

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA INFORMÁTICA

**REDES NEURONALES WAVELET CON PSO PARA  
PRONÓSTICO DE CAPTURAS MENSUALES DE ALTA  
Y BAJA FRECUENCIA DE SARDINAS EN LA ZONA  
NORTE DE CHILE**

**EDUARDO ANDRÉS MENDOZA CARVAJAL**

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

Enero 2014.-

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA INFORMÁTICA

**REDES NEURONALES WAVELET CON PSO PARA  
PRONÓSTICO DE CAPTURAS MENSUALES DE ALTA  
Y BAJA FRECUENCIA DE SARDINAS EN LA ZONA  
NORTE DE CHILE.**

**EDUARDO ANDRÉS MENDOZA CARVAJAL.**

Profesor Guía: **Nibaldo Rodríguez Agurto.**

Profesor Correferente: **José Rubio León.**

Carrera: **Ingeniería Civil Informática.**

Enero 2014.-

## *Dedicatoria*

Dedicada a mí madre por todo su cariño y apoyo incondicional, a mi padre que desde el cielo ha sido mi inspiración, a mi familia y amigos que han estado conmigo y me apoyaron en este largo camino.

## *Agradecimientos*

A Dios, mi familia, a la Universidad y profesor guía.

## Resumen

Esta memoria de título representa un modelo de pronóstico de Redes Neuronales Artificiales (RNA) basado en el algoritmo evolutivo Optimización por Enjambre de Partículas (PSO), para pronosticar el nivel de captura mensual de sardinas en la zona norte de Chile, que comprende el periodo entre Enero 1976 a Diciembre de 2007. Este pez es una de las especies pelágicas más importantes de nuestro país, es por esto que debido a la caída que sufrió en la población esta especie en 1983, la extracción de la sardina está sujeta a regulaciones. A consecuencia de esto es que el pronóstico de capturas toma un rol muy importante en el nivel de stocks, previo a la toma de decisiones. El modelo que se utilizó para dar solución a la problemática consistió en una Red Neuronal Wavelet (RNW) con función de activación Derivada de la Gaussiana, y una configuración de parámetros internos mediante la utilización del algoritmo PSO. Finalmente, los resultados indican que el modelo RNW con PSO, ajustado con peso dinámico, y con una topología de red configurada en cuatro nodos de entrada, tres nodos ocultos y un nodo de salida, presenta el mejor desempeño logrando un 97,56% de la varianza explicada y un 18,85% de SMAPE.

**Palabras claves:** Redes Neuronales Artificiales, Redes Neuronales Wavelet, Optimización por Enjambre de Partículas, Inteligencia de Enjambre.

## Abstract

This thesis represents a forecast model of Artificial Neural Networks (ANN) based on the evolutionary algorithm: Particle Swarm Optimization (PSO), to forecast the monthly catch level of sardines in the northern Chile that covers the period from January 1976 to December 2007. This fish is one of the most important pelagic species of our country, which is why due to the fall in the population that suffered this species in 1983; the extraction of sardine is subject to regulations. A consequence of this is that the forecast catches plays an important role in the stocks level, previous to making decisions. The model was used to solve the problem consisted of a Wavelet Neural Network (WNN) with Derivative of Gaussian activation function, and a internal parameters configuration using PSO algorithm. Finally, results indicate that RNW model with PSO, dynamic weight adjusted, and a network topology configured in four inputs nodes, three hidden nodes and one output node, presents the best performance achieving a 97.56% of the explained variance and 18.85% of SMAPE.

**Keywords:** Artificial Neural Networks, Wavelet Neural Networks, Particle Swarm Optimization, Swarm Intelligence.

# Índice

<b>LISTA DE FIGURAS</b> .....	<b>vii</b>
<b>LISTA DE TABLAS</b> .....	<b>viii</b>
<b>LISTA DE ABREVIATURAS O SIGLAS</b> .....	<b>ix</b>
<b>CAPÍTULO 1 INTRODUCCIÓN</b> .....	<b>1</b>
1.1. DEFINICIÓN DE OBJETIVOS .....	2
1.1.1. <i>Objetivo General</i> .....	2
1.1.2. <i>Objetivos Específicos</i> .....	2
1.1.3. <i>Organización del Texto</i> .....	2
<b>CAPÍTULO 2 PROBLEMA DE INVESTIGACIÓN</b> .....	<b>3</b>
<b>CAPÍTULO 3 MARCO TEÓRICO</b> .....	<b>4</b>
3.1. REDES NEURONALES ARTIFICIALES .....	4
3.1.1. <i>Reseña Histórica de Aplicación de RNA aplicadas a Predicción</i> .....	5
3.1.2. <i>Características de las RNA aplicadas a Predicción</i> .....	5
3.1.3. <i>Aplicaciones de RNA como herramientas de Predicción</i> .....	6
3.1.4. <i>Redes Neuronales Wavelet</i> .....	7
3.2. OPTIMIZACIÓN POR ENJAMBRE DE PARTÍCULAS.....	9
3.2.1. <i>Algoritmo PSO Modelo Tradicional</i> .....	10
3.2.2. <i>Ajuste de los Parámetros de PSO</i> .....	11
3.2.3. <i>Exploración y Explotación con PSO</i> .....	12
3.2.4. <i>Modelo Óptimo Local y Modelo Óptimo Global</i> .....	13
3.2.5. <i>Variaciones del Algoritmo PSO</i> .....	13
3.2.5.1. PSO Quantum (QPSO).....	14
3.2.5.2. PSO con factor de inercia exponencial (EPSO) .....	14
<b>CAPÍTULO 4 MODELO PROPUESTO</b> .....	<b>15</b>
4.1 MÉTRICAS DE EVALUACIÓN .....	16
4.2 SELECCIÓN DE TOPOLOGÍA Y PARÁMETROS DEL MODELO .....	17
4.2.1 <i>Selección de topología para Alta Frecuencia</i> .....	17
4.2.1.1 Neuronas de Entrada .....	17
4.2.1.2 Neuronas Ocultas.....	18
4.2.1.3 Número de Iteraciones .....	19
4.2.1.4 Número de Partículas .....	19
4.2.2 <i>Selección de topología para Baja Frecuencia</i> .....	20
4.2.2.1 Neuronas de Entrada .....	20
4.2.2.2 Neuronas Ocultas.....	20
4.2.2.3 Número de Iteraciones .....	21
4.2.2.4 Número de Partículas .....	21
4.2.3 <i>Selección parámetros PSO</i> .....	22
4.3. ANÁLISIS DE RESULTADOS.....	22
4.3.1. <i>RNW-PSO Alta Frecuencia (HF)</i> .....	22
4.3.2. <i>RNW-PSO Baja Frecuencia (LF)</i> .....	23
4.3.3. <i>RNW-PSO (HF+LF)</i> .....	24
4.3.4. <i>RNW-QPSO</i> .....	24
4.3.5. <i>RNW-EPSO</i> .....	24
4.4. COMPARACIÓN DE RENDIMIENTO DE LOS RESULTADOS. ....	25
4.4.1. <i>Etapa Alta Frecuencia</i> .....	25
4.4.2. <i>Etapa Baja Frecuencia</i> .....	25
4.4.3. <i>Etapas sumadas (HF + LF)</i> .....	25
<b>CONCLUSIONES</b> .....	<b>26</b>

**REFERENCIAS BIBLIOGRÁFICAS ..... 28**  
**ANEXOS.....**

# Lista de Figuras

Figura 3-1 Representación de una neurona básica.....	4
Figura 3-2 Arquitectura red neuronal Wavelet .....	8
Figura 4-1 Descripción del modelo.....	15
Figura 4-2 MSE v/s Nodos Entradas HF.....	18
Figura 4-3 MSE v/s Nodos Ocultos HF .....	18
Figura 4-4 MSE v/s Iteraciones HF .....	19
Figura 4-5 MSE v/s Partículas HF.....	19
Figura 4-6 MSE v/s Nodos Entradas LF .....	20
Figura 4-7 MSE v/s Nodos Ocultos LF.....	20
Figura 4-8 MSE v/s Iteraciones LF.....	21
Figura 4-9 MSE v/s Partículas LF .....	21
Figura 4-10 Gráficos observados para HF.....	22
Figura 4-11 Gráficos observados para LF.....	23
Figura 4-12 Gráficos observados para la suma de las componentes (HF+LF) .....	24



# Lista de Tablas

Tabla 4-1 Mejor configuración HF ..... 22

Tabla 4-2 Mejor configuración LF..... 23

Tabla 4-3 Resultados RNW-PSO v/s RNW-QPSO v/s RNW-EPSO (HF)..... 25

Tabla 4-4 Resultados RNW-PSO v/s RNW-QPSO v/s RNW-EPSO (LF) ..... 25

Tabla 4-5 Resultados RNW-PSO v/s RNW-QPSO v/s RNW-EPSO (HF+LF) ..... 25

## Lista de Abreviaturas o Siglas

ARIMA	Autoregressive Integrated Moving Average (Modelo Autorregresivo Integrado de Media Móvil).
EPSO	Exponential Particle Swarm Optimization (Optimización por Enjambre de Partículas Exponencial).
HF	High Frequency (Alta Frecuencia).
LF	Low Frequency (Baja Frecuencia).
MAE	Mean Absolute Error (Error Absoluto Medio).
MSE	Mean Square Error (Error Cuadrático Medio).
PSO	Particle Swarm Optimization (Optimización por Enjambre de Partículas).
QPSO	Quantum Particle Swarm Optimization (Optimización por Enjambre de Partículas Cuántica).
RMSE	Root Mean Square Error (Raíz Error Cuadrático Medio).
RNA	Redes Neuronales Artificiales.
RNW	Redes Neuronales Wavelet.
SI	Swarm Intelligence (Inteligencia de Enjambre).
SMAPE	Symmetric Mean Absolute Percentage Error (Porcentaje Error Absoluto Medio Escalado).

# Capítulo 1 Introducción

El hombre desde sus comienzos ha buscado la manera de predecir con mayor certeza los fenómenos a los cuales se ve afectado, por lo cual intenta disminuir la incertidumbre que tiene del futuro a través de distintos métodos, soluciones y ciencias. Debido a esto, es que en la historia podemos encontrar ejemplos de diferentes ciencias de la humanidad que han implementado diversas técnicas de predicción para obtener resultados que le ayuden a enfrentar los distintos sucesos.

La Inteligencia Artificial ha surgido como un área importante de las Ciencias de la Computación e Informática. Está conformada por un conjunto de técnicas y métodos, los cuales intentan imitar computacionalmente habilidades relacionadas con la inteligencia humana [1]. Una de estas técnicas que trata de simular el pensamiento humano mediante conexión de neuronas, y que ha tomado gran prestigio y buenos resultados en el área de las predicciones con series de tiempo, es la denominada “Redes Neuronales Artificiales” (denominadas habitualmente como RNA o en inglés como: "ANN"). Las RNA utilizadas en pronósticos, han demostrado gran éxito en áreas tales como la ingeniería, economía, meteorología, finanzas, y sismología.

Sin embargo, el uso de una RNA requiere de la elección de una serie de parámetros y una arquitectura apropiada para la solución del problema. Modelos híbridos inteligentes que son una combinación de diferentes métodos inteligentes son propuestos con el fin de lograr un mejor rendimiento en predicción, con el intento de combinar los puntos fuertes de varios algoritmos dentro de un sistema simple. Hay algunos estudios en la literatura que usan estas ideas, donde un método de búsqueda inteligente es combinado con una RNA para mejorar un sistema predictivo. [2] combinó un modelo por optimización por enjambre de partículas (PSO) con una RNA para predecir niveles de polución en el centro de la ciudad de Hong Kong. [3] utilizaron RNA y PSO para predecir la condición de tiempo en el futuro, teniendo en cuenta como parámetros la temperatura máxima, temperatura mínima, precipitaciones, humedad, etc. Los resultados indicaron que el enfoque propuesto es útil para pronóstico de tiempo.

El trabajo presentado a continuación se encuentra estructurado en cuatro capítulos. El primero da cuenta del planteamiento del problema investigativo y se dan a conocer el objetivo general y los objetivos específicos del proyecto. El capítulo segundo constituye el marco teórico de la investigación. En él se expone el estado del arte del proyecto. Se comprenden las RNA, el resumen histórico de las RNA aplicadas a la predicción. Posteriormente se abordan todos los conceptos de la Optimización por Enjambre de Partículas (PSO), describiendo su funcionamiento y variantes a lo que su algoritmo respecta, que básicamente se relacionan a la variación e integración de parámetros. El capítulo tercero explica el modelo general empleado para dar solución a la problemática de la investigación. Posteriormente se presentan los resultados al aplicar el modelo, como también los resultados al aplicar dos variantes para comparar el rendimiento predictivo del modelo. Se presenta el análisis y discusión de los resultados obtenidos. Finalmente se presentan las conclusiones generales obtenidas de este proyecto.

## **1.1. Definición de Objetivos**

### **1.1.1. Objetivo General**

Desarrollar y evaluar un modelo de predicción utilizando Red Neuronal Wavelet basado en el algoritmo Optimización por Enjambre de Partículas (PSO) para predecir las capturas mensuales de alta y baja frecuencia de sardinas en el norte de Chile para el periodo de Enero de 1976 a Diciembre de 2007.

### **1.1.2 Objetivos Específicos**

- Explicar la estructura y el estado del arte de las Redes Neuronales Artificiales y del algoritmo PSO.
- Diseñar la estructura y estimar los parámetros del modelo de pronóstico neuroevolutivo utilizando como función de transferencia la función Wavelet Derivada de la Gaussiana y el uso del algoritmo PSO.
- Evaluar el rendimiento predictivo del pronosticador neuronal utilizando cuatro métricas de evaluación.

