En cambio, para el fondo E es sólo un 5% y 35% respectivamente. En la figura 1.2, se puede apreciar el límite de inversión en renta fija y renta variable por fondo de pensión. La inversión de cada fondo está diversificada, de manera tal que un eventual impacto negativo no afecte significativamente la seguridad ni la rentabilidad del total de las inversiones. A junio de 2015, para el fondo A, la cartera está diversificada en renta variable nacional 13,7%, renta fija nacional 8,7%, renta variable extranjera 64,8% y renta fija extranjera 13%.

Es un sistema de capitalización individual, invirtiendo lo acumulado por el afiliado en el

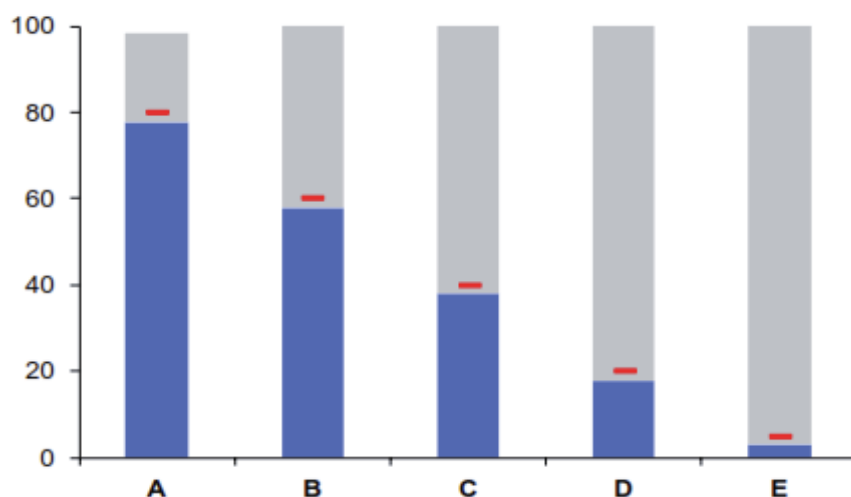


Figura 1.2 Diferenciación de los multifondos

mercado de valores de acuerdo al fondo de pensión al que está sujeto, teniendo por finalidad obtener una cierta rentabilidad. El valor cuota, es la unidad mínima con la que se expresa el valor de cada fondo, los recursos acumulados de cada afiliado se miden en un número de cuotas, cuyo valor se expresa en pesos y que varía día a día. El precio de esta cuota no es de mayor relevancia, sino su variación, la cual se traduce en la rentabilidad de los fondos invertidos de cada afiliado. La rentabilidad real para junio de 2015 es de un 3,8% y los últimos 12 meses es de un 22,06%.

El sistema de pensiones no ha estado exento de polémicas y es cuestionado desde hace un par de años. Las pensiones que reciben los afiliados distan mucho de las promesas iniciales y están muy por debajo de las cifras entregadas. El promedio de las pensiones pagadas no supera los \$180.000. Bajo el ojo crítico de los expertos, en el mejor de los casos la pensión recibida

corresponde a la mitad del promedio del sueldo mensual de los últimos 10 años. La causa de esto se debe a la intermitencia en las cotizaciones o por las bajas remuneraciones recibidas, incluyendo también en algunos casos malas decisiones en los cambios de fondo, generando una pérdida en el capital del afiliado.

El promedio anual de la rentabilidad real de los fondos A y E, desde su creación hasta la fecha es de 7,49% y 4,39% respectivamente. Esto se puede contrastar con la teoría de los expertos de que es una inversión a largo plazo. Al ser más volátil el fondo A, está más expuesto a sufrir los impactos de crisis económicas, como es el caso de la crisis de 2008. En la figura 1.3, se ilustra el retorno real de los fondos, incluyendo el impacto de esta crisis. A partir de este hecho, pareciera existir un creciente interés por parte de los afiliados de manejar de mejor forma

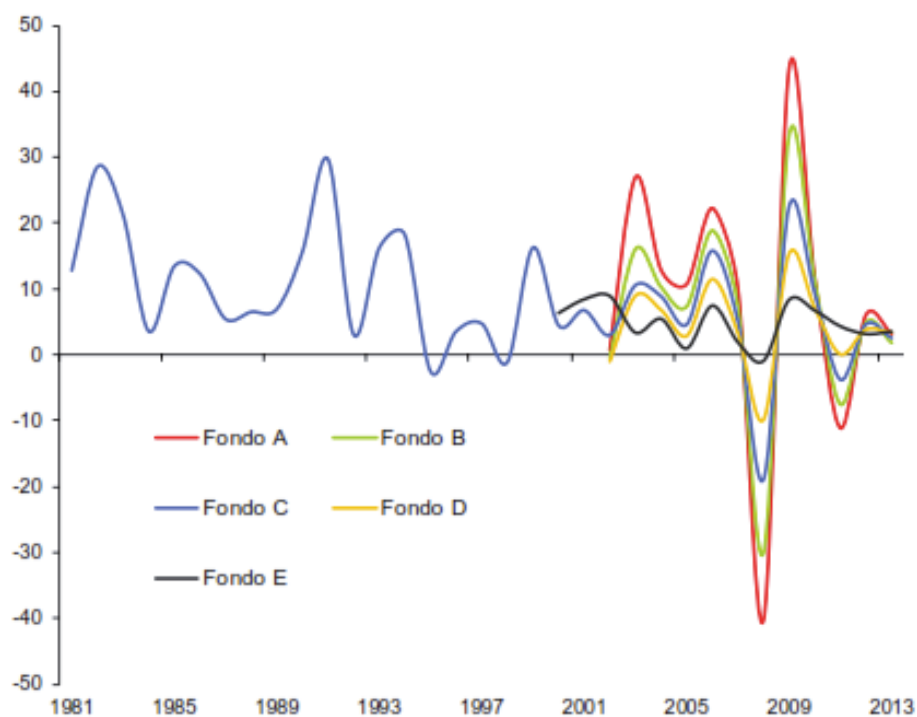


Figura 1.3 Retorno real de los fondos de pensión

sus fondos previsionales, que han o ahorrarán durante su vida laboral. Esto podría beneficiar a los afiliados siempre y cuando tomen decisiones conscientes e informados sobre el riesgo asociado al cambio de fondo, y lograr así, invertir su capital en el tipo de fondo más adecuado según su horizonte de inversión y tolerancia al riesgo.

Son muy pocas las herramientas disponibles que ayuden a la elección de un cambio de fondo de pensión. Para su utilización se necesitan conocimientos técnicos avanzados, ya que están basados en modelos matemáticos y técnicas de mercado. Si existiera un pronóstico confiable de la rentabilidad que entregará cada fondo de pensión, el sistema podría entregar pensiones más dignas a los afiliados según las decisiones que estos tomaron.

2. Objetivos

2.1. Objetivo general

Desarrollar y evaluar un modelo de regresión basado en redes neuronales artificiales para la predicción del sentido de la rentabilidad, basado en el precio del valor cuota del fondo A de AFP Cuprum.

2.2. Objetivos específicos

- Explicar el funcionamiento de las redes neuronales y su aplicación al pronóstico de series de tiempo, así como también el del algoritmo Cuckoo Search (CS) y Modified Cuckoo Search(MCS).
- Definir la estructura y los parámetros de los pronosticadores neuronales que mejor se ajusten a los requerimientos del problema.
- Definir las métricas que se utilizarán para evaluar y comparar el rendimiento de los modelos propuestos para elegir el óptimo.
- Realizar predicciones con los modelos siguiendo dos estrategias distintas de pronóstico, estrategia iterativa y estrategia DirRec, explicando su funcionamiento, similitudes y diferencias.

2.3. Organización del texto

El presente documento se encuentra organizado en capítulos. El Capítulo 3, presenta el estado del arte de las redes neuronales artificiales y sus características esenciales, se exponen investigaciones recientes que las utilizan en problemáticas de pronóstico, y además, una descripción de las redes neuronales con función de base radial y su arquitectura.

En el capítulo 4, se presentan los métodos de aprendizaje con la que se entrena una red neuronal, entregando una pequeña reseña sobre aprendizaje supervisado y las metaheurísticas Cuckoo Search y Modified Cuckoo Search, que fueron utilizadas para entrenar los modelos propuestos.

En el capítulo 5, se exponen las dos estrategias de pronóstico utilizadas que se combinan con redes neuronales, incluyendo una explicación de sus características, estas son la estrategia iterativa y la estrategia DirRec.

En el capítulo 6, se presentan los modelos propuestos y su función de activación. También, se dan a conocer las métricas que se utilizaron para medir el rendimiento cada una y su posterior comparación entre ellas.

En el capítulo 7, se realizan los experimentos de la investigación, mostrando los datos de entrada para los modelos y la configuración de cada metaheurística. Posteriormente, se presenta un análisis de resultados obtenidos por los modelos y una discusión sobre estos. Para finalizar, en el último capítulo se exponen las conclusiones obtenidas en esta investigación.

3. Redes neuronales

Las redes neuronales artificiales pueden ser descritas como un modelo de regresión no lineal, cuya estructura se inspira en el funcionamiento del sistema nervioso. Representan un esquema de cómputo distribuido, compuesta por un gran número de unidades simples procesadoras adaptativas interconectadas entre ellas, a través de vínculos ponderados y que actúan de manera paralela[3][12].

El cerebro humano está formado por unos cien mil millones de neuronas. Cada una de ellas consta de un cuerpo celular del cual surgen 2 tipos de ramificaciones: el axón (Fibra principal) y las dendritas (varias ramas fibrosas). Cada señal de entrada es captada por las dendritas, luego integradas y procesadas en su cuerpo celular para ser emitidas por el axón. Entre este terminal y el siguiente conjunto de dendritas se produce la sinapsis (transmisión de la excitación).

Análogamente, cada elemento procesador (neurona) de la red neuronal artificial recibe parámetros de entrada (dendritas), los cuales son procesados por una función interna (en su soma o cuerpo celular), para generar una única salida que luego será enviada hacia los demás elementos de la red (a través del axón). Las neuronas se agrupan en capas, las cuales están compuestas por múltiples neuronas procesadoras, cada una con comportamiento similar, constituyendo varias capas una red neuronal.

Existen tres elementos básicos que ayudan a modelar una neurona:

- Cada neurona recibe entradas desde las otras neuronas, generando una salida que depende de los ponderadores de las conexiones. Si son de valor positivo actuarán como excitadores, de ser negativos actúan como inhibidores.
- Función de activación, que corresponde generalmente a una función no lineal que normaliza el estado interno de la neurona, limitando la amplitud de salida de ésta.
- Función de transferencia, generalmente es un sumador o multiplicador, se encarga de procesar las señales entrantes a la neurona según su ponderador.

