

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

**Red Neuronal Sigmoidal para el Pronóstico de la alta y baja
Frecuencia de Captura Mensual de la Anchoveta en la Zona Norte
de Chile usando Inteligencia de Enjambre de Partículas**

FRANCISCO SEBASTIÁN AGUILERA CASTILLO

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

MAYO, 2014

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

**Red Neuronal Sigmoidal para el Pronóstico de la alta y baja
Frecuencia de Captura Mensual de la Anchoqueta en la Zona Norte
de Chile usando Inteligencia de Enjambre de Partículas**

FRANCISCO SEBASTIÁN AGUILERA CASTILLO

PROFESOR GUÍA
NIBALDO RODRIGUEZ AGURTO
PROFESOR CORREFERENTE
WENCESLAO PALMA MUÑOZ

MAYO, 2014

Dedicatoria

Dedicado a mi madre, a mi padre y hermano por ser los pilares fundamentales en mi vida, por su apoyo incondicional, paciencia, amor y sabiduría.

En la memoria de Luis Alberto Aguilera Latín (*Q.E.P.D.*)

Agradecimientos

A Dios y a la Virgen de Andacollo, a mi familia, a la familia Arcos Ugalde, a mis amigos, y Profesor Guía.

Resumen

El sector pesquero entrega una importante contribución a las exportaciones nacionales, 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. Con el objetivo de la conservación de estos recursos, las entidades gubernamentales del país han establecido regulaciones en la captura de estas especies.

Este proyecto tiene como finalidad servir de apoyo en la fijación de las cuotas mensuales de anchoveta para la zona Norte de Chile. Esta investigación ha desarrollado una Red Neuronal Sigmoidal que utiliza la Optimización de Enjambre de Partículas (PSO), modelo que analiza los datos de alta y baja frecuencia de captura mensual de la anchoveta, con objetivo pronosticar los desembarques de la Anchoveta mensualmente.

En función de los resultados, se ha observa que el modelo propuesto al ser aplicado a los datos de Alta Frecuencia, arroja una varianza explicada de un 93,13 %, mientras que al ser aplicado a los datos de baja frecuencia, arroja una varianza explicada de un 98,76 %. Finalmente, sumando los datos analizados por el modelo propuesto (datos analizados de alta frecuencia + datos analizados de baja frecuencia), se obtiene una varianza explicada de un 96,46 %, superando los mejores resultados obtenidos en estudios anteriores que obtuvieron un 94 % de varianza explicada aproximadamente.

Palabras Clave: Red Neuronal Sigmoidal, Optimización de Enjambre de Partículas, Alta Frecuencia, Baja Frecuencia.

Abstract

The fishing sector has an important contribution to exports (nationally), being a very important activity for the economy of Chile. However, this importance is due to the availability of fishery resources, becoming one of the main factors causing uncertainty in this activity, in turn, can hinder and restrict sustainable development of the sector. In order to conserve these resources, government agencies have set catch regulations for these species around the country.

This project aims to provide support in setting monthly catch quotas for the anchovy in the North Zone of Chile. This research has developed a Sigmoid Neural Network using Swarm Particles Optimizations Algorithms. This model can predict the monthly landings of this resource analyzing the High Frequency data and Low Frequency Data of the anchovy monthly catch quota.

According to the results, it is observed that proposed model applied to High Frequency Data, achieves an explained variance of 93.13 %, while with the Low Frequency Data, achieves an explained variance of 98.76 %. Finally, adding the data analyzed by the proposed model (High Frequency Data Analyzed + Low Frequency Data Analyzed), achieves an explained variance of 96,46 %, beating the best results obtained in previous studies reporting 94 % of explained variance.

Keywords: Sigmoid Neural Network, Particle Swarm Optimization, High Frequency, Low Frequency.

Índice

1. Introducción	1
2. Objetivos del Proyecto	3
2.1. Objetivo General	3
2.2. Objetivos Específicos	3
2.3. Organización del Texto	3
3. Pronóstico de Redes Neuronales	4
3.1. Pronóstico	4
3.2. Áreas de Aplicación	5
3.3. Función Sigmoidal	6
3.4. Ejemplos	7
4. Optimización por Enjambre de Partículas	8
4.1. Introducción	8
4.2. Fundamentos del PSO	8
4.2.1. Fundamentos del Movimiento de Partículas	8
4.2.2. Exploración y Explotación	9
4.3. Algoritmo PSO	9
4.4. Parámetros PSO	10
4.5. Topologías de Vecindad Local y Global	11
4.6. PSO con actualizaciones Síncronas y Asíncronas	12
4.7. Variaciones del Algoritmo PSO	12
4.7.1. PSO con Búsqueda Local Adaptativa (LPSO)	12
4.7.2. PSO con Peso Entorno Adaptativo (LAPSO)	13
4.7.3. PSO con Factor de Inercia Exponencial (EPSO)	14
4.8. Aplicaciones	14
5. Pronosticador Neuronal	16
5.1. Métricas de Evaluación	16
5.2. Modelo Genérico	16
5.3. Selección de Topología y Parámetros del Modelo	18
5.4. Resultados Datos de Alta Frecuencia	18
5.4.1. Nodos de Entrada	18
5.4.2. Nodos Ocultos	19
5.4.3. Partículas	20
5.5. Resultados Datos de Baja Frecuencia	21
5.5.1. Nodos de Entrada	21
5.5.2. Nodos Ocultos	21
5.5.3. Partículas	22
6. Análisis de Resultados	24
6.1. Análisis de Resultados Datos de Alta Frecuencia	24
6.2. Análisis de Resultados Datos de Baja Frecuencia	25
6.3. Suma de Resultados	28
7. Conclusión	30

Lista de Figuras

- 1. Gráfico de la Función Sigmoidal 7
- 2. Topologías de la Población 11
- 3. Modelo Pronosticador Neuronal 17
- 4. Diagrama de Bloque del Modelo Propuesto 18
- 5. Gráficos de Resultados - Alta Frecuencia 25
- 6. Gráficos de Resultados - Baja Frecuencia 27
- 7. Gráficos de Resultados - Alta + Baja 29

Lista de Tablas

1.	Parámetros para obtener Nodos Entrada - Alta Frecuencia	19
2.	Valor Métricas al variar Nodos de Entradas - Alta Frecuencia	19
3.	Parámetros para obtener Nodos Ocultos - Alta Frecuencia	19
4.	Valor Métricas al variar los Nodos Ocultos - Alta Frecuencia	20
5.	Parámetros para obtener N° de partículas - Alta Frecuencia	20
6.	Valor Métricas al variar N° Partículas - Alta Frecuencia	20
7.	Parámetros Fijos para obtener Nodos Entrada - Baja Frecuencia	21
8.	Valor de las Métricas al variar los Nodos de Entradas - Baja Frecuencia	21
9.	Parámetros Fijos para obtener Nodos Ocultos - Baja Frecuencia	22
10.	Valor de las Métricas al variar los Nodos Ocultos	22
11.	Parámetros Fijos para obtener N° de partículas - Baja Frecuencia	22
12.	Valor de las Métricas al variar N° Partículas - Baja Frecuencia	23
13.	Mejor Configuración - Alta Frecuencia	24
14.	Valor de las Métricas Mejor Configuración - Alta Frecuencia	24
15.	Mejor Configuración - Baja Frecuencia	25
16.	Valor de las Métricas mejor Configuración - Baja Frecuencia	26
17.	Valor de las Métricas al Sumar Alta + Baja	28

1. Introducción

El sureste del Océano Pacífico, en especial, el mar del norte de Chile y sur del Perú, es un ecosistema altamente productivo en cuanto a especies pelágicas se refiere, dentro de las cuales se encuentra la especie *Engraulis Ringens* o Anchoveta. A partir de mediados de la década de los 50 comienza a desarrollarse una intensa actividad pesquera [1], basada principalmente en la captura de la anchoveta, la cual se extiende hasta nuestros días. Tal fue la intensidad de esta actividad que desde 1983, como consecuencia de la caída de la población de anchovetas, la pesca en el área norte de Chile se encuentra sujeta a regulaciones durante los periodos de reproducción más importantes.

En este escenario, el pronóstico de las capturas es un tópico básico, debido a que juegan un rol fundamental en el papel de los stocks, previo a la toma de decisiones. Para alcanzar esta meta, es necesario pronosticar eventos incontrollables como la posible abundancia o aumento de la biomasa. La variabilidad de su nivel de abundancia se ve afectada directamente por una serie de variables ambientales, entre las cuales se encuentran: La corriente del Niño, temperatura superficial del mar y concentración de alimento (plancton) [2].

Según [3], en Chile el sector pesquero es uno de los sectores con mayor exportación ubicándose detrás del Cobre y de las exportaciones frutícolas. Desde el año 2007 al año 2012, se ha registrado un incremento de un 3,6 % en los ingresos generados por las exportaciones pesqueras. Además, en el año 2012 el porcentaje de las exportaciones pesqueras corresponde a un 5,8 % respecto de todas las exportaciones en el país, y sin considerar la proporción correspondiente a las exportaciones mineras, el porcentaje de exportaciones pesqueras asciende a un 12,6 %.

Otro punto importante que se toma en cuenta en [3] en relación al sector pesquero es la empleabilidad que se genera, debido a que entre los años 2011 y 2012 se generó un total de 140.864 empleos entre pesca artesanal, centros de cultivos, plantas de procesos y flota industrial.

Los mayores volúmenes de desembarque de especies pelágicas se registran entre la XV y la X región del país, siendo una de sus especies más abundantes la anchoveta, recurso que se caracteriza por una explotación conjunta por parte de la flota industrial y artesanal. Esta especie presenta un rápido crecimiento, además posee un ciclo de vida corto y una elevada tasa de mortalidad natural, siendo su distribución espacial, biomasa y reclutamiento fuertemente influenciados por las condiciones ambientales del medio en el que habitan, y que redundan en una alta variabilidad en los desembarques. En el periodo comprendido entre el año 2011 y 2012, los desembarques de anchoveta alcanzaron un promedio anual de 1 millón 47 mil toneladas, mientras que en el año 2011, las capturas para esta especie fueron de 1 millón 191 mil toneladas. La importancia de la anchoveta radica en la elaboración de aceite de pescado, conservas, harina de pescado, salados y otros productos [4].

En el presente proyecto de título, se tomarán como series temporales, los datos relacionados con los volúmenes de captura mensual de anchovetas en el norte de Chile y mediante el uso de Redes Neuronales Sigmoidal con aprendizaje basado en la técnica de Optimización por Enjambre de Partículas (PSO) Continuo [5].

Las Redes Neuronales Artificiales son modelos que intentan reproducir el comportamiento del cerebro humano [6] y consisten en un conjunto de elementos de procesamiento, llamados neuronas, los cuales se conectan entre sí.

La organización y disposición de las neuronas dentro de una red neuronal se denomina topología, y viene dada por el número de capas, la cantidad de neuronas por capa, el grado de conectividad, y el tipo de conexión entre neuronas.