### **1.1.3 Organización del Texto**

El proyecto continúa en el Capítulo 2 con el Problema de Investigación, que delimita de manera clara y precisa el objeto de investigación, y por qué es tan importante este proyecto para la toma de decisiones. Luego el Capítulo 3 con el Marco Teórico, el cual describe el estado del arte de las RNA, su historia, características y aplicaciones de las RNA como herramientas de predicción. Además se describe al algoritmo PSO, definiendo sus elementos básicos, ajustes al algoritmo, técnicas y variantes. El Capítulo 4 presenta el modelo propuesto para esta investigación, la selección de la mejor topología para la red y parámetros de PSO, y concluyendo con el análisis de los resultados al aplicar el modelo. Finalmente se presentan las conclusiones del presente proyecto.

## Capítulo 2 Problema de investigación

El sector pesquero presenta una importante contribución en las exportaciones a nivel nacional, siendo una de las actividades más trascendentes dentro de la economía del país. Sin embargo, su relevancia se debe principalmente a la disponibilidad de los recursos pesqueros en el tiempo, siendo éste uno de los principales factores que generan incertidumbre dentro de la actividad y que a su vez, pueden entorpecer y restringir el desarrollo sustentable del sector.

Chile tiene acceso, junto con Perú, a uno de los cinco “oasis” pesqueros más importantes del mundo, al poseer una larga costa de 4.500 Km que recibe los efectos favorables de la corriente de Humboldt, es por ello, que cuenta con una gran dotación de recursos marítimos a lo largo de sus costas.

Las sardinas son pequeños recursos pelágicos altamente importantes para el desarrollo económico en el norte de Chile. La pesca de sardina comenzó a desarrollarse en Chile a comienzos de 1973. Sin embargo, la abundancia de sardina se redujo drásticamente después de 1985 y, al mismo tiempo, la abundancia de anchoveta se recuperó. Estas fluctuaciones históricas de stocks en sardinas y anchovetas están asociadas con factores de variabilidad climáticos, como los fenómenos de El Niño, y La Niña [4]. El Niño es un fenómeno de ocurrencia normal en esta parte del océano Pacífico, que corresponde a la fase cálida de un ciclo que también posee una fase fría, conocida como La Niña, que se alternan en ciclos de alrededor de 11 años [5]. Después de 1975, un periodo de calentamiento tomó lugar y la sardina, la cual se alimenta con exceso de los huevos de anchovetas, aumentó. El regreso de las condiciones más frías después de 1985 favoreció la recuperación de anchovetas y la disminución de las sardinas, a pesar de los eventos de El Niño en 1987, 1991-92 y 1997-98 [6], [7].

Con el fin de desarrollar políticas sostenibles de explotación, el pronóstico de la existencia y capturas de especies pelágicas en el norte de Chile es uno de los objetivos principales de la industria pesquera y el gobierno. Sin embargo, fluctuaciones en las variables del entorno complican esta tarea. A consecuencia de lo expuesto anteriormente, las disminuciones en las capturas totales de sardina afecta directamente a la producción y exportación de la especie, por lo que obtener una cuota anual de desembarques que se ajuste a la realidad del sector pesquero es gravitante para el futuro de la actividad.

Por lo tanto, a consecuencia de esto, es que el presente proyecto pretende modelar las capturas mensuales de sardinas a través de series de tiempo, con el objetivo de elaborar un modelo predictivo utilizando RNA, que permita y sea capaz de apoyar a entidades gubernamentales llevando un control y estableciendo una fijación en las cuotas anuales de desembarque de sardina en el norte del país, y de esta manera permitir la disponibilidad de la especie en el futuro.

# Capítulo 3 Marco teórico

## 3.1. Redes Neuronales Artificiales

Las Redes Neuronales Artificiales (RNA) surgieron como solución a los principales objetivos y desafíos de los científicos a lo largo de la historia, la cual consistía en dar con las ideas de cómo conseguir, diseñar y construir máquinas o sistemas capaces de realizar procesos complejos con cierta “inteligencia”, los cuales no son sino procesos interconectados que operan en forma paralela, y de esta manera emular, con la mayor exactitud posible, el funcionamiento del cerebro humano, y a su vez, los sistemas neuronales de los animales.

En otras palabras, las RNA emulan las redes neuronales biológicas con ciertas aptitudes en común, pero no alcanzan la complejidad del cerebro; las representaciones y procedimientos de que se sirve el cerebro, son aún desconocidas.

Las RNA son sistemas que, a diferencia de la mayoría que necesitan programarse para realizar las tareas que requieren ejecutar, las RNA generalizan y aprenden de la experiencia, basándose en comportamientos típicos de patrones.

Según [8], el modelo de una RNA presenta tres elementos básicos:

- Una serie de sinapsis, donde cada una está especificada junto a un peso, el cual puede tomar tanto valores positivos como negativos.
- Una función de propagación, que suma las señales de entrada pesadas por la respectiva sinapsis de la neurona.
- Una función de activación, para limitar la amplitud de la salida de la neurona. Cabe destacar que ésta corresponde a una función no lineal que normaliza el estado interno de la neurona a un intervalo cerrado  $[0,1]$  o alternativamente  $[-1,1]$ .

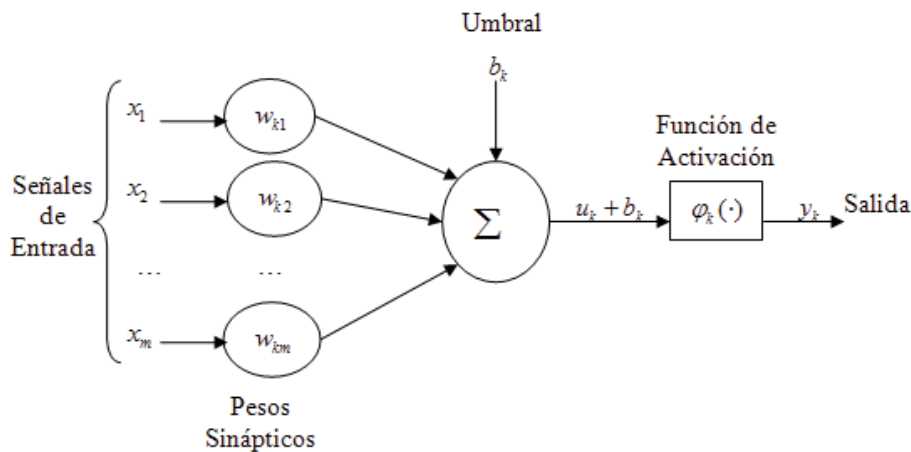


Figura 3-1 Representación de una Neurona Básica

### **3.1.1. Reseña Histórica de Aplicación de RNA aplicadas a Predicción**

La idea de utilizar RNA aplicadas a la predicción no es nueva. La primera aplicación se remonta a 1964. Hu [9], en su tesis, utiliza red lineal adaptativa de Widrow para predicción de tiempo. Debido a la falta de algoritmos de entrenamiento para redes multicapas en general (*para mayor detalle de arquitectura de RNA diríjase a [10]*), la investigación fue bastante limitada. No es hasta 1986 cuando el algoritmo de propagación hacia atrás (del inglés backpropagation) fue introducido [11], que había habido gran desarrollo en el uso de RNA para el pronóstico. Werbos [12], [13] formula en primer lugar el algoritmo backpropagation y encuentra que las RNA entrenadas con backpropagation superan a los métodos estadísticos tradicionales, tales como los métodos de regresión y de Box-Jenkins. Lapedes y Farber [14], aplicaron RNA en forma de perceptrón para predecir el comportamiento de un sistema dinámico no lineal con comportamiento caótico descrito por la ecuación diferencial no lineal Glass-Mackey, dando por resultado una mayor precisión en la predicción con redes neuronales en comparación a métodos y modelos estadísticos, concluyendo de esta forma que las RNA pueden ser utilizadas para el modelado y predicción de series de tiempo no lineales. En [15], [16] y [17] abordan el problema de la estructura de la red para la predicción de series de tiempo real. En [18], [19] y [20] presentaron resultados de varias comparaciones de predicción entre Box-Jenkins y modelos con RNA. En el año 1993, en una competencia de pronóstico organizado por Weigend y Gershenfeld a través del Instituto de Santa Fe, los ganadores de cada conjunto de datos utilizaron modelos de RNA [21]. Los esfuerzos en investigación para predicción utilizando RNA son considerables. La literatura es vasta y creciente. [22] y [23] revisan la literatura comparando RNA con modelos estadísticos en predicción de series temporales y regresión basada en predicción.

### **3.1.2. Características de las RNA aplicadas a Predicción**

Las RNA como herramientas para la predicción han sido de interés reciente para muchos investigadores y profesionales que han intentado hacer predicciones sobre valores futuros de alguna variable relevante, ya que éstas poseen varias características distintivas que las hacen valiosas y atractivas al momento de predecir, tales como:

Primero, a diferencia de los métodos basados en los modelos tradicionales, las RNA son métodos auto-adaptativos basados en datos en que hay pocas suposiciones a priori sobre los modelos para los problemas bajo estudio. Las RNA aprenden a partir de ejemplos y capturan sutiles relaciones funcionales entre los datos incluso si las relaciones subyacentes son desconocidas o difíciles de describir. Por lo tanto las RNA son adecuadas para problemas cuya solución requiere conocimiento que es difícil de especificar pero para los cuales existen suficientes datos u observaciones [24], [25].

Segundo, las RNA pueden generalizar. Después de aprender los datos que se les presentaron (una muestra), esto es finalizada su etapa de entrenamiento, las RNA pueden inferir de manera apropiada el futuro o la parte no vista de una población, incluso si los datos de la muestra contienen información ruidosa. Como el pronóstico es realizado a través de la predicción del comportamiento futuro (parte invisible) de ejemplos del comportamiento pasado, es un área de aplicación ideal para redes neuronales, al menos en principio.

Tercero, las RNA son aproximadores funcionales universales. Se ha demostrado que una red puede aproximar cualquier función continua a una precisión deseada [26], [27]. Cualquier modelo de predicción asume que existe una relación subyacente (conocida o desconocida) entre las entradas (los valores pasados de las series temporales y/u otra variable relevante) y las salidas (los valores futuros). Frecuentemente, los modelos de predicción estadísticos tradicionales tienen limitaciones en la estimación de esta función subyacente debido a la complejidad de los sistemas reales. Las RNA pueden ser un buen método alternativo para identificar esta función.

Finalmente, las RNA son no lineales. El pronóstico ha sido durante mucho tiempo el dominio de las estadísticas lineales. Los enfoques tradicionales para predicción de series temporales, tales como el método ARIMA [28], [29], asumen que las series temporales estudiadas son generadas de procesos lineales. Los modelos lineales tienen ventajas en que pueden ser entendidas y analizadas en gran detalle, y son fáciles de explicar e implementar. Sin embargo, pueden ser totalmente inapropiadas si el mecanismo fundamental no es lineal. Es irrazonable asumir a priori que una realización particular de una serie temporal dada es generada por un proceso lineal. De hecho, la mayoría de los problemas que existen en el mundo son no lineales [30]. Las RNA son capaces de realizar modelado no lineal sin un conocimiento a priori sobre las relaciones entre variables de entrada y de salida. De esta manera superan las limitaciones que poseen los métodos estadísticos clásicos, y así, son apropiadas para la comprensión y mejor entendimientos de los problemas complejos.

### **3.1.3. Aplicaciones de RNA como herramientas de Predicción**

Los problemas de predicción surgen en muchas disciplinas distintas y la literatura en pronóstico usando RNA está dispersa en diversos campos lo que hace difícil para un investigador tener en cuenta todo el trabajo realizado a la fecha en el área. Siguiendo al trabajo exitoso de Lapedes y Farber, una serie de papers se dedicaron a la utilización de RNA para analizar y predecir series temporales caóticas determinísticas con y/o sin ruido. Las series temporales caóticas ocurren en su mayoría en las ciencias de ingeniería y físicas ya que la mayoría de los fenómenos físicos son generados por sistemas caóticos no lineales. A consecuencia, muchos autores en modelamientos de series temporales caóticas y pronóstico provienen del campo de la física. En [31] se discute la relación entre sistema dinámico e interpolación funcional con RNA. En [32] proponen un modelo RNA entrenado jerárquicamente en el cual un mejoramiento dramático en precisión es logrado para predicción de dos sistemas caóticos.

La serie de mancha solar ha sido durante mucho tiempo un punto de referencia y ha sido bien estudiada en la literatura estadística. Dado que los datos se creen que son no lineales, no estacionarios y no Gaussiano, a menudo son usados como criterio para evaluar y comparar nuevos métodos de predicción. Algunos autores se centran en cómo usar RNA para mejorar la precisión en la predicción de actividades de manchas solares sobre los métodos tradicionales [33], [34], mientras que otros usan los datos para demostrar un método [15], [35].



Hay una extensa literatura en aplicaciones financieras usando RNA. Las RNA han sido utilizadas para pronosticar quiebra y fracaso empresarial [36], [37], [38], tasa de cambio extranjero [16], [39], [40], precios de las acciones [41], [42].

Otra aplicación importante de predicción con RNA es en el estudio del consumo de carga eléctrica. La predicción de carga es un área que requiere de una gran precisión ya que el suministro de electricidad es altamente dependiente del pronóstico de la demanda de carga. En [43] informan que simples RNA con entradas de información de temperatura realizan un desempeño mucho mejor que la técnica basada en regresión usada actualmente en pronóstico por hora, peak y consumo de carga total. En [44] discuten por qué las RNA son adecuadas para pronóstico de carga y proponen un sistema de subredes en cascada.

Muchas investigaciones utilizan los datos de la famosa competencia-M [45] para comparar el rendimiento de modelos RNA con modelos estadísticos tradicionales. Los datos de la competencia-M son en su mayoría de negocios, económicos y financieros. En la competencia de predicción Santa Fe [46], seis series temporales no lineales de diferentes disciplinas tales como física, psicología, astrofísica, finanzas e incluso música son usadas. Todos los conjuntos de datos son muy extensos en comparación con la competencia-M en la donde todas las series de tiempo son bastante cortas.

