

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

**REDES NEURONALES ARTIFICIALES DE BASE
RADIAL CON ALGORITMO DE APRENDIZAJE PSO
PARA EL PRONÓSTICO DEL VALOR CUOTA DEL
FONDO A DE LA AFP CUPRUM.**

FELIPE ALEXIS GARRIDO VARGAS

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

DICIEMBRE 2014

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

**REDES NEURONALES ARTIFICIALES DE BASE
RADIAL CON ALGORITMO DE APRENDIZAJE PSO
PARA EL PRONÓSTICO DEL VALOR CUOTA DEL
FONDO A DE LA AFP CUPRUM.**

FELIPE ALEXIS GARRIDO VARGAS

Profesor Guía: **Nibaldo Rodríguez Agurto**

Profesor Co-referente: **José Miguel Rubio León**

Carrera: **Ingeniería Civil Informática**

DICIEMBRE 2014

Dedicatoria

Dedicado a mis padres por su gran amor y apoyo incondicional durante esta importante etapa, a mis hermanos y familia por su confianza depositada, y a mi amada por darme fuerzas y ánimos para seguir adelante.

Índice

Índice	ii
Lista de Figuras	iv
Lista de Tablas.....	v
Resumen	vi
1. Introducción	1
2. Objetivos	4
2.1. Objetivo General	4
2.2. Objetivos Específicos.....	4
3. Pronóstico de series de tiempo financieras.	5
3.1. Estrategias de pronóstico para varias etapas futuras	6
3.1.1. Estrategia iterativa o recursiva.....	7
3.1.2. Estrategia directa	7
3.1.3. Estrategia múltiples entradas múltiples salidas	7
4. Redes Neuronales Artificiales.....	9
4.1. Redes Neuronales Para Pronóstico.....	10
4.2. Redes Neuronales De Base Radial	12
4.3. Arquitectura de la RNBR	13
5. Aprendizaje de la red	15
5.1. Aprendizaje supervisado	15
5.2. Optimización por enjambres de partículas PSO.....	15
5.2.1. Algoritmo PSO Continuo	16
5.2.2. PSO Adaptativo	17
6. Modelos Propuestos	19
6.1. Modelo de pronóstico Iterativo utilizando redes neuronales de base radial	19
6.2. Modelo de pronóstico Directo utilizando redes neuronales de base radial	20
6.3. Modelo de pronóstico múltiples entradas y múltiples salidas utilizando redes neuronales de base radial	21
7. Experimentos.....	22
7.1. Datos de entrada.....	23
7.2. Resultados	23
7.2.1. Pronóstico Iterativo.....	25

7.2.2. Pronóstico Directo	26
7.2.3. Pronóstico Múltiples Entradas y múltiples Salidas	29
7.3. Extrapolación de los resultados.....	32
7.4. Discusión de Resultados.....	33
8. Conclusiones	35
9. Referencias	37

Lista de Figuras

Figura 1.1 Rentabilidad anual de los fondos A y E.	2
Figura 4.1 Neurona Biológica.	9
Figura 4.2 Neurona Artificial.	10
Figura 4.3. Red Neuronal Base Radial con Función Gaussiana	13
Figura 5.1 aprendizaje supervisado.	15
Figura 6.1 Modelo de pronóstico iterativo.	20
Figura 6.2 Modelo de pronóstico iterativo.	21
Figura 6.3 Modelo de pronóstico múltiples entradas y múltiples salidas	21
Figura 7.1 Decaimiento del error a través de las iteraciones.	25
Figura 7.2 Datos estimados V/S observados.	26
Figura 7.3 Correlación entre los datos observado y estimado.	26
Figura 7.4 Decaimiento del error a través de las iteraciones para cada día estimado.	27
Figura 7.5 Evolución del MAPE a medida que se avanza en el horizonte a estimar.	28
Figura 7.8 Decaimiento del error a través de las iteraciones para cada día estimado.	30
Figura 7.9 Evolución del MAPE a medida que se avanza en el horizonte a estimar.	31
Figura 7.10 Datos estimados V/S observados para cada día estimado	32
Figura 7.11 Resultado extrapolación pronóstico iterativo	33
Figura 7.12 Mejor resultado obtenido por el pronóstico directo.	34

Lista de Tablas

Tabla 7.1. Resultados obtenidos por las distintas funciones de activación de base radial.....	24
Tabla 7.2. Resultados obtenidos usando distintas cantidades de nodos ocultos.	25
Tabla 7.3 Resultados obtenidos usando distintas cantidades de nodos ocultos para los distintos días.....	27
Tabla 7.4 Resultados obtenidos usando distintas cantidades de nodos ocultos para los distintos días.....	30
Tabla 7.5 Resultados promedios obtenidos por cada modelo	32
Tabla 7.6 Resultados obtenidos por el pronóstico directo en la etapa de extrapolación.	32

Resumen

En la presente memoria de título se propone un modelo de pronóstico para la estimación del valor cuota del fondo A, el cual está basado en Redes Neuronales de Base Radial (RBFN), y un algoritmo de aprendizaje fundamentado en la optimización por enjambres de partículas (PSO), el cual está inspirado en el comportamiento social de individuos dentro de un enjambre en la naturaleza. Para ello, se realiza un estudio acabado del pronóstico de series de tiempo financieras, además de la tecnología empleada (RBFN y PSO), las cuales han otorgado muy buenos resultados en las distintas áreas que han sido utilizadas. Se evaluaron tres modelos de pronóstico, los cuales se diferenciaban en la estrategia empleada para el pronóstico del valor cuota para múltiples días futuros, ya que no resulta de mucha utilidad conocer el valor de la cuota para un solo día, sino más bien para un periodo de tiempo, e identificar si la tendencia va a alza o a la baja. La estrategia de pronóstico directo fue la que otorgó mejores resultados, obteniendo un promedio de 0.5312% de MAPE y un 69.33% de acierto en la tendencia, en la fase de testing para 5 días futuros a pronosticar.

Palabras claves: Redes Neuronales Artificiales, Optimización por enjambres de partículas, series de tiempo financieras, Pronóstico.

1. Introducción

En Chile las instituciones encargadas de administrar los fondos y ahorros de pensiones de las personas son las AFP (Administradoras de Fondos de Pensiones). Estas son sociedades anónimas de carácter privado cuyo objetivo exclusivo es el de administrar los fondos de pensiones para un futuro otorgamiento de las prestaciones por vejez, invalidez y sobrevivencia a sus asociados. La cobertura del sistema de las AFP es obligatoria para todos los trabajadores que posean un contrato de trabajo, por lo cual deben realizar depósitos periódicos con el fin de acumular recursos para la vejez. Estos depósitos son llamados cotizaciones y corresponden al 10% de las remuneraciones y rentas imponibles mensuales. Cada afiliado debe tratar de aumentar al máximo su fondo personal para que así pueda recibir una pensión digna al momento de jubilarse o sufrir un accidente.

El capital acumulado por el trabajador en su cuenta individual es expresado en cuotas. El valor de la cuota se modifica diariamente y esta variación es producto de la rentabilidad de las inversiones de los fondos de pensiones. El valor de la cuota no es de gran importancia ya que lo primordial es la variación que sufre dicho valor, el cual se traduce en la rentabilidad que están obteniendo los recursos de cada afiliado.

Cada afiliado tiene una cuenta individual única en la cual deposita sus cotizaciones, y estas son capitalizadas e invertidas en el mercado de valores de acuerdo al Fondo de Pensiones al que se está sujeto, con el objeto de obtener una cierta rentabilidad. Estos fondos corresponden al A, B, C, D y E, siendo el fondo A el que cuenta con una mayor proporción de renta variable y el fondo E el que cuenta con un mayor componente de renta fija. El afiliado es libre de elegir el fondo en el cual quiere invertir sus recursos, pudiendo permanecer hasta en dos tipos de fondos distintos, además éste es libre de cambiarse de fondo en el momento que estime conveniente.

Hoy en día el sistema de las AFP se encuentra en crisis. El promedio de pensiones pagadas a la fecha por las administradoras de Fondos de Pensiones no superan los \$180.000. A juicio de expertos el sistema es deficiente, ya que en general, en el mejor de los casos la pensión corresponde a la mitad del promedio del sueldo mensual de los últimos 10 años. La causa de estas paupérrimas pensiones se debe al escaso capital acumulado por el trabajador durante sus años de trabajo ya sea por intermitencia de sus cotizaciones o por las bajas remuneraciones percibidas por este. Malas decisiones en los cambios de fondos ha generado grandes pérdidas en el capital acumulado. Por otra parte, el aumento significativo de la expectativa de edad de las personas en Chile acrecienta aún más el problema.

En la actualidad las herramientas y servicios disponibles para el apoyo de las personas al momento de tomar la decisión de un cambio de fondo de pensión son escasas, muchas veces de difícil acceso, y habitualmente es necesario poseer conocimientos técnicos avanzados para su utilización. Por lo general están basadas en modelos matemáticos y técnicas de trading, valiéndose de la utilización de indicadores económicos para la detección temprana de la tendencia de la rentabilidad.

Si los afiliados tuvieran acceso a un pronóstico confiable del valor cuota de cada fondo, sería posible conocer la rentabilidad otorgada por cada uno, permitiéndoles elegir aquel que les otorgue la mayor rentabilidad. El fondo tipo A, al poseer una mayor parte de renta variable, es el que otorga mayores rentabilidades, pero a su vez, también es el que

genera mayores pérdidas al capital de cada cotizante. De hecho muchos afiliados al fondo tipo A han perdido cantidades enormes de dinero sin siquiera saberlo. Con la información correcta es posible usufructuar de la rentabilidad que ofrece el fondo A, y de la estabilidad que ofrece el fondo E en el caso de que la rentabilidad del A fuese negativa. Lo anterior con el objeto de maximizar las ganancias y reducir las pérdidas, provocando un aumento del capital para que, llegado el momento, se perciban mejores pensiones.

En el siguiente gráfico se puede apreciar la mayor rentabilidad otorgada por el fondo A y la estabilidad que caracteriza al fondo E.

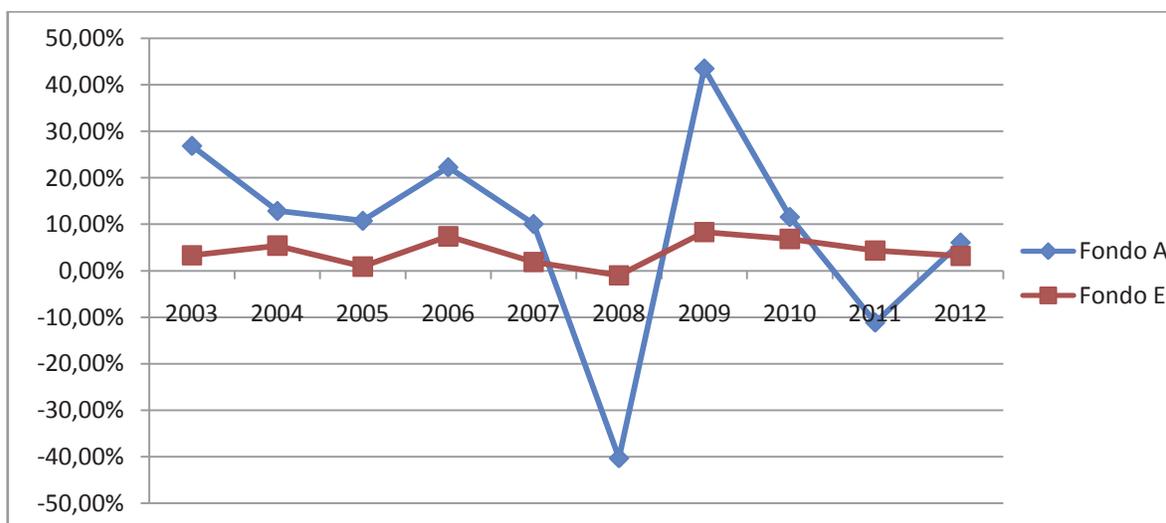


Figura 1.1 Rentabilidad anual de los fondos A y E.

Pese a que existen periodos en los cuales el fondo A produce rentabilidades negativas, el promedio de rentabilidad otorgado por este fondo desde el 2002 es positiva y corresponde a un 6,7% de rentabilidad promedio anual y posee un 94,9% de rentabilidad real acumulada. Esto avala la teoría de los expertos de que los fondos de pensión son inversiones a largo plazo y que a mayor riesgo mayor retorno.

Existen periodos en que lo anterior no se cumple, como es el periodo comprendido entre el año 2007 y 2008, en el cual el fondo A tiene una gran caída. Si nos hubiésemos cambiado en ese momento al fondo E, la rentabilidad conseguida sería mayor a la obtenida en el caso de habernos mantenidos en el fondo A (pese a que la teoría sostiene que a mayor riesgo mayor rentabilidad), y si luego en el 2008 se retoma el fondo A, se aprovecharía la gran rentabilidad otorgada a partir de ese periodo. De esta manera la rentabilidad acumulada y promedio hubiese sido mucho mayor a la percibida si nos hubiésemos mantenido sólo en el fondo A.

Esto demuestra que con una buena estrategia de cambio de fondo y con la información adecuada es posible no sólo evitar las pérdidas, sino que además, es posible aumentar el capital acumulado en una proporción considerable.

Fruto de esta problemática surge la necesidad de crear una herramienta de predicción para el apoyo en la administración de los fondos de pensiones, la que analizará datos de mercado y pronosticará el valor cuota para el fondo A en un determinado periodo, y en base a la información obtenida, elegir el fondo que nos otorgue más ganancias o en su defecto,

menos pérdidas. Para el desarrollo de esta herramienta se utilizarán algoritmos de inteligencia artificial como forma de describir las relaciones subyacentes de los valores cuota, y de esta manera poder evaluar el rendimiento de este tipo de algoritmos. Específicamente se emplearán Redes Neuronales Artificiales de Base Radial con algoritmo de aprendizaje basado en la optimización por enjambre de partículas PSO, dado que en la teoría y la práctica han demostrado ser una herramienta potente para la solución de este tipo de problemas no lineales y no estacionarios.

En la primera sección del documento se presentará el estado del arte del pronóstico financiero y de las redes neuronales artificiales, seguidamente de la arquitectura propuesta para su implementación. Luego se definirá el algoritmo de aprendizaje híbrido que servirá como método de entrenamiento de la red, seguido de una descripción del modelo utilizado para el pronóstico. Finalmente se presentarán los resultados y conclusiones del experimento.

2. Objetivos

A continuación se presentan los objetivos planteados para este proyecto. Primero se describe el objetivo general del proyecto seguido por los objetivos específicos que sirven de apoyo para la obtención del objetivo general.

2.1. Objetivo General

Desarrollar un modelo de regresión basado en Redes Neuronales Artificiales de Base Radial para el pronóstico de la variación del valor cuota de los fondos de pensiones, utilizando como método de entrenamiento algoritmos de inteligencia de enjambre.