Una vez determinada la topología de la red neuronal es necesario entrenarla. En la etapa de entrenamiento la red es capaz de aprender relaciones complejas entre entradas y salidas mediante el ajuste de los pesos de las conexiones entre neuronas

Existen distintas técnicas para entrenar una red neuronal, algunas implican un aprendizaje su-

pervisado, mientras que otras implican uno no supervisado. En el presente proyecto de título, se realiza un aprendizaje supervisado de la red mediante la utilización del algoritmo Optimización por Enjambre de Partículas (PSO) Continuo [7, 8, 9].

Optimización por Enjambre de Partículas es un método estocástico de cómputo evolutivo inspirado en el comportamiento social de organismos tales como las bandadas de aves o los bancos de peces, erigiéndose en alternativa eficiente y atractiva a otros métodos estocásticos mucho más extendidos, tales como los Algoritmos Genéticos. [10]

La utilización de esta técnica para definir el estado interno de la red se debe a sus grandes cualidades de exploración y explotación, de manera que se pueda abarcar todo el espacio de búsqueda y así obtener las mejores regiones donde poder encontrar un óptimo local que se acerque lo más posible a una solución óptima de la red. Cabe destacar que existen estudios anteriores realizados en la Pontificia Universidad Católica de Valparaíso, en donde estudiantes de Ingeniería Civil Informática, Pablo Quezada [11] y Francesco Villarroel [12], utilizaron en sus proyectos de título Redes Neuronales con algoritmos de Optimización de Enjambre de Partículas para realizar pronósticos sobre la captura de anchovetas, alcanzando como resultados un $R^2 = 92\%$ y $R^2 = 94\%$ respectivamente; por otro lado, también existen otras investigaciones llevadas a cabo por estudiantes de la misma carrera, Juan Osorio [13] y Carlos Cosming [14], que desarrollaron en su proyecto de título, Soporte de Máquinas Vectoriales con algoritmos de Optimización de Enjambre de Partículas, obteniendo ambos un $R^2 = 94\%$.

2. Objetivos del Proyecto

2.1. Objetivo General

- Desarrollar un Modelo de Predicción de Capturas de Alta y Baja Frecuencia de Anchovetas en la Zona Norte de Chile, utilizando una Red Neuronal Sigmoidal combinada con algoritmos de Optimización de Enjambre de Partículas (PSO).

2.2. Objetivos Específicos

- Explicar la funcionalidad de las Redes Neuronales Artificiales y del Algoritmo PSO.
- Diseñar la estructura y estimar los parámetros de la Red Neuronal utilizando algoritmos PSO.
- Evaluar y Contrastar el Rendimiento del Pronosticador propuesto, con otros métodos utilizados en investigaciones realizadas por estudiantes de la Escuela de Ingeniería Informática de la Pontificia Universidad Católica de Valparaíso.

2.3. Organización del Texto

El presente documento está compuesto por 7 Capítulos, en los cuales se verán de manera detallada el Estado del Arte del proyecto de título, los resultados obtenidos con el modelo pronosticador propuesto, finalizando con las conclusiones obtenidas en base a los resultados obtenidos.

Específicamente, el capítulo 3 presenta principalmente información relacionada al Pronóstico y al comportamiento que presentan las Redes Neuronales ante el presente tema. A su vez, se muestran las diversas áreas en las que las Redes Neuronales son aplicadas para resolver problemas en el ámbito de la predicción. Además, el capítulo termina mostrando el comportamiento matemático de la función Sigmoidal, que será utilizada como función de transferencia de la Red Neuronal propuesta.

En el capítulo 4, se muestran los aspectos generales del PSO, donde se describen los fundamentos básicos del algoritmo y el funcionamiento de éste. A continuación, se detallan las topologías para el enjambre y además se explican algunas variaciones existentes del algoritmo. Finalizando el capítulo, se exhiben diversas áreas en donde el algoritmo PSO ha sido aplicado con éxito.

Durante el capítulo 5, se destacan las métricas de evaluación que serán consideradas para medir la calidad predictora del modelo propuesto en el proyecto de título. Además, se explica detalladamente el funcionamiento del modelo predictor propuesto, junto al modelo general y la forma en la que se lleva a cabo la selección de su topología y sus parámetros. En el término del capítulo, se muestran los resultados obtenidos con la mejor configuración alcanzada, tanto para los datos de Alta y Baja frecuencia.

En el capítulo 6, se lleva a cabo el análisis de los mejores resultados que se alcanzaron para los datos de Alta y Baja Frecuencia, además, de las métricas de evaluación que se obtuvieron al sumar los datos estimados de los datos dichos anteriormente.

3. Pronóstico de Redes Neuronales

3.1. Pronóstico

Predecir es hacer declaraciones sobre algo que va a suceder, a menudo se basan en información del pasado y del estado actual. Por ejemplo, cada persona hace diversos pronósticos en su diario vivir, ya sea sobre el clima, el tiempo se demora en llegar a un cierto destino, etcétera. En el aspecto técnico, los parámetros predecibles de un sistema generalmente pueden ser expresados y evaluados por medio de ecuaciones, por lo tanto, la predicción es simplemente la evaluación o solución de una posible ecuación. Sin embargo, existen problemas muy complejos y quizás imposibles de solucionar. Además, la solución computacional podría ser muy complicada y el resultado podría ser generado una vez que ya sucedió el evento a predecir.

Las redes neuronales artificiales corresponden a un modelo de aprendizaje y procesamiento automático inspirado en la forma en que funciona el sistema nervioso en los animales. Consiste en un sistema de interconexión de neuronas artificiales formando una red que colabora para producir un estímulo de salida. Estas neuronas corresponden a unidades de procesamiento que se comunican unas a otras mediante el envío de señales por medio de una serie de conexiones las cuales tienen un peso asociado.

Una de las mayores aplicaciones de las redes neuronales es el pronóstico. Las redes neuronales proveen una atractiva herramienta alternativa para los investigadores ya que estas poseen varias características que las hacen valiosas para las tareas de predicción. Primero, al contrario de los métodos tradicionales estadísticos, las redes neuronales son métodos adaptativos basado en datos, donde a veces se tiene muy poco conocimiento del problema bajo estudio. Ellas aprenden en base a ejemplos capturando relaciones funcionales sutiles entre los datos aun cuando estas relaciones sean desconocidas o difíciles de describir. Además son muy adecuadas para aquellos problemas cuyas soluciones requieren un conocimiento difícil de especificar, pero para los cuales existen suficientes datos u observaciones.

Segundo, las redes neuronales pueden generalizar, luego de la fase de entrenamiento pueden correctamente inferir el futuro de la población aun cuando la data contenga información ruidosa. Como el pronóstico se realiza vía la predicción del comportamiento futuro en base al comportamiento pasado, el pronóstico es una aplicación ideal para las redes neuronales al menos en principio.

Tercero, las redes neuronales son aproximadores funcionales universales. La experiencia de trabajos previos ha demostrado que una red puede aproximar cualquier función continua a una precisión deseada. Las redes neuronales tienen una estructura más flexible que los métodos estadísticos tradicionales. Cualquier modelo de pronóstico asume que existe una relación fundamental entre las variables de entrada (valores pasados de una serie de tiempo en este caso u otras variables relevantes) y las salidas (valores futuros). Frecuentemente los métodos estadísticos tradicionales tienen limitaciones a la hora de realizar las estimaciones de estas relaciones fundamentales debido a la complejidad de los problemas reales.

Por último las redes neuronales no son lineales, por mucho tiempo el pronóstico ha estado dentro del dominio de la estadística lineal, los cuales asumen que las series de tiempo bajo estudio son generadas por procesos lineales. Los procesos lineales tienen la ventaja de que pueden ser entendidos y analizados en gran detalle, lo que los hace fáciles de explicar e implementar. Sin embargo pueden ser inapropiados si el proceso fundamental bajo estudio es no lineal. No es razonable, a priori, asumir que la realización de cierta serie de tiempo es generada por un proceso lineal. De hecho en la vida real los sistemas a menudo son no lineales.

Es por esta razón que la función de transferencia que ocupará la red neuronal en este estudio será la Función Sigmoidal.

3.2. Áreas de Aplicación

En [15] se dice que una de las primeras aplicaciones exitosas en el pronóstico con redes neuronales, es reportada por Lapedes y Farber (1987 y 1988 respectivamente). Usando dos series de tiempo caótico determinista, generados por el mapa logístico y la ecuación Glass-Mackey, ellos diseñaron la red neuronal feedforward, las que pueden imitar con precisión y predecir el comportamiento no lineal del sistema. Sus resultados muestran que las redes neuronales pueden ser usadas para modelar y pronosticar series de tiempo con alta precisión. Según [16], las redes neuronales artificiales han sido ampliamente utilizadas en los últimos años en el marco de predicción de series temporales. Esto debido a características propias de las redes neuronales, como son:

- La capacidad de las redes neuronales para aproximar y capturar relaciones a partir de un conjunto de ejemplos, sin necesidad de conocer información adicional sobre la distribución de los datos.
- La capacidad de las redes neuronales para construir relaciones no lineales.
- La capacidad de las redes neuronales para construir relaciones a partir de información incompleta o información con ruido.
- Los modelos basados en redes neuronales son fáciles de construir y utilizar.

Sin embargo, en [16] se plantean problemas de predicción de series temporales a partir de un conjunto de muestras de la serie, conocer los valores en el futuro, es decir, su evolución o comportamiento a lo largo del tiempo. En este contexto se suelen distinguir dos casos:

- Predicción en un paso de tiempo.
- Predicción en múltiples pasos de tiempo.

También [15] dice que las series de manchas solares han servido como punto de referencia y han sido bien estudiadas en la literatura estadística. Ya que los datos que son no lineales, no estacionarios y no gaussianos, a menudo se utilizan como un criterio para evaluar y comparar los nuevos métodos de predicción. Algunos autores se centran en el uso de redes neuronales para mejorar la precisión en la predicción de las actividades de las manchas solares a través de los métodos tradicionales, mientras que otros utilizan los datos para ilustrar un método. Otro campo que se toma en cuenta en [15], es el pronóstico de las redes neuronales usado en la economía, ya que han sido utilizadas para prevenir quiebres de empresas, comportamiento del precio de acciones, para saber los posibles tipos de cambio. En [17], se realiza un pronóstico de variables macroeconómicas durante una crisis económica, utilizando una red neuronal auto regresiva de una sola capa oculta. Lo interesante de este modelo en el contexto de la investigación, es que forma una clase de aproximadores universales que podrían trabajar bien durante periodos excepcionales, tales como las grandes crisis económicas. La investigación en sí considera realizar previsiones en distintos periodos, utilizando dos modelos: el primero es para especificar y estimar un modelo único para generar las previsiones; el segundo es un modelo separado para cada horizonte de pronóstico.