Muchos otros problemas de predicción han sido resueltos usando RNA. De entre éstos podemos nombrar: Polen en el aire [47], precios de commodity [48], temperatura ambiental [49], carga de componentes de helicóptero [50], tráfico de pasajeros de aerolínea internacional [51], índices macroeconómicos [52], nivel de ozono [53], inventario de personal [54], lluvia [55], caudal de los ríos [56], demanda de agua [57], y presión del viento [58].

Cabe mencionar además, que ex alumnos de la Escuela de Ingeniería Informática de la Pontificia Universidad Católica de Valparaíso en sus investigaciones se enfocaron principalmente en desarrollar modelos con RNA que fueron de carácter relevante para esta investigación. [59] y [60] utilizaron RNA con algoritmo PSO para el pronóstico de captura de anchovetas en el norte de Chile. [61] desarrolló una RNW híbrida para el pronóstico de captura de anchovetas.

### 3.1.4. Redes Neuronales Wavelet

Las Redes Neuronales Wavelet (RNW) fueron propuestas por Zhang y Benveniste [62] en 1992. Estas redes combinan la teoría de procesamiento wavelet con las redes neuronales. En la arquitectura para esta investigación, la capa oculta está formada por *wavelons* cuyos parámetros de entrada están definidos por peso de entrada, parámetro de traslación ( $\tau$ ) y dilatación ( $\lambda$ ). La salida de la red es una combinación lineal de las funciones wavelet de cada wavelon. La arquitectura se puede observar en la figura 3-2.

En donde  $\mathbf{u}$  representa cada una de las entradas que tiene la red.  $\Psi$  representa cada uno de los *wavelons* que se encuentran en la capa oculta.  $\mathbf{y}$  es la salida de la red.

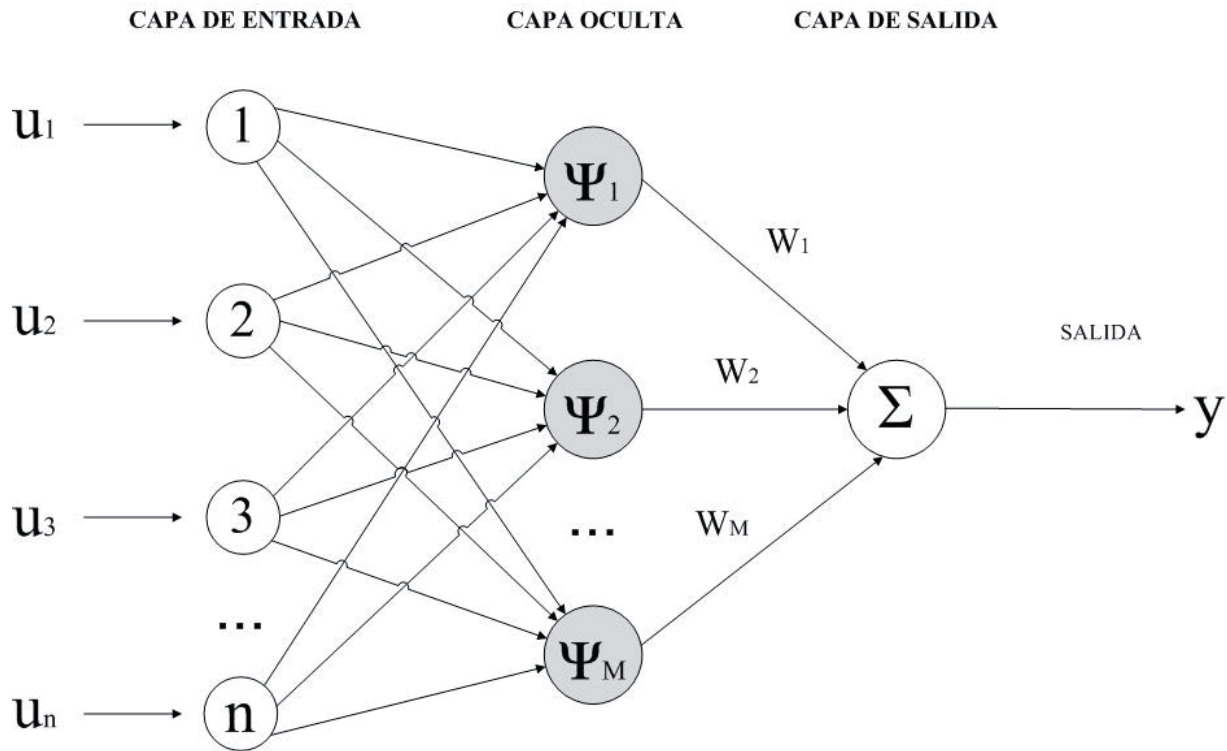


Figura 3-2 Arquitectura Red Neuronal Wavelet

El factor de dilatación denotado por la letra  $\lambda$ , es una cantidad entera positiva que tiene un efecto compresor mientras más pequeño sea o un efecto de estiramiento al adoptar un valor grande. Los valores pequeños en el factor de escalamiento sirven para detectar cambios rápidos en la función que se está analizando mientras que los valores grandes provocan una baja frecuencia que detecta cambios suaves en la señal. El factor de traslación denotado por la letra  $t$ , tiene un efecto de retraso o avance en la señal. Cuando se efectúa el cálculo de los coeficientes, el factor de traslación se modifica para que cubra toda la señal con el valor de escalamiento específico. La salida de la red, está dada por la siguiente función:

$$y = \sum_{i=1}^M W_i \Psi_i(u_1, \dots, u_n) \quad (3.1)$$

$$\Psi(u_1, \dots, u_n) = \prod_{i=1}^n \psi_{\lambda_i, t_i}(u_i) \quad (3.2)$$

$$\psi_{\lambda, t} = \psi\left(\frac{u-t}{\lambda}\right) \quad (3.3)$$

Donde  $\Psi$  corresponde al *wavelon*,  $\psi$  es la función *wavelet madre*. La función madre a utilizar será la función derivada de la Gaussiana descrita en la siguiente ecuación:

$$f(x) = -x * e^{\left(-\frac{1}{2} * x^2\right)} \quad (3.4)$$

## 3.2. Optimización Por Enjambre De Partículas

A lo largo de la historia se han desarrollado diversas formas de entregar soluciones a problemas de optimización con distintos métodos y a su vez cada una con su propia perspectiva o fuente inspiradora. Intentar abordar problemas de optimización es una tarea difícil que en muchos casos se vuelve impracticable debido a su complejidad o desconocimiento. Una alternativa a estos problemas son los algoritmos de optimización metaheurísticos que permiten encontrar soluciones de buena calidad en tiempos de ejecución computacionalmente aceptables.

Especial énfasis ha adquirido la Optimización por Enjambre de Partículas o PSO (del inglés Particle Swarm Optimization), algoritmo que nace en un intento por imitar y mimetizar el comportamiento de los procesos naturales. Es uno de los métodos más utilizados y es considerado uno de los más emblemáticos en el área de la Inteligencia de Enjambre o SI (del inglés Swarm Intelligence). Este proceso trata de imitar los comportamientos sociales de una población a partir de la interacción de los individuos entre sí y con el entorno.

El origen de PSO se remontan a las investigaciones realizadas por Kennedy y Eberhart [63] que intentaron simular el movimiento sincronizado de grupos de especies tales como cardumen de peces o bandada de aves, ya que estos grupos, al observarlos, notaron que poseían una capacidad asombrosa al momento de separarse, reagruparse o encontrar alimento. Sus conclusiones se relacionan directamente con la capacidad que tienen estas especies para compartir información y la experiencia que acumula y aporta cada miembro del grupo.

Kennedy y Eberhart introducen el concepto de partícula o agente para representar a un individuo del que se pretenda emular su comportamiento social. De acuerdo a los fundamentos teóricos del método, el movimiento de cada una de estas partículas hacia la consecución de un objetivo en común depende de dos factores: la memoria autobiográfica de la partícula y la influencia social de todo el enjambre. Cada partícula se dice que sobrevuela el espacio de decisión en busca de soluciones óptimas. Generalizando, la posición instantánea de cada una de éstas en el espacio vectorial de  $D$  dimensiones representa una solución potencial. Cuando se habla de un espacio  $D$ -dimensional, se cuentan con  $d$  incógnitas a solucionar. El proceso evolutivo se basa en mover a cada una de las partículas dentro del espacio de soluciones, con una velocidad que varía de acuerdo a la memoria de la partícula, información global y su velocidad actual, utilizando una función de aptitud (fitness) que se encargará de cuantificar la calidad de cada partícula en función de la posición que ésta ocupe [64].

### 3.2.1. Algoritmo PSO Modelo Tradicional

**Definición:** PSO es un algoritmo evolutivo, iterativo y estocástico, que se basa en buscar soluciones en un espacio vectorial  $N$  dimensional desplazando a un conjunto de partículas sobre el mismo, siendo cada partícula una posible solución.

Este algoritmo parte de una población (o enjambre) para iniciar un proceso de búsqueda, donde cada partícula está agrupada dentro una población con posiciones y velocidades aleatorias. Una partícula es considerada como un punto en el espacio de búsqueda  $D$ -dimensional, donde cada dimensión representa una de las incógnitas del problema. El enjambre está representado por  $X = (x_{i1}, x_{i2}, \dots, x_{id})$ , donde  $i=1, 2, \dots, n$  corresponde a la  $i$ -ésima partícula y  $d = 1, 2, \dots, D$  corresponde a sus dimensiones.

Ya teniendo inicializadas las partículas comienza el proceso iterativo. Por cada iteración, cada partícula actualizará el valor de su mejor posición encontrada, denominada mejor personal  $p_{best}$ . Para representar al  $p_{best}$  de una  $i$ -ésima partícula se tiene la siguiente expresión  $P_i = (p_{i1}, p_{i2}, \dots, p_{id})$ . Además de actualizar el  $p_{best}$ , la partícula actualiza el valor de la mejor posición encontrada en el enjambre, denominado mejor global  $p_{gbest}$ . Estos valores se utilizan para ajustar la velocidad con la que se mueve una partícula  $i$ -ésima en cada una de las dimensiones. Además, con el valor de la velocidad, la partícula podrá calcular su nueva posición para la siguiente iteración. La influencia que tiene la mejor posición personal sobre la velocidad de una partícula se conoce como el factor o componente de cognición, y la influencia de la mejor posición global o del enjambre se conoce como componente social [65].

Las ecuaciones que determinan la posición y la velocidad de cada partícula dentro del enjambre está dada por:

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (3.5)$$

$$v_{id}(t+1) = w \cdot v_{id}(t) + c_1 r_1 (p_{id}(t) - x_{id}(t)) + c_2 r_2 (p_{gd}(t) - x_{id}(t)) \quad (3.6)$$

Donde los parámetros  $c_1$  y  $c_2$  corresponden a las variables que controlan las influencias de los componentes de cognición y social. Los valores de  $r_1$  y  $r_2$  corresponden a números aleatorios independientes pertenecientes al intervalo  $[0,1]$ . El parámetro  $w$  surge de una investigación previa [63], en donde se realizó una modificación al algoritmo agregando un componente denominado factor de inercia  $w$  en la ecuación que determina la velocidad, y así lograr balancear la velocidad de la partícula respecto a la búsqueda local y global.

Para prevenir que las velocidades de las partículas incrementen infinitamente, estas se acotan a un rango determinado por el valor  $V_{max}$ , el cual delimita el rango de velocidad al intervalo  $[-V_{max}, V_{max}]$ .

A continuación se presentarán los pasos a seguir en el PSO tradicional:

1. Se inicia con una población de partículas de tamaño  $n$ , con posiciones y velocidades aleatorias en un espacio  $D$ -dimensional.
2. Para cada partícula, evaluar su función fitness de acuerdo a algún criterio de optimización (función de costo, mínimos cuadrados, error cuadrático medio).
3. Se compara el valor de la función fitness de la partícula con el fitness del mejor personal ( $p_{best}$ ). Si el valor del fitness actual resulta ser mejor, entonces se actualiza  $p_{best}$  al valor de la posición actual  $x_{id}$ .
4. Se compara el fitness actual de cada partícula con el mejor fitness encontrado por el enjambre. Si el fitness actual resulta ser mejor, entonces se actualiza la mejor posición global ( $p_{gbest}$ ) al valor de la posición actual  $x_{id}$ .
5. Modificar las posiciones y velocidades de las partículas de acuerdo a las ecuaciones 3.1 y 3.2 respectivamente descritas en este capítulo, verificando que las partículas no sobrepasen la velocidad máxima definida el intervalo generado por  $Vmax$ .
6. Si el criterio de parada no se ha cumplido, volver al paso 2.

Las condiciones de términos más recurridas en la literatura para finalizar el proceso de optimización son:

- Número de iteraciones sin mejora: El proceso termina después de alcanzar un número fijo de iteraciones sin presentar mejoras en la solución.
- Número máximo de iteraciones sin mejora: El proceso termina después de realizar un número fijo de iteraciones preestablecidas.
- Fitness logrado: El proceso termina cuando llega a un valor menor o igual al fitness establecido.

El éxito del algoritmo depende en la medida que se ajusten las posiciones de las partículas en el espacio de búsqueda, estos ajustes dependen de un número reducido de parámetros lo que se traduce en la gran potencia de PSO.

### 3.2.2. Ajuste de los Parámetros de PSO

El ajuste de los parámetros condiciona el rendimiento posterior del algoritmo de optimización. Existen múltiples estudios [66] que intentan generalizar la selección de estos parámetros, sin embargo dicha selección está directamente relacionada a la naturaleza del problema [67], siendo preciso lograr un balance óptimo entre la exploración y la convergencia. La definición de la función fitness cumple un papel no menor, dado que una función que no introduzca una métrica adecuada para pesar la bondad de cada partícula, hará emerger las carencias del PSO, las cuales erróneamente pueden asociarse con el algoritmo.

Para acotar la velocidad de la partícula se define un valor máximo  $V_{\max}$ , el cual restringe a la velocidad en cada dimensión del intervalo  $[-V_{\max}, V_{\max}]$ . Si el valor de  $V_{\max}$  tiende a ser demasiado pequeño, las partículas explorarán el espacio de soluciones muy lentamente y podrán quedar atrapadas en un óptimo local, este fenómeno se conoce como convergencia prematura del algoritmo [67]. Por otro lado, la no restricción de la velocidad llevaría a la no convergencia del enjambre en un punto, ocurriendo lo que se conoce como explosión del PSO, el cual provoca la ineficiencia del PSO como algoritmo de optimización [63].