2.2. Objetivos Específicos

- Explicar el funcionamiento de las redes neuronales y su aplicación al pronóstico de series de tiempo, así como también el de los algoritmos de optimización por enjambres de partículas.
- Estimar la estructura y los parámetros del pronosticador neuronal que mejor se ajusten a los requerimientos del problema.
- Evaluar y comparar el rendimiento de los pronosticadores neuronales propuestos.

3. Pronóstico de series de tiempo financieras.

Las series de tiempo corresponden a una secuencia de datos, observaciones o valores, medidos en determinados momentos y ordenados cronológicamente. El pronóstico de series de tiempo busca desarrollar un modelo que en base a un análisis describa las relaciones subyacentes entre los datos presentes en esta y que permitan extrapolar o interpolar los datos y así predecir el comportamiento de la serie en momentos no observados [34].

En el último tiempo se le ha dado gran importancia e interés al pronóstico de series temporales financieras dado que constituyen una información valiosa a la hora de realizar inversiones o en la toma de decisiones financieras, por lo que han llamado la atención considerablemente y se han convertido en un área de investigación activa. Sin embargo el pronóstico de este tipo de series de tiempo es considerado como uno de los problemas más complejos, dado que estas son inherentemente ruidosas y además por su naturaleza no estacionaria y determinísticamente caótica [1]. Por otra parte estas series están normalmente sujetas a régimen de cambios, por lo cual sus propiedades estadísticas no permanecen constantes en el tiempo. Además, una gran cantidad de variaciones aleatorias del día a día se encuentran presentes en estas series, lo que las hace aún más impredecibles. En muchas ocasiones, ciertos comunicados de prensa de algunas empresas, las medidas fiscales, los informes de trabajos o algún evento político, influyen en el comportamiento de una serie de tiempo económica, incrementando aún más su carácter ruidoso.

El pronóstico de series de tiempo estuvo influenciado durante mucho tiempo por métodos estadísticos lineales, tal es el caso de los modelos ARIMA. Sin embargo entre los años 1970 y 1980 se hizo cada vez más evidente que los modelos lineales no se adaptaban a problemas reales. En este mismo periodo varios modelos no lineales fueron propuestos, como es el caso del modelo bilineal diseñado por Poskitt y Tremayne en 1986, el modelo umbral autorregresivo planteado por Tong y Lim, y el modelo de heterocedasticidad condicional autorregresivo (ARCH) formulado por Engle en 1982. Cabe destacar que el estudio del análisis y pronóstico de series temporales no lineales está aún en su infancia en contraste con el estudio de series de tiempo lineales [34].

Los resultados obtenidos por modelos híbridos sugieren que la combinación de pronósticos realizados por más de un modelo a menudo presentan mejores desempeños de predicción [35]. La principal razón se debe a la posibilidad de que un proceso de generación de datos cambie su estructura a lo largo del periodo de observación entre una estructura lineal y no lineal.

En las últimas dos décadas, los modelos de aprendizaje automático han sido muy utilizados y se han establecidos como serios aspirantes a modelos estadísticos clásicos en el ámbito del pronóstico. Estos modelos también llamados modelos de caja negra, son del tipo no lineal y no paramétricos, que utilizan sólo los datos históricos para conocer la dependencia estocástica entre el pasado y el futuro. Werbos determinó que las redes neuronales artificiales superan a los métodos estadísticos clásicos, como la regresión lineal y los enfoques de Box-Jenkins. Los árboles de decisión, las máquinas de soporte vectorial, y la regresión del vecino más cercano son otros ejemplos de este tipo de modelos [36].

Box y Jenkins han combinado los modelos ARIMA con redes neuronales artificiales y la lógica difusa con el fin de superar las limitaciones lineales y de las muestras de datos de la

metodología ARIMA. Sus resultados han demostrado la superioridad presentada por los modelos de pronósticos combinados [37].

Investigadores en Taiwan establecieron un nuevo modelo de predicción de series de tiempo financieras basado en la evolución y agrupación de árboles de decisión difusa para las acciones en la bolsa de Taiwan Corporation [38]. Este modelo de predicción integra una técnica de agrupación de datos, un árbol de decisión difuso y algoritmos genéticos para la construcción de un sistema de toma de decisiones basado en datos históricos e índices técnicos. El conjunto de datos históricos se divide en K subgrupos adoptando el algoritmo K-means. Los algoritmos genéticos son aplicados para evolucionar el número de términos difusos para cada índice de entrada en el árbol de decisión difuso, por lo que la exactitud predictiva del modelo puede ir mejorando cada vez. Un modelo de predicción diferente se genera para cada subgrupo. Esta metodología obtuvo un buen rendimiento alcanzando una tasa de un 82% de éxito promedio en comparación a otros enfoques.

Por otra parte las máquinas de soporte vectorial también han dado buenos resultados en el pronóstico de estas series temporales, debido a que utilizan una función de riesgo consistente en el error empírico y un término de regularización que deriva del principio de minimización del riesgo estructural. Los resultados experimentales muestran que esta tecnología es una alternativa prometedora en la predicción del mercado de valores [39].

Francis Tay y L.J. Cao propusieron una versión modificada de las máquinas de soporte vectorial (MSV), llamada Maquinas de soporte vectorial con C-ascendente para el modelado de series temporales financieras [40]. Este se obtiene mediante una simple modificación de la función regularizadora de riesgos, en la cual los errores insensibles recientes están penalizados en mayor medida que los errores insensibles lejanos. Este procedimiento se basa en el conocimiento previo de que en las series de tiempo financieras no estacionarias la dependencia entre las variables de entrada y de salida cambia gradualmente con el tiempo, es decir, los últimos datos del comportamiento pasado pueden proporcionar información más relevante que los datos de un pasado distante. Este tipo de metodología fue aplicada a los datos provenientes de Chicago Mercantile Market y se obtuvieron mejores pronósticos que los otorgados por las MSV tradicionales. Por otra parte las MSV C-ascendente utilizan menos vectores de soporte que las MSV estándar.

Recientemente se realizó un estudio en el cual se incluyen los sentimientos del mercado en el modelo de predicción [41]. Teniendo en consideración que el sentimiento del mercado contiene información útil para la predicción, en este trabajo se utiliza información textual para ayudar en el pronóstico. Esta metodología consta de tres etapas: primero la representación textual de los sentimientos, expresados como vectores de características, seguido del uso de ARIMA para modelar la parte lineal y finalmente el desarrollo de vectores de soporte de regresión basados en los vectores de características para la parte no lineal. Este enfoque proporciona una alternativa prometedora para el pronóstico de series de tiempo financieras.

3.1. Estrategias de pronóstico para varias etapas futuras

Dado que el pronóstico de una sola etapa (año, día, semana) en las series de tiempo no es de gran utilidad, ya que por lo general se desea saber el comportamiento de la serie en un horizonte mayor, es necesario realizar el pronóstico para varias etapas futuras. Esto resulta una tarea compleja e implica un gran reto para los problemas de predicción de series de tiempo

producto de las incertidumbres crecientes derivadas de los errores acumulados y la falta de información [45]. Existen varias estrategias que podrían ser utilizadas para la construcción de modelos de predicción para varias etapas futuras, a continuación se detallan algunas.

3.1.1. Estrategia iterativa o recursiva

Este tipo de estrategia es la más común e intuitiva y consiste en implementar un único modelo de pronóstico capacitado para predecir solo la siguiente etapa. Si se desea pronosticar las N etapas siguientes, el primer paso consiste en realizar el pronóstico para la primera etapa. Posteriormente se utiliza dicho valor como variable de entrada en el mismo modelo con el fin de predecir la etapa subsiguiente, y así sucesivamente hasta pronosticar todo el horizonte deseado.

El método iterativo puede sufrir de bajo rendimiento en pronósticos de horizontes largos, o en series de tiempo en donde la relación subyacente existente entre los datos es muy sensible. El motivo de este bajo rendimiento se debe a que esta estrategia va acumulando errores cada vez que se avanza en el horizonte pronosticado. Los errores presentes en las estimaciones intermedias se propagan hacia adelante ya que estas estimaciones se utilizan como datos de entrada para determinar los valores posteriores [46].

Este tipo de estrategia presenta una gran limitación ya que el horizonte a pronosticar no puede ser mayor al número de datos de entrada del modelo, ya que si llegase a suceder esta situación, en algún momento todos los valores de entradas serían estimaciones y no valores reales.

3.1.2. Estrategia directa

Esta estrategia, también conocida como estrategia independiente, consiste en pronosticar cada horizonte independiente de los otros, es decir, si se desea pronosticar N etapas siguientes se deben implementar N modelos distintos, uno para cada etapa a pronosticar.

Esto implica que esta estrategia no utiliza valores aproximados para realizar los pronósticos, por lo que no se ve afectada por la acumulación de errores. Sin embargo, como los modelos se aprenden de forma independiente, el pronóstico resultante no es capaz de dar cuenta de las dependencias existentes entre los valores del horizonte pronosticado, afectando la precisión del pronóstico [47]. Además, esta estrategia requiere un gran tiempo de cálculo, ya que hay que determinar tantos modelos como el tamaño del horizonte.

3.1.3. Estrategia múltiples entradas múltiples salidas

Las estrategias anteriores pueden considerarse como estrategias de una única salida, modelando múltiples datos de entrada pero convergiendo a una sola salida. Como forma de evitar dejar de lado la existencia de dependencias estocásticas entre los valores futuros a pronosticar, se ha propuesto el modelo de múltiples entradas y múltiples salidas, en donde el número de salidas es igual al horizonte a pronosticar, y cada salida corresponde a una etapa de dicho horizonte. Esta estrategia evita el supuesto de independencia condicional hecha por la estrategia directa, así como la acumulación de errores presente en la estrategia iterativa o recursiva.

Sin embargo, la necesidad de preservar las dependencias estocásticas utilizando un solo modelo tiene un inconveniente, ya que restringe todos los horizontes para ser pronosticadas con la misma estructura del modelo. Esta limitación podría reducir la flexibilidad del enfoque de predicción [48].

Se han realizado múltiples investigaciones y experimentos con distintos tipos de redes neuronales artificiales aplicados al pronóstico de series temporales financieras obteniendo buenos resultados en la predicción, por ende se puede inferir que esta tecnología es apta para la solución de este tipo de problemas, por lo cual se espera obtener resultados favorables en el pronóstico del valor cuota para el fondo A en la AFP CUPRUM.

4. Redes Neuronales Artificiales

Las redes neuronales artificiales representan un esquema de computación distribuida compuesto de unidades procesadoras adaptativas con una alta interconexión entre ellas. Estas son un paradigma de aprendizaje y procesamiento automático que se inspira en el funcionamiento del sistema nervioso en los animales y se comporta de una forma similar a las neuronas biológicas del cerebro.

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 y 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).

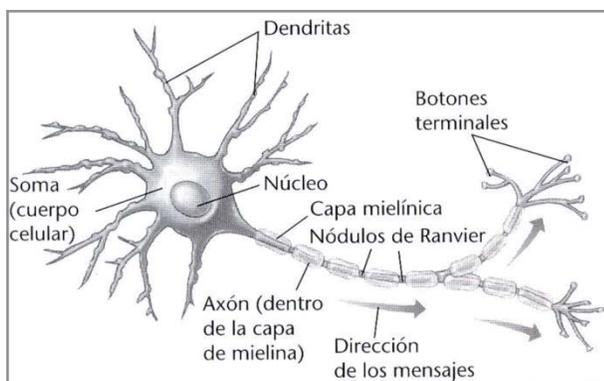


Figura 4.1 Neurona Biológica.

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

Las señales de entrada se encuentran moduladas por un factor llamado peso (Sinapsis), que gradúa la importancia de la conexión existente entre la neurona receptora y el emisor de la señal. En definitiva podemos señalar que una neurona es en realidad un procesador muy simple con una capacidad limitada de cómputo, restringida a un conjunto de instrucciones elementales y una pequeña memoria para almacenar pesos y activaciones.

Es posible modelar una neurona artificial en base a tres elementos básicos:

- Una serie de sinapsis o entrada, cada una caracterizada por un peso sináptico w , el cual define la fuerza de una conexión sináptica entre dos neuronas. Estos pesos pueden tomar valores positivos, en cuyo caso actúan como excitador, como también puede tomar valores negativos, actuado en este caso como inhibidor.
- Una función de transferencia, usualmente es un sumador o un multiplicador, la cual procesa las señales de entrada hacia la neurona en base a su peso sináptico.

- Una función de activación para limitar la amplitud de la salida de la neurona. Esta corresponde habitualmente a una función no lineal que normaliza el estado interno de la neurona a un intervalo cerrado $[0,1]$ o alternativamente $[-1,1]$. Ejemplos de función de activación son: la función escalón, sigmoideal, polinomial y las de base radial.

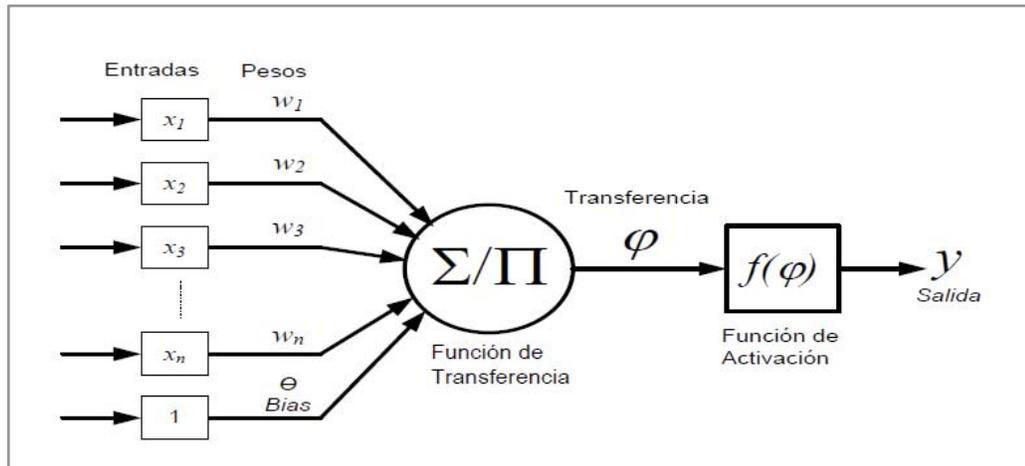


Figura 4.2 Neurona Artificial.

En general las RNA son un sistema de interconexión de múltiples procesadores elementales (neuronas artificiales) que colaboran entre sí con el objetivo de producir un estímulo de salida. Es importante destacar que la propiedad más importante y llamativa de las redes neuronales artificiales es su capacidad de aprender a partir de un conjunto de patrones de entrenamiento siendo capaz de encontrar un modelo que ajuste los datos. Además permite resolver problemas que no son linealmente separables, encontrando una relación estocástica entre los datos de entrada.

Las redes neuronales han sido aplicadas con bastante éxito en diversas áreas, ya que constituyen un método práctico y general para aprender funciones objetivas a partir de ejemplos, es decir, son capaces de encontrar patrones de forma inductiva por medio de algoritmos de aprendizajes basado en los datos existentes.