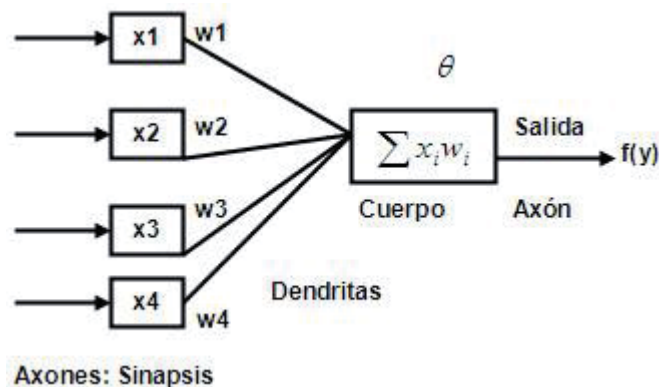


Figura 3.1 Neurona artificial

3.1. Redes neuronales para pronóstico

Un estudio realizado en India, pronostica el precio de equilibrio de mercado de la electricidad[6] con una red neuronal de tres capas backpropagation. Basándose en experiencia de análisis de mercado y de análisis de correlación, toma como variables de entrada el precio histórico de la electricidad, la carga del sistema y el precio del petróleo. El conjunto de datos es de ocho meses, de los cuales ocupa seis para el entrenamiento. Para el testing utiliza varios días de un mes en particular, dentro de estos días se encuentran días con una demanda normal, días con una pequeña alza en la demanda y días con una alta alza en la demanda. Cada día se divide en segmentos de media hora y esto ingresa a la red. La red neuronal cuenta con dos capas ocultas y el horizonte a pronosticar es de dos días. Las métricas ocupadas son MAE y RMSE, obteniendo un MAE de 0,682 y un RMSE de 1,129 para los días con una pequeña alza, MAE de 2,655 y RMSE de 0,525 para los días con una demanda normal y un MAE de 9,282 y RMSE de 4,105 para los días con una alta alza en la demanda.

Se presenta en esta investigación, un modelo para el pronóstico de precios a corto plazo de la electricidad para días similares y precios históricos[4]. Este estudio presenta un modelo para pronosticar el precio de equilibrio mediante un perceptrón multicapa, basándose en la optimización de la arquitectura y los pesos de la red. Para la determinación tanto del número de capas ocultas y de neuronas en estas capas, se utiliza algoritmos genéticos; siendo la mejor configuración una red neuronal feedforward de tres capas, con veintinueve neuronas en la capa oculta. Se utilizaron cuatro técnicas evolutivas diferentes como algoritmo de entrenamiento, para ajustar los pesos de la red y estas técnicas se comparan entre sí, en todas ellas se ha utilizado el mismo número de población inicial igual a cuarenta. El modelo propuesto, es optimizado finalmente con Cuckoo Search usando una configuración de P_a de 0,3 y lambda igual a 2,5. Los conjuntos de datos utilizados son de los mercados de electricidad de Nord Pool 2012 y del mercado iraní. Para el primer caso, el conjunto de entrada consistió en 17 precios escalonados desde uno hasta ciento sesenta y nueve horas atrás, la salida se obtiene por recursión y se predicen veinticuatro horas hacia adelante. El período de entrenamiento consiste en cuarenta y ocho días, contando con mil ciento cincuenta y dos datos. En la tabla 3.1, se comparan los distintos algoritmos con cincuenta iteraciones independientes y la métrica utilizada es el MSE. Para el segundo caso, se eligieron cuatro semanas diferentes, una por cada estación del año en el mercado de la electricidad de Irán entre 2012-2013. Cada semana seleccionada tiene siete días para cubrir todos los tipos de días, es decir, día de semana, fines de semana y festivos. Posteriormente se eligieron cuatro días, uno por semana y se pronosticó. La métrica utilizada es el MAPE, dando como resultados para primavera un MAPE de 1,6089, verano igual a 0,91588, otoño de 1,023 e invierno un MAPE de 1,1414.

Tabla 3.1 Resultados obtenidos [4]

Solution approach	Best (fitness)	Ave. (fitness)	Worst (fitness)	Ave. convergence iteration
GA	2324.84	2603.032	2902.885	159
PSO	2000.71	2485.54	2962.05	184
ICA	1747.08	2312.68	2665.59	171
CS	1503.57	1802.11	2256.65	139

Un estudio de pronóstico bursátil realizado por dos investigadores iraníes, proponen una red neuronal con función de base radial local lineal (LLRBFN), para pronosticar el precio de las acciones de Irankhodro[5]. En este tipo de red, los pesos de conexión entre las neuronas de la capa oculta y de salida se sustituyen por un modelo lineal local. Esta red se compara con otros modelos, como un perceptrón multicapa y una red neuronal local lineal wavelet (LLWNN). Como algoritmo de aprendizaje para el modelo propuesto, se utilizó una modificación de PSO con partículas cazadoras, los parámetros fueron seleccionados a través de ensayo y error, siendo estos una población de cuarenta partículas, cincuenta iteraciones y coeficientes de confianza C_1 y C_2 de 2 y 5, respectivamente. El conjunto de datos fue de mil setecientos, que corresponden al precio de las acciones diarias de Irankhodro, de los cuales mil fueron para training. Todos los datos se normalizaron entre [-1, 1]. La red cuenta con tres entradas, que equivalen a los días previos al día que se pronosticará, diez nodos ocultos y un nodo de salida. Los resultados obtenidos por el modelo fueron para training un MSE de 0,0040 y para testing de 0,0042, la correlación entre los objetivos y los resultados del modelo es de 0.9887.

En este trabajo, se presenta una red neuronal para el pronóstico de m días en los precios de la electricidad en el European Energy Exchange (EEX)[12]. Se utilizan mil noventa y seis datos diarios, registrados desde el año 2002 hasta el año 2005. Se llevaron a cabo más de cincuenta experimentos para determinar la mejor configuración para el índice de aprendizaje, el impulso, el número de nodos de entrada y el de nodos ocultos. Para esta tarea, la métrica utilizada fue el RMS, siendo los valores óptimos encontrados de 0,08 en la tasa de aprendizaje, 0,1 para impulso, cinco nodos de entrada y siete nodos ocultos. Esta investigación, propone una red neuronal con un único nodo de salida y utiliza el enfoque de predicción directa. El método de validación cruzada con quince períodos de predicción, se utiliza para reducir el efecto de la variación de muestreo. En el entrenamiento, se aplica la regla de aprendizaje de datos generalizada y como función de transferencia la función sigmoideal. El rendimiento se mide mediante tres métricas, RMSE, MAE y MAPE, estas también se aplicaron a los modelos autorregresivos que se utilizan para comparar el modelo propuesto. Se tiene como objetivo cinco horizontes de pronóstico $m= 7, 14, 21, 28$ y 91 días. El modelo propuesto se desempeña mejor que los modelos autorregresivos, en cada horizonte de pronóstico. Los valores medios de las tres métricas para los 15 períodos de predicción, en los modelos autorregresivos es más grande que las del modelo propuesto. Para el pronóstico de tres meses, el modelo propuesto todavía tiene valores medios más pequeños para las métricas. En la tabla 3.2 se muestran los resultados obtenidos.

Tabla 3.2 Resultados obtenidos [12]

Forecasting horizon	One week		Two weeks		Three weeks		Four weeks		Three months	
	ANN	AUTO	ANN	AUTO	ANN	AUTO	ANN	AUTO	ANN	AUTO
	<i>The averages of RMSE, MAE, and MAPE for 15 forecasting periods</i>									
RMSE	3.21	5.70	3.16	6.26	3.25	7.50	3.13	7.73	3.89	9.17
MAE	2.71	4.79	2.47	5.18	2.38	6.15	2.59	6.19	3.06	6.96
MAPE	9.02	15.16	8.24	15.24	8.59	16.39	8.36	16.98	8.85	20.55
<i>The standard deviations of RMSE, MAE, and MAPE for 15 forecasting periods</i>										
RMSE	1.25	2.51	1.13	3.22	1.20	5.45	1.16	5.11	1.12	4.56
MAE	1.09	2.09	1.01	2.78	1.05	4.28	1.02	3.85	0.98	3.17
MAPE	3.18	7.59	2.41	6.04	2.33	6.55	2.30	5.97	2.05	4.42

El objetivo principal de este estudio, es examinar si el modelo de redes neuronales es adecuado para la predicción a corto plazo del precio de productos agrícolas, y cómo su precisión se compara con un modelo de series de tiempo[14]. Se utiliza la predicción histórica y futura. La histórica es principalmente para evaluar el rendimiento de los dos métodos basados en datos históricos, y la futura, es la de examinar la veracidad de los resultados de los pronósticos sobre la base de los dos modelos. Se utilizó el tomate debido a que es uno de los productos agrícolas más populares y conocidos en China. Los datos históricos empleados son del Ministerio de Agricultura de China (MOA), que está conformado por el precio diario de tomates de quinientos once mercados mayoristas desde 1996. Los datos utilizados fueron los precios diarios, semanales y mensuales, pero para algunos mercados mayoristas no existían y se utilizó un promedio de estos mercados. Como los datos no eran adecuados para desarrollar buenos modelos, se normalizaron en la etapa de preprocesamiento y los datos se dividieron para training y testing. Para evaluar el rendimiento de los diferentes modelos se utilizó el error relativo. Como modelo propuesto se utilizó una red neuronal feedforward de tres capas. El número de neuronas de la capa de entrada es n y de la capa de salida m , para predecir el precio de los próximos m días sobre la base del precio de últimos n días. De acuerdo con esto, el número de neuronas de la capa oculta se establece como $2n + 1$, basado en el teorema de Kolmogorov. Como métrica de rendimiento del modelo se utiliza el MSE, y la fase de entrenamiento se detendrá cuando se alcance el MSE deseado o se alcance el número máximo de iteraciones. Para el entrenamiento de la red en el pronóstico diario, se ocuparon datos entre 1996 y 2008, y del año 2009 para testing. La distribución de la precisión de predicción histórica entre 90% y 95% fue de 29,9%, y para más de 95% fue de 68% de error relativo. Para el pronóstico semanal, los datos de entrenamiento fueron de 1996 a 2006, para testing se utilizaron datos de 2007 a 2009. La distribución de la precisión de predicción entre 90% y 95% fue 19,2%, y para más de 95% de 61,6% de error relativo. En el pronóstico mensual los datos de entrenamiento y testing estuvieron entre los años 1996 a 2009. Entre 90% y 95% de la distribución de precisión tuvo un 21,7% de error relativo, en cuanto para más del 95% fue de un 31,7%. Posteriormente, se comparan modelos ARIMA con la red neuronal en pronósticos para siete días, cuatro semanas y tres meses adelante. El modelo ARIMA y el modelo neuronal son eficaces para la predicción de los precios diarios y la precisión es más del 90%, pero la precisión del modelo neuronal disminuye a mayores ciclos de predicción, pero aun así está por sobre un 80%.

Un estudio aplica redes neuronales para la predicción del crecimiento económico de Zhejiang[8], una provincia ubicada en la costa sureste de China que ha obtenido un gran crecimiento en los últimos años. Como variable de entrada, se eligió la proporción del PIB que aporta el sector terciario al PIB de China, debido a que es el sector que más aportó al crecimiento económico. Los datos son normalizados para acelerar la convergencia, ocupando la fórmula $x_i = (x_i - x_{min}) / (x_{man} - x_{min})$, de estos, de 1971 a 1994 se ocupan para training y de 1995 a 1997 para testing. La cantidad de nodos de entrada es de tres, ya que entran los primeros tres años previos para generar como salida el año actual, siendo el único nodo de salida. Tiene una sola capa oculta, y la cantidad de nodos ocultos se rige por la ley de Kolmogorov, $2n + 1$, pero con pruebas posteriores se encontró que con cinco nodos se obtenían mejores resultados. Es una red feedforward que utiliza como algoritmo de entrenamiento backpropagation. Los parámetros de entrenamiento fueron una tasa de aprendizaje de 0,85 y la constante de momentum de 0,6. Esta red se compara con *gray forecasting model*, una técnica muy utilizada para el pronóstico del crecimiento económico, siendo las redes neuronales las que obtuvieron los mejores

resultados, obteniendo un error medio comparativo de 3% y un error máximo comparativo de 10,77%.