El peso de inercia  $w$ , regula la compensación entre la capacidad de exploración global y locales del enjambre. A mayor  $w$ , facilita la exploración global y en caso contrario facilita la exploración local. En un comienzo  $w$  se asigna como una constante, sin embargo, resultados experimentales recomiendan la asignación de mayores valores en un comienzo para que en una primera instancia se examine de manera global el espacio de búsqueda y gradualmente se disminuya para comenzar a conseguir soluciones más refinadas.

Los parámetros  $c_1$  y  $c_2$ , luego de una buena sintonización, pueden acelerar la convergencia del algoritmo y aliviarlo de los mínimos locales. Típicamente a estos parámetros se les asigna 2, pero algunos resultados experimentales asignan a ambos parámetros el valor de 1.49. Trabajos recientes revelan que podrían ser un parámetro cognitivo  $c_1$  más grande que el social  $c_2$ , siempre y cuando se cumpla que  $c_1 + c_2 \leq 4$  [64].

El tamaño de la población (o enjambre) debe ser seleccionado con rigurosidad, dado que valores muy grandes pueden explorar de manera minuciosa el espacio de búsqueda pero el costo computacional se eleva de manera considerable. Generalmente se utilizan poblaciones que oscilan entre 10 y 50 partículas, o entre 100 y 200 para problemas complejos.

### 3.2.3 Exploración y Explotación con PSO

Uno de los problemas de los algoritmos evolutivos, es que en ocasiones quedan atrapados en óptimos locales, lo cual en la literatura se conoce como convergencia prematura del algoritmo y corresponde a la incapacidad de salir la zona en la cual se encontró el mínimo local, impidiendo que se exploren otras regiones del espacio de búsqueda.

Lo ideal es que las partículas representen soluciones de diferentes zonas del espacio de búsqueda. Esto se conoce como diversificación y en términos de los algoritmos corresponde a que las partículas sobrevuelen diferentes zonas del espacio de soluciones antes de que se comprometan con alguna en especial. Por otro lado, las partículas, una vez encontrada una zona comprometedoras deben ser capaces de mejorar la solución encontrada. Esto se conoce como convergencia de la población.

Se define como exploración a la búsqueda de la solución del problema en el espacio de búsqueda de manera amplia, privilegiando la diversificación del enjambre. Por otro lado, la explotación consiste en la búsqueda de la solución del problema en un rango (sub-espacio)

dentro del espacio de búsqueda donde el enjambre trate de tomar todos los valores posibles, privilegiando la convergencia del enjambre.

Estudios realizados en [64] revelan que altos valores para el coeficiente de inercia, factores de cognición y  $V_{\max}$  ( $0.75 < w < 1$  y  $2 < c_1, c_2 < 4$ ), favorecen el modo de exploración del PSO, mientras que valores bajos ( $0.4 < w < 0.75$  y  $0.1 < c_1, c_2 < 2$ ) favorecen el modo de explotación del PSO.

La idea es ajustar los parámetros de tal manera que el algoritmo funcione en modo de exploración en las primeras etapas y una vez transcurridas unas cuantas iteraciones cambie a modo explotación.

### **3.2.4 Modelo Óptimo Local y Modelo Óptimo Global**

Eberhart y Kennedy diseñaron dos modelos de PSO. El primero corresponde al modelo del óptimo global, en donde la trayectoria de búsqueda de cada partícula está directamente influenciada por la mejor posición encontrada por cualquier miembro de la población. En cambio, el modelo del óptimo local una partícula sólo puede ser influenciada por un número reducido de miembros adyacentes de la población.

Por lo tanto, se concluye que los modelos de óptimos global convergen rápidamente hacia las soluciones del problema, pero presentan una desventaja de poder quedar atrapado en un óptimo local. Por el contrario, el modelo del óptimo local tiene la desventaja de que converge en forma lenta hacia la solución del problema. En [65] los autores recomiendan usar el modelo de óptimo global para funciones unimodales y utilizar el modelo local para funciones multimodales.

### **3.2.5. Variaciones del Algoritmo PSO**

Durante la primera década de la utilización del algoritmo PSO, aparecieron numerosos trabajos que, partiendo del enjambre original, proponían modificaciones con el fin de mejorar sus características, normalmente de forma empírica.

Sin embargo al hacer esto se agrega un mayor nivel de complejidad al algoritmo tradicional. Actualmente existen una serie de versiones alternativas al algoritmo PSO tradicional, las cuales incorporan nuevos parámetros, nuevos métodos de actualización, e híbridos con otros algoritmos. Estas versiones tienen por objeto mejorar algunas de las características del PSO tales como mejora de la solución encontrada y mejora de velocidad de convergencia. Las variaciones que se pueden hacer son infinitas, sin embargo no todas tienen porque tener un buen comportamiento. A continuación se describirán algunas variaciones del algoritmo PSO tradicional que han demostrado tener resultados competitivos.



### 3.2.5.1. PSO Quantum (QPSO)

Esta propuesta consiste en eliminar el concepto de velocidad e incorporar una función de movimiento. Esta función consiste en comparar la posición actual de una partícula con el promedio de las mejores posiciones del enjambre, a diferencia del PSO tradicional que compara la posición de una partícula con la mejor posición del enjambre [68].

En QPSO, las partículas se mueven de acuerdo a las siguientes funciones:

$$x(t + 1) = p \pm \beta \cdot |m_{best} - x(t)| \cdot \ln\left(\frac{1}{\mu}\right) \quad (3.7)$$

$$p = r \cdot P_{id} + (1 - r) \cdot P_{gd} \quad (3.8)$$

$$\beta = (1 - 0.5) \cdot \frac{T_{max} - iter}{T_{max}} + 0.5 \quad (3.9)$$

$$m_{best} = \frac{\sum_{i=1}^n p_i}{n} \quad (3.10)$$

Donde el parámetro  $p$  depende de la mejor posición de la partícula  $P_{id}$  y de la mejor posición global  $P_{gd}$ , y donde  $r$  corresponde a un valor aleatorio entre [0,1].  $\beta$  es un coeficiente de expansión-contracción, perteneciente al rango [0,1.7].  $m_{best}$  representa el vector con las mejores posiciones encontradas por las partículas, y  $\mu$  corresponde a un valor aleatorio entre [0,1]. Esta propuesta fue probada en 3 funciones obteniendo resultados superiores en términos de velocidad de convergencia al PSO tradicional [69].

### 3.2.5.2. PSO con factor de inercia exponencial (EPSO)

En [70], se propone una modificación al PSO tradicional, donde las partículas tienden a acercarse hacia la mejor posición encontrada por el enjambre. Esta situación ayuda a descubrir soluciones óptimas rápidamente. Sin embargo, también se obtienen partículas que se estancan en óptimos locales y no convergen a un óptimo global. Los autores plantean EPSO, el cual tiene un gran impacto tanto en la exploración global como local, con el objetivo de que el comportamiento llevado a cabo en la búsqueda sea de forma rápida e inteligente, y así evitar que las partículas queden atrapadas en un óptimo local mediante un factor de inercia que varía exponencialmente, donde el movimiento de las partículas es más rápido y distante entre ellas. Este factor de inercia exponencial viene dado por la siguiente ecuación:

$$w = (w_{max} - w_{min}) \cdot e^{\frac{(iter_{max} - k)}{iter_{max}} \cdot 1} + w_{min} \quad (3.11)$$

Los autores expresan, a partir de los diversos experimentos realizados, que el factor de inercia disminuya exponencialmente dentro del intervalo [0.9 - 0.4], donde  $w_{max} = 0.9$ , mientras que  $w_{min} = 0.4$ . Por otro lado,  $iter_{max}$  corresponde al número máximo de iteraciones del algoritmo y  $k$  hace referencia al número actual de iteraciones.



## Capítulo 4 Modelo propuesto

Para obtener un modelo de predicción para el problema propuesto se utilizó información obtenida del Servicio Nacional de Pesca ([www.sernapesca.cl](http://www.sernapesca.cl)). Esta información contiene los niveles de captura mensuales de sardina en la zona norte de Chile. Estos datos, medidos en tonelada, comprenden el periodo entre Enero de 1976 a Diciembre de 2007, sumando un total de 384 datos.

El modelo propuesto en esta investigación toma los datos que corresponden a los niveles de captura mensual de sardinas y los divide en un grupo correspondiente a la Alta Frecuencia (HF) y otro correspondiente a la Baja Frecuencia (LF). Cabe mencionar que la data histórica será entregada ya dividida, es decir, los datos ya vienen separados en grupos de alta y baja frecuencia, cada uno con 384 datos.

Ya teniendo dividida y normalizada la data en el rango  $[0,1]$ , se aplicará la RNA en ambos grupos, la función de activación que utilizará la RNA será la función Wavelet Derivada de la Gaussiana, y para su entrenamiento, se aplicará el algoritmo PSO. El primer grupo corresponde a los datos de HF y el segundo a los datos de LF. Cada etapa entregará un “pronosticador”, correspondiente a la mejor solución entregada por la RNA, los cuales se sumarán y se compararán con el valor del dato deseado ( $X$ ) para evaluar el error del modelo. En la figura 4.1 se visualiza el modelo en forma general.

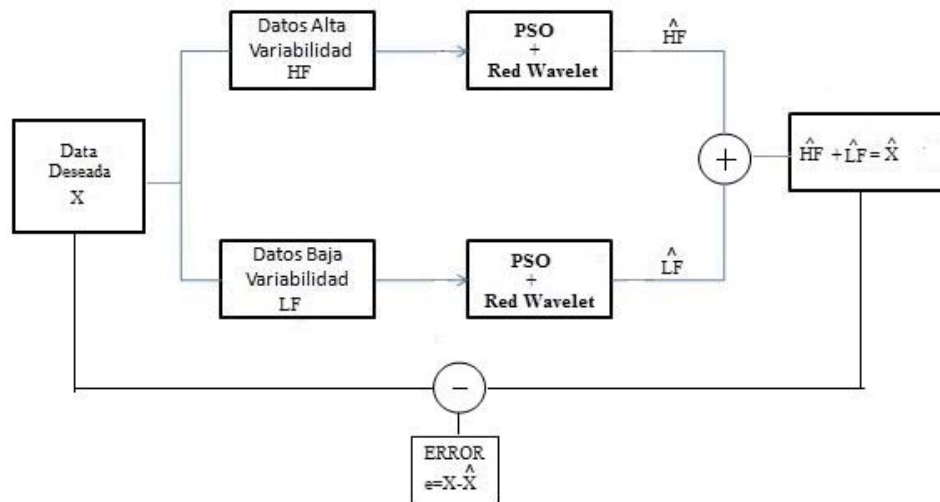


Figura 4 -1 Descripción del Modelo

En ambos grupos se obtuvieron en cada uno dos subconjuntos, uno de ellos fue utilizado para el entrenamiento de la red, y el segundo conjunto fue utilizado para la validación del modelo propuesto. Para el primer subconjunto de datos se utilizó el 65% del total, y para el segundo subconjunto, se utilizó el 35% restante.

## 4.1 Métricas de Evaluación

En el presente estudio, para la selección de las topologías y, por consiguiente, de los modelos que presentan mejores resultados se utilizaron un conjunto de métricas de exactitud calculadas entre los datos observados (valores reales) y los datos pronosticados (entregados por la red). Estas métricas se nombran a continuación.

- Raíz Error Cuadrático Medio (RMSE). Consiste en la suma de las diferencias entre los datos observados y los datos proyectados por el modelo.

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (d_i - y_i)^2}{N}} \quad (4.1)$$

- Coeficiente de Determinación ( $R^2$ ). Mide la dependencia entre los datos reales y los pronosticados. El 0 muestra independencia y el 1 lo contrario.

$$R^2 = 1 - \frac{\sum_{i=1}^N (d_i - y_i)^2}{\sum_{i=1}^N (d_i - \bar{d})^2} \quad (4.2)$$

- Error Absoluto Medio (MAE). Se usa para medir que tan cerca está el valor estimado del valor real o esperado. Es un promedio de todos los errores absolutos.

$$MAE = \frac{\sum_{i=1}^N |d_i - y_i|}{N} \quad (4.3)$$

- Porcentaje de Error Absoluto Medio Escalado (SMAPE). Es una alternativa a MAPE cuando la demanda de los datos es cero o cercano a cero. La diferencia absoluta entre  $d_i$  y  $y_i$  se divide por la mitad de la suma del valor real  $d_i$  y el valor pronosticado de  $y_i$ .

$$SMAPE = \frac{\sum_{i=1}^N \frac{|d_i - y_i|}{(d_i + y_i)/2}}{N} \times 100 \quad (4.4)$$

Se define para las formulas presentadas a  $d_i$  al valor observado en el mes  $i$ ,  $y_i$  al valor pronosticado en el mes  $i$ ,  $\bar{d}$  como la media de la data observada y  $N$  como el número total de meses medidos.

Finalmente, cabe mencionar que para implementar los modelos y sus correspondientes gráficos se utilizó el software MATLAB.

## 4.2 Selección de topología y parámetros del modelo

El método de selección de las configuraciones de prueba fueron hechas empleando la siguiente modalidad:

- **Neuronas de Entrada:** Para obtener el número óptimo de neuronas de entradas primero se fija el número de neuronas ocultas en **10**, el número de iteraciones en **1000** y finalmente el número de partículas en **20**, a continuación se prueba el número de entradas desde un rango de **2 - 12** y el que presente mejor **MSE** será el indicado.
- **Neuronas Ocultas:** Para obtener el número óptimo de neuronas ocultas, se emplea el mismo mecanismo que se utilizó para las neuronas de entrada con la diferencia que éstas se fijan con la mejor obtenida en el proceso anterior, luego se varían las neuronas ocultas desde un rango de **1-10** dejando la que presente el mejor **MSE**.
- **Número de Iteraciones:** Luego de obtener el número óptimo de neuronas de entrada y ocultas, además del número de partículas fijo en **20**, se varía el número de iteraciones en un rango de **100 – 3000**, dejando la que presente el mejor **MSE**.
- **Número de Partículas:** Finalmente al obtener el número óptimo de neuronas de entrada y ocultas, además del número óptimo de iteraciones se varía el número de Partículas en un rango de **10-40**, dejando la que presente el mejor **MSE**.