4.1. Redes Neuronales Para Pronóstico

En gran parte de los problemas de pronóstico la relación entre las variables implicadas es más compleja de lo que hacen suponer los modelos teóricos habituales. Es aquí donde las redes neuronales presentan una gran ventaja ya que permiten hacer relaciones no lineales entre las variables e incluso encontrar relaciones que son implícitas. Esta es una de las razones por las cuales este tipo de metodología ha sacado buenos resultados en el ámbito de las predicciones.

Durante décadas muchas investigaciones y publicaciones han sido dedicadas al desarrollo de sistemas para predecir el precio de valores de mercado.

En Irán dos investigadores implementaron una red neuronal local lineal de base radial para predecir el valor de las acciones de Iranxhodro [6]. En este tipo de red los pesos que conectan la capa oculta con la capa de salida son reemplazados por un modelo local lineal. Para el entrenamiento de la red se utilizaron 1700 datos, que corresponden al valor diario de

las acciones de Irankhodro. La red consta de 3 nodos de entradas, que equivalen al valor de la acción de los 3 días previos al valor a estimar, 10 nodos ocultos y un nodo de salida el cual otorgará el valor de la acción estimada por la red. Para el aprendizaje se empleó la optimización por enjambre de partículas con partículas fantasmas. La población del enjambre es de 40 partículas y se utilizaron 50 iteraciones, los coeficientes de confianza $c1$ y $c2$ equivalentes a 2 y 5 respectivamente. La mejor predicción dada por esta red tiene un error cuadrático medio de 0.0040 en la fase de training y un 0.0042 en el testing.

Recientemente se introdujo una arquitectura híbrida de red neuronal basada en la optimización por enjambres de partículas y una función de base radial adaptativa (ARBF-PSO) para la predicción de los tipos de cambio de Euro a distintas monedas [9]. En esta, el peso de inercia y los coeficientes cognitivo y social van variando a medida que avanzan las iteraciones con el fin de tener una buena exploración y explotación del espacio de búsqueda. En este trabajo se introdujo una modificación al PSO convencional incluyendo dos operadores más, cuya función es la de agregar o eliminar nodos ocultos según lo exija la complejidad del problema. Se utilizó un enjambre de 30 partículas y 100 iteraciones para converger al óptimo. La data utilizada consiste en el histórico de 10 años de los valores de los tipos de cambio. La arquitectura óptima de la red se consiguió con 7 nodos de entrada, 14 nodos ocultos dando como resultado un error cuadrático medio de 0.0064 en la predicción del tipo de cambio de Euro a Dólar.

La predicción de series de tiempo constituye una tarea bastante compleja por lo que se han realizado muchas investigaciones en torno al tema. Se desarrolló un modelo para la solución de esta problemática fundamentado en redes neuronales de base radial con el algoritmo de aprendizaje PSO y una función evolutiva no lineal variante en el tiempo [11] para ajustar los pesos de inercia y los coeficientes de aceleración mostrada en la ecuación 1, 2 y 3. Para el entrenamiento se utilizaron los datos de la demanda diaria de electricidad correspondientes al periodo de un mes. La mejor performance se consiguió con 40 partículas y 3000 iteraciones. La arquitectura de la red consta de ocho nodos de entradas, 20 nodos ocultos, determinados a partir de la técnica de ensayo y error, y un nodo de salida correspondiente a la predicción de la demanda eléctrica de un día. El mejor error conseguido fue de un 0.69% de MAPE.

$$\omega(Iter) = \omega_{\min} + \frac{(Iter_{\max} - Iter)^{\alpha}}{Iter_{\max}} * (\omega_{\max} - \omega_{\min}) \quad (1)$$

$$c1(Iter) = c1_{\min} + \frac{(Iter_{\max} - Iter)^{\beta}}{Iter_{\max}} * (c1_{\max} - c1_{\min}) \quad (2)$$

$$c2(Iter) = c2_{\max} + \frac{(Iter_{\max} - Iter)^{\delta}}{Iter_{\max}} * (c2_{\min} - c2_{\max}) \quad (3)$$

Además las RNA, parecen ser de las más flexibles y precisas en la predicción temprana de crisis bancarias [8]. El tipo de RNA perceptrón multicapa se ha revelado como un excelente

predicador de las situaciones de crisis bancarias a corto plazo, con índices de acierto cercanos al 94% [18].

Pero no solo en las predicciones financieras las redes neuronales se desempeñan de buena manera, se han conseguido muy buenos rendimientos en la predicción de la demanda en las industrias, de hecho los resultados apoyan la utilización de las RNA como técnicas confiables de pronóstico y demuestran su factibilidad de instauración dentro de sus estrategias de negocios.

Por otro lado se desarrolló una aplicación para predecir 7 propiedades térmicas de distintas sustancias [7]. Para esto se empleo una red neuronal con la tangente hiperbólica como función de transferencia y el algoritmo PSO para el aprendizaje de la red. El enjambre se compone de 50 partículas y el número de iteraciones es de 1500. Su coeficiente cognitivo y social (c_1 y c_2) es de 1.494 mientras que el W_{min} y el W_{max} es de 0.5 y 0.7 respectivamente. La arquitectura óptima de esta red se consiguió con 4 nodos de entrada, que representan las características de cada sustancia (Masa molecular, momento de inercia, polaridad y estructura molecular), 8 nodos ocultos y un nodo de salida. En esta investigación se logro alcanzar un coeficiente de determinación R^2 de 0.9923

Un inteligente modelo es propuesto para la predicción de la proporción de gas condensando en las reservas de dicho gas [10]. Este modelo está basado en una red neuronal feed-forward con el algoritmo de aprendizaje PSO. En el entrenamiento se utilizo una data de 295 registros históricos. Los mejores resultados se obtuvieron con 3 nodos de entrada que corresponden a características de dicho gas: temperatura, presión del punto de rocío y peso molecular. La capa oculta se compone de 8 nodos ocultos, dando como salida la estimación de la proporción de gas condensado. El coeficiente social y cognitivo del algoritmo de aprendizaje equivale a 1.4 y 2.4 respectivamente mientras que el W_{min} y el W_{max} son de 0.4 y 0.5. El enjambre se compuso de 21 partículas y se emplearon 500 iteraciones para converger a la solución deseada con un error cuadrático medio de 0.041.

Uno de los aspectos más importantes antes de aplicar cualquier herramienta de predicción es el análisis previo de los datos disponibles, para determinar de esta forma la correlación con la variable de estudio. La determinación correcta de los datos de entrada juega un rol importante ya que demasiada información puede hacer lento el proceso de entrenamiento, o dar información redundante que no aporta el mejoramiento de la calidad de la salida deseada.

4.2. Redes Neuronales De Base Radial

Para la presente investigación se utilizarán Redes Neuronales de Base Radial dado que permiten resolver problemas que no son linealmente separables, como sucede en nuestro caso de estudio. Estas redes han adquirido una gran popularidad en los últimos años, junto al perceptrón multicapa, habiéndose aplicado en una gran variedad de problemas.

Una de las grandes cualidades de este tipo de redes es su capacidad para aproximar las salidas deseadas sin necesidad de una descripción matemática de cómo las salidas funcionalmente dependen de las entradas. Reciben información sobre el dominio del problema e intentan ajustar los pesos para aprender de las muestras de entrenamiento.

El aprendizaje realizado por este tipo de redes, se basa en la determinación de los centros y desviaciones de la función de activación, además de los pesos de la capa oculta a la

capa de salida. Por el hecho de que las capas de la red realizan diferentes tareas, se separan los parámetros de la capa oculta de la capa de salida para optimizar el proceso. Dada esta situación los centros y desviaciones se guían por una optimización en el espacio de entrada, mientras que los pesos siguen una optimización en base a las salidas que se esperan obtener.

4.3. Arquitectura de la RNBR

Las RNBR son redes del tipo perceptrón multicapa con conexión feed-forward, donde la propagación de las señales se da en un solo sentido, es decir, sólo hacia la siguiente capa sin la posibilidad de bucles de retroalimentación.

Esta red está compuesta por tres capas: La capa de entrada, una capa oculta y la capa de salida. La capa de entrada se constituye de unidades sensoriales o neuronas, que introducen los datos de entrada a la red. La siguiente capa, la capa oculta, es la encargada de aplicar una transformación no lineal del espacio de entrada al espacio oculto de alta dimensionalidad. Esta capa se compone de unidades o neurona ocultas, cada una de las cuales representa una función de base radial y son las encargadas de otorgar las propiedades de generalización, extracción de características y adaptabilidad para la solución del problema. Por último en la capa de salida se encuentran las neuronas que otorgan la salida de la red. Estas son las encargadas de realizar una combinación lineal de los datos resultantes de la activación de las neuronas ocultas.

A continuación se presenta un esquema de la red neuronal de base radial:

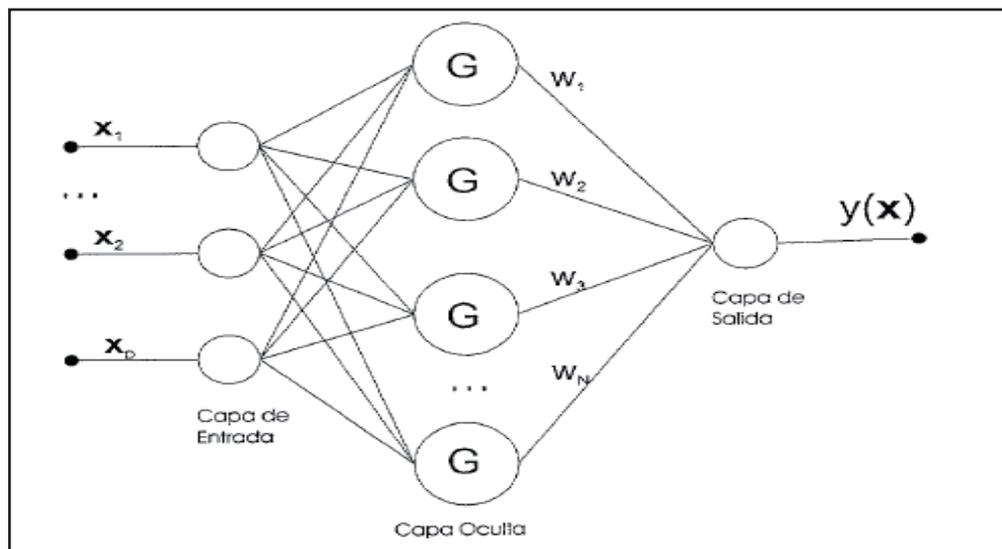


Figura 4.3. Red Neuronal Base Radial con Función Gaussiana

El número de neuronas de entrada y ocultas presentes en la RNBR depende de la naturaleza del problema. Estos valores son los que determinan el grado de complejidad de la red. La función de propagación que conectará la capa de entrada con la capa oculta viene dada por la ecuación (4), y es la encargada de calcular la distancia euclídea del vector de entrada con respecto del centro.

$$z = \left\| \frac{x_i - t_k}{d_k} \right\|; \quad (4)$$

donde x corresponde al vector de datos de entrada, y las componentes t y d corresponden al vector de los centros y desviaciones de la función de activación. Estas tres variables se representan de la siguiente manera:

$$x_i = (x_{i,1}, x_{i,2}, \dots, x_{i,n}), i\text{-ésima muestra}; \quad (5)$$

$$t_k = (t_{k,1}, t_{k,2}, \dots, t_{k,n}), \quad d_k = (d_{k,1}, d_{k,2}, \dots, d_{k,n}), k\text{-ésimo nodo oculto} \quad (6)$$

La función de activación es la encargada de recibir las señales de entrada y determinar la salida de la neurona. Para este tipo de redes existen distintas funciones de activación de base radial, entre ellas encontramos:

- **Función Gaussiana**

$$f(x) = \exp(-0.5 \cdot z^2) \quad (7)$$

- **Función Mexican Hat**

$$f(x) = -z \cdot \exp(-0.5 \cdot z^2) \quad (8)$$

- **Función Morlet Wavelet**

$$f(x) = \cos(5z) \cdot \exp(-0.5 \cdot z^2) \quad (9)$$

- **Función Multicuadrática inversa**

$$f(x) = \frac{1}{\sqrt{(1 + z^2)}} \quad (10)$$

La capa de salida está formada por una unidad lineal conectada a todas las unidades ocultas, de forma que la salida de la red es una combinación lineal de las salidas de las unidades ocultas con los pesos de salida, y viene determinada por la siguiente ecuación:

$$y = \sum_{k=1}^h w_k G_k \quad (11)$$

Donde w_k representa los pesos sinápticos entre la capa oculta y la capa de salida. La solución proporcionada por esta red será un interpolador óptimo, en el sentido que otorgara la salida con el menor error alcanzado.

5. Aprendizaje de la red

Una de las características más destacables de las redes neuronales es su capacidad de aprendizaje. Esta permite que la red vaya modificando sus parámetros, adaptándose hasta conseguir un algoritmo de ejecución óptimo.

El proceso de aprendizaje se puede describir como el proceso de buscar en un espacio n-dimensional una superficie que se ajuste de la mejor manera posible a los datos (puntos del espacio) de entrenamiento. Es importante destacar que la superficie encontrada debe poseer la característica de generalización, es decir, debe servir para interpolar datos distintos a los entrenados.

5.1. Aprendizaje supervisado

En este trabajo se empleara el método de aprendizaje supervisado en el cual se conocen las salidas deseadas, las cuales se comparan con las obtenidas por el estimador. Se mide la distancia entre el valor deseado y el estimado, y se modifican los parámetros de la red hasta que la distancia sea mínima. Cada vector de entrada es sometido a la red de forma secuencial, repitiendo el proceso hasta que la red se estabiliza y deja de sufrir modificaciones.

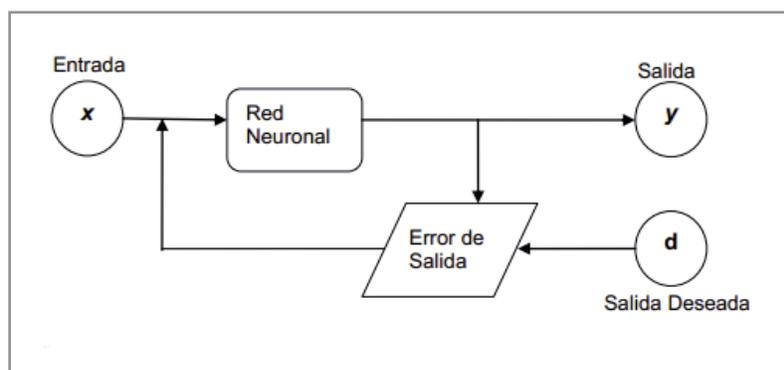


Figura 5.1 aprendizaje supervisado

5.2. Optimización por enjambres de partículas PSO

La optimización por enjambres de partículas PSO (Particle Swarm Optimization) es una metaheurística poblacional de optimización fundamentada en la inteligencia colectiva y que fue propuesta por Kennedy y Eberhart en 1995. Este método utiliza un simple mecanismo que imita el comportamiento social de organismos biológicos, como bandadas de pájaros y cardúmenes de peces, en cuanto a la forma en que estos exploran el espacio y guían sus desplazamientos en base al individuo que está más cerca del alimento, considerado como su objetivo primordial [4]. Un enjambre hace alusión a una bandada de pájaros por lo cual una partícula representa un ave que está siempre en continuo movimiento.