La estrategia energética de un país es extremadamente importante para su desarrollo, sobre todo para aquellos países en vías de desarrollo, debido a que su economía va creciendo rápidamente, la demanda por el consumo energético también aumenta, en [18] afirman que realizar un pronóstico acertado acerca del consumo en el futuro permitiría a aquellos que hacen las políticas energéticas poder elaborar estrategias críticas para enfrentar esta problemática. En su experimento utilizaron una red neuronal de retro propagación obteniendo un MAPE de 3.8 % para los datos de prueba.

En una rama mucho menos explorada, varios investigadores han estado utilizando redes neuronales para poder predecir la magnitud de los terremotos. En [19], proponen un modelo basado en redes neuronales recurrentes con capas ocultas múltiples, para pronosticar las magnitudes de futuros terremotos utilizando las magnitudes de eventos telúricos pasados en el Norte del Mar Rojo, península del Sinaí, el golfo de Aqaba y golfo de Suez. En sus resultados afirman que su red neuronal obtuvo una ganancia del 32 % respecto a otros métodos estadísticos tradicionales. Otro campo de aplicación es la predicción del clima basándose principalmente en los datos históricos de series de tiempo. Los parámetros básicos para este tipo de predicción son la temperatura máxima, temperatura mínima, precipitación, humedad, entre otros. En [20] se intenta predecir las condiciones climáticas futuras en base a los parámetros nombrados anteriormente usando una Red Neuronal Artificial y la técnica de optimización por enjambre de partículas. Los resultados experimentales indican que el enfoque propuesto es útil para la predicción del tiempo.

En [21] se propone una nueva estrategia en la predicción de la velocidad del viento basándose en la lógica difusa y las redes neuronales artificiales. La nueva estrategia de la lógica difusa no sólo proporcionó significativamente menos base de reglas, sino que también aumentó la precisión estimada de la velocidad del viento, en comparación con un modelo tradicional. Mientras tanto, la aplicación del enfoque propuesto para la red neuronal artificial conduce a menos números de neuronas y menos tiempo de proceso de aprendizaje junto con la precisión de los resultados de predicción de la velocidad del viento. Los resultados experimentales demuestran que el método propuesto no sólo proporciona menos tiempo computacional sino también una mejor predicción de rendimiento en la velocidad del viento.

En [22], se propone un sistema que pronostique de manera precisa y fiable los contaminantes en el aire, con el objetivo de crear un sistema de alerta temprana para el control de la calidad del aire. En el estudio, se proponen 3 modelos distintos utilizando redes neuronales, los cuales alcanzando una buena configuración entregan resultados satisfactorios.

En [23], se propone una técnica no lineal basada en redes neuronales para la predicción del tiempo de viaje de los tsunamis en el Océano Índico. El modelo propuesto utiliza una regresión no lineal, donde se utiliza un perceptrón de múltiples capas para enfrentar la no linealidad de los tiempos de espera de llegada computarizada. Los resultados del modelo propuesto presentan un alto grado de correlación, demostrando su solidez en el desarrollo de un sistema de alerta de tsunamis en tiempo real para el Océano Índico.

3.3. Función Sigmoidal

Existen muchos procesos naturales y curvas de aprendizaje de sistemas complejos que muestran una progresión temporal con bajos niveles al inicio, y que al pasar el tiempo, se acercan a su clímax.

La función sigmoideal es usada generalmente cuando no se dispone de una descripción detallada. Ésta función, es producida por una función matemática que al graficarla tiene forma de S. A menudo, esta función se refiere al caso particular de la función logística, cuya gráfica está presente en la siguiente figura: 1

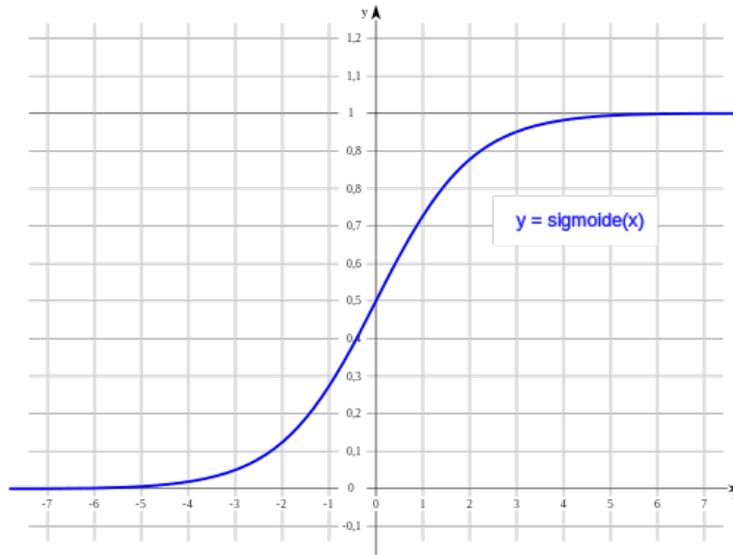


Figura 1: Gráfico de la Función Sigmoideal

Esta función, está definida por la siguiente ecuación:

$$\text{sig}(t) = \frac{1}{1 + e^{-t}} \quad (1)$$

Una amplia variedad de funciones sigmoideales han sido utilizadas como función de activación de las redes neuronales, incluyendo las funciones tangente hiperbólica y logísticos. Las curvas sigmoideales también son comunes en las estadísticas como funciones de distribución acumulada, como las integrales de la distribución logística, la distribución normal, y las funciones de densidad de probabilidad T de Student.

Propiedades En general, una función sigmoideal es de valor real y diferenciable, que tiene ya sea una primera derivada no negativo, que es en forma de campana. También hay un par de asíntotas horizontales como $t \rightarrow \pm\infty$. Las funciones logísticas son sigmoideales y se caracterizan como las soluciones de la ecuación diferencial.

3.4. Ejemplos

Además de la función logística, sin funciones sigmoideas el arco tangente, la tangente hiperbólica, y la función de error, pero también la función logística generalizada y funciones algebraicas como la siguiente:

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

La integral de cualquier función suave, positiva, con forma de "protuberancia" será sigmoideal, con lo que las funciones de distribución acumulada para muchas distribuciones de probabilidad comunes son sigmoideal. El más famoso de estos ejemplos es la función de error. Cabe destacar que la Función Sigmoideal, es una de las funciones de transferencias más comunes en el trabajo de las redes neuronales. Actualmente las aplicaciones de las redes neuronales se extienden ampliamente en los campos de los negocios, industrias y ciencias, llevando a cabo con éxito investigaciones en variados áreas del conocimiento humano, a continuación se presentaran algunas aplicaciones recientes de las redes neuronales aplicadas al pronóstico.

4. Optimización por Enjambre de Partículas

4.1. Introducción

La optimización por enjambre de partícula, la que en la literatura científica es llamada Particle Swarm Optimization (PSO), nace, al igual que otras técnicas estocásticas del cálculo evolutivo, en un intento por imitar y mimetizar el comportamiento de los procesos naturales. Siendo uno de los métodos más utilizados en la inteligencia computacional, busca imitar comportamientos sociales de un colectivo a partir de la interacción entre sus individuos y de éstos con su entorno.

Sus orígenes se remontan de los estudio realizados por Kennedy y Eberhart en el año 1995, quienes intentaron simular de manera gráfica el movimiento sincronizado e impredecible de bancos de peces o bandadas de aves, intrigados por la capacidad que tienen estos grupos para separarse, reagruparse o encontrar su alimento. Paralelamente con trabajos previos en el ámbito de la biología y la sociología, concluyen que el comportamiento, inteligencia y movimiento de estas agrupaciones está relacionado directamente con la capacidad para compartir información y aprovecharse de la experiencia acumulada por cada uno de los integrantes del grupo.

Kennedy y Eberhart introducen el término partícula, que representa a cualquier tipo de individuo que muestre algún tipo de comportamiento social como grupo en forma de una colección de agentes o partículas que interactúan entre sí. En base a los fundamentos teóricos del método, el movimiento de cada una de las partículas hacia la consecución de un objetivo en común depende de 2 factores, que son descritos a continuación: el primero es la memoria autobiográfica de la partícula o nostalgia y el segundo factor es la influencia social de todo el enjambre. Esto se puede extender, dependiendo del problema bajo análisis, a nivel computacional de D -dimensiones, siendo N el número de incógnitas. Básicamente, el proceso evolutivo se reduce a mover cada partícula dentro del espacio de soluciones con una velocidad que irá cambiando de acuerdo a su velocidad actual, a la memoria de la partícula y a la información global que comparte el resto del enjambre, usando una función de fitness para cuantificar la calidad de cada partícula en función de la posición en que ésta se encuentre.

Existen diversos tipos de esquemas para la implementación de PSO. Dependiendo de cómo se actualicen las posiciones de las partículas surgen versiones sincronas y asíncronas del método. Además, se puede distinguir entre PSO local y global dependiendo de cómo fluya la información a través del enjambre.

En el presente capítulo se comienza con la base de PSO relacionada con el fundamento del movimiento de partículas, para luego dar a conocer el algoritmo básico. Posteriormente se estudian los parámetros y las topologías más comunes. Finalmente se presentan 2 variantes del método y se termina con un listado de aplicaciones.

4.2. Fundamentos del PSO

4.2.1. Fundamentos del Movimiento de Partículas

Para aplicaciones relacionadas con el ámbito de la vida artificial se deben respetar cinco principios básicos de lo que se conoce como inteligencia de grupo, estos se conocen como: proximidad, calidad, diversidad de respuesta, estabilidad y adaptabilidad.

- **Proximidad:** la población debiera ser capaz de realizar cálculos sencillos de espacio y tiempo, lo cual en PSO se traduce a movimientos en D -dimensiones llevados a cabo durante un intervalo de tiempo que coinciden con movimientos de la población a una determinada velocidad.
- **Calidad:** Los factores de calidad en PSO se consiguen en base a la memoria de la partícula junto con la historia o conocimiento social que comparten todos los congéneres.

- **Diversidad de Respuesta:** Representadas por las tendencias marcadas por la memoria personal de cada partícula y por la historia de la mejor posición visitada por el conjunto.
- **Estabilidad y Adaptabilidad:** Resaltan aspectos contrapuestos, dado que por un lado, la población solo cambia su comportamiento como grupo cuando se actualiza la mejor posición históricamente visitada por alguno de los miembros que lo integran (principio de estabilidad) y, por otro lado, desde el punto de vista de ahorro computacional o mejora de precisión, la población debe ser capaz de cambiar su comportamiento y movimiento cuando hay alguna señal que así lo recomienda. En PSO esto se consigue cuando alguna partícula alcanza una solución global que mejora el resultado, en ese instante la población cambia de rumbo.

4.2.2. Exploración y Explotación

Uno de los principales problemas de los algoritmos evolutivos, incluido PSO, es que en ocasiones se quedan atrapados en óptimos locales, lo cual genera una convergencia prematura del algoritmo, por lo que no se pueden explorar otras regiones del espacio de soluciones.