### 4.2.1 Selección de topología para Alta Frecuencia

#### 4.2.1.1 Neuronas de Entrada

La siguiente figura muestra el número de entradas óptimo para este modelo, que corresponden a **4 nodos de entrada**.

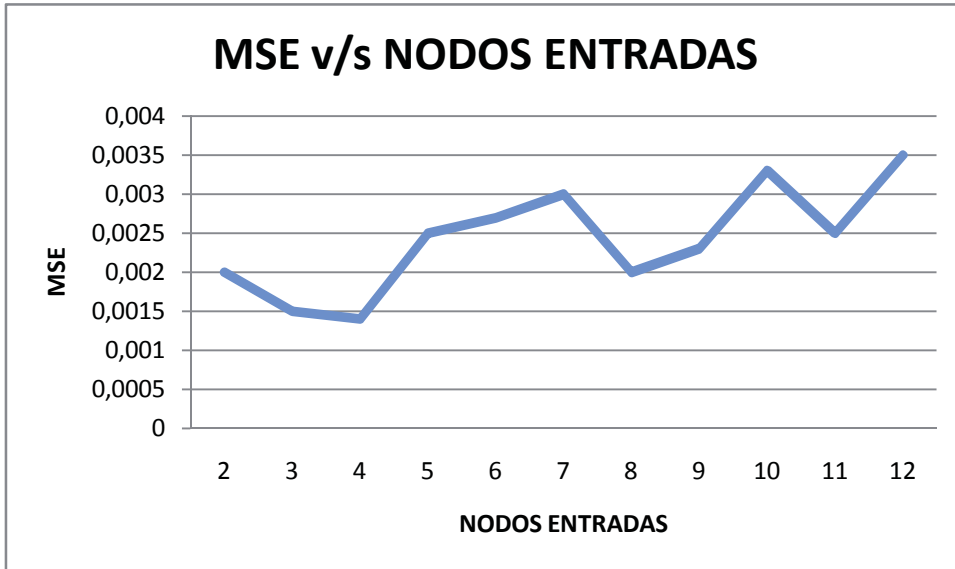


Figura 4-2 MSE v/s Nodos Entradas HF

#### 4.2.1.2 Neuronas Ocultas

La siguiente figura muestra el número de neuronas ocultas que tiene el mejor MSE, que corresponden a **3 nodos ocultos**.

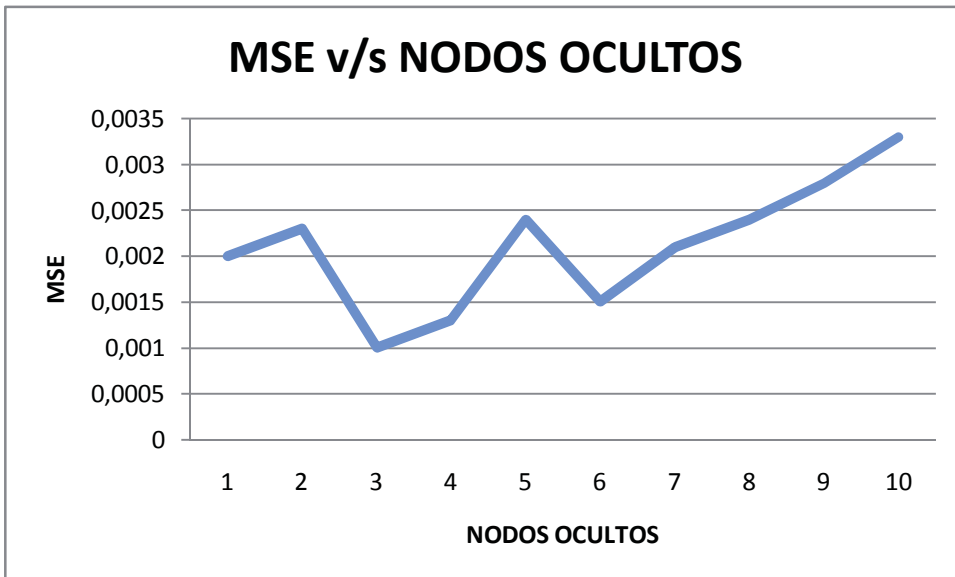


Figura 4-3 MSE v/s Nodos Ocultos HF

### 4.2.1.3 Número de Iteraciones

La siguiente figura muestra el número de iteraciones óptimo para este modelo que corresponde a **2000 iteraciones**.

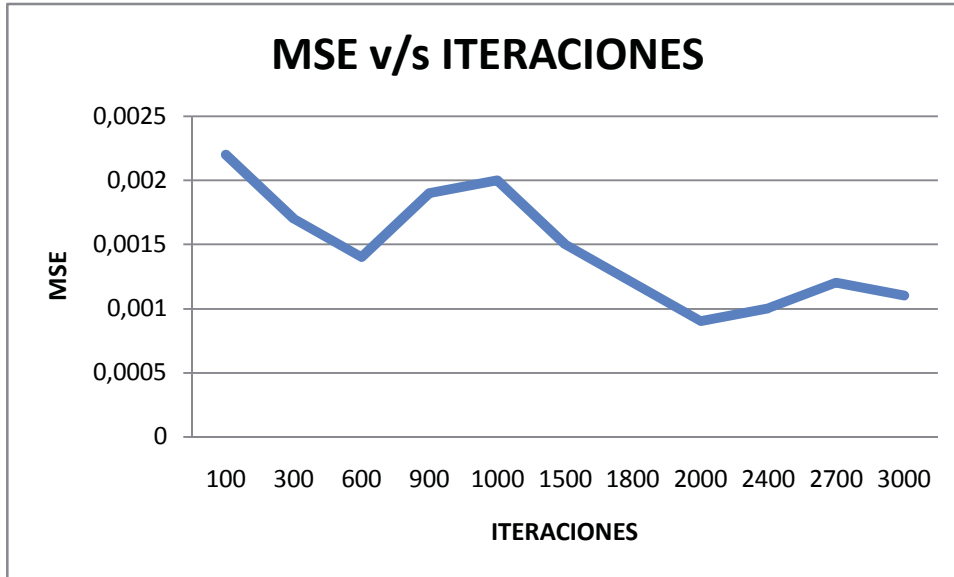


Figura 4-4 MSE v/s Iteraciones HF

### 4.2.1.4 Número de Partículas

La siguiente figura muestra el número de partículas óptimo para este modelo, que corresponden a **30 partículas**.

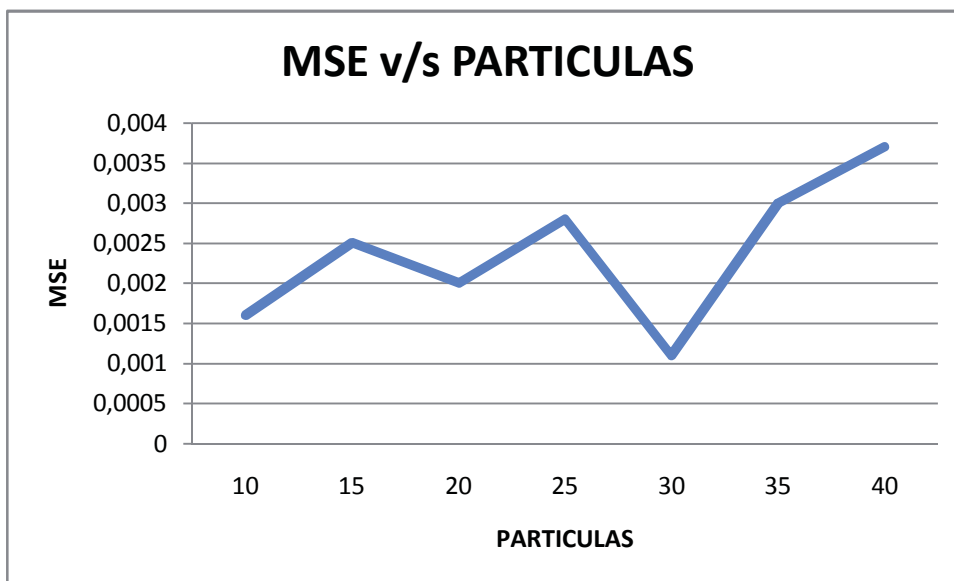


Figura 4-5 MSE v/s Partículas HF

## 4.2.2 Selección de topología para Baja Frecuencia

### 4.2.2.1 Neuronas de Entrada

La siguiente figura muestra el número de entradas óptimo para este modelo, que corresponden a **4 nodos de entrada**.

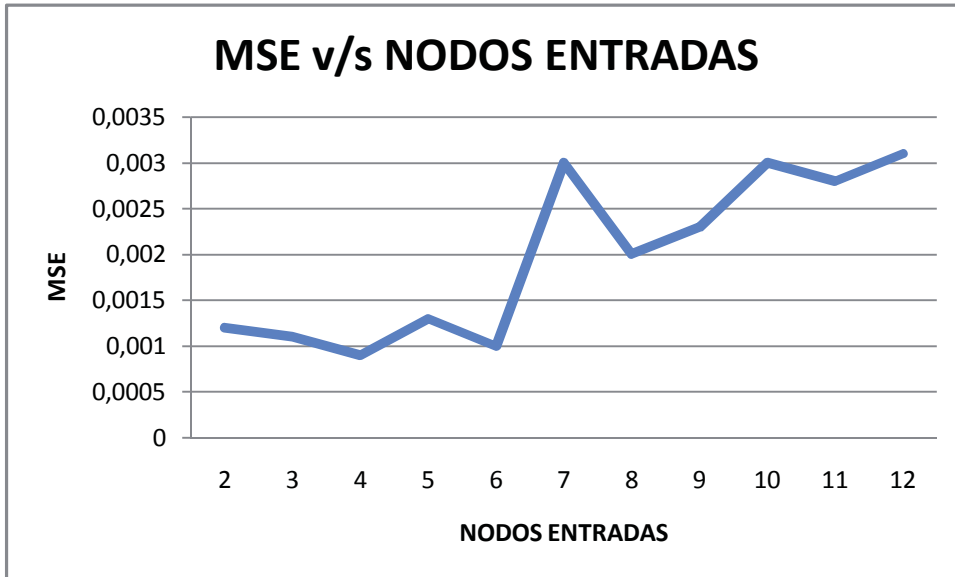


Figura 4-6 MSE v/s Nodos Entradas LF

### 4.2.2.2 Neuronas Ocultas

La siguiente figura muestra el número de neuronas ocultas que tiene el mejor MSE, que corresponden a **3 nodos ocultos**.

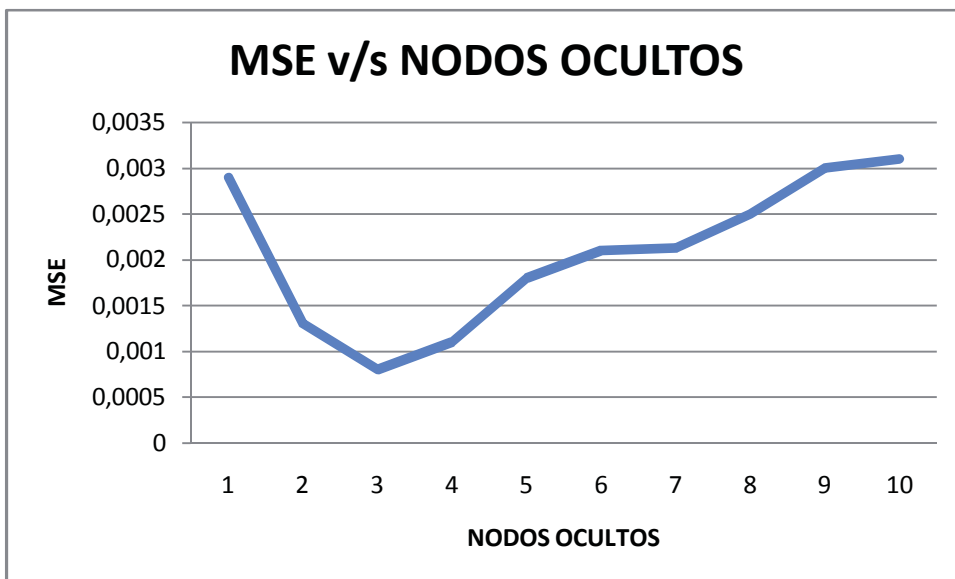


Figura 4-7 MSE v/s Nodos Ocultos LF

### 4.2.2.3 Número de Iteraciones

La siguiente figura muestra el número de iteraciones óptimo para este modelo que corresponde a **2000 iteraciones**.

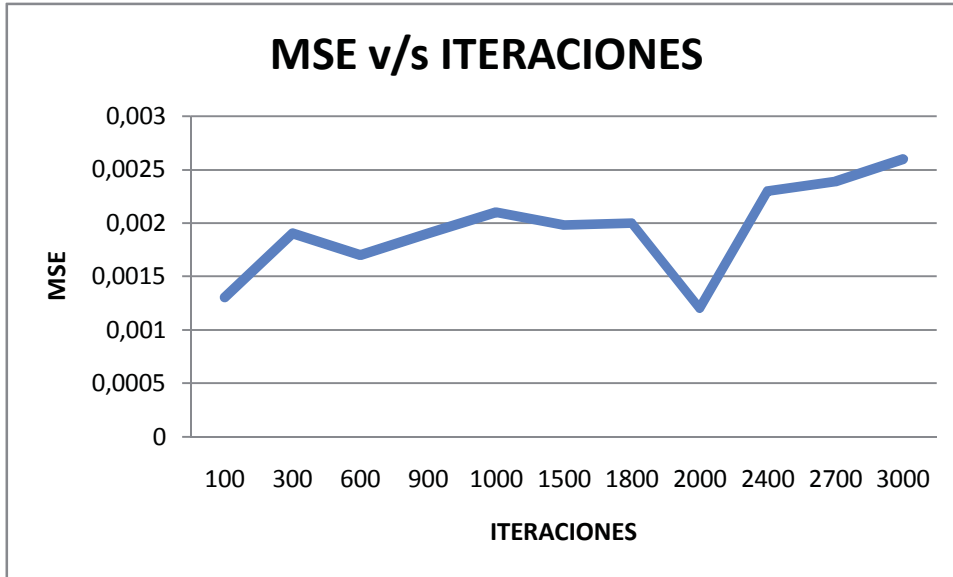


Figura 4-8 MSE v/s Iteraciones LF

### 4.2.2.4 Número de Partículas

La siguiente figura muestra el número de partículas óptimo para este modelo, que corresponden a **30 partículas**.