El enjambre de partículas es un sistema multiagente en donde sus partículas son agentes simples que exploran el espacio de búsqueda guardando y comunicando la mejor solución parcial hasta llegar a su objetivo, El movimiento de las partículas en el espacio

está guiado por aquellas que tiene la mejor solución del momento, por lo tanto, cada posición de la partícula es considerada una solución [1]. Esta solución es del tipo cooperativa ya que su posición es el resultado del intercambio de experiencias exitosas entre las partículas vecinas del enjambre durante la ejecución del algoritmo.

El enjambre explora el espacio de búsqueda en un número determinado de movimientos, o en relación a alguna condición de término, siendo la mejor partícula la solución al problema. Para saber cuan conveniente es la posición de una partícula, esta es sometida a evaluación mediante una función de aptitud (fitness), la cual estima que tan buena es respecto al objetivo y al ambiente del problema.

PSO ha demostrado ser un procedimiento muy eficiente en la solución de problemas de optimización del tipo complejo y gracias a su fácil implementación se ha aplicado con éxito en diferentes campos de investigación, como por ejemplo en la optimización de funciones numéricas, aprendizajes de sistemas difusos, ingeniería química y en el entrenamiento de redes neuronales ha conseguido muy buenos resultados.

A continuación se realiza una descripción formal de las dos variantes del algoritmo PSO, junto a sus principales factores y aspectos de su desarrollo.

5.2.1. Algoritmo PSO Continuo

El algoritmo PSO es un proceso iterativo y estocástico que actúa sobre un enjambre de partículas. Una solución potencial es representada por la posición de la partícula, la cual tiene asociada tres vectores y dos valores de fitness. Cada una de estas partículas i tiene la misma dimensión n y se representa de la siguiente manera:

$$X(i) = (x_{(i,1)}, x_{(i,2)}, \dots, x_{(i,n)}) \quad (12)$$

El vector $X(i)$ almacena la posición actual de la partícula en el espacio de búsqueda n -dimensional. Esta posición va variando en el transcurso del proceso de acuerdo a una combinación lineal de su inercia, su propia experiencia y del conocimiento del enjambre.

Cada agente almacena en un vector la posición de la mejor solución encontrada hasta el momento t . La experiencia de la partícula se denota:

$$Pbest(i) = (p_{(i,1)}, p_{(i,2)}, \dots, p_{(i,n)}) \quad (13)$$

Cada elemento de $X(i)$ tiene asociada una velocidad o razón de cambio:

$$V(i) = (v_{(i,1)}, v_{(i,2)}, \dots, v_{(i,n)}) \quad (14)$$

Este vector de velocidad almacena el gradiente (dirección) según el cual se moverá la partícula. Para realizar dicho desplazamiento, la partícula determina la velocidad considerando su propia inercia (para evitar la convergencia prematura), su memoria de conocimiento y su confianza en el enjambre.

Los valores de fitness asociados a la partícula corresponden al $fitness(X_i)$, que almacena el valor de aptitud de la solución actual $X(i)$, y el $fitness(Pbest_i)$ que almacena el valor de aptitud de la mejor solución local encontrada hasta el momento $Pbest(i)$.

El enjambre de partículas se inicializa generando las posiciones $X(i)$ y las velocidades $V(i)$ de forma aleatoria en el intervalo $[-1,1]$. Inmediatamente se calcula el fitness de cada partícula y se actualizan los valores de $fitness(X_i)$ y $fitness(X_i)$.

Una vez inicializado el enjambre, las partículas se mueven en el espacio de búsqueda durante el proceso iterativo. El desplazamiento de la partícula es ocasionado simplemente por la adición del vector velocidad al vector posición:

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (15)$$

Luego de que la posición de la partícula es actualizada, esta es evaluada por la función de aptitud (fitness) actualizando el valor de $fitness(X_i)$. Si este nuevo valor es mejor que el mejor fitness parcial, se actualizan los valores de la mejor posición $Pbest(i)$ además del valor del $fitness(Pbest_i)$.

En cada iteración también es modificado el vector velocidad en base a su velocidad actual, su inercia ω , un componente cognitivo y un componente social. La ecuación resultante para dicha ecuación, y que representa la base del algoritmo PSO es la siguiente:

$$V_i(t+1) = \omega \cdot V_i(t) + c_1 \cdot r_1 \cdot (Pbest_i(t) - X_i(t)) + c_2 \cdot r_2 \cdot (Pbest_g(t) - X_i(t)) \quad (16)$$

La ecuación (13) refleja la actualización del vector velocidad de cada partícula i en cada iteración t . El componente cognitivo viene dado por la expresión: $c_1 \cdot r_1 \cdot (Pbest_i(t) - X_i(t))$ y representa la distancia entre la posición actual y la mejor posición conocida por esa partícula, de manera más simple el componente cognitivo permite que la nueva posición de la partícula sea influenciada por su propia experiencia a lo largo de su vida. Por otra parte el componente social esta modelado por $c_2 \cdot r_2 \cdot (Pbest_g(t) - X_i(t))$ y representa la distancia entre la posición actual y la mejor posición del vecindario, es decir, este componente proporciona la influencia que el resto del enjambre ejerce sobre ella.

El grado de confianza lo determinan los coeficientes aleatorios r_1 y r_2 junto con los coeficientes de confianza c_1 y c_2 , también llamados constantes de aceleración. Estos últimos son los términos que atraen a cada partícula hacia las posiciones $Pbest_i$ y $Pbest_g$ [15]. $Pbest_g$ corresponde a la posición de la mejor partícula del entorno de P_i o de todo el enjambre.

El factor de inercia ω es el encargado de evitar la convergencia prematura y contrarrestar el poco control sobre el balance entre la exploración y explotación. El coeficiente ω se va reduciendo gradualmente a medida que avanzan las iteraciones del algoritmo.

5.2.2. PSO Adaptativo

Como forma de obtener un mejor control y balance entre la exploración y explotación del espacio de búsqueda, se ha utilizado una versión modificada del algoritmo de optimización por enjambres de partículas, el PSO adaptativo.

Para que el enjambre explore el espacio de búsqueda a fondo durante las primeras iteraciones es necesario que los valores de las variables c_1 y ω sean altos. En las iteraciones

finales el enjambre debe converger a una solución óptima por lo que el área de la mejor solución debe ser explorada a fondo. Para esto c_2 debe ser relativamente alto y c_1 y ω debe mantenerse con valores bajos. Con la finalidad de lograr que el enjambre se comporte de esta manera, evitando el atascamiento temprano de óptimos locales se ha introducido una modificación al PSO mediante valores adaptativos para los parámetros c_1 , c_2 y ω . Es por esta razón que en esta investigación se utilizará un PSO adaptativo, que difiere del PSO normal en el hecho de que en este último c_1 , c_2 y ω son constante, mientras que en la versión adaptativa estos parámetros van variando (adaptándose) a medida que avanzan las iteraciones con el objeto de conseguir el comportamiento descrito anteriormente. Las ecuaciones (17), (18) y (19) describen el cambio de dichos factores.

$$\omega(Iter) = \left(0.4 / Iter_{\max}^2\right) * (Iter - Iter_{\max})^2 + 0.4 \quad (17)$$

$$c1(Iter) = -2 * Iter / Iter_{\max} + 2.5 \quad (18)$$

$$c2(Iter) = 2 * Iter / Iter_{\max} + 0.5 \quad (19)$$

6. Modelos Propuestos

Para la presente investigación se propusieron tres modelos de pronóstico, diferenciados en la estrategia escogida para pronosticar varios días futuros. Esto es de gran importancia ya que el pronóstico del valor cuota para un solo día no es de utilidad para los usuarios, debido a que las AFP demoran aproximadamente 4 días hábiles en hacer efectivo el cambio de fondo [Superintendencia de valores].

Los tres modelos emplean la tecnología de las redes neuronales artificiales de base radial para determinar la relación subyacente entre los datos presentes en la serie de tiempo de valores cuota. Esta red es del tipo perceptrón multicapa, la cual consta de tres capas: una capa de entrada compuesta por 7 nodos de entrada, los cuales representan los valores cuotas de los 7 días previos al día que se desea pronosticar, una capa oculta en donde la cantidad de nodos ocultos se determinará a través de la técnica “ensayo y error”, y finalmente la capa de salida con un nodo de salida en el caso del pronóstico iterativo y el pronóstico directo, el cual corresponde al pronóstico del valor cuota para un solo día, y N nodos de salida en el caso del pronóstico de múltiples y entradas y múltiples salidas, donde N corresponde al número de días deseados a pronosticar.

La conexión existente entre las distintas capas presentes en la red neuronal es del tipo feed-forward, donde la propagación de las señales se da en un solo sentido, solo hacia la siguiente capa, no permitiendo bucles de retroalimentación.

El aprendizaje de la red será del tipo supervisado y se empleará el algoritmo de optimización por enjambres de partículas PSO en su versión adaptativa. Este será el encargado de encontrar los coeficientes del polinomio que producirán la transformación no lineal de la capa de entrada a la capa oculta y los coeficientes implicados en la transformación lineal de la capa oculta a la capa de salida y que darán como resultado el pronóstico del valor cuota. Este tipo de PSO fue escogido ya que demostró otorgar mejores resultados que el PSO normal y que el PSO local lineal, los cuales se compararon en la primera etapa de esta investigación.

La posición y velocidad inicial de los enjambres de partículas son inicializadas con valores aleatorios pertenecientes al intervalo $[-1,1]$. La calidad de cada partícula se determina en base al cálculo efectuado del error absoluto porcentual medio MAPE, presentada en la ecuación (23).

6.1. Modelo de pronóstico Iterativo utilizando redes neuronales de base radial

Este modelo básicamente consiste en encontrar el mejor pronosticador neuronal para el valor cuota de un solo día y luego iterar el pronóstico para los N días deseados. Para que esto sea posible primero se pronosticará el día n con los datos observados en los 7 días previos. Una vez obtenido el pronóstico del día n este pasará a formar parte de la data de entrada a la red y se procede a pronosticar el día siguiente $n+1$. Una vez obtenido el valor cuota del día $n+1$, este vuelve a ingresar a la red como dato de entrada para el pronóstico del día $n+2$ y así iteramos este proceso hasta conseguir el pronóstico de los siguientes N días.

La dimensión de cada partícula del enjambre para este modelo viene dada por la ecuación (20) y cada partícula se compone de los vectores de centro y desviación de la función de activación de base radial, además de los pesos implicados en la transformación lineal de la capa oculta a la capa de salida y se describe en la ecuación (21).

$$\mathcal{G} = n \cdot 2h + h \quad (20)$$

$$X = [t_{11}, \dots, t_{1h}, t_{21}, \dots, t_{2h}, \dots, t_{n1}, \dots, t_{nh}, d_{11}, \dots, d_{1h}, d_{21}, \dots, d_{2h}, \dots, d_{n1}, \dots, d_{nh}, w_1, \dots, w_h] \quad (21)$$

, donde n corresponde a la cantidad de nodos de entrada, en nuestro caso 7, y h corresponde a la cantidad de nodos ocultos presentes en la capa oculta del pronosticador neuronal.

En la figura 6.1 podemos observar como la salida de una red es vuelta a introducir al modelo como dato de entrada para pronosticar el día subsiguiente.

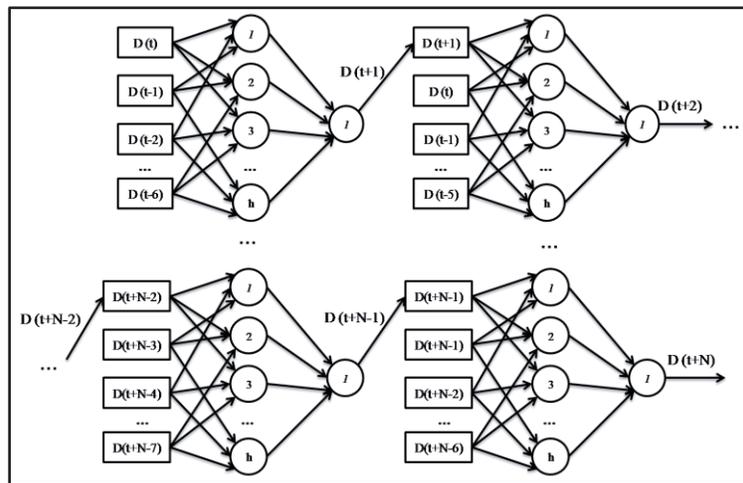


Fig 6.1 Modelo de pronóstico iterativo.

6.2. Modelo de pronóstico Directo utilizando redes neuronales de base radial

Este modelo consiste en encontrar un pronosticador neuronal para cada día futuro a pronosticar. Si se desean pronosticar N días siguientes es necesario determinar N pronosticadores neuronales distintos. Este modelo requiere de mucho tiempo para su implementación ya que es necesario definir tantos pronosticadores neuronales como días deseados a pronosticar.

Para este modelo la dimensión y la posición de la partícula se definen al igual que en el modelo anterior y viene dado por las ecuación (20) y (21) respectivamente.

En la figura 6.2 podemos observar como cada modelo pronostica el valor para un día futuro

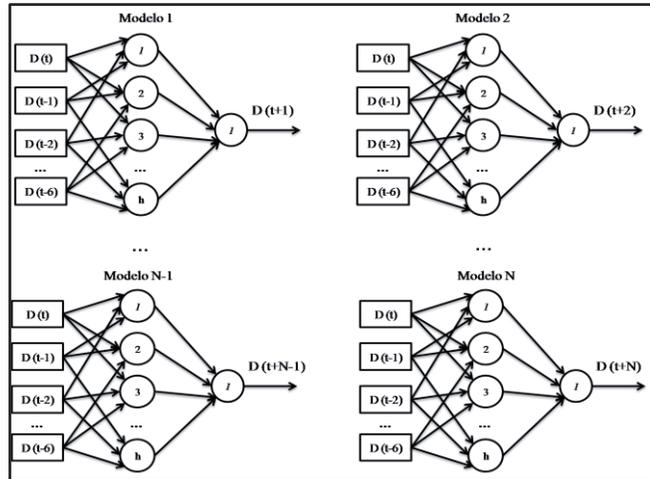


Fig 6.2 Modelo de pronóstico iterativo.