Tabla 3.3 Resultados obtenidos [8]

	Year	x_t	y_o	y_{ann}	e_{ann}	y_{gm}	e_{gm}
Training sample	1971	21.2	0.2029	19.828	-6.47	21.200	0.00
	1972	18.3	0.2245	20.181	10.28	16.165	11.67
	1973	18.2	0.0793	17.801	-2.19	16.704	8.22
	1974	18.6	0.0936	18.035	-3.04	17.261	7.20
	1975	19.5	0.1449	18.876	-3.21	17.836	8.53
	1976	19.5	0.1767	19.399	-0.52	18.431	5.48
	1977	19.3	0.1553	19.046	-1.31	19.045	1.32
	1978	18.6	0.1270	18.583	-0.09	19.680	-5.81
	1979	16.5	0.1083	18.277	10.77	20.336	-23.25
	1980	17.2	0.0962	18.078	5.11	21.014	-22.18
	1981	19.9	0.1663	19.227	-3.38	21.715	-9.12
	1982	21.4	0.3229	21.796	1.85	22.439	-4.85
	1983	23.5	0.3861	22.832	-2.84	23.187	1.33
	1984	23.7	0.4696	24.202	2.12	23.960	-1.10
	1985	24.5	0.4798	24.369	-0.54	24.759	-1.06
	1986	26.6	0.5726	25.891	-2.67	25.584	3.82
	1987	27.0	0.6397	26.991	-0.03	26.437	2.08
	1988	28.1	0.7056	28.072	-0.10	27.319	2.78
	1989	29.2	0.8261	30.048	2.90	28.229	3.32
	1990	29.5	0.8485	30.415	3.10	29.171	1.12
Test sample	1991	31.6	0.8899	31.094	-1.60	30.143	4.61
	1992	32.9	0.9168	31.536	-4.15	31.148	5.32
	1993	32.0	0.9299	31.750	-0.78	32.187	-0.58
	1994	31.3	0.9571	32.196	2.86	33.260	-6.26
	1995	32.1	0.9637	32.305	0.64	34.369	7.07
	1996	32.2	0.9557	32.174	-0.08	34.981	8.64
	1997	32.2	0.9521	32.114	-0.27	35.485	10.20

x_t is true value of the tertiary industry proportion in Zhejiang; y_o is output value of ANN; y_{ann} is forecast value of ANN; and y_{gm} is forecast value of GM.

En este estudio, se propone un modelo de pronóstico híbrido llamado Wavelet Denoising-based Back Propagation (WDBP)[7]. La transformada wavelet, se utiliza en el análisis de series de tiempo no estacionarias para la generación de información, tanto en el dominio del tiempo y como en el de frecuencia. Descompone una señal en versiones desplazadas y escaladas de una onda madre, en este estudio se utilizó wavelet Daubechies (DB3). A diferencia de otros modelos de redes backpropagation, que utilizan directamente los datos originales, primero se descompusieron los datos originales en una señal de baja y de alta frecuencia. La señal de baja frecuencia se normaliza y es la que se utiliza como entrada para la red, está descompuesta en seis capas, estableciendo seis modelos WDBP. Los datos de los experimentos son los precios de cierre de SCI (Shanghai Composite Index), recogidos desde la Bolsa de Valores de Shanghai (SSE). La cantidad total de precios de cierre de SCI es de doscientos cuatro meses, desde enero de 1993 hasta diciembre de 2009. El conjunto de datos se divide en un 80% para entrenamiento, y en un 20% en pruebas para su validación. En este trabajo, se optó por una red neuronal backpropagation de tres capas, que tiene tres neuronas en la capa de entrada, diez neuronas en

la capa oculta y tres neuronas en la capa de salida. La red neuronal WDBP, se compara con una red neuronal backpropagation con los datos originales. Las métricas para evaluar los modelos fueron MAE, RMSE y MAPE. El modelo BP4 tiene los errores más pequeños entre estos modelos, por lo tanto, fue el elegido. Además, supera significativamente al modelo backpropagation con los datos originales.

Tabla 3.4 Resultados obtenidos [7]

Errors	BP	BP1	BP2	BP3	BP4	BP5	BP6
MAE	930.2404	1620.6	1038.4	1340.3	675.9543	1067.1	1377.7
RMSE	1174.3	2006.2	1412.7	1984.4	847.5841	1349.7	1619.1
MAPE	0.2492	0.4364	0.2584	0.4831	0.1948	0.3056	0.4276

Se utilizan redes neuronales artificiales, para la predicción del precio de las acciones y la mejora del índice de predicción direccional, para las acciones de Petrobras PETR4 que se transan en BM&FBOVESPA[20]. Este estudio se sustenta en teoría económica y financiera, combinando análisis fundamental, técnico y de series de tiempo. Las variables de entrada, se eligieron según la disponibilidad y acceso a los datos, para poder construir el conjunto de muestra. De éstas, quince son de visión técnica y once de visión fundamental, posteriormente, fueron analizadas usando la función de correlación cruzada entre las variables y PETR4, seleccionando finalmente dieciocho. Las fuentes utilizadas son Economatica, Banco Central de Brasil, BM&FBOVESPA y Thomson Reuters. Los datos, están compuestos por observaciones mensuales que van desde enero del 2000 a diciembre del 2011, contando con ciento cuarenta y cuatro registros en total. Éstos se normalizaron entre 0,2 y 0,8, a pesar de que las funciones de activación no lineal van de 0 a 1, estos valores no son alcanzados. Para el modelo, se utilizó el método directo para pronosticar con un horizonte igual a 1. Alrededor de dos tercios de los datos son utilizados para entrenar y un tercio para testear. Es una red feedforward con tres capas, y la cantidad de nodos de entrada, se determina por la multiplicación entre el tamaño de la ventana y el número de series en el conjunto de muestras. Los nodos ocultos se rigen por la ecuación $2\sqrt{(m + 2)N}$, donde m representa el número de nodos en la capa de salida y N los nodos en la capa de entrada. Los nodos de la capa de salida son fijados por el tamaño del horizonte de pronóstico. Como función de activación se utilizó la función sigmoïdal, y como algoritmo de entrenamiento resilient backpropagation, una adaptación del estándar backpropagation que trata de eliminar la influencia negativa de la derivada parcial en el ajuste de los pesos. El número de iteraciones se fijó en diez mil y la tasa de error mínima en 0,0001. Las métricas utilizadas son MAPE, RMSE, THEIL, que evalúa el rendimiento de la predicción en relación a la predicción trivial, lo que significa que el valor futuro estimado es igual al valor actual, y POCID, predicción direccional que mide el porcentaje de exactitud relativa de la tendencia de la serie. Se realizaron experimentos para determinar el mejor tamaño de la ventana de entrada para las variables, para el valor de 5 se obtuvo el MSE más bajo, pero en testing el valor fue de 3, obteniendo un POCID de 93,62%, y MAPE de 5,45, siendo este el valor elegido para la ventana y utilizado en las demás pruebas. Para validar el modelo se realizaron pruebas en el intervalo de enero a noviembre de 2012, con un total de once registros. Aquí las métricas presentan un rendimiento más pobre en comparación con la etapa de testing, pero POCID obtuvo 87,5%, que es considerado satisfactorio. La explicación para los puntos divergentes y el rendimiento se debe a condiciones atípicas del mercado.

3.2. Redes de base radial

Las redes de base radial, son un tipo de red neuronal artificial propuestas por Broomhead y Lowe[1] como una alternativa al perceptrón multicapa, para hacer ajustes a funciones no lineales. En la capa oculta, están diseñadas con neuronas que contienen funciones radiales de carácter no lineal con sus centros gravitacionales propios, y en la capa de salida, con neuronas que contienen funciones lineales.

Las funciones de base radial tienen un carácter local. Esto debido a que son funciones que alcanzan un nivel cercano al máximo de su recorrido, cuando el patrón de entrada x está próximo al centro de la neurona. La función calcula la distancia euclídea del vector de entrada con respecto al centro. A medida que el patrón se aleja del centro, el valor de la función va tendiendo al valor mínimo de su recorrido.

Las salidas de las redes de base radial son una combinación lineal de la salida de cada neurona multiplicado por su respectivo peso.

3.2.1. Arquitectura de la red de base radial

Las redes de base radial son redes del tipo multicapa con conexión feedforward, esto es, la conexión se realiza sólo hacia adelante, sin posibilidades de retroalimentación[2].

Estas redes están conformadas por tres capas: capa de entrada, capa oculta y la capa de salida. La capa de entrada, está compuesta por neuronas que introducen los datos de entrada a la red y transmiten estas señales a la capa oculta. La segunda capa, la capa oculta, es la encargada de aplicar una transformación no lineal del espacio de entrada al espacio oculto de alta dimensionalidad. Cada neurona en esta capa tiene asociada una función de base radial, de tal manera que representa una clase o categoría. El primer cálculo que realiza cada elemento es hallar la distancia existente entre la entrada al centro, dada por la ecuación (1).

$$z = \left\| \frac{x_i - t_k}{d_k} \right\| \quad (1)$$

Siendo x_i el vector de entrada, y las componentes t_k y d_k los vectores de centro y desviación de la función, que se representan de la siguiente forma:

$$x_i = (x_{i,1}, x_{i,2}, \dots, x_{i,n}), i - \text{ésima muestra} \quad (2)$$

$$t_k = (t_{k,1}, t_{k,2}, \dots, t_{k,n}), d_k = (d_{k,1}, d_{k,2}, \dots, d_{k,n}), k - \text{ésimo nodo oculto} \quad (3)$$

Posteriormente a este resultado se le aplica la función de base radial, que determinará la salida de la neurona. Existen distintos tipos de funciones de activación, entre las que podemos encontrar están:

- Función Gaussiana

$$f(x) = \exp(-0.5 \cdot z^2) \quad (4)$$

- Función Spline

$$f(x) = z^2 \ln(z) \quad (5)$$

- Función Multicuadrática

$$f(x) = \sqrt{1 + z^2} \quad (6)$$

- Función Multicuadrática inversa

$$f(x) = \frac{1}{\sqrt{1+z^2}} \quad (7)$$

La capa de salida, consta de una unidad lineal interconectada con las neuronas de la capa oculta. Realiza una combinación lineal de la resultante de cada función de base radial y los pesos de salida asociado a cada neurona. Esta capa actúa como la salida de la red y el resultado se obtiene de la siguiente manera:

$$y = \sum_{i=1}^h w_i G_i \quad (8)$$

Donde w_i , son los pesos sinápticos entre la neurona oculta i y la neurona de la capa de salida. G_i es la salida de la capa oculta, que se corresponde con la función radial aplicada a la distancia euclidiana en cada una de las unidades ocultas. La solución proporcionada por esta red será un interpolador óptimo, debido a que entregará la salida con el menor error alcanzado.

4. Aprendizaje de la red

El aprendizaje de una red neuronal es una de las características más importantes. Durante este proceso, se van ajustando adecuadamente los centros, desviaciones y pesos sinápticos para obtener la salida deseada, de forma que la red pueda responder después por sí sola a situaciones diferentes a las aprendidas.