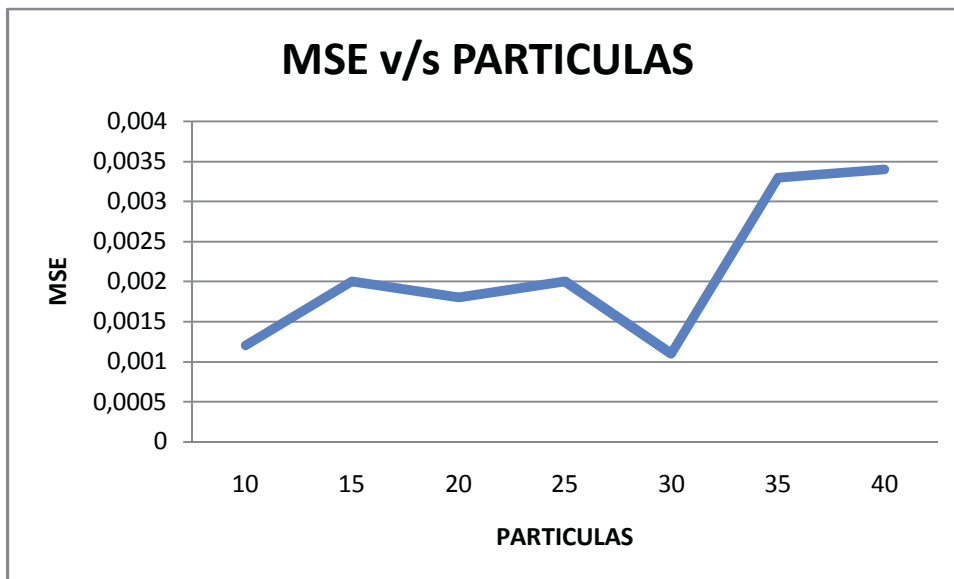


Figura 4-9 MSE v/s Partículas LF

### 4.2.3 Selección parámetros PSO

La configuración elegida para PSO se basó en iniciar con una configuración que favoreciera la exploración y conforme avanzaban las iteraciones se iba disminuyendo y aumentando la explotación. El decaimiento fue aplicado al parámetro peso de inercia  $w$  de manera lineal, además a los parámetros  $c_1$  y  $c_2$  se les otorgó la misma importancia.

$$W = W_{min} - \left( \frac{W_{min} - W_{max}}{MaxIter} \right) \times k \quad (4.5)$$

Donde  $W_{max} = 0.9$ ,  $W_{min} = 0.4$ ,  $c_1 = c_2 = 1.8$ ,  $MaxIter$  = número máximo de iteraciones y  $k$  es la  $k$ -ésima iteración.

### 4.3. Análisis de Resultados

A continuación se analizarán los resultados obtenidos de la mejor configuración de la RNA, tanto para los datos de HF como LF y ambas componentes sumadas.

#### 4.3.1. RNW-PSO Alta Frecuencia (HF)

Nodos Entradas	4
Nodos Ocultos	3
Partículas	30
Iteraciones	2000

Tabla 4-1 Mejor Configuración HF

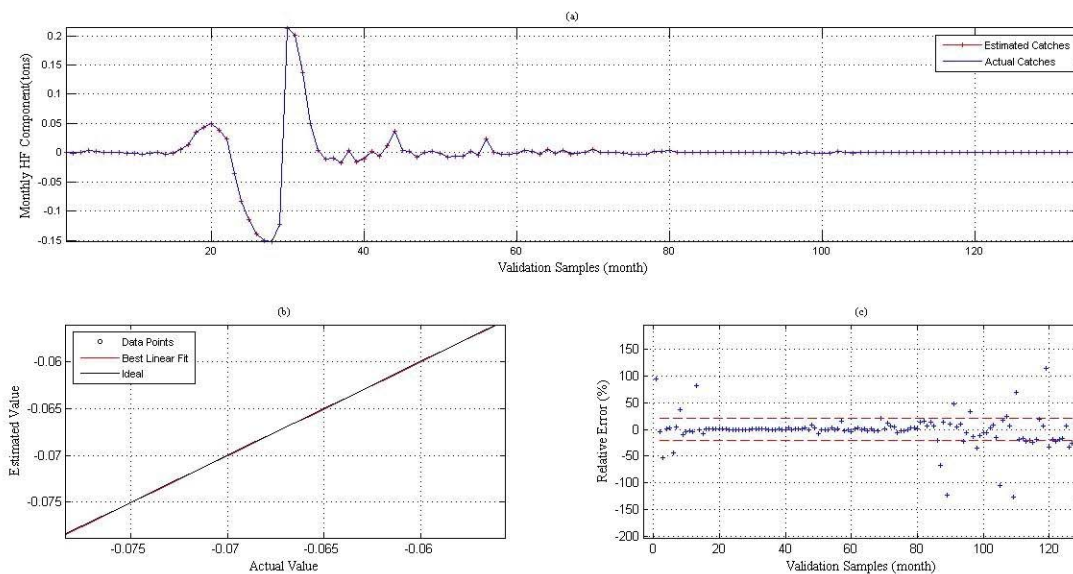


Figura 4-10 Gráficos observados para HF



En la figura 4-10, en el gráfico superior se observa que la curva pronostica por la red es muy similar a la curva deseada, lo que garantiza un excelente pronóstico entregando un valor de  $RMSE = 0.000049$ . Esto se reafirma con el gráfico de la izquierda que muestra al Gráfico de Correlación con un coeficiente de determinación ( $R^2$ ) = 99,99 %, y un valor  $SMAPE = 4,24\%$ .

### 4.3.2. RNW-PSO Baja Frecuencia (LF)

Nodos Entradas	4
Nodos Ocultos	3
Partículas	30
Iteraciones	2000

Tabla 4-2 Mejor Configuración LF

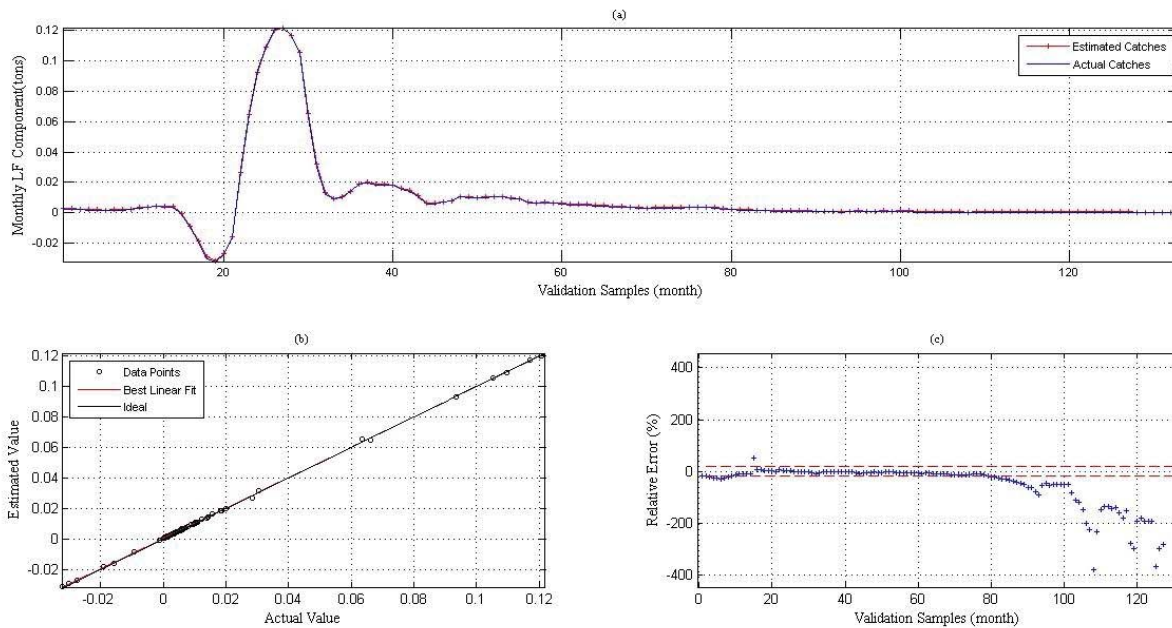
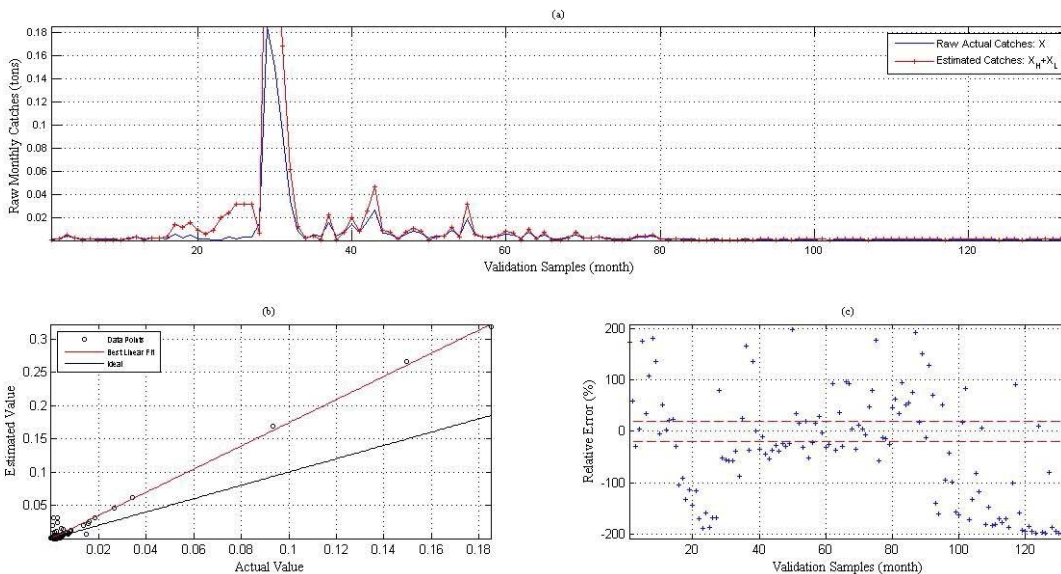


Figura 4-11 Gráficos Observados para LF

En la figura 4-11, el gráfico superior muestra a la curva pronosticada similar a la curva deseada, aunque a partir de la muestra 60 app. la curva pronosticada tiende a desviarse un poco, entregando un valor de  $RMSE = 0.000510$ , un  $R^2 = 99,98\%$  y un aumento también en el error  $SMAPE = 9,97\%$ .

### 4.3.3. RNW-PSO (HF+LF)



**Figura 4-12 Gráficos Observados para la suma de las componentes (HF + LF)**

En la figura 4-12, muestra los gráficos observados al momento de sumar las componentes HF y LF. Como se aprecia en el gráfico superior, la curva pronosticada por la red pierde precisión en los peaks presentados por la curva deseada, haciendo que disminuya sutilmente el coeficiente de determinación  $R^2 = 97,56\%$ , aumente el RMSE = 0,02123 y un aumento en el SMAPE = 18,85%

### 4.3.4. RNW –QPSO

El algoritmo QPSO presenta sólo un parámetro configurable,  $\beta$ , que varía en el intervalo [0.5, 1.0] El valor del coeficiente de expansión-contracción  $\beta_i$  queda en función de la  $i$ -ésima iteración y del máximo de iteraciones. La mejor configuración, determinada mediante ensayo y error, corresponde a 2 nodos de entrada, 4 nodos ocultos y 25 partículas, con un máximo de 1500 iteraciones. Para visualizar los gráficos RNW-QPSO, diríjase al anexo A de este informe adjuntado al final del mismo.

### 4.3.5. RNW –EPSO

El algoritmo EPSO, para llevar a cabo una búsqueda más rápida, varía el parámetro peso de inercia  $W$  exponencialmente, único parámetro configurable. La mejor configuración, determinada mediante ensayo y error, corresponde a 8 nodos de entrada, 6 nodos ocultos y 20 partículas, con un máximo de 500 iteraciones. Para visualizar los gráficos RNW-EPSO, diríjase al anexo B de este informe adjuntado al final del mismo.