6.3. Modelo de pronóstico múltiples entradas y múltiples salidas utilizando redes neuronales de base radial

Para este modelo se utiliza una arquitectura distinta de red neuronal a la empleada por los dos modelos anteriores, ya que esta posee D nodos de salidas, donde D corresponde al número deseado de días a pronosticar. De esta manera si deseamos pronosticar el valor cuota para los próximos 5 días, se debe implementar una red neuronal con 5 nodos de salida, en donde cada nodo corresponderá al valor cuota estimado de uno de los días.

En este caso la dimensión de la partícula viene dado por la ecuación (22), y la posición de cada partícula viene dada por la ecuación (23).

$$\mathcal{G} = n \cdot 2h + s * h \quad (22)$$

$$X = [t_{11}, \dots, t_{1h}, t_{21}, \dots, t_{2h}, \dots, t_{n1}, \dots, t_{nh}, d_{11}, \dots, d_{1h}, d_{21}, \dots, d_{2h}, \dots, d_{n1}, \dots, d_{nh}, w_{11}, \dots, w_{1h}, w_{21}, \dots, w_{2h}, \dots, w_{s1}, \dots, w_{sh}] \quad (23)$$

, donde n corresponde a la cantidad de nodos de entrada, h corresponde a la cantidad de nodos ocultos presentes en la capa oculta y s a la cantidad de nodos de salida de la red, es decir al número de días a pronosticar.

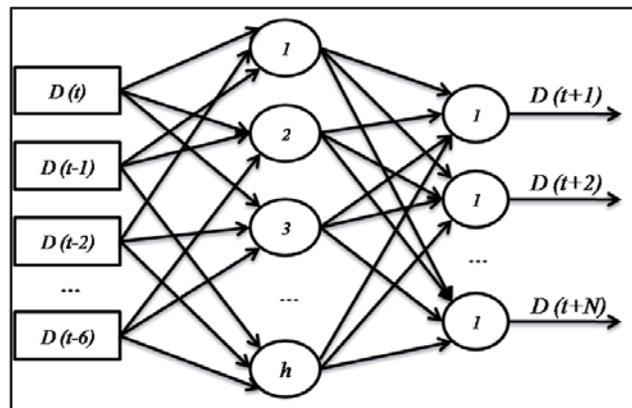


Fig 6.3 Modelo de pronóstico múltiples entradas y múltiples salidas

7. Experimentos

En esta sección se describen los experimentos realizados para determinar la mejor topología de red neuronal para cada modelo de pronóstico, el cual debe ser capaz de pronosticar el valor cuota del Fondo A para varios días futuros.

Los distintos pronosticadores neuronales propuestos se implementaron y probaron en el entorno de desarrollo MATLAB 7.12.0. Como forma de comparar el rendimiento de los pronosticadores propuestos, se implementaron casos de pruebas para cada modelo, donde la bondad de las soluciones se evaluó en base a las siguientes métricas:

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

$$MAPE = \frac{\sum_{i=1}^n \left| \frac{Y_i - \hat{Y}_i}{Y} \right|}{N} \times 100 \quad (24)$$

Donde Y es la salida deseada, \hat{Y} es la salida estimada y N el tamaño de la muestra. El Error Porcentual Absoluto Medio es una medida de precisión comúnmente usado en series temporales. Representa la precisión en forma de porcentaje.

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

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (Y_i - \hat{Y}_i)^2} \quad (25)$$

Donde Y es la salida deseada, \hat{Y} es la salida estimada y N el tamaño de la muestra.

- **Coefficiente de Determinación (R^2).**

El Coeficiente de Determinación mide el porcentaje de variabilidad en Y que se puede explicar con el conocimiento de la variabilidad (diferencias) en la variable independiente X (26).

$$R^2 = 1 - \frac{\sum (Y_i - \hat{Y}_i)^2}{\sum (Y_i - \bar{Y})^2} \quad (26)$$

Donde Y es la salida deseada, \hat{Y} es la salida estimada por el modelo.

- **Sentido de la Tendencia.**

Esta métrica determina el porcentaje de aciertos en el sentido de la tendencia del valor cuota, es decir si el valor cuota observado va a alza al siguiente día, el valor estimado debiese imitar este comportamiento. Se determina en base al siguiente algoritmo:

```
cont=0;
for i=1:N
    if ((Yobs(i)-Yobs(i+1))>0&&(Yest(i)-Yest(i+1))>0) || ((Yobs(i)-Yobs(i+1))<0&&(Yest(i)-Yest(i+1))<0)
        cont=cont+1;
    end
end
Sentido=cont/(N-1);
```

Done N corresponde a la cantidad de datos, Yobs a la salida deseada y Yest a la salida estimada por el modelo neuronal.

A continuación se otorga una descripción de los datos que serán utilizados como entrada a la red neuronal, luego se presentan los resultados de los experimentos realizados y finalmente se comparan los resultados obtenidos por los diferentes modelos.

7.1. Datos de entrada

Para la implementación del pronosticador neuronal se utilizaron como datos de entrada a la red los valores cuotas diarios del fondo A de la AFP CUPRUM, datos públicos disponibles en el sitio web de la Superintendencia de Pensiones.

Para la etapa de entrenamiento se utilizaron 245 datos correspondientes a los valores cuotas diarios entre los meses de enero y agosto. Para la fase de pruebas se emplearon 30 datos, que corresponden a los valores cuotas del mes de septiembre. Adicionalmente se utilizaron 77 datos para una etapa de extrapolación de resultados, correspondientes a los valores cuotas diarios entre los meses de octubre y el 16 de diciembre.

Los datos son normalizados con el objetivo de que estos queden en un rango acotado entre 0 y 1 manteniendo las relaciones existentes entre los patrones de entrada original. Para conseguir la acotación, cada dato será dividido por el valor mayor del conjunto.

Luego de que se tienen los datos procesados estos son desfasados según el número de días deseados. Este proceso consiste en convertir el vector con los datos de los valores cuota diarios a una versión retardada o desplazada, con el objeto de tener una matriz de regresión de nuestra variable explicativa que servirá como datos de entrada a nuestra red. El número de días de desfase corresponderá a la cantidad de días previos al día que se desea pronosticar, en nuestro caso es de 7 días.

7.2. Resultados

Para determinar la topología de red neuronal que mejor describa las relaciones subyacentes existentes entre los datos de la serie de tiempo para cada modelo, se realizaron una serie de pruebas para cada configuración del estimador neuronal.

Para determinar el mejor estimador, no sólo se tomó en cuenta el error obtenido, sino que además se considera el sentido de la predicción, es decir, evaluar si nuestro valor cuota estimado sigue la tendencia de alza o baja del valor cuota observado. Este último criterio resulta de gran importancia ya que no es de mucha utilidad saber el valor exacto de la cuota para cierto día, sino que es más importante conocer si esta va al alza o a la baja, ya que producto de este comportamiento, el fondo de pensión produce rentabilidades negativas o positivas, información crucial para tomar la decisión de cambio de fondo.

El rendimiento de cada estimador neuronal propuesto es evaluado en base a los resultados obtenidos en la etapa de pruebas, ya que de esta manera comprobaremos su capacidad de generalización, es decir, la capacidad de respuesta a valores que no fueron vistos en la etapa de entrenamiento

El primer paso a realizar fue identificar la función de activación más adecuada para determinar la salida deseada. Para esto se implementaron distintas configuraciones de redes

neuronaes, diferenciándose en la función de activación y la cantidad de nodos ocultos presentes, a las cuales se les realizaron 30 pruebas a cada una.

Las funciones de base radial evaluadas para la activación de las neuronas artificiales son las siguientes:

- Función Gaussiana
- Función Mexican Hat
- Función Morlet Wavelet
- Función Multicuadrática inversa

La siguiente tabla muestra el mejor resultado obtenido por cada configuración en la etapa de prueba.

	Gaussiana			MexicanHat		
	MAPE	R2	Sentido	MAPE	R2	Sentido
H=4	0,3173	0,8756	66,66%	0,3437	0,8921	66,66%
H=6	0,3199	0,8885	66,66%	0,2942	0,8930	60,00%
H=8	0,3614	0,8624	60,00%	0,2960	0,8982	63,33%
H=10	0,2835	0,8922	63,33%	0,3041	0,8922	60,00%
H=12	0,3263	0,8791	63,33%	0,2914	0,8791	66,66%
H=14	0,3189	0,8889	60,00%	0,3528	0,8889	63,33%
H=16	0,3250	0,8797	53,33%	0,2519	0,8797	63,33%
Promedio	0,3218	0,8809	61,90%	0,3049	0,8976	62,85%

	MorletWavelet			Multicuadratica		
	MAPE	R2	Sentido	MAPE	R2	Sentido
H=4	0,3466	0,8646	56,66%	0,2942	0,8921	50,00%
H=6	0,3599	0,8902	53,33%	0,3018	0,8930	50,00%
H=8	0,3158	0,8931	70,00%	0,3249	0,8982	50,00%
H=10	0,2895	0,8711	63,33%	0,2846	0,8922	63,33%
H=12	0,3148	0,8849	66,66%	0,2947	0,8791	56,66%
H=14	0,3049	0,9104	60,00%	0,3233	0,8889	56,66%
H=16	0,3047	0,8937	63,33%	0,3027	0,8797	60,00%
Promedio	0,3194	0,8868	61,90%	0,3074	0,8914	55,24%

Tabla 7.1. Resultados obtenidos por las distintas funciones de activación de base radial

En la tabla 7.1 se puede observar que la función Mexican Hat es la que otorga el mejor resultado en términos de error y acierto del sentido de la tendencia. El valor promedio obtenido para cada métrica de calidad por esta función de activación fue de un 0,304% de MAPE, un coeficiente de determinación de 0,8914 y un 62% de acierto en la tendencia, los cuales son superiores al promedio obtenido por el resto de las funciones, por lo tanto esta función será utilizada para los estimadores neuronales de cada modelo.

7.2.1. Pronóstico Iterativo

Producto de que el número de neuronas ocultas determina la complejidad del problema a resolver, se evaluó el rendimiento del estimador neuronal con distinta cantidad de nodos ocultos. Cada configuración fue sometida a 30 experimentos registrando sus resultados. La capa de entrada de la red contiene 7 nodos de entrada, se utilizó un enjambre de 40 partículas y se realizaron 100 iteraciones, estos valores fueron determinados en base a la técnica de ensayo y error. La tabla 7.2 muestra el mejor resultado obtenido por cada configuración en la etapa de prueba, donde H representa la cantidad de nodos ocultos.

	MexicanHat			
	MAPE	RMSE	R2	Sentido
H=4	0,3235	0,0049	0,8786	53,33%
H=6	0,2952	0,0046	0,8934	60,00%
H=8	0,2974	0,0045	0,8982	63,33%
H=10	0,3055	0,0045	0,8963	60,00%
H=12	0,2933	0,0046	0,8933	66,66%
H=14	0,2926	0,0046	0,8916	66,66%
H=16	0,2531	0,0040	0,9169	63,33%

Tabla 7.2. Resultados obtenidos usando distintas cantidades de nodos ocultos.

Como se puede observar en la tabla 7.2 el mejor resultado se obtuvo 14 nodos ocultos, ya que es la configuración que presenta un mejor balance entre el error obtenido y el acierto de la tendencia. En la figura 7.1 se muestra el comportamiento del MAPE a través de las iteraciones para las distintas configuraciones de nodos ocultos.

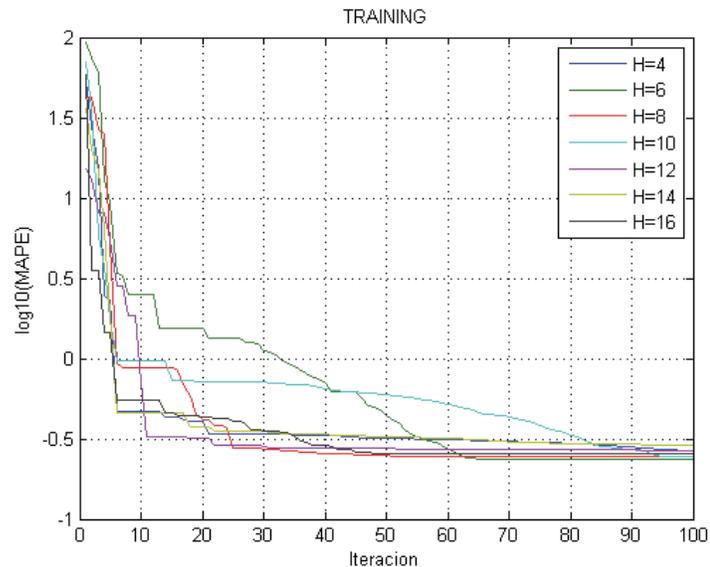


Figura 7.1 Decaimiento del error a través de las iteraciones.

En la figura 7.2 se presentan la gráfica comparativa entre los valores cuotas observados y estimados. En la figura 7.3 se muestra la correlación lineal existente entre ambos, en base a los resultados obtenidos por el mejor estimador en la etapa de pruebas.

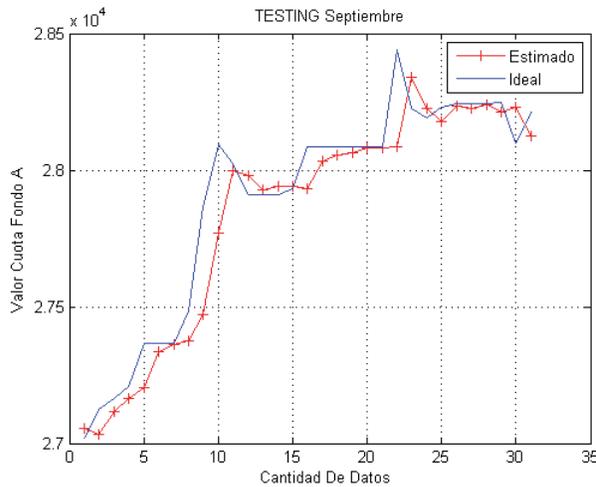


Figura 7.2 Datos estimados V/S observados.

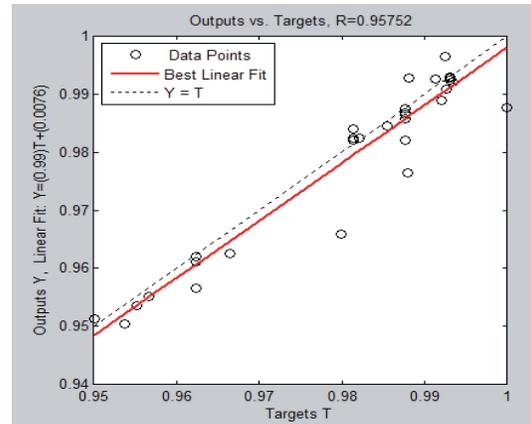


Figura 7.3 Correlación entre los datos observado y estimado.

7.2.2. Pronóstico Directo

Dado que en este tipo de pronóstico es necesario determinar un estimador neuronal para cada día futuro a pronosticar, es necesario encontrar la cantidad de nodos ocultos para cada estimador, ya que la complejidad de cada uno es distinta. Cada configuración fue sometida a 30 experimentos registrando sus resultados. La capa de entrada de la red se compone de 7 nodos, se utilizó un enjambre de 40 partículas y se realizaron 100 iteraciones, estos valores fueron determinados en base a la técnica de ensayo y error al igual que en el modelo anterior.