En rigor, es la búsqueda de un hiperplano definido en un espacio n-dimensional que describa las entradas a la red. Ésta debe poseer las características de generalización, ya que debe servir para interpolar datos distintos a los utilizados para su entrenamiento. Lo que pretende una red de base radial, es realizar una correspondencia no lineal, entre los patrones de entrenamiento que definen el espacio de entrada, al espacio oculto definido por la capa oculta. La determinación de los parámetros de esta capa, se realiza mediante la optimización en el espacio de entrada, ya que cada neurona representa una zona diferente en dicho espacio. También busca una correspondencia lineal, desde el espacio oculto al espacio de salida. Los parámetros de la capa de salida, se determinan mediante la optimización en base a las salidas que se desean obtener, es decir, las salidas deseadas.

4.1. Aprendizaje supervisado

En esta investigación se ocupará el método de aprendizaje supervisado, que consiste en utilizar un conjunto de entrenamiento de entradas y salidas deseadas. La salida estimada por la red, se compara con la salida deseada, de esta diferencia se obtiene el error cometido por el estimador, los pesos son ajustados en función de éste y el proceso se realiza iterativamente hasta que este error llega a un nivel aceptable[13].

4.2. Técnicas metaheurísticas de optimización

Las técnicas metaheurísticas, son algoritmos no exactos que se fundamentan en la combinación de diferentes métodos heurísticos a un nivel más alto, para conseguir una exploración del espacio de búsqueda más eficiente y efectiva. Esta combinación se realiza para obtener una solución óptima. Se caracterizan por ser estrategias generales que guían el proceso de búsqueda, sus procedimientos no dependen del problema a resolver y utilizan funciones fitness para cuantificar el grado de adecuación de una determinada solución. Durante su evolución, han incorporado estrategias para evitar la convergencia a óptimos locales y la exploración de regiones no óptimas en espacios complejos. Además, permiten abordar problemas con espacios de búsqueda de gran tamaño, por esto es de especial interés el correcto equilibrio entre diversificación e intensificación[11].

La aplicación de técnicas metaheurísticas es interesante en casos de problemas de optimización combinatoria, en que las variables de decisión son enteras o discretas, y el espacio de soluciones está formado por ordenaciones de valores de dichas variables. Éstas técnicas también pueden ser utilizadas en otros tipos de problemas, por ejemplo, con variables continuas.

Estas técnicas son conocidas como algoritmos de caja negra, debido a que explotan un conocimiento limitado sobre el problema a resolver. La forma de operar es similar entre ellas, el punto de partida es una solución o un conjunto de ellas, que por lo general no es óptima. A partir de esta solución y mediante modificaciones en sucesivas iteraciones, se obtienen otras similares, de entre las cuales se elige una que satisfaga algún criterio, comenzando un nuevo proceso. Este proceso se detiene cuando se cumple alguna condición de término.

Existen diversas maneras de clasificar y definir las técnicas metaheurísticas. Dependiendo de las características que se seleccionen se pueden obtener distintas clases: inspiradas en la naturaleza o no inspiradas en esta, basadas en memoria o sin memoria, con función objetivo estática o dinámica. También se pueden clasificar en si la técnica en cada paso utiliza un único punto del espacio de búsqueda o trabaja sobre un conjunto o población. Según esto, se clasifican en basados en trayectoria o basados en población. En esta investigación se ocuparán las basadas en población, ya que el algoritmo de aprendizaje pertenece a esta clasificación.

Las metaheurísticas basadas en población trabajan con un conjunto de individuos, también llamada matriz de patrones, que representan otras tantas soluciones. Su eficiencia y resultado depende fundamentalmente de la manera en que se manipula la población en cada iteración. Cada uno de estos patrones es un agente individual, en la optimización por enjambre de partículas (PSO), la matriz de patrones se conoce como enjambre y cada patrón corresponde a una partícula. Del mismo modo, en Cuckoo Search (CS), cada agente representa un nido y la matriz de patrones es un conjunto de nidos.

4.2.1. Algoritmo Cuckoo Search

Es un algoritmo de optimización metaheurístico desarrollado por Yang y Deb en 2009. Ha sido aplicado para resolver problemas de optimización de diseño ingenieril, incluidos el diseño de muelles y vigas. Según sus autores, los resultados obtenidos por Cuckoo Search (CS) son mejores que los resultados obtenidos por PSO[10].

Se basa en el comportamiento de reproducción agresiva de algunas especies de la familia de los cucúlidos. Cada huevo en un nido representa una solución, y un huevo de cuco representa una nueva solución. El objetivo es emplear las nuevas y potenciales mejores soluciones para reemplazar las soluciones no tan óptimas en los nidos. En la forma más simple, cada nido tiene un huevo. El algoritmo se puede extender a casos más complicados en los que cada nido tiene varios huevos que representan un conjunto de soluciones [9][10]. Cuckoo Search se basa en tres reglas idealizadas:

- Cada cuco pone un huevo a la vez, y lo coloca en un nido elegido al azar.
- Los mejores nidos con alta calidad de los huevos (soluciones) se trasladarán a las siguientes generaciones.
- El número de nidos huésped disponibles es fijo, el pájaro huésped puede descubrir un huevo con una probabilidad $P_a \in [0, 1]$. En este caso, puede lanzar el huevo o abandonar el nido para construir uno nuevo otra ubicación.

Para simplificar el algoritmo, el último supuesto se aproxima a una fracción P_a de los n nidos siendo reemplazados por nuevos nidos, con nuevas soluciones al azar. Para un problema de maximización, el fitness de una solución simplemente puede ser proporcional a la función objetivo o definirse de manera similar a la de los algoritmos genéticos [9].

Cuando se generan nuevas soluciones $x_i^{(t+1)}$ para el cuckoo i , se realiza un vuelo de Lévy

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus Lévy(\lambda) \quad (9)$$

donde $\alpha > 0$ es el tamaño del paso, y que debe estar relacionado con la magnitud del problema de interés. La elección de α puede tener un impacto significativo en la capacidad de búsqueda, en la mayoría de los casos podemos usar $\alpha = 1,5$ por recomendación del autor. El

producto \oplus significa multiplicación elemento a elemento. Los vuelos de Lévy esencialmente proporcionan un camino aleatorio, mientras sus pasos aleatorios se han extraído de una distribución Lévy para grandes pasos

$$Lévy \sim u = t^{-\lambda}, \quad (1 < \lambda \leq 3) \quad (10)$$

que contiene una varianza y una media infinita. Aquí los consecutivos saltos o pasos de un pájaro cucú, forman esencialmente un proceso de caminata aleatoria que obedece a una distribución de ley de potencias con cola pesada [9][10].

Cuckoo Search es un algoritmo relativamente sencillo con sólo dos parámetros ajustables, P_a y el tamaño de la población n . Según Yang[10] la velocidad de convergencia del algoritmo no se ve muy afectada por el valor de P_a y se sugiere usar un valor de 0,25. Un estudio comparativo de CS con un pequeño conjunto de funciones de prueba, diseñadas por Yang y Deb, contra PSO y GA mostró que el comportamiento de convergencia de CS, en términos de media y desviación estándar, de las funciones de evaluación es superior a GA y PSO. El uso de vuelos de Lévy en CS, sugiere que es posible encontrar todos los óptimos en un espacio de búsqueda para problemas de funciones multimodales. También ha sido aplicado con éxito para resolver problemas con múltiples mínimos globales, tanto para entornos sin ruido y con ruido aditivo blanco gaussiano (AWGN)[11].

En la figura 5.1 se muestra el pseudo código de Cuckoo Search.

```

Objective function  $f(\mathbf{x})$ ,  $\mathbf{x} = (x_1, \dots, x_d)^T$ ;
Initial a population of  $n$  host nests  $\mathbf{x}_i$  ( $i = 1, 2, \dots, n$ );
while ( $t < \text{MaxGeneration}$ ) or (stop criterion);
    Get a cuckoo (say  $i$ ) randomly by Lévy flights;
    Evaluate its quality/fitness  $F_i$ ;
    Choose a nest among  $n$  (say  $j$ ) randomly;
    if ( $F_i > F_j$ ),
        Replace  $j$  by the new solution;
    end
    Abandon a fraction ( $p_a$ ) of worse nests
        [and build new ones at new locations via Lévy flights];
    Keep the best solutions (or nests with quality solutions);
    Rank the solutions and find the current best;
end while
Postprocess results and visualisation;

```

Figura 4.1 Pseudocódigo Cuckoo Search

4.2.2. Algoritmo Modified Cuckoo Search

Modified Cuckoo Search (MCS), es un potente algoritmo de optimización de gradiente que fue presentado por S. Walton en 2011[15]. El objetivo de esta herramienta, consiste en que el proceso de acoplamiento de MCS para una determinada aplicación sea más simple. Cuckoo Search siempre halla el óptimo, pero como la búsqueda se basa enteramente en las caminatas aleatorias, una convergencia rápida no puede ser garantizada.

MCS es un algoritmo de optimización rápido, en términos de número de evaluaciones de la función objetivo requeridos para alcanzar una solución. Es capaz de manejar funciones no suavizadas, multimodales y de muchas dimensiones.

Se realizan dos modificaciones en el método con el objetivo de aumentar la tasa de convergencia, haciendo que el método sea más práctico para una gama más amplia de aplicaciones, pero sin perder las características del método original[16].

La primera modificación se hace con el tamaño del paso del vuelo de Lévy. En CS, α es constante y el valor es igual a 1,5. En MCS, el valor de α disminuye a medida que el número de generaciones aumenta. Esto se hace por las mismas razones que la constante de inercia se reduce en PSO, es decir, fomenta la búsqueda más localizado cuando los huevos están más cerca de la solución. Se elige un valor inicial del tamaño del paso del vuelo de Lévy $A = 1$, y en cada generación, un nuevo paso se vuelve a calcular utilizando $\alpha = A/\sqrt{G}$, donde G es el número de la generación. Esta búsqueda de exploración sólo se realiza en la fracción de nidos que serán abandonados.

La segunda modificación es agregar el intercambio de información entre los huevos, con el propósito de acelerar la convergencia a un mínimo. En CS, no hay intercambio de información entre éstos, debido a que las búsquedas se realizan de manera independiente. En MCS, una fracción de los huevos con el mejor fitness, se pone en un grupo con las mejores soluciones. Para cada solución de este grupo, se escoge al azar una segunda, y un nuevo huevo se genera en la línea que conecta estas dos soluciones. La distancia a lo largo de esta línea, en la que el nuevo huevo se encuentra, se calcula aplicando la inversa de la razón áurea, $\varphi = (1 + \sqrt{5})/2$, de tal manera que está más cerca de la solución con el mejor fitness. En el caso de que ambos huevos tengan el mismo fitness, el nuevo huevo se genera en el punto medio.

Mientras se desarrolló el algoritmo, se utilizó una fracción aleatoria en lugar de la proporción áurea, encontrándose que ésta mostró mayor rendimiento que una fracción aleatoria. Existe una posibilidad de que en este paso se haya seleccionado el mismo huevo dos veces. En este caso, una búsqueda local a través de vuelos de Lévy se realiza desde un nido elegido aleatoriamente con tamaño de paso $\alpha = A/G^2$. Hay dos parámetros que son ajustados en MCS, la fracción de los nidos a ser abandonados y la fracción de nidos en el grupo con las mejores soluciones. A través de pruebas en la investigación con los problemas de referencia, se encontró que el mejor valor para la fracción de nidos a ser abandonado es igual 0,75 y la fracción de nidos colocados en el grupo de mejores soluciones igual a 0,25.