## 4.4. Comparación de rendimiento de los resultados.

### 4.4.1. Etapa Alta Frecuencia

Métricas	RNW-PSO	RNW-QPSO	RNW-EPSO
R <sup>2</sup>	99.993494	99.994038	99.328542
RMSE	0.000049	0.015400	0.077391
MAE	0.000049	0.015397	0.077100
MAPE	27.417197	8453.222875	14218.624730
SMAPE	4.246215	24.416216	52.633802

Tabla 4-3 Resultados RNW-PSO v/s RNW-QPSO v/s RNW-EPSO (HF)

### 4.4.2. Etapa Baja Frecuencia

Métricas	RNW-PSO	RNW-QPSO	RNW-EPSO
R <sup>2</sup>	99.985286	99.242449	99.259534
RMSE	0.000510	0.061130	0.013184
MAE	0.000458	0.059637	0.012885
MAPE	40.250682	12340.824039	2627.290545
SMAPE	9.976731	18.3965067	48.926027

Tabla 4-4 Resultados RNW-PSO v/s RNW-QPSO v/s RNW-EPSO (LF)

### 4.4.3. Etapas sumadas (HF + LF)

Métricas	RNW-PSO	RNW-QPSO	RNW-EPSO
R <sup>2</sup>	97.563866	96.289873	87.563867
RMSE	0.021230	0.022714	0.095707
MAE	0.014540	0.010657	0.092443
MAPE	1931.212772	29402.569212	40897.547330
SMAPE	18.859796	37.638057	67.316044

Tabla 4-5 Resultados RNW-PSO v/s RNW-QPSO v/s RNW-EPSO (HF+LF)

Al comparar los resultados de los tres modelos anteriores, se observa que el modelo RNW-PSO presenta mejor desempeño que las otras dos variantes escogidas para llevar a cabo el objetivo de esta investigación. Probablemente se den estos resultados debido a la configuración de los parámetros de PSO, ya que estos parámetros fueron elegidos al azar apoyado por la literatura, y estuvieron mucho más acertados que los otros dos modelos. El que presenta resultados más alejados fue el modelo RNW-EPSO, si bien es cierto esto, este modelo presentó convergencia a la iteración 500 en la fase de entrenamiento a diferencia de los otros dos modelos, sin embargo, puede ser que la naturaleza y el comportamiento de los datos no son adecuados para este modelo.

Como se puede apreciar el valor MAPE para los tres modelos es bastante elevado, esto se debe a que los datos aplicados a los modelos son iguales o muy cercanos al valor cero. Para solucionar este problema es conveniente utilizar el SMAPE, en donde en el caso de RNW-PSO para las componentes sumadas presenta un valor de 18,85%, obteniendo una ganancia de 18,78% con respecto a RNW-QPSO y un 48,46% con respecto a RNW-EPSO.

## Conclusiones

En la presente memoria de título, se investigó, desarrolló y evaluó un modelo de pronóstico para la captura de sardinas en la zona norte de Chile que comprende el periodo de Enero de 1976 a Diciembre 2007, este modelo está conformado por una Red Neuronal de tres capas, una de entrada, una oculta y una de salida, con función de activación Wavelet Derivada de la Gaussiana. El algoritmo de aprendizaje aplicado a la red consistió en algoritmo PSO.

En base al estado del arte de las Redes Neuronales investigado en este proyecto, junto con las aplicaciones encontradas en la literatura, han dejado claro que las RNA corresponden a un modelo computacional capaz de resolver una gran cantidad de problemas, y dependiendo de lo que se desee resolver se debe configurar una red que satisfaga dichos requerimientos.

Las RNA han sido una herramienta potente en la resolución de problemas aplicados al pronóstico, dada sus excelentes características. Desde el trabajo pionero de Lapedes y Farber, numerosos trabajos y artículos han sido expuestos para analizar y dar solución a problemas que involucren series temporales, en donde la mayoría son caóticas, ruidosas y no lineales. Áreas tales como las matemáticas, ingeniería, física, economía, han visto grandes avances y cosechas gracias a las RNA

En los últimos quince años, las técnicas de computación evolutiva se han consolidado como un conjunto de metaheurísticas efectivas y robustas como consecuencia de su alta aplicabilidad a la resolución de problemas en múltiples áreas. Es debido a esto que PSO, surge como una técnica potente y ventajosa al momento de la resolución de problemas de predicción. Aunque una de sus mayores desventajas es el poco fundamento matemático en lo que respecta a la elección de sus parámetros, debido a que se configuran aleatoriamente, por lo que la inspiración divina muchas veces se hace presente al momento de encontrar la solución óptima. Obviamente como todo problema de optimización, la existencia de varias soluciones al problema hacen que la tarea de encontrar el óptimo global sea más difícil, porque existe la posibilidad de que el algoritmo converja hacia un óptimo local y no pueda salir de ese sub-espacio de soluciones. Es por esto que el enjambre de partículas presenta algunas variaciones en su funcionamiento con el afán de recorrer todo el espacio de solución en busca del óptimo global sin caer en los óptimos locales. Es por esto que en esta investigación se escogieron dos variantes de PSO: QPSO y EPSO.

Luego del análisis de los distintos modelos propuestos, se puede concluir que el modelo RNW-PSO obtiene mejores resultados que las otras variantes aplicadas, RNW-QPSO y RNW-EPSO. Estos resultados se observaron en las métricas de rendimiento, porque en lo que respecta a la velocidad de convergencia el modelo RNW-EPSO obtuvo un mejor desempeño, no obstante, el que haya presentado mejor desempeño en la velocidad de convergencia no mejora los resultados en el modelo predictivo.

Además, cabe mencionar que el actual proyecto ha sido parte del proyecto Fondecyt Regular N°1131105, por lo que los resultados obtenidos son de carácter relevante para el desarrollo científico y tecnológico del país.

Se concluye también que los resultados obtenidos de esta investigación permiten apoyar la toma de decisiones y planificación para el gobierno de Chile, y de esta manera establecer cuotas anuales de desembarque con el fin de permitir la existencia de la especie en el futuro.

Finalmente, se concluye que las Redes Neuronales Wavelet sí son una alternativa para obtener buena calidad en pronóstico frente a la necesidad de predecir el nivel de captura mensual de sardinas en el norte de Chile, presentándose al modelo RNW-PSO con un 97,56% de la varianza explicada y un 18,78% de SMAPE como el mejor modelo de esta investigación.

## Referencias bibliográficas

- [1] Leake, D., *Artificial Intelligence*, Van Nostrand Scientific Encyclopedia, ninth Edition, 2002.
- [2] Lu, W. Z.; Fan, H. Y.; Lo, S. M.; *Neurocomputing* 2003, 51, 387.
- [3] Asis Kumar Tripathy, Suvendu Mohapatra, Shradhananda Beura, Gunanidhi Pradhan. *Weather Forecasting using ANN and PSO*. International Journal of Scientific & Engineering Research Volume 2, Issue 7, July-2011.
- [4] Cañón, J. “*El Niño 1997-1998: Sus efectos en el sector pesquero industrial de la zona norte de Chile*”. Corpesca.
- [5] Glantz, M., R. W. Kats & N. Nicholls (1991). *Teleconnections linking worldwide climate anomalies*. Cambridge University Press. Cambridge.
- [6] Chavez, F. P., Ryan, J., Lluch-Cota, S., Ñiquen, M. (2003). *From anchovies to sardines and back: multidecadal change in the Pacific Ocean*. *Science* 299: 217-221.
- [7] Yañez E., M. A. Barbieri & L. Santillán. (1992). *Long term environmental variability and pelagic fisheries in Talcahuano, Chile*. En Paine et al., Editors. *Benguela Trophic Functioning*. *S. Afr. J. Mar. Sci.* 12, 175-188.
- [8] Haykin, S. (1999). *Neural Networks: A Comprehensive Foundation*.
- [9] Hu, M.J.C., 1964. Application of the adaline system to weather forecasting. Master Thesis, Technical Report 6775-1, Stanford Electronic Laboratories, Stanford, CA, June.
- [10] Soria, E. y Blanco, A. (2001). *Redes Neuronales Artificiales*. ACTA.
- [11] Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning internal representation by back-propagating errors. In: Rumelhart, D.E., McClelland, J.L., the PDP Research Group (Eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. MIT Press, MA.
- [12] Werbos, P.J., 1974. Beyond regression: new tools for prediction and analysis in the behavioral sciences. Ph.D. thesis, Harvard University.
- [13] Werbos, P.J., 1988. Generalization of back propagation with application to a recurrent gas market model. *Neural Networks* 1, 339–356.
- [14] Lapedes A.& Farber R. (1987) "Nonlinear signal processing using neural networks: prediction and system modeling". Proc. Conference on Neural Information Processing Systems - Natural and Synthetic, IEEE, November.

- [15] Weigend, A.S., Huberman, B.A., Rumelhart, D.E., 1990. Predicting the future: A connectionist approach. *International Journal of Neural Systems* 1, 193–209.
- [16] Weigend, A.S., Huberman, B.A., Rumelhart, D.E., 1992. Predicting sunspots and exchange rates with connectionist networks. In: Casdagli, M., Eubank, S. (Eds.), *Nonlinear Modeling and Forecasting*. Addison-Wesley, Redwood City, CA, pp. 395– 432.
- [17] Cottrell, M., Girard, B., Girard, Y., Mangeas, M., Muller, C., 1995. Neural modeling for time series: a statistical stepwise method for weight elimination. *IEEE Transactions on Neural Networks* 6 (6), 1355–1364.
- [18] Tang, Z., Almeida, C., Fishwick, P.A., 1991. Time series forecasting using neural networks vs Box-Jenkins methodology. *Simulation* 57 (5), 303–310.
- [19] Tang, Z., Fishwick, P.A., 1993. Feedforward neural nets as models for time series forecasting. *ORSA Journal on Computing* 5 (4), 374–385.
- [20] Sharda, R., Patil, R.B., 1992. Connectionist approach to time series prediction: An empirical test. *Journal of Intelligent Manufacturing* 3, 317–323.
- [21] Weigend, A.S., Gershenfeld, N.A., 1993. *Time Series Prediction Forecasting the Future and Understanding the Past*. Addison-Wesley, Reading, MA.
- [22] Hill, T., Marquez, L., O'Connor, M., Remus, W., 1994. Artificial neural networks for forecasting and decision making. *International Journal of Forecasting* 10, 5–15.
- [23] Marquez, L., Hill, T., O'Connor, M., Remus, W., 1992. Neural network models for forecast a review. In: *IEEE proceedings of the 25th Hawaii International Conference on System Sciences.*, 4, pp. 494–498.
- [24] White, H., 1989. Learning in artificial neural networks: A statistical perspective. *Neural Computation* 1, 425–464.
- [25] Ripley, B.D., 1993. Statistical aspects of neural networks. In: Barndorff-Nielsen, O.E., Jensen, J.L., Kendall, W.S. (Eds.), *Networks and Chaos-Statistical and Probabilistic Aspects*. Chapman and Hall, London, pp. 40–123.
- [26] Cybenko, G., 1989. Approximation by superpositions of a sigmoidal function. *Mathematical Control Signals Systems* 2, 303–314.
- [27] Irie, B., Miyake, S., 1988. Capabilities of three-layered perceptrons. In: *Proceedings of the IEEE International Conference on Neural Networks, I*, pp. 641–648.
- [28] Box, G. and Jenkins, G. (1970) *Time series analysis: Forecasting and control*, San Francisco: Holden-Day.



- [29] Pankratz, A., 1983. Forecasting with Univariate Box-Jenkins Models: Concepts and Cases. John Wiley, New York.
- [30] Granger, C.W.J., Terasvirta, T., 1993. Modelling Nonlinear Economic Relationships. Oxford University Press, Oxford.
- [31] Lowe, D., Webb, A.R., 1990. Time series prediction by adaptive networks: A dynamical systems perspective. IEE proceedings-F138 (1), 17–24.
- [32] Deppisch, J., Bauer, H.-U., Geisel, T., 1991. Hierarchical training of neural networks and prediction of chaotic time series. Physics Letters 158, 57–62.
- [33] Li, M., Mehrotra, K., Mohan, C., Ranka, S., 1990. Sunspots numbers forecasting using neural networks. In: Proceedings of the 5th IEEE International Symposium on Intelligent Control, pp. 524–529.
- [34] De Groot, C., Wurtz, D., 1991. Analysis of univariate time series with connectionist nets: a case study of two classical examples. Neurocomputing 3, 177–192.
- [35] Ginzburg, I., Horn, D., 1992. Learning the rule of a time series. International Journal of Neural Systems 3 (2), 167–177.
- [36] Odom, M D & Sharda, R. (1990). "A neural network model for bankruptcy prediction". Proc. IJCNN International Joint Conference on Neural Networks. San Diego. II: 163-168.
- [37] Coleman, K.G., Graettinger, T.J., Lawrence, W.F., 1991. Neural networks for bankruptcy prediction: The power to solve financial problems. AI Review 5, July /August, 48–50.
- [38] Wilson, R., Sharda, R., 1994. Bankruptcy prediction using neural networks. Decision Support Systems 11, 545–557.
- [39] Borisov, A.N., Pavlov, V.A., 1995. Prediction of a continuous function with the aid of neural networks. Automatic Control and Computer Sciences 29 (5), 39–50.
- [40] Hann, T.H., Steurer, E., 1996. Much ado about nothing? Exchange rate forecasting: Neural networks vs. linear models using monthly and weekly data. Neurocomputing 10, 323–339.
- [41] Kimoto, T., Asakawa, K., Yoda, M. and Takeoka, M. (1990). "Stock market prediction system with modular neural networks". Proc. IJCNN International Joint Conference on Neural Networks. San Diego. I: 1-6.
- [42] Grudnitski, G., Osburn, L., 1993. Forecasting S and P and gold futures prices: An application of neural networks. The Journal of Futures Markets 13 (6), 631–643.
- [43] Park, J., Sandberg, I.W., 1993. Approximation and radial basis function networks. Neural Computation 5, 305–316.