En PSO, se busca que las partículas del enjambre puedan recorrer gran parte del espacio de búsqueda antes de comprometerse con alguna solución en particular, esto es lo que se conoce como diversificación. Por otro lado, cuando las partículas encuentren una zona comprometedoras deben ser capaces de mejorar la solución encontrada, lo que es conocido como convergencia de la población.

Dentro de los parámetros de PSO, existen dos términos a tener en consideración: exploración y explotación. El primero, se refiere a la búsqueda de la solución del problema de manera extensa dentro del espacio, privilegiando la diversificación del enjambre. El segundo término, consiste en la búsqueda de la solución en un rango dentro del espacio de búsqueda, privilegiando la convergencia del enjambre debido a que las partículas deberán tomar la máxima cantidad de valores posibles. Por lo tanto, los parámetros del algoritmo deben ser ajustados para que en un principio funcione en modo de exploración y luego de varias iteraciones varíe a modo de explotación.

4.3. Algoritmo PSO

Una partícula se define como un vector formado por la velocidad, posición y memoria de la partícula. PSO se inicializa con partículas aleatorias y optimiza sus soluciones a medida que aumentan las iteraciones. El enjambre es 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.

En cada iteración cada una de las partículas del enjambre se actualiza en base a dos valores importantes:

- Mejor posición encontrada por una partícula, conocida como P_{best} . Donde el P_{best} de una i -ésima partícula se representa por $P_i = (p_{i1}, p_{i2}, \dots, p_{id})$.
- Mejor posición que ha encontrado el enjambre, conocida como P_{best} .

Ambos valores se actualizan en cada iteración y se utilizan para ajustar la velocidad con la que se mueve una determinada partícula en cada una de las dimensiones. La influencia que tiene la mejor posición personal sobre la velocidad de una partícula se conoce como el factor de cognición y la influencia de la mejor partícula del enjambre se conoce como componente social.

En [24] se incorpora el factor de inercia w en la ecuación que determina la velocidad, de tal manera de que se balancee la búsqueda global y local.

$$v_{id}(t+1) = w * v_{id}(t) + c_1 * r_1 * (p_{id}(t) - x_{id}(t)) + c_2 * r_2 * (p_{gd}(t) - x_{id}(t)) \quad (3)$$

$$x_{id}(t + 1) = x_{id}(t) + v_{id}(t + 1) \quad (4)$$

Donde $r1$ y $r2$ corresponden a valores aleatorios independientes en el intervalo $[0,1]$, y $c1$ y $c2$ corresponden a las variables que controlan la influencia de las componentes cognitiva y social.

Para prevenir que las velocidades de las partículas incrementen infinitamente, se incorpora un parámetro $Vmax$ que delimita el rango de velocidad $[-Vmax, Vmax]$ que puede tomar una partícula.

$$\begin{aligned} &if(v_{ij} > V_{max})or(v_{ij} < -V_{max}) \\ &v_{ij} = sign(v_{ij} * V_{max}) \end{aligned} \quad (5)$$

A continuación se presentan los seis pasos relevantes que constituyen el algoritmo de PSO Tradicional:

- **Paso 1:** Se debe definir el tamaño n del enjambre y se deben inicializar las posiciones de las velocidades de las partículas en un espacio D-dimensional.
- **Paso 2:** Evaluar fitness de cada partícula en base a algún criterio de optimización (función de costo, mínimos cuadrados, error cuadrático medio).
- **Paso 3:** Comparar el fitness de cada partícula obtenido en el Paso 2 con el fitness de su mejor posición personal. Si el fitness actual es mejor que el histórico, se debe actualizar esta mejor posición por la nueva obtenida.
- **Paso 4:** Comparar el fitness actual de cada partícula con el mejor fitness encontrado por el enjambre, si el actual es mejor, entonces se actualiza la mejor posición global con el valor de la posición actual.
- **Paso 5:** Ajustar la velocidad y posición de la partículas en base a las ecuaciones de velocidad y posición descritas en este capítulo. Se debe verificar que las partículas no sobrepasen la velocidad máxima definida V_{max} .
- **Paso 6:** Si se cumple algún criterio de termino término, en caso contrario se debe volver al Paso 2. Los criterios de términos más conocidos pueden ser:
 - Número máximo de iteraciones.
 - Numero de iteraciones sin mejoras.
 - Error mínimo de la función objetivo.

4.4. Parámetros PSO

El ajuste de los parámetros permitirá condicionar el rendimiento posterior del algoritmo de optimización. La selección está directamente relacionada a la naturaleza del problema que se desea optimizar, siendo preciso lograr un balance óptimo entre la exploración y la convergencia. La definición de la función fitness es una decisión muy importante, dado que una función que no introduzca una métrica adecuada para pesar la bondad de cada partícula, hará emerger las carencias del PSO, que erróneamente pueden asociarse con el algoritmo.

Para acotar la velocidad de la partícula se define un valor máximo V_{max} , que 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 de manera lenta y podrán quedar atrapadas dentro de soluciones locales. 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 corresponde a un comportamiento oscilatorio y creciente de la posición de las partículas provocando la ineficiencia del PSO como algoritmo de optimización.

El peso de inercia w , regula la compensación entre la capacidad de exploración global y local del enjambre. A mayor w , es más fácil 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. Por lo general, a estos parámetros se les asigna 2 o 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 $c_1 + c_2 \leq 4$ [24].

El tamaño de la población debe ser seleccionado de manera rigurosa, dado que valores muy grandes pueden explorar de manera minuciosa el espacio de búsqueda pero el costo computacional se eleva de manera considerable debido al aumento del número de evaluaciones de la función fitness. Generalmente se utilizan poblaciones de entre 10 y 50 o 100 y 200 para problemas completos.

4.5. Topologías de Vecindad Local y Global

En PSO los individuos mejoran sus fitness dado que en cada iteración imitan los comportamientos y tendencias que encuentran en los mejores congéneres de la población. Establecer cómo se define la vecindad de un individuo tiene una trascendencia vital en el rendimiento del algoritmo. Dependiendo de la topología que adquiera la población, la transmisión de la información entre individuos puede acelerarse o ralentizarse, lo cual está íntimamente relacionado con la velocidad de convergencia y con la capacidad del algoritmo para escapar de soluciones locales.

La topología más extendida es la topología de vecindad global, donde todos los individuos están interrelacionados y tienen acceso inmediato a los hallazgos de sus semejantes. Sin embargo, esta estructura social es vulnerable a soluciones locales, dado que dependiendo de la distribución puntual de las partículas sobre el espacio de soluciones, una de ellas apuntando a una solución local puede llegar a dominar al resto.

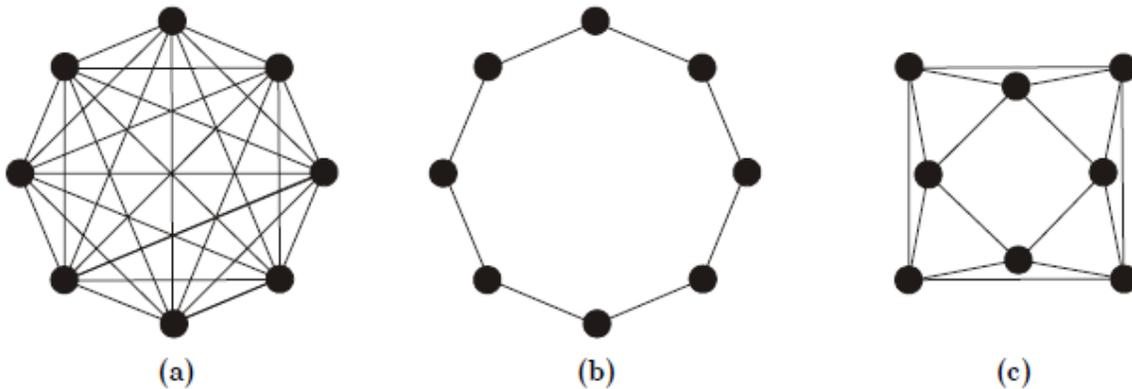


Figura 2: Topologías de la Población
 (a)Global, (b) Local con $N=2$, (c) Local con $N=4$

El modelo del óptimo local permite que cada individuo sea influenciado por un reducido número de miembros adyacentes. Estas partículas vecinas no tienen relación directa con el resto de las partículas que pertenecen a otra vecindad. Por lo general, como se visualiza en la Fig. 2, el óptimo local tienen 2 vecinos.

En [25], los autores declaran que el modelo del óptimo global converge rápidamente hacia las soluciones del problema, pero tiene la debilidad de quedar atrapado en mínimos locales, mientras que el óptimo local tienen la desventaja de converger lentamente hacia la solución del problema.

4.6. PSO con actualizaciones Síncronas y Asíncronas

La diferencia entre una actualización asíncrona con una síncrona, radica en el instante en el cual se realiza la actualización de la memoria de cada partícula y el conocimiento social del grupo.

En el modelo síncrono, todas las partículas se mueven en paralelo. El fitness de las partículas se evalúa en cada iteración, se actualiza su memoria p_{id} y el conocimiento social p_{gd} . Por otro lado, en el PSO asíncrono al momento de desplazarse, cada partícula aprovecha la información actualizada por sus inmediatos predecesores. Es decir, en cada iteración k , la i -ésima partícula se desplaza hacia un nuevo punto utilizando la información de los vectores p_{id} y p_{gd} , actualizados por las $i-1$ partículas previas. Posteriormente, la partícula evalúa la calidad del nuevo punto y actualiza, si procede, las variables p_{id} y p_{gd} . Esta información se transmite a las restantes partículas. Al actualizar la información partícula a partícula, el modelo asíncrono acelera la optimización, aunque la naturaleza del modelo síncrono lo hace susceptible de ser ejecutado en paralelo, sobre múltiples procesadores.

4.7. Variaciones del Algoritmo PSO

No es de extrañarse que desde su invención en 1995 el algoritmo PSO haya sido objetivo de una serie de cambios desde su primera propuesta, estos cambios buscan aliviar el estancamiento en la optimización. A continuación se mostrara algunas de las variantes de PSO.

4.7.1. PSO con Búsqueda Local Adaptativa (LPSO)

En el PSO estándar, las partículas son en gran parte influenciadas por su mejor posición alcanzada en su recorrido y por la mejor posición del enjambre. Una vez que la mejor partícula, no presenta cambios en un óptimo local, el resto de las partículas convergerán rápidamente a la posición presentada por la mejor partícula, esto se puede lograr a través de búsquedas sobre la vecindad de la mejor partícula del enjambre en cada generación.