5. Estrategias de pronóstico

El pronóstico para un sólo valor en adelante para una serie de tiempo no es de utilidad, ya que lo que se desea saber es el comportamiento de dicha serie en un horizonte mayor. Para esto, es necesario el pronóstico de varias etapas futuras. La predicción en series de tiempo utiliza los valores anteriores como base para obtener los valores futuros[17]. Normalmente, esta tarea enfrenta distintas incertidumbres provenientes de varias fuentes, como por ejemplo, la acumulación de errores y la falta de información, que hacen que la predicción sea más difícil[18]. Existen varias estrategias que se utilizan para la predicción en etapas futuras, siendo las estrategias recursiva y DirRec las que se implementarán.

5.1. Estrategia recursiva o iterativa

La estrategia recursiva, también llamada iterativa, utiliza los valores pronosticados como datos conocidos para predecir los próximos[8][21]. Consiste en un único modelo que está capacitado para pronosticar el siguiente paso, es decir, al predecir N pasos por delante, lo primero que pronosticamos es el primer paso. Posteriormente, se utiliza el valor obtenido como parte de las variables de entrada para pronosticar el subsiguiente paso. Continuamos de esta manera hasta que hemos pronosticado todo el horizonte deseado.

Dependiendo del ruido presente en la serie de tiempo y del horizonte a pronosticar, la estrategia recursiva puede sufrir de bajo rendimiento en varios pasos por delante. La razón de la inexactitud potencial, es que la estrategia recursiva es sensible a la acumulación de errores con el horizonte de predicción. Los errores presentes en las predicciones intermedias se propagarán hacia adelante, ya que estos valores se utilizan para determinar los valores siguientes.

Esta estrategia presenta una limitación debido a que, si el horizonte a pronosticar excede el número de datos de entrada, en algún momento todas las entradas serán valores pronosticados en lugar de valores reales.

$$\hat{y}_{t+1} = f(y_t, y_{t-1}, \dots, y_{t-r+1}) \quad (11)$$

$$\hat{y}_{t+2} = f(\hat{y}_{t+1}, y_t, \dots, y_{t-r+2}) \quad (12)$$

$$\hat{y}_{t+r+1} = f(\hat{y}_{t+r}, \hat{y}_{t+r-1}, \dots, \hat{y}_{t+1}) \quad (13)$$

5.2. Estrategia DirRec

La estrategia DirRec, combina las arquitecturas y los principios subyacentes a las estrategias directa y recursivas[19][22]. DirRec calcula los pronósticos con diferentes modelos para cada horizonte, como la estrategia directa. En cada paso de tiempo, se amplía el conjunto de entradas mediante la adición de variables, correspondientes a los pronósticos de la etapa previa, como la estrategia iterativa. Sin embargo, hay que tener en cuenta que, a diferencia de las otras dos estrategias, el tamaño del conjunto de entrada no es el mismo para todos los horizontes. La estrategia DirRec aprende H modelos f de la serie de tiempo.

$$\hat{y}_{t+1} = f_1(y_t, y_{t-1}, \dots, y_{t-r+1}) \quad (14)$$

$$\hat{y}_{t+2} = f_2(\hat{y}_{t+1}, y_t, y_{t-1}, \dots, y_{t-r+1}) \quad (15)$$

$$\hat{y}_{t+p} = f_p(\hat{y}_{t+p-1}, \dots, \hat{y}_{t+1}, y_t, y_{t-1}, \dots, y_{t-r+1}) \quad (16)$$

6. Modelo propuesto

Para la presente investigación se propusieron tres modelos, diferenciándose por la estrategia de pronóstico y el algoritmo de entrenamiento. Estos modelos emplean redes neuronales artificiales del tipo perceptrón multicapa, con función de activación de base radial, componiéndose por tres capas. La interconexión entre estas capas es del tipo feedforward, la propagación de la señal es en un sólo sentido, hacia la siguiente capa y no permite la retroalimentación. El aprendizaje de los modelos es del tipo supervisado y pronosticarán siete días hacia adelante.

Se utilizaron dos estrategias de pronóstico, la estrategia recursiva y la estrategia DirRec. Para la primera estrategia se utilizaron los algoritmos de aprendizaje Cuckoo Search y Modified Cuckoo Search, comparando entre ellos su rendimiento. Para la segunda estrategia sólo se utilizó Cuckoo Search. Estos algoritmos se encargarán de encontrar los parámetros del polinomio que producirán la transformación no lineal, de la capa de entrada a la capa oculta, y los coeficientes de la transformación lineal de la capa oculta a la capa de salida, que entregarán como resultado el pronóstico de la rentabilidad del valor cuota.

El modelo recursivo CS cuenta con 5 nodos de entrada y 4 nodos ocultos. El modelo recursivo MCS con 3 y 2, respectivamente. El modelo DirRec CS, parte en 5 hasta los 12 nodos de entrada y 4 nodos ocultos. Todos ellos cuentan con un solo nodo en la capa de salida.

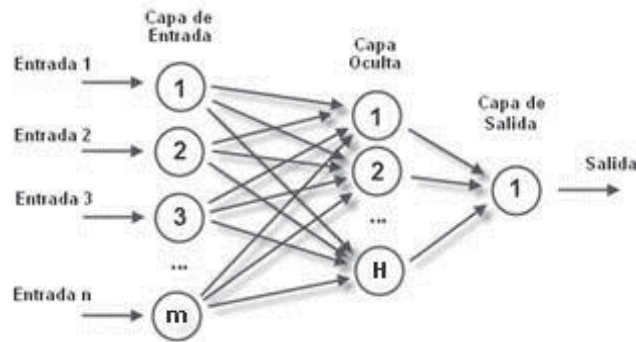


Figura 6.1 Modelo propuesto

6.1. Función de activación

Para el modelo propuesto se utiliza una función de activación de tipo radial, como lo es la función multicuadrática, que viene dada por la siguiente ecuación:

$$f(x) = \sqrt{1 + z^2} \quad (17)$$

Donde z es la distancia euclídea del vector de entrada con respecto del centro, dada por la ecuación:

$$z = \left\| \frac{x_i - t_k}{d_k} \right\| \quad (18)$$

Siendo x el vector de entrada, y las componentes t y d los vectores de centro y desviación de la función de activación, que se representan de la siguiente forma:

$$x_i = (x_{i,1}, x_{i,2}, \dots, x_{i,n}), i - \text{ésima muestra} \quad (19)$$

$$t_k = (t_{k,1}, t_{k,2}, \dots, t_{k,n}), d_k = (d_{k,1}, d_{k,2}, \dots, d_{k,n}), k - \text{ésimo nodo oculto} \quad (20)$$

6.2. Métricas

Las siguientes métricas son frecuentemente utilizadas para la evaluación de modelos y su comparación.

- **Raíz del Error Cuadrático Medio (RMSE)**

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (Y_i - \hat{Y}_i)^2} \quad (21)$$

Mide la magnitud del error. Donde Y es el valor observado, \hat{Y} es el valor estimado y N es el tamaño de la muestra.

- **Error Medio Absoluto Porcentual (MAPE)**

$$MAPE = \frac{\sum_{i=1}^N \left| \frac{Y_i - \hat{Y}_i}{Y} \right|}{N} \times 100 \quad (22)$$

Mide la precisión en porcentaje que busca evaluar el rendimiento de modelos. Donde Y es el valor observado, \hat{Y} es el valor estimado y N es el tamaño de la muestra.

- **Validación Cruzada Generalizada (GCV)**

$$GCV = \frac{MAPE}{\left(1 - \left(\frac{K}{N}\right)\right)^2} \quad (23)$$

Es una medida para evaluar el trade-off entre precisión y complejidad del modelo. Donde N es el tamaño de la muestra y K es el número de parámetros del modelo.

- **Sentido de la Tendencia**

Determina el porcentaje de aciertos en el sentido de la tendencia, si el valor observado va al alza, el valor pronosticado debiese seguir ese comportamiento.

- **Coficiente de Determinación (R^2)**

$$R^2 = 1 - \frac{\sum_{i=1}^N (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^N (Y_i - \bar{Y})^2} \quad (24)$$

Indica que tan ajustados están los resultados obtenidos a la línea de regresión de los resultados reales. Se espera que su valor esté cercano a 1. Donde Y es el valor observado e \hat{Y} es el valor estimado.

7. Experimentos

En esta sección, se describen los experimentos realizados para determinar el mejor modelo utilizando la estrategia iterativa, para posteriormente, comparar esta estrategia con la estrategia DirRec. El mejor modelo debe ser capaz de pronosticar la rentabilidad del fondo A de AFP Cuprum para varios días futuros, tomando como principal métrica para su elección el sentido de la curva.

7.1. Datos de entrada

Como datos de entrada, se utilizó el precio diario del valor cuota del fondo A. Estos datos fueron extraídos desde el sitio web de la superintendencia de pensiones. Para la etapa de entrenamiento, se utilizaron 3701 datos, correspondientes a los precios del fondo A desde la creación de los multifondos. Para la validación, se utilizaron 925 datos, que corresponde al 20% del total de la muestra, considerando hasta el 31 de mayo de 2015. Previo a ingresar a la red los datos son normalizados entre 0 y 1, manteniendo las relaciones existentes entre los patrones de entrada original. Para conseguir esto, se divide por el valor mayor toda la muestra. Posterior a este procesamiento, los datos son desfasados según el número de días deseados, esto consiste en convertir el vector de datos a una versión desplazada, con el objetivo de obtener una matriz de regresión de la variable explicativa.

Primero la red se entrena con los precios, posteriormente se calcula la rentabilidad con la fórmula $Rentabilidad_t = \frac{Precio_t - Precio_{t-1}}{Precio_{t-1}}$, y en base a esto, se calculan las métricas en validación. El siguiente paso es el pronóstico de los precios de los 7 días siguientes, y sobre estos, se calcula la rentabilidad y se obtienen las métricas.

7.2. Resultados

Para determinar el mejor modelo, se consideró el sentido de la predicción, si los valores estimados siguen la tendencia de alza o baja del valor observado. Esto se debe a que es de mayor importancia saber su dirección que el valor exacto, ya que es este comportamiento el que indica una rentabilidad positiva o negativa y que sirve para la decisión de cambio de fondo.

7.2.1. Estrategia recursiva

Para la estrategia recursiva, se presentaron dos modelos que se comparan entre sí, ambos se entrenaron y testearon 30 veces. Para cada uno de ellos se analizan tres gráficos, correspondientes a dispersión, valor estimado v/s observado y error relativo. El primer modelo utilizó como algoritmo de entrenamiento Cuckoo Search. Los parámetros propios del algoritmo son 100 iteraciones, 60 nidos y $P_a = 0,3$.