La cantidad de días futuros a pronosticar es de 5 días, por lo que se tienen que determinar 5 estimadores distintos para realizar el proceso de pronóstico directo. Por el hecho de que este modelo ocupa el mismo estimador que el modelo anterior para el pronóstico del primer día futuro, los resultados de este día estimado no serán incluidos ya que fueron presentados en el punto anterior. La siguiente tabla muestra el mejor resultado obtenido por cada configuración de cada estimador en la etapa de prueba.

	Día 2				Día 3			
	MAPE	RMSE	R2	Sentido	MAPE	RMSE	R2	Sentido
H=4	0,4429	0,0062	0,7689	58,62%	0,5512	0,004	0,6292	64,28%
H=6	0,4342	0,0063	0,7675	58,62%	0,5369	0,0074	0,6232	71,43%
H=8	0,4153	0,0063	0,7627	62,06%	0,5488	0,0072	0,6429	67,86%
H=10	0,4342	0,0062	0,7724	68,96%	0,5194	0,0076	0,6027	64,28%
H=12	0,4308	0,0062	0,7746	68,96%	0,5635	0,008	0,5619	67,86%
H=14	0,4345	0,0064	0,7584	68,96%	0,5596	0,008	0,5664	67,86%
H=16	0,4546	0,0068	0,7223	58,62%	0,5263	0,0076	0,6062	64,28%

	Día 4				Día 5			
	MAPE	RMSE	R2	Sentido	MAPE	RMSE	R2	Sentido
H=4	0,6583	0,0084	0,4337	59,25%	0,7037	0,0091	0,1816	65,38%
H=6	0,6321	0,0083	0,4386	55,55%	0,756	0,009	0,0082	61,54%
H=8	0,6769	0,0085	0,4175	66,66%	0,7225	0,0094	0,1304	65,38%
H=10	0,7011	0,009	0,3487	62,96%	0,7667	0,0094	0,1131	53,85%
H=12	0,7018	0,0088	0,3712	59,25%	0,7411	0,0095	0,0949	61,54%
H=14	0,6629	0,0085	0,415	66,66%	0,735	0,0095	0,1022	69,23%
H=16	0,6607	0,0085	0,4129	70,37%	0,658	0,0093	0,1318	65,38%

Tabla 7.3 Resultados obtenidos usando distintas cantidades de nodos ocultos para los distintos días.

Como se puede observar el mejor resultado se obtuvo 14 nodos ocultos para el primer día, 12 nodos ocultos para el segundo día, 6 nodos ocultos para el tercer día, 16 nodos ocultos para el cuarto día y 14 nodos ocultos para el quinto día, ya que dicha configuración presenta un mejor balance entre el error y el acierto de la tendencia. El promedio de error obtenido en los 5 días a pronosticar fue de un 0.5312% de MAPE y un 69.33% de acierto en el sentido de la tendencia. En la figura 7,4 se muestra el comportamiento del MAPE a través de las iteraciones para las distintas configuraciones de nodos ocultos.

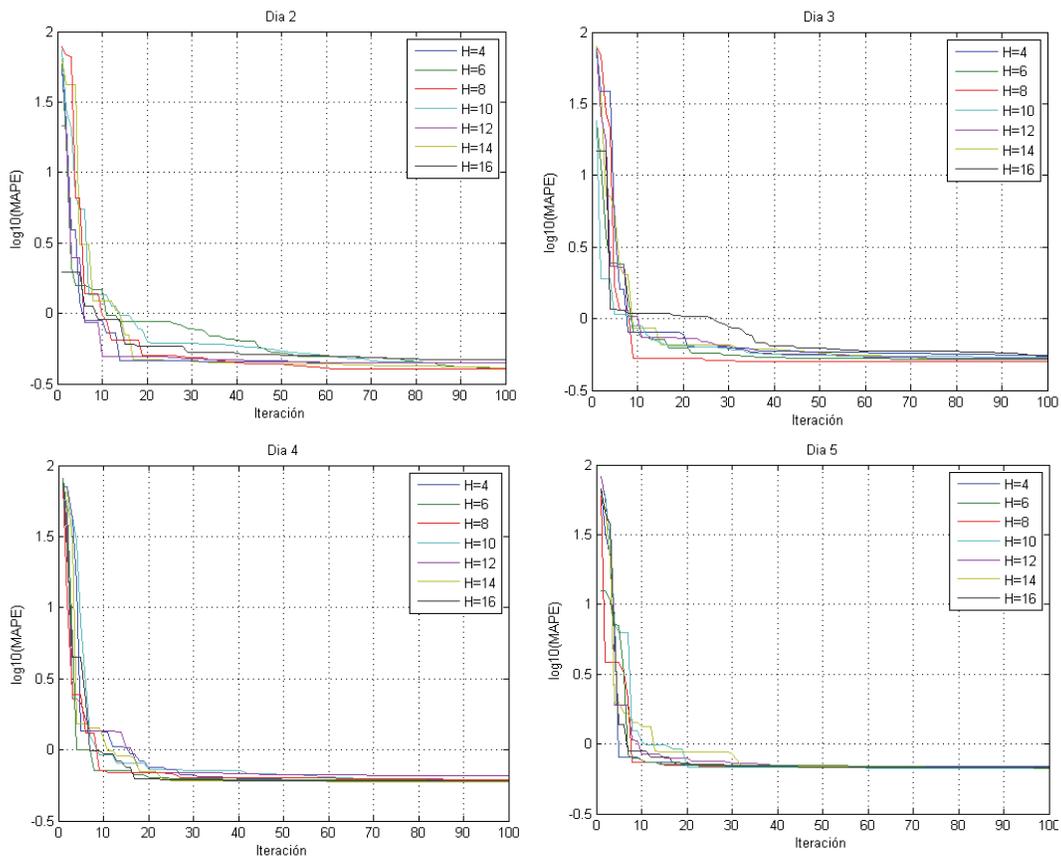


Figura 7.4 Decaimiento del error a través de las iteraciones para cada día estimado.

En la figura 7.5 se muestra como incrementa el error a medida que se avanza en el horizonte a estimar, y se puede observar que tiene un comportamiento casi lineal.

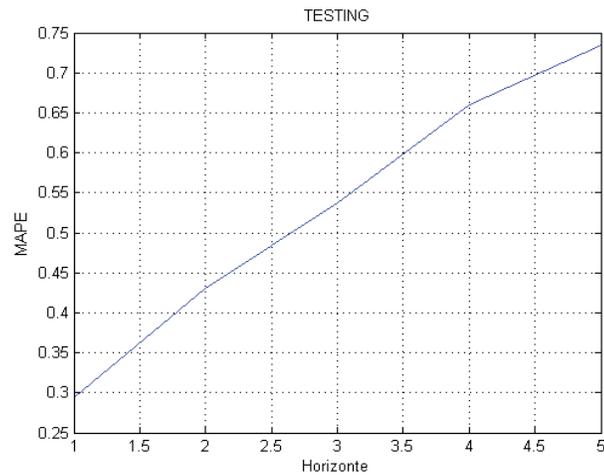


Figura 7.5 Evolución del MAPE a medida que se avanza en el horizonte a estimar.

En la figura 7.6 se presentan la gráfica comparativa entre los valores cuotas observados y estimados, y en la figura 7.7 la correlación lineal existente entre ambos en la etapa de prueba para cada día futuro a pronosticar.

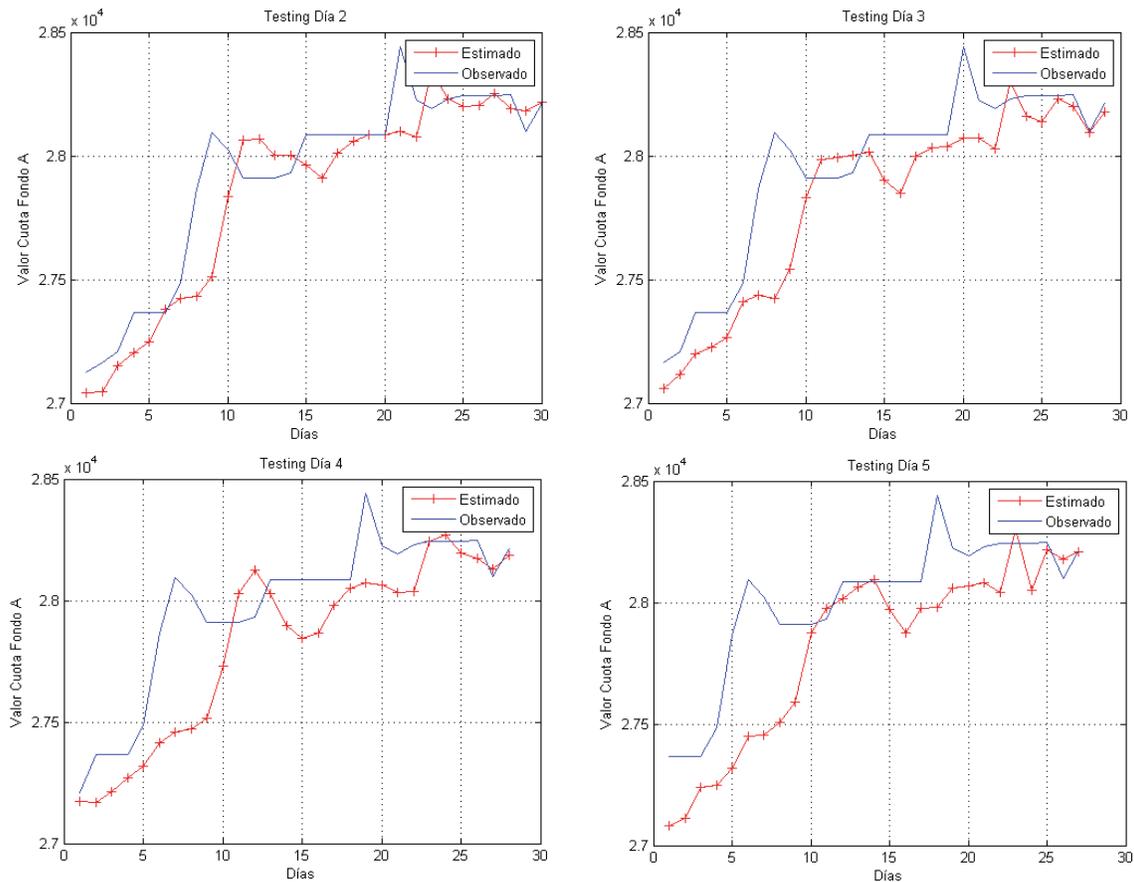


Figura 7.6 Datos estimados V/S observados para cada día estimado

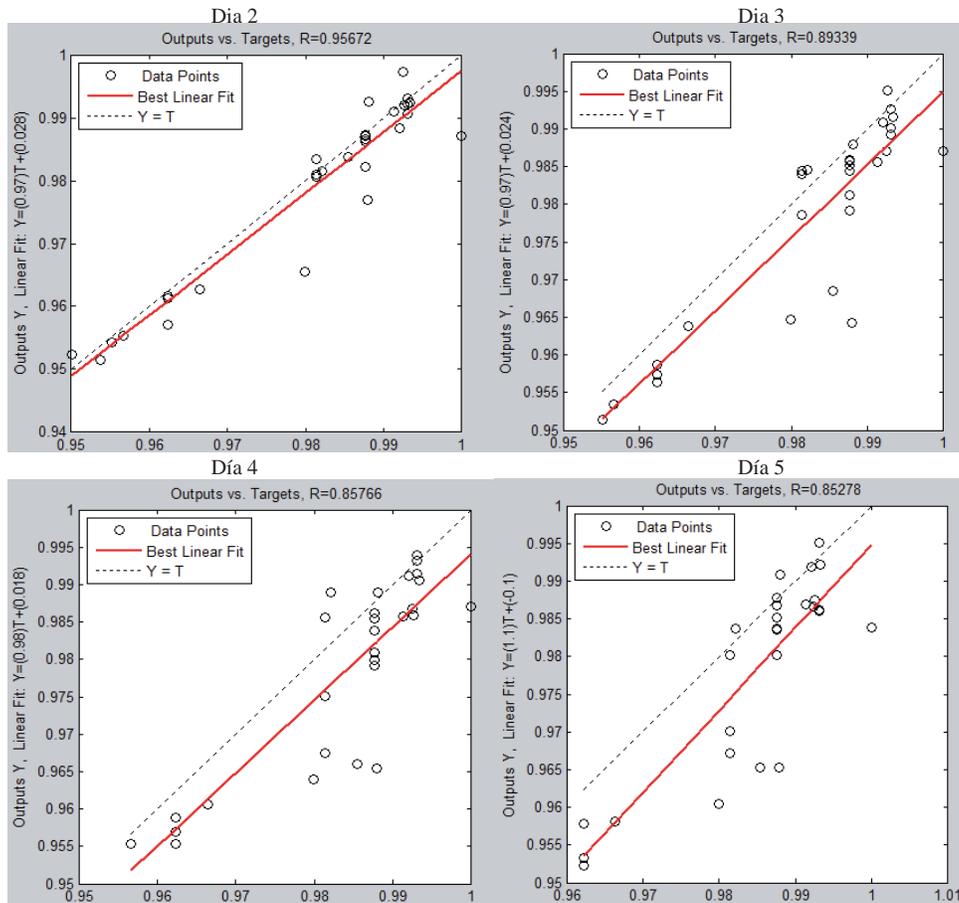


Figura 7.7 Correlación entre los datos observado y estimado para cada día estimado

7.2.3. Pronóstico Múltiples Entradas y Múltiples Salidas

Ya que la complejidad del estimador neuronal para este modelo es distinta a la de los dos modelos anteriores, se evaluó el rendimiento del estimador neuronal con distinta cantidad de nodos ocultos. Cada configuración fue sometida a 30 experimentos registrando sus resultados. La capa de entrada de la red contiene 7 nodos de entrada, se utilizó un enjambre de 30 partículas y se realizaron 100 iteraciones. La siguiente tabla muestra el mejor resultado obtenido por cada configuración en la etapa de prueba.