En [26], se propone un nuevo operador de búsqueda local adaptativa, llamado también operador de mutación, el cual consiste en mutar en cada iteración la mejor partícula encontrada hasta ese momento, la cual variará respecto a las ecuaciones presentadas a continuación:

$$gBest_j(t+1) = gBest_j(t) + [\beta_j(t) - \alpha_j(t)] * rand() \quad (6)$$

$$\alpha_j(t) = \min(x_{ij}(t)); \beta_j(t) = \max(x_{ij}(t)); \quad (7)$$

$$i = 1, 2, \dots, ps; j = 1, 2, \dots, n$$

donde $gBest_j$ es el vector de la partícula que presenta la mejor solución del enjambre, $\alpha_j(t)$ y $\beta_j(t)$ son los valores mínimo y máximo de la dimensión j en el espacio de búsqueda, mientras que $rand()$ es un número aleatorio entre 0 y 1, ps corresponde al tamaño de la población y t indica en número de iteraciones.

El tamaño de este operador ($\beta_j(t) - \alpha_j(t)$) puede ser considerado como el diámetro del espacio de búsqueda de la población. Al principio de la evolución, el espacio de búsqueda inicial es grande,

por ende, el valor del operador es demasiado alto, siendo esto beneficioso para una búsqueda global, acelerando la convergencia del algoritmo. Sin embargo, con el aumento de las iteraciones, la población poco a poco converge a la mejor posición encontrada hasta ese momento. Por lo que el espacio de búsqueda de la población disminuye, así como también, el valor para el operador de búsqueda. Con esta situación, se beneficiará una búsqueda local en el espacio de soluciones, lo que conlleva a explotar de mejor manera la vecindad de la mejor partícula del enjambre.

4.7.2. PSO con Peso Entorno Adaptativo (LAPSO)

En [27], realizan una hipótesis basada en que la distribución de todo el enjambre puede proporcionar información adicional importante para la actualización de la nueva velocidad. Durante el proceso de búsqueda en PSO, la distribución de las partículas cambia continuamente. Los autores, definen un vector \vec{D} de distribución para describir el espacio de las partículas en un determinado momento:

$$D_j = \frac{\max_{i=1}^n(x_{ij}) - \min_{i=1}^n(x_{ij})}{|\max_{i=1}^n(x_{ij})| + |\min_{i=1}^n(x_{ij})|}; j = 1, 2, \dots, m; \quad (8)$$

donde m es la dimensión de las partículas y n es el tamaño de la población. La dimensión de \vec{D} es m .

El valor de cada componente del vector D de distribución es entre 0 y 1. Comparando a esta variante con la actualización de velocidad dada por IW-PSO, el reajuste de la velocidad a partir del vector de distribución no afecta solamente a valor número de la velocidad, sino que también a la dirección de las partículas. De esta manera, la actualización de la velocidad viene dada por:

$$v_{ij}^{k+1} = D[v_{ij}^k + \varphi_1 \cdot rand_1() \cdot (p_{ij} - x_{ij}^k) + \varphi_2 \cdot rand_2() \cdot (p_{gj} - x_{ij}^k)] \quad (9)$$

A partir de la ecuación anterior, se puede apreciar que si muchas partículas quedan atrapadas en un mínimo local, el vector de distribución será muy pequeño, casi cero, y por lo tanto la posibilidad de ajustar las velocidades se perderá. Esto llevaría a la convergencia prematura del algoritmo. Sin embargo, si la velocidad es demasiado grande puede provocar oscilaciones en la búsqueda y que sea poco probable que el PSO pueda encontrar valores mínimos.

Con el objetivo de evitar la situación mencionada, los autores modifican en vector de distribución en cada iteración del algoritmo de la siguiente manera:

$$D_j = D_{max}; Si D_j = D_{min} \quad (10)$$

$$D_j = D_{min}; Si D_j = D_{max} \quad (11)$$

$$D_j = D_j; Enotro caso. \quad (12)$$

En [27], por medio de los experimentos realizados, definen los valores de D_{max} y D_{min} en 0.95 y 0.4, respectivamente.

En el LAPSO, la velocidad en la siguiente iteración se genera a partir de la distribución de las partículas presentes dentro del vector de distribución, garantizando de esta manera, una mejor propagación de las partículas según su distribución en el espacio de búsqueda. El límite inferior y límite superior para el vector de distribución se proporciona para evitar una convergencia local y una oscilación inestable.

Los autores proponen una pequeña modificación en caso de que determinadas partículas se encuentren fuera del espacio de solución. A lo largo de las iteraciones, algunas partículas abandonarán

el área de búsqueda permitida. Para ello, recomiendan una redistribución de las partículas, bajo las siguientes ecuaciones:

$$\vec{X}(\vec{X} < Xmin) = Xmin + rand() \quad (13)$$

$$\vec{X}(\vec{X} > Xmax) = Xmax - rand() \quad (14)$$

4.7.3. PSO con Factor de Inercia Exponencial (EPSO)

En [28], se propone una modificación al PSO convencional, 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. Una vez que se obtiene la mejor posición del enjambre, las partículas son atraídas hasta esa posición en las iteraciones siguientes, hasta que se encuentre otra mejor solución. Por lo tanto, el estancamiento del algoritmo PSO se debe a la velocidad de la mejor solución encontrada. Una mejora a esta situación, está dada por la variante presentada en la sección 4.5.1, donde el factor de inercia disminuye linealmente hasta un valor mínimo.

En [28], los autores plantean el PSO exponencial (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, así evitar que las partículas se estancuen 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:

$$w = (w_{max} - w_{min}) * e^{\frac{(iter_{max}-k)}{iter_{max}} - 1} + w_{min} \quad (15)$$

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.

4.8. Aplicaciones

A medida que avanza el tiempo, las aplicaciones de PSO han recibido una gran atención en muchas aplicaciones y áreas de investigación durante los últimos años. Su carácter evolutivo junto con la gran cantidad de mejoras que presenta la literatura han dado pie para una gran cantidad de aplicaciones, como por ejemplo: en problemas electromagnéticos, en la síntesis de agrupaciones lineales de antenas y más aún en el aprendizaje de las Redes Neuronales. Este último es de gran importancia para motivos del presente caso de estudio.

A continuación se presentan las aplicaciones listadas en el párrafo anterior, destacando que el uso de PSO en algunos de estos casos resulta como híbrido con GA.

- **Electromagnetismo:** [29] presenta un algoritmo híbrido evolutivo llamado GSO (Genetical Swarm Optimization) producto de la combinación de GA con PSO. En general, debido a la rápida búsqueda global en el espacio sin quedar atrapado en mínimos locales, GSO puede ser adoptado para resolver adecuadamente problemas de optimización en la síntesis de arrays lineales y planos, en materiales EBG o en el diseño de estructuras de protección.
- **Síntesis de Arrays:** [10] compara PSO y Gas en la síntesis de arrays concluyendo que la aplicación de ambos metidos aplicados a la síntesis de alimentaciones en agrupaciones lineales

de antenas demuestra la mutua validez pero que la sencillez de implementación, ajuste, rapidez de convergencia priman a PSO frente a los Gas.

- **Entrenamiento de Redes Neuronales:** En [30] se comenta sobre un híbrido, formado por 2 variantes de PSO, para calcular los pesos de una RNA y definir la estructura, ambas entrenadas de manera simultánea demostrando una rápida convergencia.

Además en [7] se plantean otros campos de investigación en los cuales el algoritmo PSO ha sido utilizado. A continuación se presentan algunas de las áreas:

- **Biomedicina:** Se afirma que en el área de la Biomedicina y Médico y Farmacéutico, el uso del PSO en las investigaciones es muy alto, destacando: Análisis de vibración humana para el diagnóstico de la enfermedad de Parkinson [8], Reconstrucción de árboles filogenéticos [9], Clasificación del cáncer [31] y Predicción de la supervivencia a éste [32], y la Predicción de la estructura [33] y acoplamiento de las Proteínas [34].
- **Entretenimiento:** Se muestra que para la música y los juegos existe un pequeño grupo de aplicaciones en donde se aplica el PSO. Entre las aplicaciones que se pueden encontrar en el ámbito de los juegos se pueden observar: Análisis de nodos hoja en los árboles del juego [35], Aprendizaje de juegos de mesa [36, 37, 38]. Respecto de la música, se puede ver la Improvisación de música interactiva [39, 40].
- **Redes Neuronales:** Se destacan las investigaciones que combinan el PSO con las redes neuronales, tales como: Redes Neuronales para el Control de Procesos No Lineales [41], Controladores Neuronales [42], Diseño de Redes Neuronales Recurrentes [43, 44]
- **Predicción y Previsión:** Se puede observar que para la Predicción y Previsión existe una amplia variedad de investigaciones que utilizan PSO. Entre las que se pueden observar, están: Predicción y Clasificación de la Calidad del agua [45], Predicciones de Series de Tiempo [46, 47], Previsión en el flujo del tráfico urbano [48].

5. Pronosticador Neuronal

En este capítulo se presenta el modelo desarrollado con el cual se realizará el pronóstico de los datos.

En una primera sección se presentan las principales métricas utilizadas para medir la calidad del modelo. Luego en las siguientes secciones se presentan los modelos desarrollados. Hay que señalar que el modelo genérico es una Red Neuronal que utiliza el algoritmo PSO. Finalmente se presentan los resultados del estudio comparando los modelos presentados de manera que se determine cuál de ellos presenta una mejor capacidad predictiva frente al pronóstico del nivel de captura de Anchovetas.

5.1. 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.

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (d_i - y_i)^2}{N}} \quad (16)$$

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

$$R^2 = 1 - \frac{\sum_{i=1}^N (d_i - y_i)^2}{\sum_{i=1}^N (d_i - \bar{d})^2} \quad (17)$$

- **Porcentaje de Error Medio Absoluto Escalonado (MAPE(E)):** Proporciona una indicación de que tan grandes son los errores de pronóstico comparados con los valores reales de la serie. También, correspondiente a la operación interna de la sumatoria se encuentra el Promedio del dato observado con el dato pronosticado.

$$MAPE(E) = \frac{\sum_{t=1}^N \left| \frac{(d_t - y_t)}{(d_t + y_t) * 0,5} \right|}{N} * 100; d_t \neq 0 \quad (18)$$

Se define para las fórmulas presentadas a d_i como 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 computados.

Finalmente, cabe destacar que el equipo en el cual se realizarán todos los entrenamientos y pruebas corresponde a un Intel Core i5 1.8 GHz y 6 GB de RAM y los modelos están desarrollados en MATLAB.

5.2. Modelo Genérico

El modelo propuesto, toma el resultado de una investigación previa, la cual en breves palabras, toma una cierta cantidad de datos que corresponden a la captura mensual de la anchoveta en el norte del país, para luego separar dichos datos en un grupo correspondiente a la alta frecuencia de

captura mensual y otro correspondiente a la baja frecuencia de captura mensual. Cabe destacar que los datos a analizar en la presente investigación, serán entregados ya separados, es decir, los datos a considerar en este estudio serán tomados ya en grupos de alta y baja frecuencia.

Ante lo anterior, se procederá a realizar una red neuronal que será aplicada por separado tanto para los datos de alta frecuencia como para los datos de baja frecuencia de captura mensual de la anchoveta en el norte de Chile.