Tabla 7.1 Resultados obtenidos testing Recursivo Cuckoo Search

	Precio			Rentabilidad			Iteración
	R2	MAPE	Sentido	R2	MAPE	Sentido	
Mínimo MSE	0.997667	0.770735	0.581692	0.041962	Inf	0.687138	14
Máximo MSE	0.988733	9.570.140	0.596756	0.007455	Inf	0.705678	2
Media	0.984780	3.009.367	0.632097	0.008918	Inf	0.680494	-
Desviación estándar	0.011636	2.150.252	0.052544	0.012070	Inf	0.032567	-

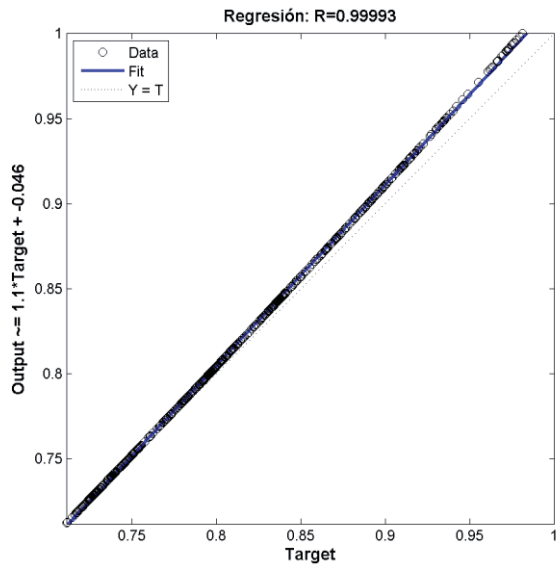


Figura 7.1 Gráfico de determinación

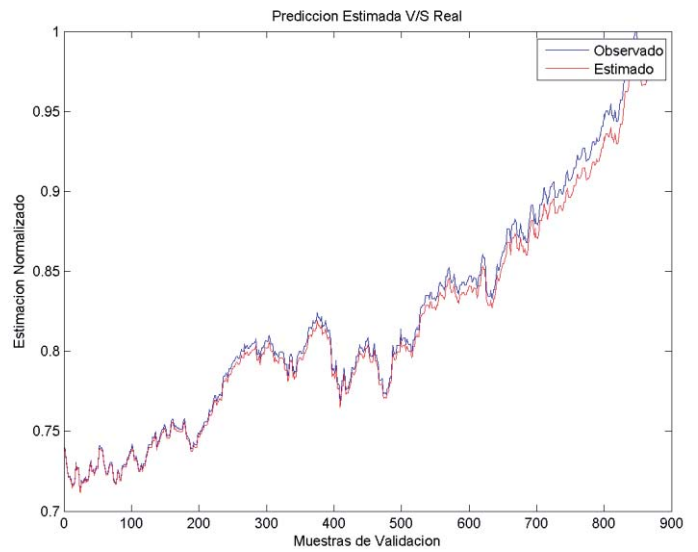


Figura 7.2 Gráfico de valor observado v/s valor estimado

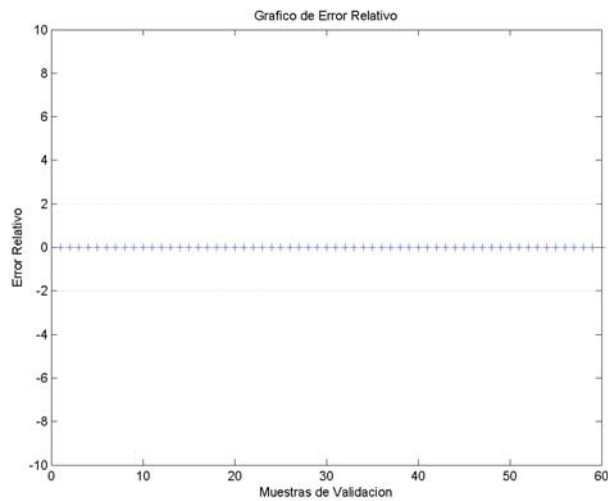


Figura 7.3 Gráfico de error relativo

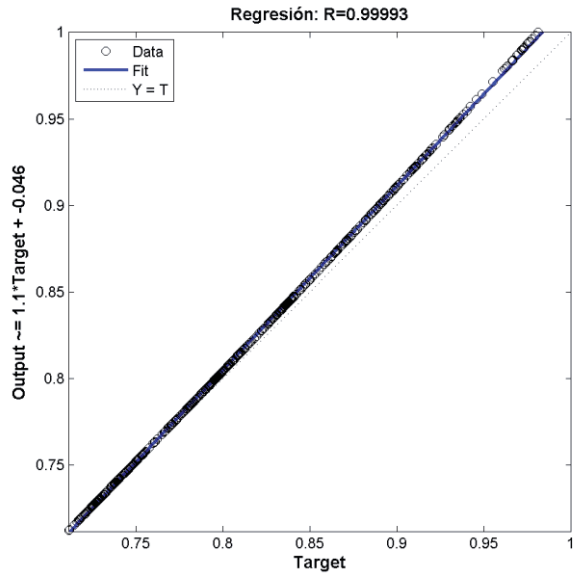


Figura 7.4 Gráfico de determinación

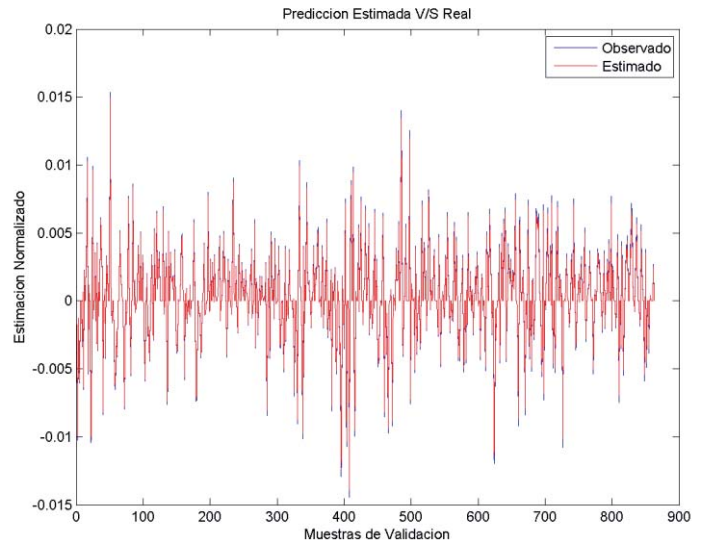


Figura 7.5 Gráfico de valor observado v/s valor estimado

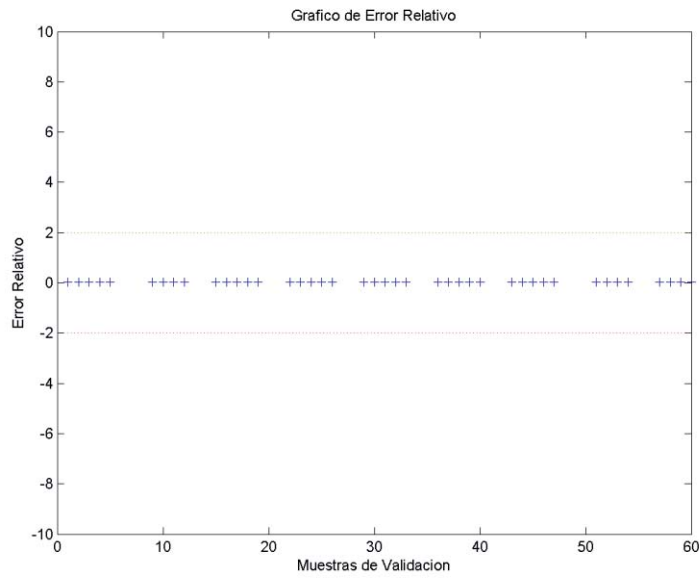


Figura 7.6 Gráfico de error relativo

Tabla 7.2 Resultados obtenidos extrapolación Recursivo Cuckoo Search

	Rentabilidad		
	R2	MAPE	Sentido
Mínimo	0.351407	Inf	0.733435

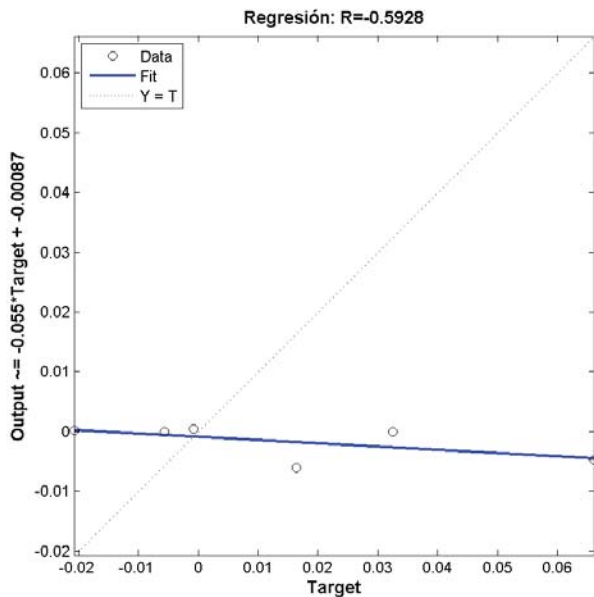


Figura 7.7 Gráfico de determinación

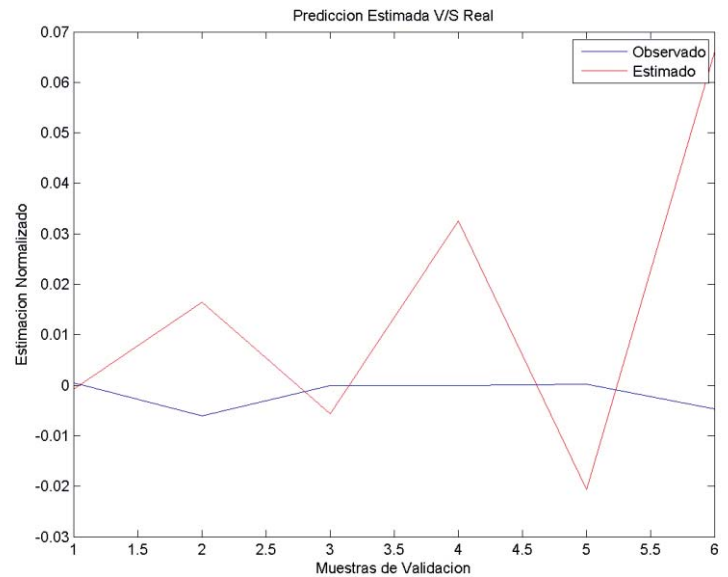


Figura 7.8 Gráfico de valor observado v/s valor estimado

El segundo modelo fue entrenado con Modified Cuckoo Search. Los parámetros del algoritmo son $P_a = 0,7$, 300 iteraciones para entrenamiento, 30 nidos, distancia máxima que puede viajar un cuckoo = 200, factor que reduce el paso en cada iteración = 0,3, número de nidos eliminados por cada iteración = 1. Se analizan tres gráficos correspondientes a dispersión, valor estimado v/s valor observado y error relativo.