	Día 1			Día 2			Día 3		
	MAPE	R2	Sentido	MAPE	R2	Sentido	MAPE	R2	Sentido
H=4	0.4268	0.8424	59,26%	0.6006	0.6154	48,15%	0.6263	0,4689	48,15%
H=6	0,5290	0,7063	74,07%	0,6815	0,4683	62,96%	0,7593	0,2022	62,96%
H=8	0,7889	0,5343	62,96%	0,8135	0,2940	66,66%	0,9385	-0,0075	55,55%
H=10	0,5383	0,6197	74,07%	0,7351	0,4139	59,26%	0,7445	0,1771	74,07%
H=12	0,3510	0,8820	70,37%	0,6371	0,5546	66,66%	0,8351	0,1754	66,66%
H=14	0,4979	0,7603	66,66%	0,4593	0,6975	62,96%	0,5863	0,4831	62,96%
H=16	0,3523	0,8738	59,26%	0,5724	0,6510	48,15%	1,0954	-0,2808	37,04%

	Día 4			Día 5			Promedios	
	MAPE	R2	Sentido	MAPE	R2	Sentido	MAPE	Sentido
H=4	0,8127	0,0409	59,26%	0,7598	-0,1343	62,96%	0,6452	55,56%
H=6	0,8632	-0,2226	59,26%	0,9916	-0,9030	62,96%	0,7649	64,44%
H=8	1,0875	-0,5117	48,15%	1,1017	-1,0866	70,37%	0,9460	60,74%
H=10	0,8916	-0,2873	74,07%	1,0270	-1,1589	59,26%	0,7873	68,15%
H=12	0,6904	0,0467	51,85%	0,5831	0,2160	59,26%	0,6194	62,96%
H=14	0,9014	-0,1815	66,66%	0,8390	-0,4202	62,96%	0,6568	64,44%
H=16	0,7190	0,1307	62,96%	1,3615	-1,6707	55,55%	0,8201	52,59%

Tabla 7.4 Resultados obtenidos usando distintas cantidades de nodos ocultos para los distintos días.

Como se puede observar el mejor resultado se obtuvo con un estimador neuronal configurado con 12 nodos ocultos, ya que dicha configuración es la que presenta un mejor balance entre el error y el acierto de la tendencia promedio para los 5 días, obteniendo un MAPE promedio de 0,6194% y un 62,69% de acierto en la tendencia. En la figura 7.8 se muestra el comportamiento del MAPE a través de las iteraciones para las distintas configuraciones de nodos ocultos.

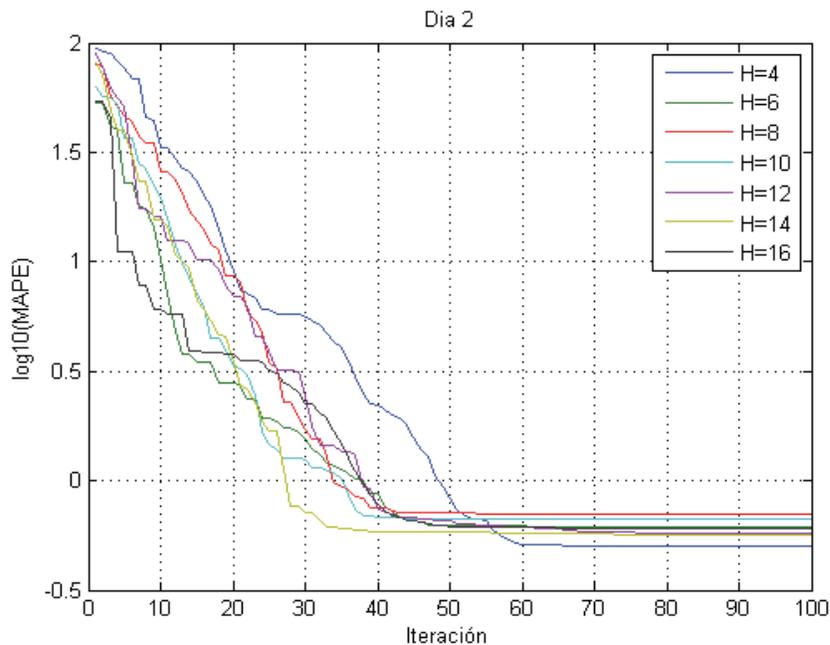


Figura 7.8 Decaimiento del error a través de las iteraciones para cada día estimado.

En la figura 7.9 se muestra el comportamiento del error a medida que se avanza en el horizonte a estimar, y se puede observar que a diferencia del modelo anterior, el error no crece a medida que se avanza en los días a estimar, si no que presenta alzas y bajas a medida que se avanza en el horizonte.

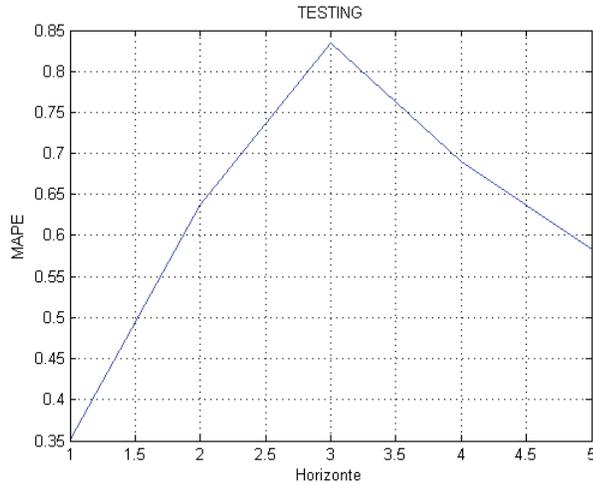
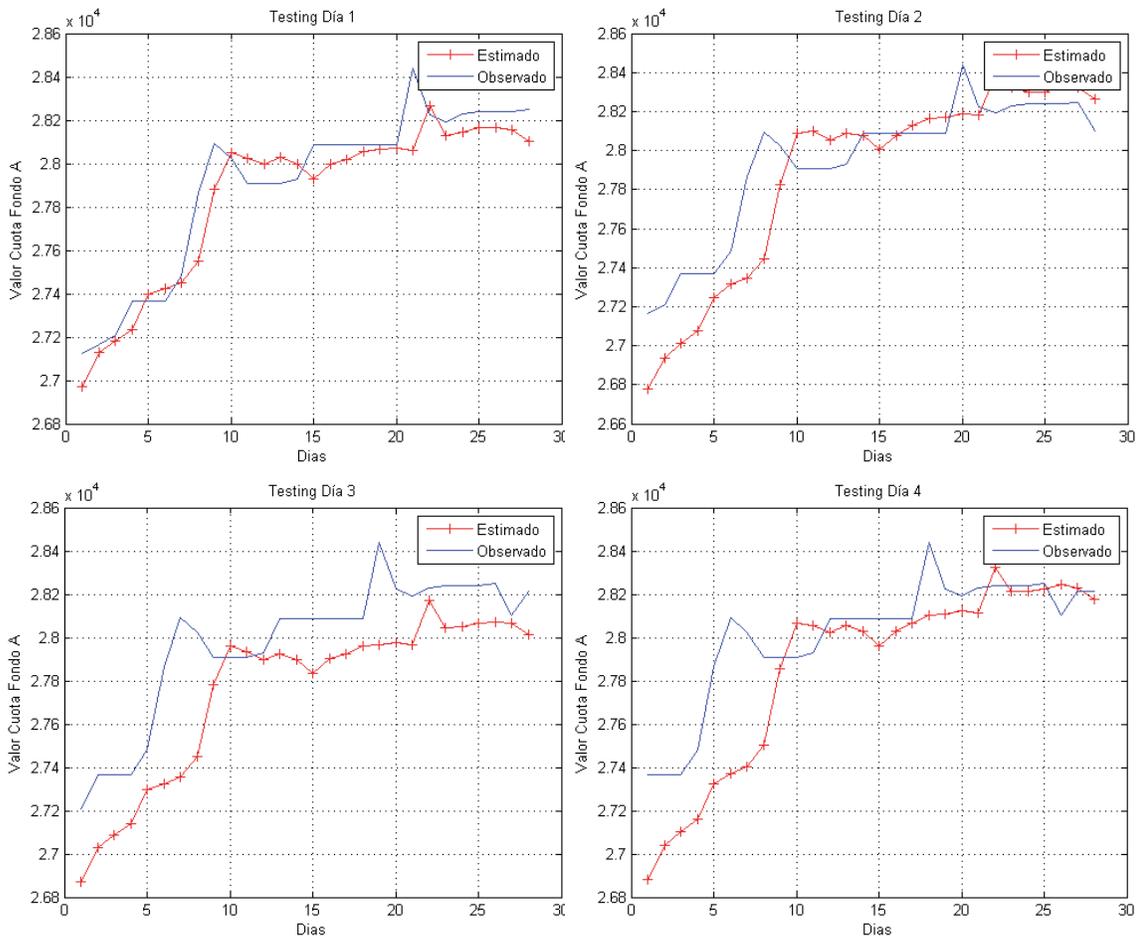


Figura 7.9 Evolución del MAPE a medida que se avanza en el horizonte a estimar.

En la figura 7.10 se presenta la gráfica comparativa entre los valores cuota observados y estimados otorgados por el presente modelo para cada día pronosticado del horizonte deseado.



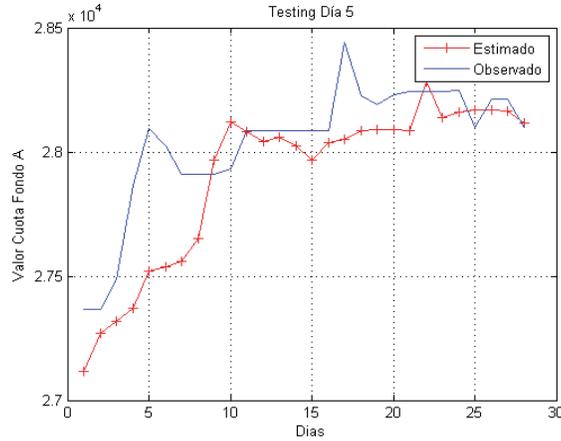


Figura 7.10 Datos estimados V/S observados para cada día estimado

7.3. Extrapolación de los resultados

Como forma de comparar el rendimiento de los modelos propuestos en el pronóstico del valor cuota para varios días futuros, se realiza una etapa de extrapolación de los datos, la cual consiste en pronosticar el valor cuota para 5 días futuros utilizando los datos destinados para esta, los cuales no han sido utilizados. Se eligieron 12 días y para cada uno se pronosticaron el valor cuota de los 5 días futuros. En la tabla 7.5 se muestran los resultados promedios obtenidos por cada modelo en esta etapa, pudiendo inferir que el pronóstico Directo es la estrategia más adecuada para la presente problemática. En la tabla 7.6 se observan los valores obtenidos por esta estrategia en cada horizonte pronosticado.

	Pronóstico Iterativo	Pronóstico Directo	Pronóstico M.E.M.S.
$\overline{\text{MAPE}}$	0.5955	0.5811	1.6772
$\overline{\text{Sentido}}$	43.18%	56.81%	47.72%

Tabla 7.5 Resultados promedios obtenidos por cada modelo

Periodo	MAPE	Sentido	Periodo	MAPE	Sentido
01/10/13 - 05/10/13	0,1464%	100%	03/11/13 - 07/11/13	0,3398%	75%
07/10/13 - 11/10/13	0,3699%	50%	10/11/13 - 14/11/13	0,1628%	25%
12/10/13 - 17/10/13	0,4737%	75%	17/11/13 - 07/11/13	1,0431%	50%
17/10/13 - 21/10/13	0,2592%	50%	24/11/13 - 07/11/13	0,2030%	50%
21/10/13 - 26/10/13	1,4989%	75%	01/12/13 - 07/12/13	0,6887%	25%
27/10/13 - 31/10/13	0,4405%	50%	08/12/13 - 07/12/13	0,6364%	50%

Tabla 7.6 Resultados obtenidos por el pronóstico directo en la etapa de extrapolación.

7.4. Discusión de Resultados

En base a los resultados obtenidos en los experimentos realizados se apreciaron las ventajas y desventajas de cada modelo propuesto. El pronóstico iterativo por el hecho de utilizar datos estimados para el pronóstico de los siguientes días futuros, va acarreado los errores de estas estimaciones a medida que se avanza en el horizonte a predecir. Para este problema lo anterior presenta es una gran desventaja, ya que este tipo de series son muy sensibles y un pequeño error puede ser determinante a la hora de tratar de encontrar las relaciones presentes en los datos de la serie. Es por esta razón que en los resultados obtenidos en el proceso de extrapolación de datos, las estimaciones del valor cuota tendían a una recta a medida que se avanzaba en el horizonte (figura 7.11), ya que los errores acumulados hacían perder las relaciones existentes entre los datos.

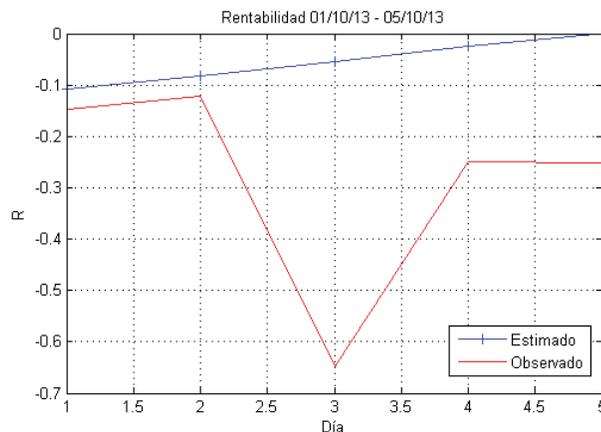


Figura 7.11 Resultado extrapolación pronóstico iterativo

En el caso del pronóstico de múltiples entradas y múltiples salidas se pudo observar que esta estrategia no sufre del aumento del error a medida que avanza en el horizonte a estimar, como sucede en el caso de las otras dos estrategias, pero los resultados mostraron que la gran ventaja de este modelo, evitar el supuesto de independencia condicional hecha por la estrategia directa, terminó convirtiéndose en una desventaja en la presente problemática. Se observa que este estimador neuronal determina la misma relación existente entre los valores cuotas del horizonte a estimar para todos los pronósticos realizados, lo cual es un gran inconveniente ya que dada la gran complejidad del problema, la alta caoticidad y la gran incertidumbre presente en la serie de tiempo, la relación existente entre los valores cuotas va cambiando en cada periodo y muchas veces de manera drástica.

Por otra parte el pronóstico directo demostró su gran flexibilidad a la hora de realizar estimaciones para múltiples días futuros, ya que permite determinar distintos estimadores neuronales para cada día a pronosticar, lo cual se tradujo en menores errores en los pronósticos. Su gran desventaja radica en la gran cantidad de tiempo dedicado para configurar y determinar cada estimador, ya que se debe definir un estimador por cada día a pronosticar. Además, a medida que se avanza en el horizonte de pronóstico, el error obtenido en la estimación va aumentando, por lo cual la cantidad de días futuros que se pueden estimar es reducida. En la figura 7.12 se muestra el mejor resultado obtenido en la

etapa de extrapolación por el pronóstico directo, obteniendo un 100% de acierto en el sentido de la tendencia.