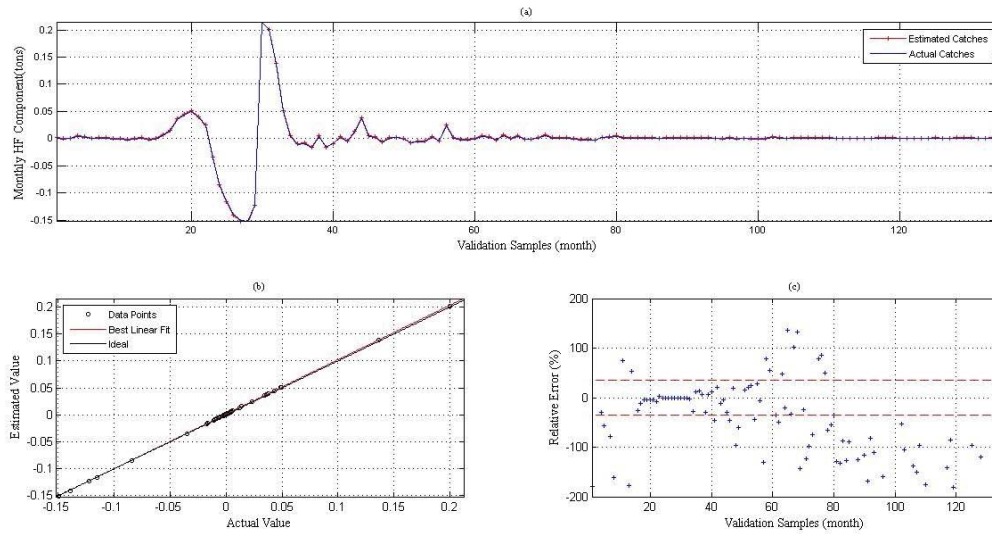
- [44] Bacha, H., Meyer, W., 1992. A neural network architecture for load forecasting. In: Proceedings of the IEEE International Joint Conference on Neural Networks, 2, pp. 442–447
- [45] Makridakis, S., Anderson, A., Carbone, R., Fildes, R., Hibdon, M., Lewandowski, R. et al., 1982. The accuracy of extrapolation (time series) methods: Results of a forecasting competition. *Journal of Forecasting* 1 (2), 111–153.
- [46] Weigend, A.S., Gershenfeld, N.A., 1993. *Time Series Prediction: Forecasting the Future and Understanding the Past*. Addison-Wesley, Reading, MA.
- [47] Arizmendi, C.M., Sanchez, J.R., Ramos, N.E., Ramos, G.I., 1993. Time series prediction with neural nets: Application to airborne pollen forecasting. *International Journal of Biometeorology* 37, 139–144.
- [48] Kohzadi, N., Boyd, M.S., Kermanshahi, B., Kaastra, I., 1996. A comparison of artificial neural network and time series models for forecasting commodity prices. *Neurocomputing* 10, 169– 181.
- [49] Balestrino, A., Bini Verona, F., Santanche, M., 1994. Time series analysis by neural networks: Environmental temperature forecasting. *Automazione e Strumentazione* 42 (12), 81–87.
- [50] Haas, D.J., Milano, J., Flitter, L., 1995. Prediction of helicopter component loads using neural networks. *Journal of the American Helicopter Society* 40 (1), 72–82.
- [51] Nam, K., Schaefer, T., 1995. Forecasting international airline passenger traffic using neural networks. *Logistics and Transportation* 31 (3), 239–251.
- [52] Maasoumi, E., Khotanzad, A., Abaye, A., 1994. Artificial neural networks for some macroeconomic series: A first report. *Econometric Reviews* 13 (1), 105–122.
- [53] Ruiz-Suarez, J.C., Mayora-Ibarra, O.A., Torres-Jimenez, J., Ruiz-Suarez, L.G., 1995. Short-term ozone forecasting by artificial neural networks. *Advances in Engineering Software* 23, 143–149.
- [54] Huntley, D.G., 1991. Neural nets: An approach to the forecasting of time series. *Social Science Computer Review* 9 (1), 27–38.
- [55] Chang, I., Rapiraju, S., Whiteside, M., Hwang, G., 1991. A neural network to time series forecasting. In: Proceedings of the Decision Science Institute., 3, pp. 1716–1718.
- [56] Karunanithi, N., Grenney, W.J., Whitley, D., Bovee, K., 1994. Neural networks for river flow prediction. *Journal of Computing in Civil Engineering* 8 (2), 201–220.
- [57] Lubero, R.G., 1991. Neural networks for water demand time series forecasting. In: Proceedings of the International Workshop on Artificial Neural Networks, pp. 453–460.

- [58] Turkkan, N., Srivastava, N.K., 1995. Prediction of wind load distribution for air-supported structures using neural networks. *Canadian Journal of Civil Engineering* 22 (3), 453–461
- [59] Villarroel, F. “*Redes neuronales aplicativas para pronóstico de captura de anchoveta en la zona norte de Chile*”. Biblioteca Pontificia Universidad Católica de Valparaíso. Junio 2012.
- [60] Quezada, P. “*Redes RRBF con aprendizaje híbrido para la predicción de captura de anchovetas*”. Biblioteca Pontificia Universidad Católica de Valparaíso. Diciembre 2009.
- [61] Trigo, A. “*Regresiones funcionales neuro-wavelet aplicadas al pronóstico de anchovetas*”. Biblioteca Pontificia Universidad Católica de Valparaíso. Septiembre 2011.
- [62] Zhang, Q. and Benveniste, A. "Wavelet Networks", *IEEE Transactions on Neural Networks*. Vol 3. No 6. July 1992.
- [63] Kennedy, J. and Eberhart, R. (1995). Particle swarm Optimization. In *Proceedings of IEEE International Conference on Neural Networks*.
- [64] Kennedy, J. y Eberhart, R. (2001). *Swarm Intelligence*, The Morgan Kaufmann Series in Artificial Intelligence, Morgan Kaufmann, San Francisco – California, pages 3-80, April 2001.
- [65] A. Abraham, H. Guo y H. Liu, *Swarm Intelligence: Foundations, Perspectives and Applications*, Studies in Computational Intelligence (SCI), Springer-Verlag, vol. 26, pages 3-25, Noviembre 2006.
- [66] Magnus Erik Hvass Pedersen, *Good Parameters for Particle Swarm Optimization*, Hvass Laboratories, Technical Report no.HL1001, 2010.
- [67] J. Gimmler, T. Stutzle, T.E. Exner *Hybrid Particle Swarm Optimization: An Examination of the Influence of Iterative Improvement Algorithms on Performance*, Lecture Notes in Computer Science (LNCS), Springer-Verlag, vol. 4150, pages 72-83, August 2006.
- [68] Xi, M., Sun, J., y Xu, W. “*Parameter optimization of PID controller based on Quantum-Behaved Particle Swarm Optimization Algorithm*,” *Complex Systems and Applications-Modelling*, vol. 14, supplement 2, pp. 603–607, 2007.
- [69] Liu, J., Sun, J. y Xu, W. *Quantum-Behaved Particle Swarm Optimization with Adaptive Mutation Operator*, Lecture Notes in Computer Science (LNCS), Springer-Verlag, vol. 4221, pages 959-967, Septiembre 2006.
- [70] Ghali, N. I., El-Dessouki, N., Mervat, A. N., and Bakrawi, L. (2009). Exponential particle swarm optimization approach for improving data clustering. *International Journal of Electrical, Computer, and Systems Engineering*, 3:208-212.

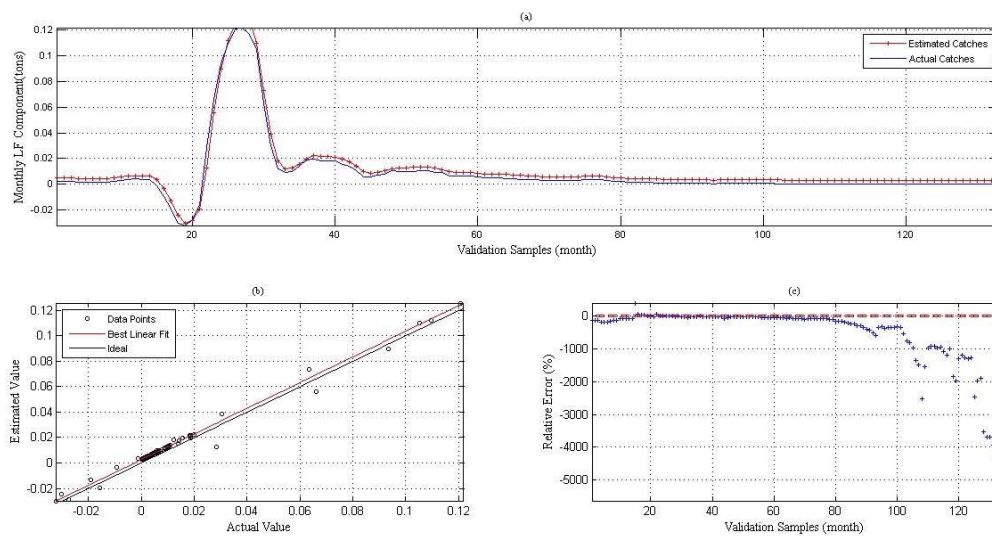
# Anexos

## Anexo 1: Gráficos Observados para RNW-QPSO

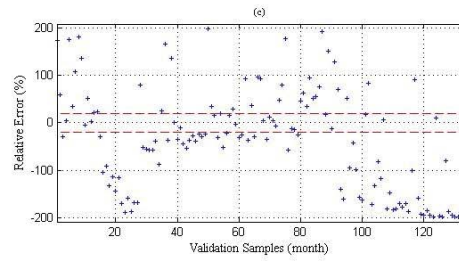
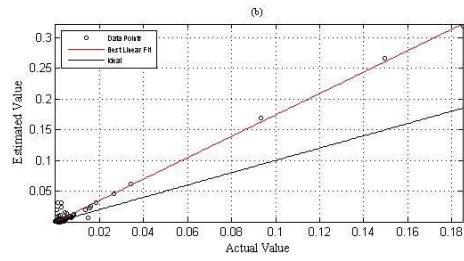
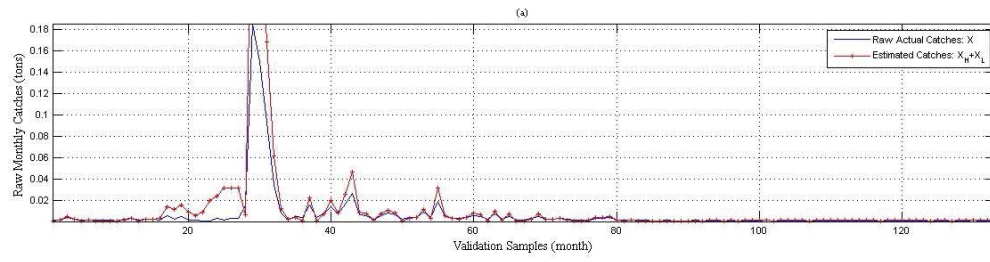
### Gráficos Observados para Alta Frecuencia (HF)



### Gráficos Observados para Baja Frecuencia (LF)

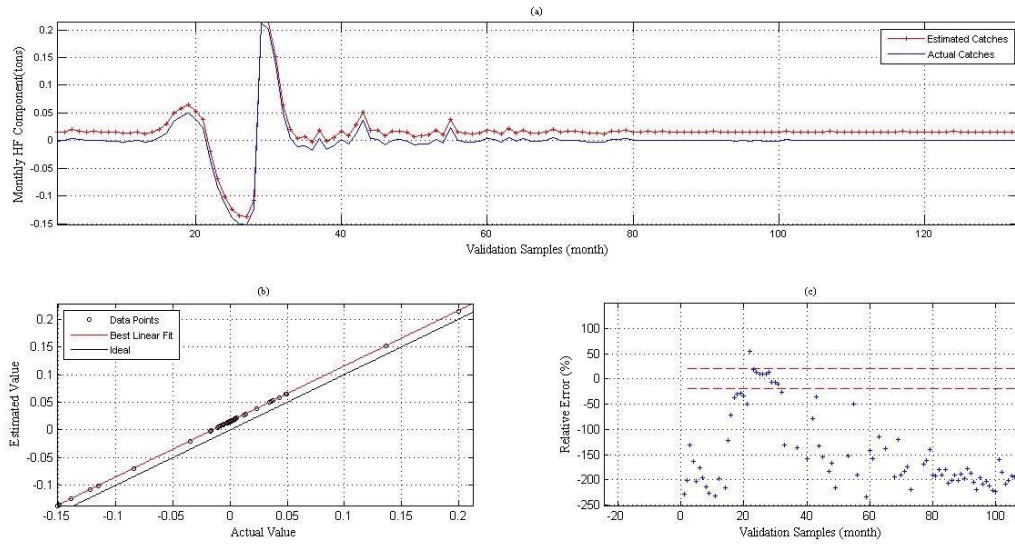


## Gráficos Observados componentes sumadas (HF+LF)

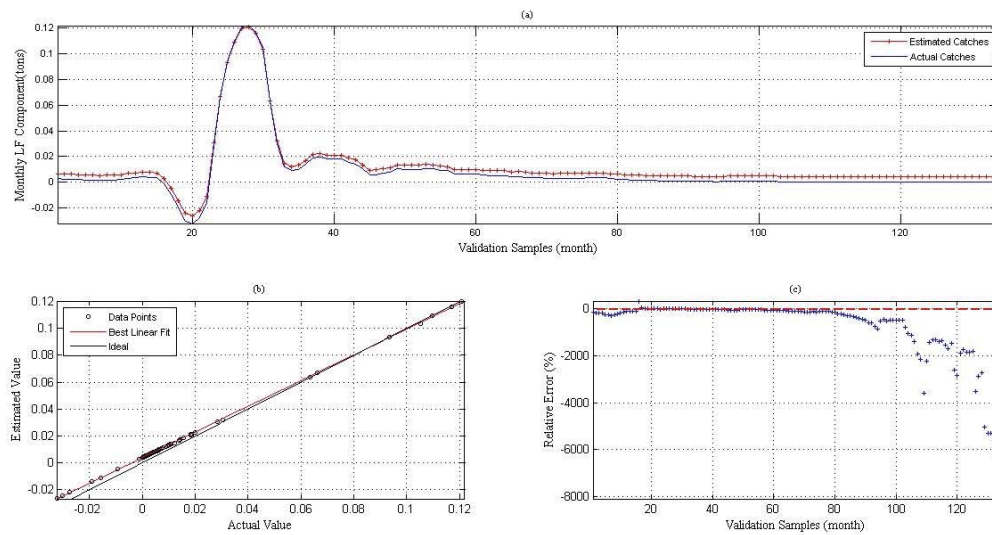


# Anexo 2: Gráficos Observados para RNW-EPPO

## Gráficos Observados para Alta Frecuencia (HF)



## Gráficos Observados para Baja Frecuencia (LF)



## Gráficos Observados para la suma de las componentes (HF+LF)