Tabla 7.3 Resultados obtenidos testing Recursivo Modified Cuckoo Search

	Precio			Rentabilidad			Iteración
	R2	MAPE	Sentido	R2	MAPE	Sentido	
Mínimo MSE	0.998261	0.294117	0.571098	0.041749	Inf	0.708671	23
Máximo MSE	0.995473	3.269.364	0.722543	0.002696	Inf	0.717919	2
Media	0.997119	0.839248	0.666358	0.014532	Inf	0.716339	-
Desviación estándar	0.000831	0.702814	0.049781	0.015601	Inf	0.013576	-

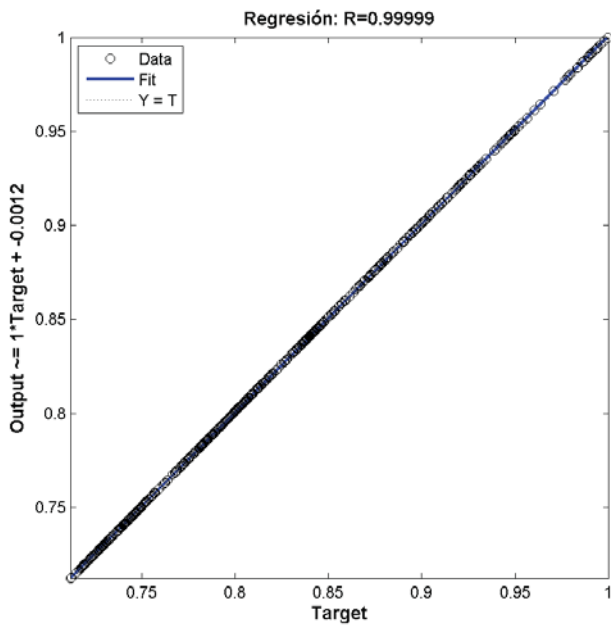


Figura 7.9 Gráfico de determinación

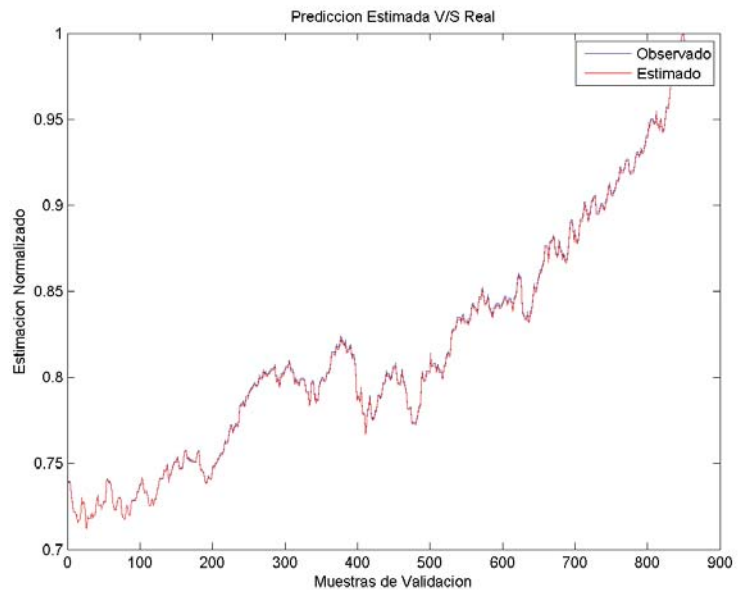


Figura 7.10 Gráfico de valor observado v/s valor estimado

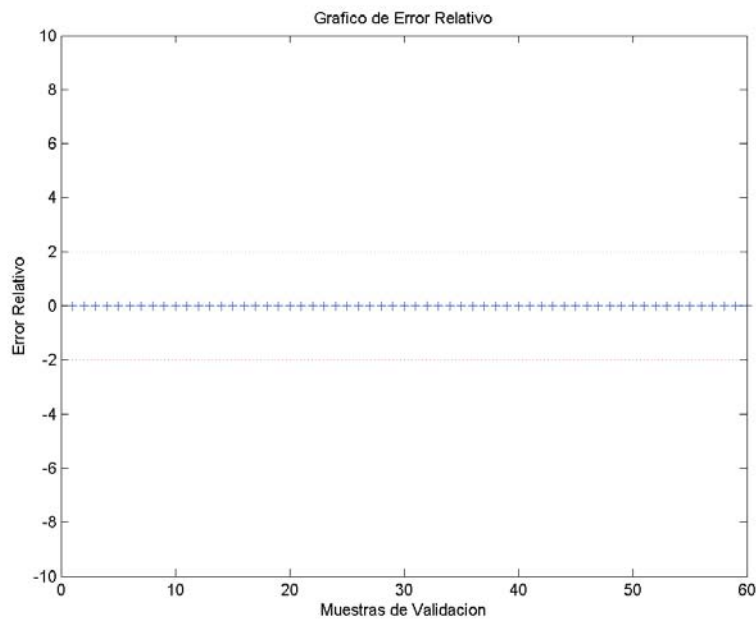


Figura 7.11 Gráfico de error relativo

Tabla 7.4 Resultados obtenidos extrapolación Recursivo Modified Cuckoo Search

	Rentabilidad		
	R2	MAPE	Sentido
Mínimo	0.216156	Inf	0.666667

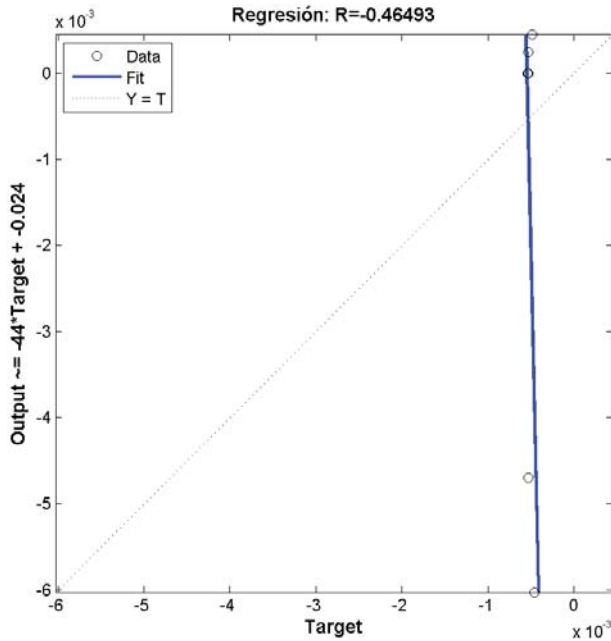


Figura 7.12 Gráfico de determinación

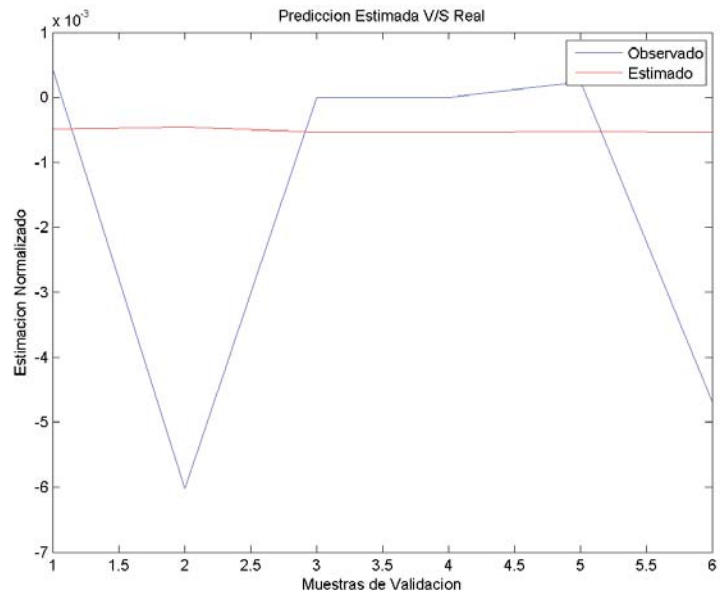


Figura 7.13 Gráfico de valor observado v/s valor estimado

De los resultados obtenidos, el mejor modelo fue el de Cuckoo Search, obteniendo en sentido en extrapolación 73,34% y R^2 de 35,14%. El modelo Modified Cuckoo Search obtuvo en sentido 66,66% y R^2 de 21,61% en extrapolación. A pesar de los resultados obtenidos por el segundo modelo, éste obtuvo mejor desempeño en testing, con un sentido de 71,79% y R^2 0.0026.

7.2.2. Estrategia DirRec

Para este tipo de estrategia es necesario determinar un pronosticador para cada día del horizonte, lo que hace necesario encontrar la cantidad de nodos ocultos para cada uno de ellos, debido a que la complejidad es distinta. Para este modelo se utilizó el algoritmo Cuckoo Search, debido a que fue el que obtuvo mejor resultado usando la estrategia recursiva. Los parámetros propios del algoritmo son 100 iteraciones, 60 nidos y $P_a = 0,3$.

Tabla 7.5 Resultados obtenidos testing DirRec Cuckoo Search

	Precio			Rentabilidad		
	R2	MAPE	Sentido	R2	MAPE	Sentido
Día 1	0,9869185	3,1231220	0,6570105	0,0019870	-	0,7068365
Día 2	0,9944455	1,7287360	0,6438515	0,0110220	-	0,6757545
Día 3	0,9512985	5,2221320	0,6672475	0,0119155	-	0,6881535
Día 4	0,9898095	7,6402050	0,6593020	0,0020650	-	0,6895345
Día 5	0,9601135	1,5423480	0,6722935	0,0093980	-	0,6839350
Día 6	0,9598000	2,9276570	0,6620045	0,0046955	-	0,6486015
Día 7	0,9766070	3,7863490	0,5980160	0,0040245	-	0,6931160
Promedio	0,9741418	3,7100784	0,6513894	0,0064439	-	0,6837045

Tabla 7.6 Resultados obtenidos extrapolación DirRec Cuckoo Search

	Rentabilidad		
	R2	MAPE	Sentido
Mínimo	0.003111	Inf	0.666667

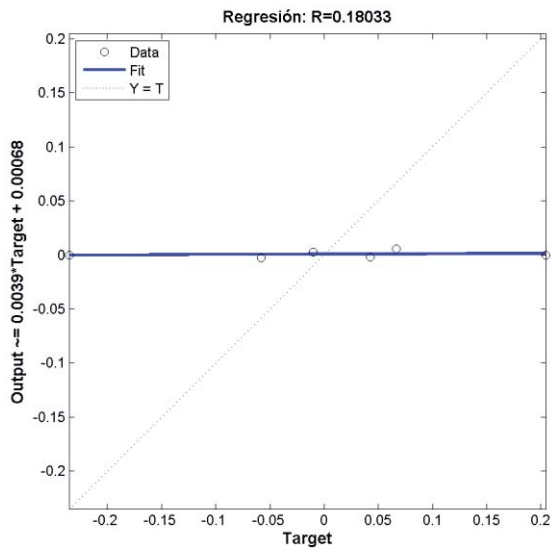


Figura 7.14 Gráfico de determinación

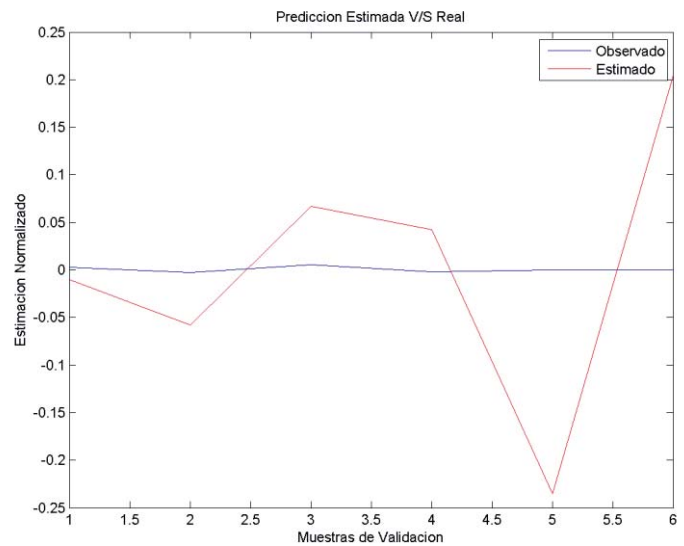


Figura 7.15 Gráfico de valor observado v/s valor estimado

7.3. Discusión de resultados

En base a los resultados obtenidos, se pueden percibir las ventajas y desventajas de cada modelo. Para la estrategia recursiva el mejor modelo fue Cuckoo Search, La dificultad de este modelo es debido a que emplea datos estimados para el pronóstico de los siguientes días, acarreando el error para la siguiente etapa. A pesar de esto, con esta estrategia se logró el mejor modelo de los tres propuestos.