La red neuronal, recibirá los datos procesados por el algoritmo de PSO Continuo, donde se evaluará cuáles serán las mejores partículas presentes en la red. La evaluación dentro de la red neuronal será realizada por medio de la Función Sigmoidal.

Una vez obtenidos las mejores soluciones de cada red neuronal, se procederá a calcular el error de las soluciones. Esto será posible sumando los resultados obtenidos de la red neuronal de los datos de alta y baja frecuencia, para posteriormente restarlos a la solución deseada. Este funcionamiento de la red, queda demostrado en la siguiente figura:

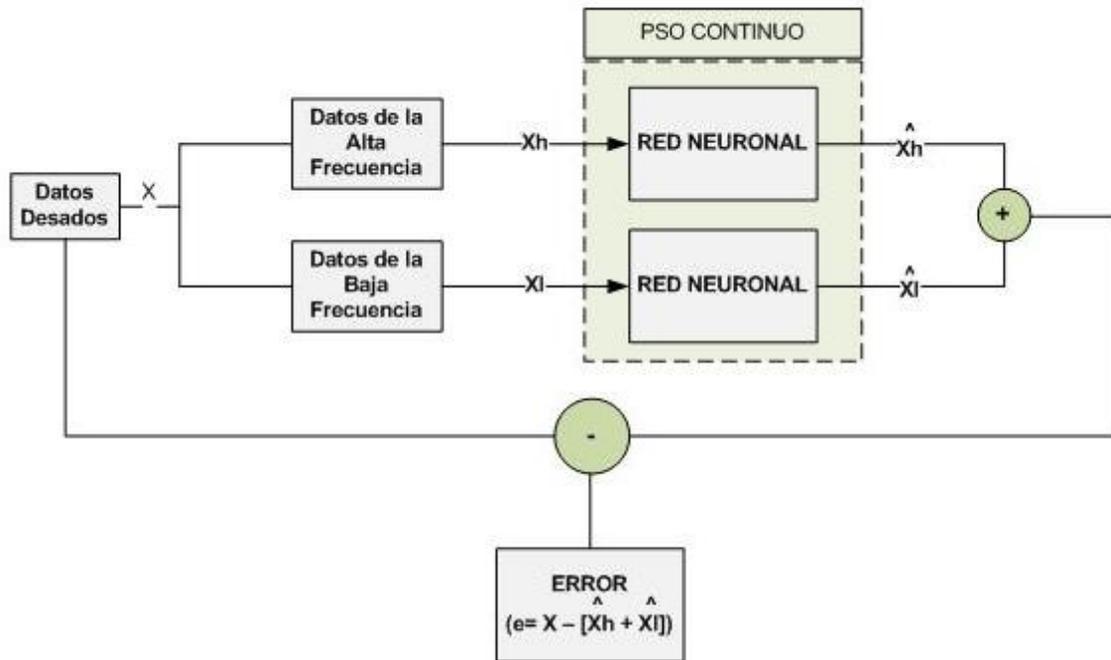


Figura 3: Modelo Pronosticador Neuronal

Centrando la atención en el modelo de la red neuronal propuesto, específicamente en la etapa de entrenamiento, se inicia con la creación de un enjambre inicial de partículas, de tamaño N , de manera aleatoria. En dicho enjambre, cada partícula contendrá los pesos de la red como posición (pesos lineales y no lineales), la velocidad determinada de manera aleatoria y una memoria personal (P_{best}) que contendrá el mismo valor de la posición. Una vez definida la posición, la velocidad y la memoria de la partícula, se procede a calcular el error inicial. Finalmente, se tendrá un enjambre inicial de N partículas, definiendo al menor error encontrado como mejor posición global (P_{gbest}).

Posteriormente, se realiza el proceso de actualización de la partícula, donde se actualizará la velocidad y la posición. Luego se efectúa la actualización del error de cada partícula, P_{best} y P_{gbest} , reemplazando las nuevas posiciones de las partículas obtenidas en la red y del cálculo del RMSE obtenido al procesar los datos de entrenamiento y compararla con los datos esperados.

Todo este procedimiento se puede ver específicamente en el Modelo de Bloque del Modelo Propuesto, el cual está representado en la siguiente figura:

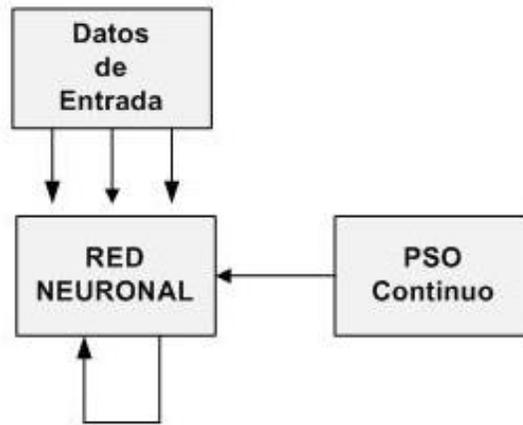


Figura 4: Diagrama de Bloque del Modelo Propuesto

5.3. Selección de Topología y Parámetros del Modelo

Tanto para los datos de Alta Frecuencia como los datos de Baja Frecuencia, el método de selección de las configuraciones de prueba han sido realizadas considerando la siguiente modalidad:

- **Nodos de Entrada:** Para obtener el número óptimo de entradas se realizan pruebas fijando el número de Nodos Ocultos en **8**, el número de iteraciones se fijará en 1000, y finalmente el número de partículas en **10**. Luego se prueba el número de entradas desde un rango de **1 a 12**, considerando los parámetros ya descritos, dejando la que presente el mejor R^2 .
- **Nodos Ocultas:** Para obtener el numero óptimo de nodos ocultas, se emplea el mismo mecanismo que se utilizó para la selección de nodos de entrada, con la diferencia que estas se fijan con la mejor obtenida en el proceso anterior, luego se varían los nodos ocultos en un rango de **1-8** dejando la que presente el mejor R^2 .
- **Número de Partículas:** Luego de obtener el número óptimo de neuronas de entrada y ocultas, y dejando fijas las iteraciones en 1000, se procederá a buscar la mejor configuración considerando un rango de 10 a 20 partículas, para finalmente elegir la que arroje un mejor resultado respecto del R^2 .

5.4. Resultados Datos de Alta Frecuencia

A continuación, se presentan los resultados obtenidos para los datos de Alta Frecuencia, en donde se considerará la elección del mejor valor de la métrica: R^2 .

5.4.1. Nodos de Entrada

Las tablas que se presentan en el presente capítulo muestran los resultados obtenidos para la selección de los nodos de entrada.

Los resultados que se presentarán corresponden a los valores que se obtienen de las métricas del modelo, y se basan en los valores de los parámetros de la tabla que aparece a continuación:

Nodos Ocultos	8
Partículas	10
Iteraciones	1000

Tabla 1: Parámetros para obtener Nodos Entrada - Alta Frecuencia

MSE	RMSE	MAPE (E)	R	R cuadrado	NE
0.0352	0.1877	0,9978	0.0481	0.0023	1
0.0157	0.1252	0,3918	0.7447	0.5545	2
0.0086	0.0926	0,2411	0.8707	0.7582	3
0.0080	0.0892	0,2382	0.8803	0.7749	4
0.0075	0.0866	0,2301	0.8878	0.7882	5
0.0049	0.0698	0,1942	0.9208	0.8479	6
0.0048	0.0696	0,1984	0.9211	0.8485	7
0.0044	0.0667	0,1887	0.9276	0.8605	8
0.0045	0.0672	0,1958	0.9268	0.8590	9
0.0038	0.0615	0,1741	0.9387	0.8811	10
0.0039	0.0628	0,1775	0.9354	0.8749	11
0.0035	0.0594	0,1679	0.9424	0.8881	12

Tabla 2: Valor Métricas al variar Nodos de Entradas - Alta Frecuencia en un rango de 1 a 12

Tal como se puede apreciar en la tabla anterior, el mejor resultado obtenido en base al R^2 es de un 88,81% correspondiente a los 12 Nodos de Entradas. Ante esto, en adelante se considerarán 12 Nodos de Entradas para las futuras pruebas.

5.4.2. Nodos Ocultos

Las tablas de resultados que se presentan a continuación, muestran los resultados de la obtención de los nodos ocultos para este modelo, conservando los Nodos de Entradas obtenidos anteriormente. A continuación se muestra una tabla con los parámetros que serán ingresados a la red neuronal:

Nodos Entradas	12
Partículas	10
Iteraciones	1000

Tabla 3: Parámetros para obtener Nodos Ocultos - Alta Frecuencia

MSE	RMSE	MAPE (E)	R	R cuadrado	NO
0.0303	0.1740	0.9023	0.4323	0.1869	1
0.0063	0.0793	0.2521	0.8950	0.8010	2
0.0085	0.0920	0.2800	0.8575	0.7354	3
0.0046	0.0682	0.2212	0.9317	0.8680	4
0.0041	0.0639	0.1879	0.9333	0.8710	5
0.0039	0.0625	0.1719	0.9374	0.8787	6
0.0036	0.0600	0.1657	0.9414	0.8862	7
0.0034	0.0587	0.1639	0.9439	0.8909	8

Tabla 4: Valor Métricas al variar los Nodos Ocultos - Alta Frecuencia en un rango de 1 a 8

Observando los resultados de la tabla anterior, el mejor resultado obtenido es de un 89,09 % de R^2 , correspondiente a los 8 Nodos de Ocultos. Por lo tanto, en adelante se considerarán 8 Nodos Ocultos para las futuras pruebas.

5.4.3. Partículas

Las tablas de resultados que se presentan a continuación, muestran los resultados de la obtención del número de partículas para este modelo, considerando los parámetros que lograron los mejores Resultados de las métricas anteriormente (Nodos de Entrada - Nodos Ocultos). A continuación se muestra una tabla con los parámetros que serán ingresados a la red neuronal:

Nodos Entradas	12
Nodos Ocultos	8
Iteraciones	1000

Tabla 5: Parámetros para obtener N° de partículas - Alta Frecuencia

MSE	RMSE	MAPE (E)	R	R cuadrado	Partículas
0,0010	0,0320	21,7501	0,8902	0,7925	10
0,0008	0,0283	105,8636	0,9232	0,8524	11
0,0005	0,0232	14,0754	0,9451	0,8932	12
0,0005	0,0218	13,6951	0,9470	0,8969	13
0,0006	0,0244	19,1846	0,9356	0,8754	14
0,0006	0,0235	50,7431	0,9389	0,8815	15
0,0008	0,0280	42,9590	0,9265	0,8583	16
0,0006	0,0236	29,5274	0,9388	0,8813	17
0,0005	0,0232	16,7618	0,9484	0,8994	18
0,0005	0,0233	16,9266	0,9432	0,8897	19
0,0005	0,0222	11,4899	0,9493	0,9012	20

Tabla 6: Valor Métricas al variar N° Partículas - Alta Frecuencia en un rango de 10 a 20

Como se puede observar en la tabla anterior, el mejor resultado obtenido fue con 19 partículas, donde el R^2 es igual a 90,12 %. Por lo tanto, para las futuras pruebas el número de partículas será fijado en 19.