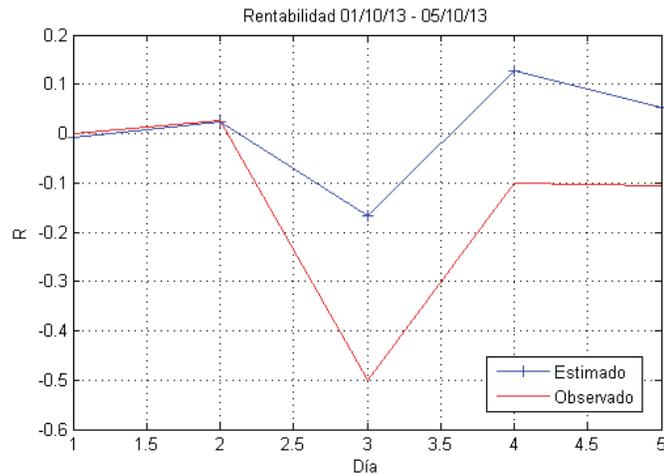


Figura 7.12 Mejor resultado obtenido por el pronóstico directo

En base a los resultado obtenidos por cada modelo y al análisis y conclusiones desprendidos del proceso de experimentación, se deduce que el mejor modelo de pronóstico para la presente problemática corresponde al Pronostico Directo empleando redes neuronales de base radial con función de activación Mexican Hat y algoritmo de aprendizaje PSO adaptativo. Por ende, este es el modelo que se propone en esta memoria de título. Las estimaciones realizadas son entregadas en términos de rentabilidad, ya que el valor cuota en si no entrega mayor información que sirva de apoyo a la toma de decisión de cambio de fondo, si no que es la rentabilidad otorgada por este, la que permite decidir si es conveniente realizar el cambio de fondo.

Para tomar la decisión sobre el cambio de fondo se propone el siguiente criterio. En el caso que la rentabilidad estimada del fondo A por el modelo de pronostico directo sea superior a la rentabilidad otorgada por el fondo E en el último mes previo al pronostico, se aconseja cambiarse al fondo A o mantenerse en este en caso de que ya se esté cotizando en este fondo. En caso contrario, si la rentabilidad estimada para el fondo A es inferior a la otorgada por el fondo E, es aconsejable cambiarse al fondo E o mantenerse en este en caso de que ya se esté cotizando en el.

En la figura 7.12 se aprecia que la estimación realizada por el modelo propuesto indica que la rentabilidad otorgada por el fondo A para la primera semana de octubre es de un 0,052% mientras que la rentabilidad real otorgada por el fondo E en la AFP CUPRUM para el mes de septiembre es de un 0,79%, por lo que en este caso se aconseja afiliarse al fondo E dado que lo más probable es que este otorgue una mayor rentabilidad en dicho periodo.

Cabe destacar que producto de la gran complejidad del problema el modelo propuesto no responde adecuadamente para todos los periodos de pronóstico, ya que como se puede observar en la tabla 7.6 existen periodos en los cuales la estimación es de gran calidad, pero en otros periodos los pronósticos obtenidos son deficientes.

8. Conclusiones

Gracias al desarrollo del presente proyecto de investigación, se comprendió el funcionamiento de las redes neuronales y algoritmos de inteligencia de enjambres, pero lo más importantes es que se adquirió un entendimiento y una visión global de los sistemas de aprendizaje automático y de los modelos de pronóstico.

Se emplearon distintas estrategias para el pronóstico del valor cuota para varios días futuros y se pudo evidenciar y apreciar, en base a los distintos experimentos, las ventajas y desventajas de cada técnica. El pronóstico directo resultó ser la estrategia que obtuvo mejores resultados en el pronóstico del valor cuota para 5 días futuros. El hecho de generar un modelo para cada día futuro a pronosticar permite disminuir considerablemente los errores conseguidos, aunque el tiempo dedicado a la calibración para cada modelo es muy elevado. Los errores obtenidos aumentan notablemente a medida que se avanza en el horizonte del pronóstico, lo cual impide hacer un buen pronóstico para horizontes mayores.

Realizar la etapa de entrenamiento en una red neuronal es una tarea bastante costosa, por el hecho de que para estas técnicas evolutivas no existe una metodología exacta para determinar los parámetros que mejor se adecuen a la complejidad del problema. Por lo que sólo nos queda recurrir a la técnica del “ensayo y error” para determinar dichos parámetros, la cual requiere mucho tiempo.

El hecho de que una gran parte del capital acumulado en el fondo A es invertida en distintos mercados de valores hace que el pronóstico de estas series de tiempo sea un problema demasiado complejo. Pese al alto ruido existente y a la naturaleza no estacionaria y determinísticamente caótica de las series de tiempo de valores cuotas, las redes neuronales junto al algoritmo PSO demostraron su gran capacidad de encontrar las relaciones presentes entre los datos, logrando describir parte de las relaciones subyacentes existentes entre los datos de la serie. Se obtuvieron pronósticos con un 0.5312% de MAPE promedio en los 5 días futuros a estimar, con un 69.33% de acierto en la tendencia. Incluso se obtuvieron pronósticos con aciertos del 100% en el sentido de la tendencia para la etapa de extrapolación, pero a su vez se registraron pronósticos con aciertos muy bajos en la tendencia para ciertos periodos. Este comportamiento se debe a la gran variación aleatoria del día a día presente en este tipo de series, ya sea por temas políticos, sociales o incluso ambientales lo que conlleva a que existan periodos más caóticos y con mayor ruido que otros.

El modelo propuesto no logra ser un sistema que por sí solo sea de real ayuda para los afiliados de las AFP, ya que se necesitan pronósticos más precisos, o de horizontes aún mayores, cosa que las estrategias de pronóstico no pudieron cumplir por la gran complejidad del problema. Con la incertidumbre presente en los resultados obtenidos se pueden conseguir comportamientos contrarios a los deseados, donde no se aprovechen rentabilidades positivas o no se prediga el momento en que las rentabilidades comiencen a bajar. Pese a lo anterior, la herramienta propuesta resulta muy útil para el apoyo y complemento a otros modelos de pronósticos o a técnicas de trading, además de otorgar gran información del comportamiento de estas series, y presenta una fuerte base para futuros proyectos de investigación en torno al pronóstico del valor cuota en las AFP.

Los objetivos del proyecto se pudieron cumplir a cabalidad, donde se profundizó y se explicó acerca de la tecnología utilizada, se definieron la estructura y parámetros del

estimador neuronal que mejor se ajustaban a los requerimientos del problema, y finalmente se evaluaron y compararon distintos modelos de pronóstico.

A futuro se pretende seguir investigando en algoritmos de aprendizaje automático y tratar de mejorar los resultados. Es posible que utilizando e integrando más de una técnica al modelo de pronóstico se pueda mejorar considerablemente los resultados, ya que cada método tiene sus pros y contras, pero al momento de generar un modelo híbrido entre distintas técnicas, hace que entre estas se compensen sus debilidades y potencien sus fortalezas.

9. Referencias

- [1] Yaser, A., Amir, A., "Introduction to financial forecasting" *Applied Intelligence*, 6(3):205-213, Julio 1996.
- [2] Kennedy, J., Eberhart, R., "A Discrete Binary Version of the Particle Swarm algorithm", *Systems, Man, and Cybernetics*, 5:4104-4108, Octubre 1997.
- [3] Kennedy, J., Eberhart, R., Shi, Y., "Swarm Intelligence", Morgan Kaufmann Publishers San Francisco, 2001.
- [4] Kennedy, J., Eberhart, R., "Particle Swarm Optimization", *Neural Networks*, 4:1942-1948, Diciembre 1995.
- [5] Shi, Y., Eberhart, R., "Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization", *Evolutionary Computation*, 1:84-88, Julio 2000.
- [6] Nekoukar, V., "A local linear radial basis function neural network for financial time-series forecasting", *Applied Intelligence*, 33(3):352-356, Diciembre 2010.
- [7] Lazzús, J., "Neural network-particle swarm modeling to predict thermal properties", *Mathematical and Computer Modelling*, 57(9-10):2408-2418, Mayo 2012.
- [8] Foix, C., Weber, R., "Pronóstico del precio del cobre mediante Redes Neuronales", *Revista Ingeniería de Sistemas*, 21:63-86, 2007.
- [9] Sermpinis, G., Theofilatos, K., "Forecasting foreign Exchange rates with adaptive neural networks using radial-basis functions and particle swarm optimization", *European Journal of Operational Research*, 225(3):528-540, Marzo 2012.
- [10] Zandehboundi, S., Ali, M., James L., "Prediction of condensate-to-gas for retrograde gas condensate reservoirs using artificial network with particle swarm optimization", *Energy Fuels*, 26(6):3432-3447, 2012.
- [11] Lee, C., Ko, C., "Time series prediction using RBF network with a nonlinear time-varying evolution PSO algorithm", *Neurocomputing*, 73(1-3):449-460, Diciembre 2009.
- [12] Zhang, W., Wang, J., "Short-term wind speed forecasting based on a hybrid model", *Applied Soft Computing*, 13(7):3225-3233, Julio 2013.
- [13] Luh, G., Lin, C., Lin, Y., "A binary particle swarm for continuum structural topology optimization", *Applied Soft Computing*, 11(2):2833-2844, Marzo 2011.
- [14] Lau, S., Lu, M., "Applying radial basis function neural networks to estimate next-cycle production rates in tunneling construction", *Tunnelling and Underground Space Technology*, 25(4):357-365, Julio 2010
- [15] Wu, A., Hsieh, W., "Neural network forecast of the tropical Pacific sea surface temperatures", *Neural Networks*, 19(2):145-154, Marzo 2006.
- [16] Guresen, E., Kayakutlu, G., "Using artificial neural network models in stock market index prediction", *Expert Systems with Applications*, 38(8):10389-10397, Agosto 2011.
- [17] Qing-liang, C., Chao-jiong, H., "Using particle swarm optimization algorithm in an artificial neural network to forecast the strength of paste filling material", *Journal of China University of Mining and Technology*, 18(4):551-555, Diciembre 2008.

- [18] Serrano, C., Martín, B., “Predicción de la Quiebra Bancaria Mediante el Empleo de Redes Neuronales Artificiales”, *Revista Española de Financiación y Contabilidad*, 23(74):153-176, Enero-Marzo 1993.
- [19] Feng, H., Chou, H., “Evolutional RBFNs prediction systems generation in the applications of financial time series data”, *Expert Systems with Applications*, 38(7):8285-8292, Julio 2011.
- [20] Ch, S., Mathur, S., “Particle swarm optimization trained neural network for aquifer parameter estimation”, *KSCE Journal of Civil Engineering*, 16(3):298-307, 2012.
- [21] Shafie, M., Moghaddam, M., “Price forecasting of day-ahead electricity markets using a hybrid forecast method”, *Energy Conversion and Management*, 52(5):2165-2169, Mayo 2011.
- [22] Shen, W., Guo, X., Wu, C., “Forecasting stock indices using radial basis function neural network optimized by artificial fish swarm algorithm”, *Knowledge-Based Systems*, 24(3):378-385, Abril 2011.
- [23] Ko, C., “Identification of non-linear systems using radial basis function neural network with time-varying learning algorithm”, *IET Signal Processing*, 6(2):91-98, Abril 2012.
- [24] Kitayam, S., yasuda, K., “Integrative optimization by RBF Network and particle Swarm Optimization”, *Electronics and communication in Japon*, 92(12):31-42, Diciembre 2009.
- [25] Li, J., Liu, X., “Melt index prediction by RBF neural network optimized with an MPSO-SA hybrid algorithm”, *Neurocomputing*, 74(5):735-740, Febrero 2011.
- [26] Mirjalili, S., Mohd, S., Moradian, H., “Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm” *Applied Mathematics and Computation*, 218(22):11125-11137, Julio 2012.
- [27] Rakitianskaia, A., Engelbrecht, A., “Training feedforward neural networks with dynamic particle swarm optimization” *Swarm Intell*, 6(3):233-270, Septiembre 2012.
- [28] Cover, T., “Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition.” *IEEE Transactions on Electronic Computers*, 14:326-334, 1965.
- [29] Khanesar, M., Teshnehlab, M., “A Novel Binary Particle Swarm Optimization.” *Control & Automation*, :1-6, Julio 2007.
- [30] Yuan, X., Nie, H., Su, A., Wang, L., Yuan, Y., “An improved binary particle swarm optimization for unit commitment problem” *Expert Systems with Applications*, 36(4):8049-8055, Mayo 2009.
- [31] Chuang, L., Tsai, J., Yang, C., “Binary particle swarm optimization for operon prediction” *Nucleic Acids Research*, 38(12), Abril 2010.
- [32] Shi, Y., Eberhart, R., “Parameter selection in particle swarm optimization” *Lecture Notes in Computer Science*, 1447:591-600, 1998.
- [33] Bai, Q., “Analysis of Particle Swarm Optimization Algorithm”, *Computer and Information Science*, 3(1):180-184, Febrero 2010.

- [34] De Gooijer, J., Hyndman, R., “25 years of time series forecasting” *International Journal of Forecasting*, 22:443-473, 2006
- [35] Palm F., Zellner, A., “To combine or not to combine? Issues of combining forecasts” *Journal of Forecasting*, 11(8):687-701, 1992.
- [36] 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.
- [37] Khashei, M., Bijari, M., Raissi Ardali, G., “Improvement of auto-regressive integrated moving average models using fuzzy logic and artificial neural networks (ANNs)” *Neurocomputing*, 72(4-6):956-967, 2009.
- [38] Lai, R., Fan, C., Huang, W., Chan, P., “Evolving and clustering fuzzy decision tree for financial time series data forecasting” *Expert Systems with Applications*, 36:3761-3773, 2009.
- [39] Kim, K., “Financial time series forecasting using support vector machines” *Neurocomputing*, 55:307-319, 2003.
- [40] Tay, F., Cao, L.J., “Modified support vector machines in financial time series forecasting” *Neurocomputing*, 48:847-861, 2002.
- [41] Wang, B., Huang, H., Wang, X., “A novel text mining approach to financial time series forecasting” *Neurocomputing*, 83:136-145, 2012.
- [42] Liang, X., Chen, R., Chen, Y., “Associating stock prices with web financial information time series based on support vector regression” *Neurocomputing*, 115:142-149, 2013.
- [43] Aznarte, J., Alcalá-Fdez, J., Benítez, J., “Financial time series forecasting with a bio-inspired fuzzy model” *Expert Systems with Applications*, 39:12302-12309, 2012.
- [44] Strozzi, F., Zaldívar, J., “Non-linear forecasting in high-frequency financial time series” *Physica A*, 353:463-479, 2005.
- [45] Sorjamaa, A., Hao, J., Reyhani, N., Ji, Y., Lendasse, A. “Methodology for longterm prediction of time series”, *Neurocomputing*, 70(16–18):2861–2869, 2007.
- [46] Ben Taieb, S., Bontempi, G., 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(8): 7067-7083, Junio 2012.
- [47] Bontempi, G., Ben Taieb, S., “Conditionally dependent strategies for multiple-step-ahead prediction in local learning”, *International Journal of Forecasting*, 27(3):689–699.
- [48] Ben Taieb, S., Sorjamaa, A., Bontempi, G. “Multiple-output modeling for multi-step-ahead time series forecasting”. *Neurocomputing*, 73(10-12):1950–1957, (2010).