Para el modelo DirRec, que contiene las ventajas y desventajas de las estrategias recursiva y directa, obtuvo el mismo resultado que el modelo Recursivo MCS, no logrando superar al modelo Recursivo CS. Esta estrategia tiene la ventaja de que existe un pronosticador distinto para cada día en el horizonte, pero tiene un alto costo en tiempo para encontrar los parámetros adecuados para cada uno de ellos.

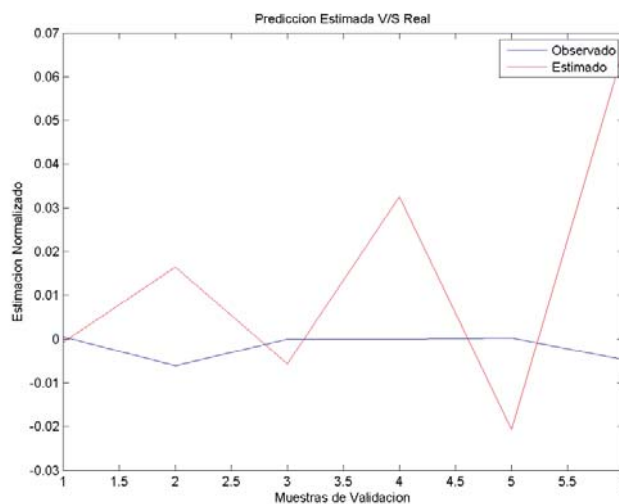


Figura 7.16 Gráfico de valor observado v/s estimado mejor modelo

En la gráfica se puede apreciar que, aunque los valores pronosticados no están cerca de los observados, estos siguen la misma tendencia en el sentido de la curva, siendo el objetivo de esta investigación.

Tabla 7.7 Comparación resultados obtenidos por los modelos propuestos

Modelos	Rentabilidad extrapolación	
	R2	Sentido
Recursivo Cuckoo Search	0.351407	0.733435
Recursivo Modified Cuckoo Search	0.216156	0.666667
DirRec Cuckoo Search	0.003111	0.666667

En base a los resultados expuestos para los tres modelos en la tabla 7.7, se concluye que el mejor modelo de pronóstico para la problemática de esta investigación, corresponde al pronóstico recursivo utilizando redes neuronales de base radial, con función de activación multicuadrática y algoritmo de aprendizaje Cuckoo Search.

8. Conclusión

Esta investigación pretende ser una ayuda para los afiliados al sistema de pensiones, entregando un pronóstico de la rentabilidad a futuro para saber cuándo cambiarse de fondo, con el objetivo de evitar pérdidas y aumentar sus pensiones.

En el desarrollo del presente proyecto de investigación, se demostró a través de estudios realizados por distintos autores, que los pronósticos mediante redes neuronales obtienen óptimos resultados en problemáticas referentes a series de tiempo financieras, lo cual implica que las redes son aptas para abordar y predecir la rentabilidad del fondo A de las AFP.

La problemática se abordó utilizando redes neuronales con función de activación de base radial, prediciendo el precio y a partir de este calculando la rentabilidad, obteniendo sobre este último las métricas para comparar los modelos. Como variable explicativa se empleó el precio diario del valor cuota del fondo A de AFP Cuprum, desde la creación de los multifondos, octubre de 2002, hasta el 31 de mayo de 2015.

También se logró obtener una comprensión del funcionamiento y modelado de las redes neuronales, de los algoritmos de aprendizaje Cuckoo Search y Modified Cuckoo Search, además del entendimiento de distintas técnicas de pronóstico y como estas se pueden integrar con las redes neuronales.

Cuckoo Search es un algoritmo metaheurístico reciente, el cual ha demostrado obtener óptimos resultados incluso sobre los más conocidos y populares algoritmos de aprendizaje, como es en el caso de PSO para problemas de pronóstico de series de tiempo. Modified Cuckoo Search en una modificación del algoritmo original realizada por S. Walton, en el que propone dos modificaciones para mejorar la rapidez de la convergencia. Realizar la etapa de entrenamiento de estas dos metaheurísticas es una tarea costosa, debido a que no existe una metodología exacta para determinar los parámetros que se adecuen mejor a la complejidad del problema.

Se presentaron tres modelos, en dos de ellos se utilizó la estrategia iterativa para realizar los pronósticos, utilizando los algoritmos de aprendizaje CS y MCS. Para el tercer modelo, se utilizó la estrategia DirRec con el algoritmo CS. Para determinar la arquitectura de cada modelo y los parámetros a usar en cada algoritmo, se recurrió a la técnica ensayo y error, debido a que para estas técnicas evolutivas no existe una metodología que sirva de guía para fijar dichos parámetros, siendo este proceso muy costoso en tiempo.

Para la primera estrategia de pronóstico, el modelo que empleó Cuckoo Search obtuvo un sentido de curva en extrapolación de 73,34%, en contraste con el otro modelo propuesto que utilizó Modified Cuckoo Search, llegando a 66,66% de sentido de la curva. A pesar de estos resultados, en testing MCS obtuvo un promedio de 71,63% de sentido en rentabilidad y un MAPE promedio de 0,8392, con un mínimo de 0,2941 para el precio. Para estas mismas métricas en testing, CS consiguió 68,04%, 3,009 y 0,7707, respectivamente.

Para la estrategia DirRec, se presentó un solo modelo tomando como algoritmo de entrenamiento al mejor de la estrategia recursiva. Éste modelo obtuvo en extrapolación un 66,66% en sentido en rentabilidad, no superando al modelo de la primera estrategia. Las otras métricas obtenidas en testing, son un MAPE promedio 3,71 para el precio y 68% de sentido. DirRec combina la estrategia recursiva y directa, creando un pronosticador por cada día en el horizonte de pronóstico, esta tarea es costosa en tiempo, pero asegura un mejor rendimiento. El

modelo propuesto finalmente es la red neuronal de base radial, con función de activación multicuadrática y algoritmo de entrenamiento Cuckoo Search.

Como trabajo futuro, se propone utilizar otras estrategias de pronóstico y algún procesamiento de los datos. Es probable que mediante algún tipo de suavizado o transformación matemática, se puedan mejorar los resultados obtenidos con las estrategias utilizadas, también añadiendo el uso de otras estrategias de pronóstico que puedan aumentar el horizonte a pronosticar.

Finalmente, se espera que esta investigación se convierta en una herramienta para los cotizantes del sistema previsional. A pesar de que para que sea una real ayuda se necesitan pronósticos más precisos y horizontes de pronóstico mayor, resulta útil como complemento a otros modelos de pronóstico o técnicas de trading, en pos de obtener una mayor pensión para su jubilación.

9. Referencias

- [1] Broomhead D. S., Lowe D., "Radial basis functions, multi-variable functional interpolation and adaptative networks", Royal Signals and Radar Establishment Memorandum 4148, 1988.
- [2] Park J., Sandberg I. W., "Universal approximation using radial-basis-function networks", *Neural computation* 3, 246-257, 1991.
- [3] Foix, C., Weber, R., "Pronóstico del precio del cobre mediante Redes Neuronales", *Revista Ingeniería de Sistemas*, 21:63-86, 2007.
- [4] Taherian H., Nazer I., Razavi E., Goldani S. R., Farshad M., Aghaebrahimi M. R., "Application of an Improved Neural Network Using Cuckoo Search Algorithm in Short-Term Electricity Price Forecasting under Competitive Power Markets", Vol. 1, No. 2, Summer & Fall, 2013.
- [5] Nekoukar V., Hamidi M., "A local linear radial basis function neural network for financial time-series forecasting", *Appl Intell*, 33: 352–356, 2010.
- [6] Deepak S., "Electricity price forecasting using artificial neural networks", *Electrical Power and Energy Systems* 33 (2011) 550–555, 2010.
- [7] Wang J., Wang J., Zhang Z., Guo S., "Forecasting stock indices with back propagation neural network", *Expert Systems with Applications* 38 14346–14355, 2011.
- [8] Taieb, S., Bontempi, G., Atiya, A., Sorjamaa, A., "A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition" *Expert systems with applications*, 39:7067-7083, 2012.
- [9] Valian E., Mohanna S., Tavakoli S., "Improved Cuckoo Search Algorithm For Feedforward Neural Network Training", *International Journal of Artificial Intelligence & Applications (IJAIA)*, Vol.2, No.3, 2011.
- [10] Yang X., Deb S., "Engineering Optimisation by Cuckoo Search", *Int. J. Mathematical Modelling and Numerical Optimisation*, Vol. 1, No. 4, 330–343, 2010.
- [11] Jamil M., Zepernick H., "Lévy flights and global optimization", *Swarm Intelligence and Bio-Inspired Computation* 49-72, 2013.
- [12] Pao H., "Forecasting electricity market pricing using artificial neural networks", *Energy Conversion and Management* 48 907–912, 2007.
- [13] Gómez F., Fernández M., López M, Alonso M, "Aprendizaje con redes neuronales artificiales", *Revista de la Facultad de Educación de Albacete* N° 9, 169-180, 1994.
- [14] Li G., Xu S., Li Z., "Short-Term Price Forecasting For Agro-products Using Artificial Neural Networks", *Agriculture and Agricultural Science Procedia* 1 278–287, 2010.
- [15] Walton S., "Gradient Free Optimisation in Selected Engineering Applications", Swansea University, 2013.
- [16] Walton S., Hassan O., Morgan K., Brown M., "Modified cuckoo search: A new gradient free optimisation algorithm", *Chaos, Solitons & Fractals* 44 710–718, 2011.
- [17] Ji Y., Hao J., Reyhani N., Lendasse A., "Direct and Recursive Prediction of Time Series Using Mutual Information Selection", *Springer Berlin Heidelberg* 3512 1010-1017, 2005.

- [18] Sorjamaa, A., Hao, J., Reyhani, N., Ji, Y., Lendasse, A. "Methodology for longterm prediction of time series", *Neurocomputing*, 70(16–18):2861–2869, 2007.
- [19] Bontempi G., Taieb S., Borge Y., "Machine Learning Strategies for Time Series Forecasting", Springer Berlin Heidelberg 138 62-77, 2013.
- [20] Oliveira F., Nobre C., Zárata L., "Applying Artificial Neural Networks to prediction of stock price and improvement of the directional prediction index – Case study of PETR4, Petrobras, Brazil" *Expert Systems with Applications*, 2013.
- [21] Grigorievskiy A., Miche Y., Ventelä A., Séverin E., Lendasse A., "Long-term Time Series Prediction using OP-ELM", *Neural Networks* 51 50–56, 2014.
- [22] Sorjamaa A., Lendasse A., "Time Series Prediction using DirRec Strategy", *European Symposium on Artificial Neural Networks* 26-28, 2006.