5.5. Resultados Datos de Baja Frecuencia

A continuación, se presentan los resultados obtenidos para los datos de Baja Frecuencia, en donde se considerará la elección del mejor valor de la métrica: R^2 .

5.5.1. Nodos de Entrada

Las tablas que se presentan en el presente capítulo muestran los resultados obtenidos para la selección de los nodos de entrada.

Los resultados que se presentarán corresponden a los valores que se obtienen de las métricas del modelo, y se basan en los valores de los parámetros de la tabla que aparece a continuación:

Nodos Ocultos	8
Partículas	10
Iteraciones	1000

Tabla 7: Parámetros Fijos para obtener Nodos Entrada - Baja Frecuencia

MSE	RMSE	MAPE (E)	R	R cuadrado	NE
0.0361	0.1899	0,2759	0.1340	0.0180	1
0.0024	0.0491	0,0487	0.9538	0.9097	2
0.0004	0.0212	0,0214	0.9917	0.9835	3
0.0003	0.0178	0,0195	0.9941	0.9882	4
0.0011	0.0325	0,0265	0.9805	0.9615	5
0.0008	0.0279	0,0310	0.9859	0.9719	6
0.0006	0.0246	0,0284	0.9897	0.9794	7
0.0006	0.0237	0,0262	0.9896	0.9793	8
0.0005	0.0213	0,0231	0.9915	0.9830	9
0.0005	0.0220	0,0242	0.9911	0.9822	10
0.0005	0.0215	0,0237	0.9915	0.9830	11
0.0007	0.0262	0,0299	0.9874	0.9749	12

Tabla 8: Valor de las Métricas al variar los Nodos de Entradas - Baja Frecuencia en un rango de 1 a 12

Tal como se puede apreciar en la tabla anterior, el mejor resultado obtenido en base al R^2 es de un 98,82 % correspondiente a los 4 Nodos de Entradas. Ante esto, en adelante se considerarán 4 Nodos de Entradas para las futuras pruebas.

5.5.2. Nodos Ocultos

Las tablas de resultados que se presentan a continuación, muestran los resultados de la obtención de los nodos ocultos para este modelo, conservando los Nodos de Entradas obtenidos anteriormente. A continuación se muestra una tabla con los parámetros que serán ingresados a la red neuronal:

Nodos Entradas	4
Partículas	10
Iteraciones	1000

Tabla 9: Parámetros Fijos para obtener Nodos Ocultos - Baja Frecuencia

MSE	RMSE	MAPE (E)	R	R cuadrado	NO
0,0021	0,0463	5,6482	0,8878	0,7882	1
0,0001	0,0091	1,1570	0,9915	0,9830	2
0,0001	0,0097	1,2123	0,9896	0,9793	3
0,0001	0,0093	1,2341	0,9902	0,9804	4
0,0001	0,0116	1,6086	0,9846	0,9694	5
0,0001	0,0080	0,9870	0,9929	0,9859	6
0,0001	0,0078	0,9935	0,9930	0,9860	7
0,0004	0,0203	2,7057	0,9523	0,9069	8

Tabla 10: Valor de las Métricas al variar los Nodos Ocultos en un rango de 1 a 8

Observando los resultados de la tabla anterior, el mejor resultado obtenido es de un 98,60 % de R^2 , correspondiente a los 7 Nodos de Ocultos. Por lo tanto, en adelante se considerarán 6 Nodos Ocultos para las futuras pruebas.

5.5.3. Partículas

Las tablas de resultados que se presentan a continuación, muestran los resultados de la obtención del número de partículas para este modelo, considerando los parámetros que lograron los mejores Resultados de las métricas anteriormente (Nodos de Entrada - Nodos Ocultos). A continuación se muestra una tabla con los parámetros que serán ingresados a la red neuronal:

Nodos Entradas	4
Nodos Ocultos	7
Iteraciones	1000

Tabla 11: Parámetros Fijos para obtener N° de partículas - Baja Frecuencia

MSE	RMSE	MAPE (E)	R	R cuadrado	Partículas
0,0001	0,0112	1,6396	0,9858	0,9718	10
0,0001	0,0089	1,0984	0,9911	0,9822	11
0,0001	0,0081	1,0300	0,9924	0,9848	12
0,0001	0,0082	1,0199	0,9922	0,9844	13
0,0001	0,0090	1,1730	0,9906	0,9813	14
0,0001	0,0091	1,2149	0,9904	0,9808	15
0,0001	0,0079	1,0269	0,9927	0,9854	16
0,0001	0,0090	1,1017	0,9906	0,9812	17
0,0001	0,0078	1,0160	0,9930	0,9861	18
0,0001	0,0073	0,9143	0,9937	0,9875	19
0,0001	0,0079	0,9936	0,9928	0,9857	20

Tabla 12: Valor de las Métricas al variar N° Partículas - Baja Frecuencia en un rango de 10 a 20

Como se puede observar en la tabla anterior, el mejor resultado obtenido fue con 19 partículas, donde el R^2 es igual a 98,75 %. Por lo tanto, para las futuras pruebas el número de partículas será fijado en 19.

6. Análisis de Resultados

En esta sección, se analizarán los resultados obtenidos de la mejor configuración de la Red Neuronal, tanto para los datos de Alta y Baja Frecuencia.

6.1. Análisis de Resultados Datos de Alta Frecuencia

Las tablas de resultados que se presentan a continuación, muestran los resultados obtenidos con distintos valores de iteraciones, considerando los parámetros que lograron los mejores resultados de las métricas en la sección anterior (Nodos de Entrada - Nodos Ocultos - Partículas). A continuación se muestra una tabla con los parámetros de la mejor configuración de la red neuronal para los datos de Alta Frecuencia:

Nodos Entradas	12
Nodos Ocultos	6
Particulas	19

Tabla 13: Mejor Configuración - Alta Frecuencia

Iteraciones	MSE	RMSE	MAPE (E)	R	R^2
2000	0,0005	0,0219	40,3995	0,9485	0,8996
2500	0,0006	0,0240	14,6033	0,9350	0,8742
3000	0,0005	0,0224	10,6057	0,9447	0,8924
3500	0,0005	0,0222	31,1307	0,9447	0,8925
4000	0,0003	0,0179	9,5912	0,9650	0,9313
Mejor	0,0003	0,0179	9,5912	0,9650	0,9313
Peor	0,0006	0,0240	40,3995	0,9350	0,8742
Promedio	0,00048	0,02168	21,26608	0,94758	0,898
σ	0,000110	0,002264	13,763517	0,010945	0,020854

Tabla 14: Valor de las Métricas Mejor Configuración - Alta Frecuencia

Como se puede observar en la tabla anterior, el mejor resultado obtenido fue con 4000 iteraciones, donde el R^2 es igual a 93,13 %.

En la siguiente figura, específicamente en el gráfico (b) correspondiente al Gráfico de Correlación, se observa que existe una dispersión entre los datos pronosticados respecto de los datos deseados, lo cual se verifica en los valores de $R = 96,5\%$, que muestra que el grado de relación entre los datos es alta, y al $R^2 = 93,13\%$ que muestra que la variación explicada es alta, por lo tanto el pronóstico es bueno. Además, en el gráfico (a) se observa la curva de datos de Valor Estimado vs Valor Observado, el cual muestra que la curva de los datos pronosticados es similar a la curva de los datos observados, lo cual se refleja en el valor del coeficiente de determinación (R^2) y el valor del error cuadrático medio ($MSE = 0,0003$). Por último, en el gráfico (c) se observa el error relativo del modelo, el cual detalla que del 100 % de los datos estimados, un 45,71 % tiene un error relativo por sobre al $\pm 20\%$.

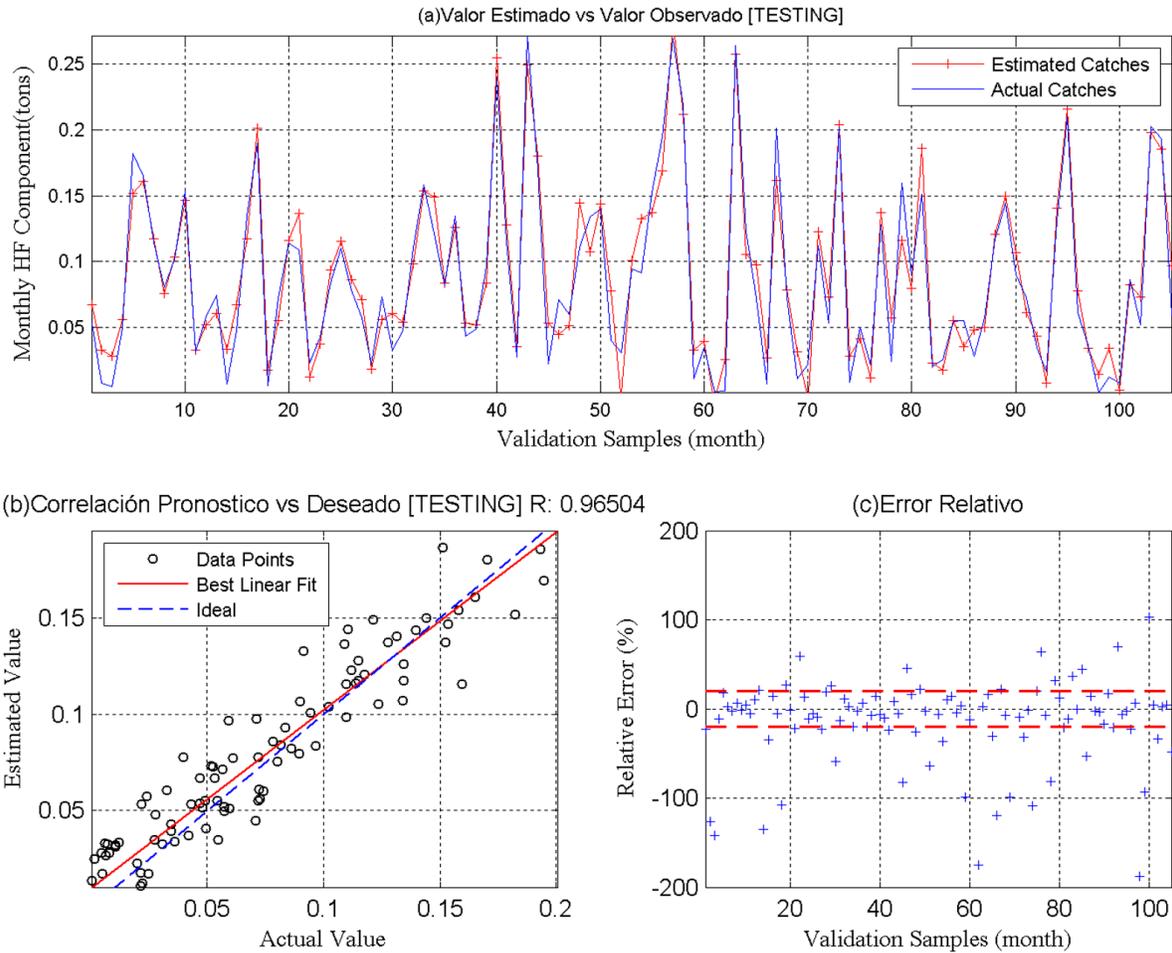


Figura 5: Gráficos de Resultados - Alta Frecuencia

(a) Gráfico Estimación vs Pronóstico (b) Gráfico Correlación (c) Error Relativo

6.2. Análisis de Resultados Datos de Baja Frecuencia

Las tablas de resultados que se presentan a continuación, muestran los resultados obtenidos con distintos valores de iteraciones, considerando los parámetros que lograron los mejores resultados de las métricas en la sección anterior (Nodos de Entrada - Nodos Ocultos - Partículas). A continuación se muestra una tabla con los parámetros de la mejor configuración de la red neuronal para los datos de Baja Frecuencia:

Nodos Entradas	4
Nodos Ocultos	7
Partículas	19

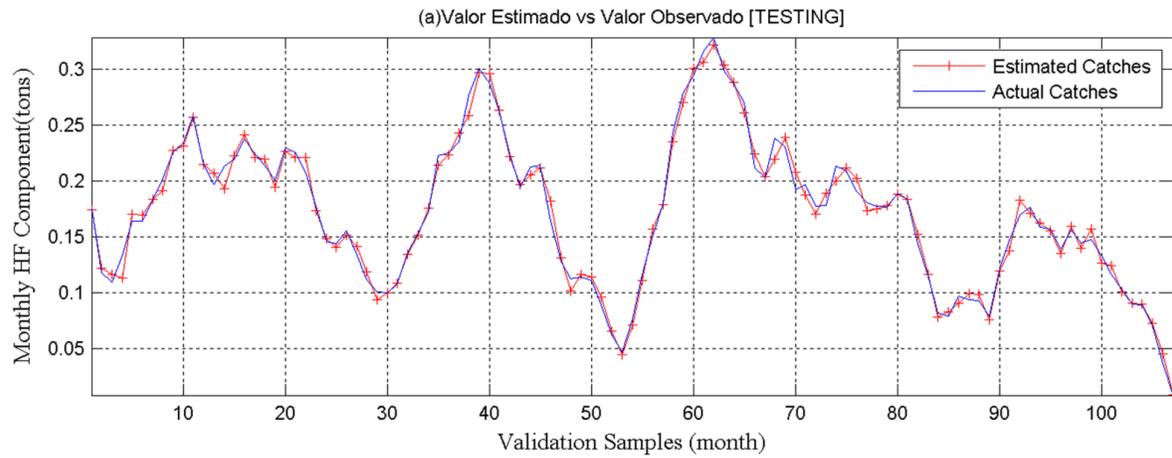
Tabla 15: Mejor Configuración - Baja Frecuencia

Iteraciones	MSE	RMSE	MAPE (E)	R	R^2
2000	0,0001	0,0090	1,2567	0,9920	0,9840
2500	0,0001	0,0075	0,9658	0,9934	0,9869
3000	0,0001	0,0076	0,9517	0,9933	0,9866
3500	0,0001	0,0073	0,9103	0,9938	0,9876
4000	0,0001	0,0073	0,9095	0,9938	0,9876
Mejor	0,0001	0,0073	0,9103	0,9938	0,9876
Peor	0,0001	0,0090	1,2567	0,992	0,984
Promedio	0,0001	0,00774	0,9988	0,99326	0,98654
σ	0,0000	0,000716	0,146310	0,000740	0,001486

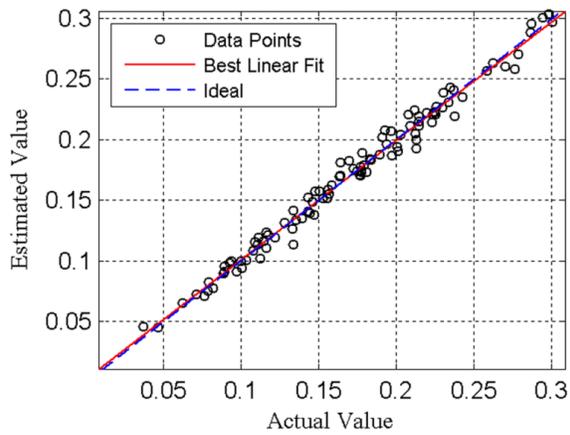
Tabla 16: Valor de las Métricas mejor Configuración - Baja Frecuencia

Como se puede observar en la tabla anterior, el mejor resultado obtenido es al probar con 3500 y 4000 iteraciones, donde el R^2 es igual a 98,76 % en ambos. Ante esto, se escogerá por medio del mejor Mape Escalonado, que en este caso corresponde a los resultados obtenidas con 3500 iteraciones, donde dicho Mape Escalonado es igual a 91,03 %.

En la siguiente figura, específicamente en el gráfico (b) correspondiente al Gráfico de Correlación, se observa que existe una dispersión entre los datos pronosticados respecto de los datos deseados, lo cual se verifica en los valores de $R = 99,38\%$, que muestra que el grado de relación entre los datos es alta, y al $R^2 = 98,76\%$ que muestra que la variación explicada es alta, por lo tanto el pronóstico es bueno. Además, en el gráfico (a) se observa la curva de datos de Valor Estimado vs Valor Observado, el cual muestra que la curva de los datos pronosticados es similar a la curva de los datos observados, lo cual se refleja en el valor del coeficiente de determinación (R^2) y el valor del error cuadrático medio ($MSE = 0,0001$). Por último, en el gráfico (c) se observa el error relativo del modelo, el cual detalla que del 100 % de los datos estimados, un 0,9345 % tiene un error relativo por sobre al $\pm 20\%$.



(b) Correlación Pronóstico vs Deseado [TESTING] R: 0.99377



(c) Error Relativo

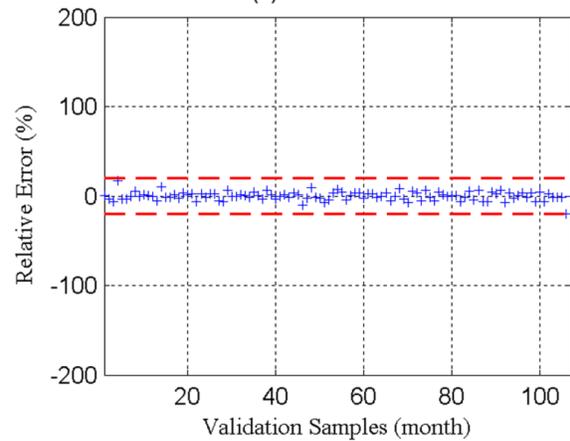


Figura 6: Gráficos de Resultados - Baja Frecuencia

(a) Gráfico Estimación vs Pronóstico (b) Gráfico Correlación (c) Error Relativo

6.3. Suma de Resultados

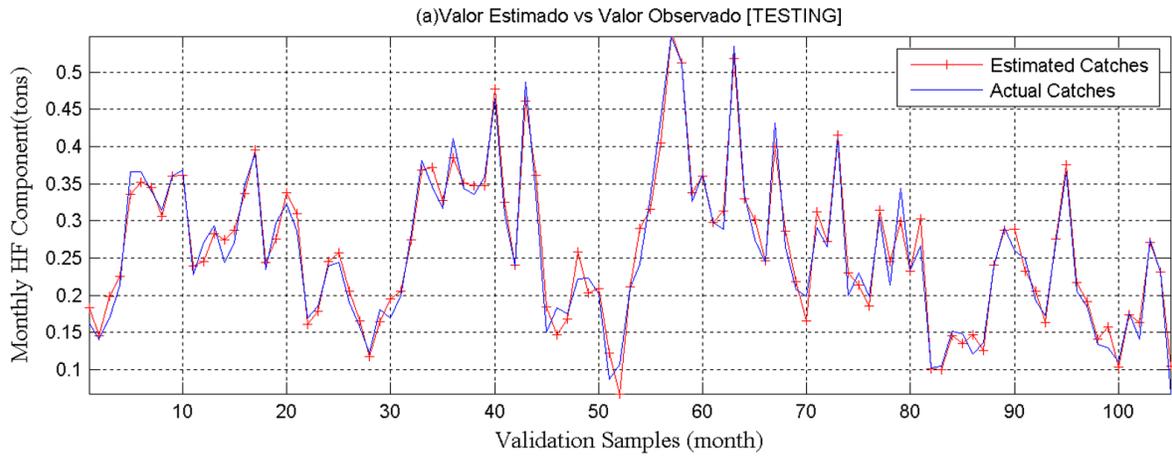
En la presente sección, se procederá a sumar los datos estimados de la Alta y Baja Frecuencia, con el objetivo de realizar un nuevo pronóstico de datos respecto a los datos observados reales.

En la siguiente tabla se observan los resultados obtenidos en las métricas al sumar los datos estimados:

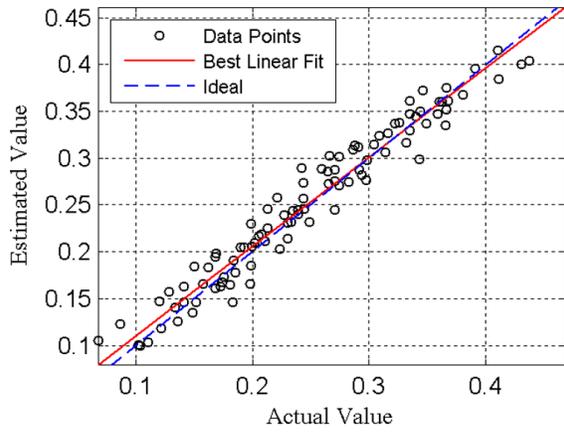
MSE	RMSE	MAPE (E)	R	R^2
0,0004	0,0193	4,7932	0,9821	0,9646

Tabla 17: Valor de las Métricas al Sumar Alta + Baja en un rango de 10 a 20

En la siguiente figura, específicamente en el gráfico *(b)* correspondiente al Gráfico de Correlación, se observa que existe una dispersión entre los datos pronosticados respecto de los datos deseados, lo cual se verifica en los valores de $R = 98,21\%$, que muestra que el grado de relación entre los datos es alta, y al $R^2 = 96,46\%$ que muestra que la variación explicada es alta, por lo tanto el pronóstico es bueno. Además, en el gráfico *(a)* se observa la curva de datos de Valor Estimado vs Valor Observado, el cual muestra que la curva de los datos pronosticados es similar a la curva de los datos observados, lo cual se refleja en el valor del coeficiente de determinación (R^2) y el valor del error cuadrático medio ($MSE = 0,0004$). Por último, en el gráfico *(c)* se observa el error relativo del modelo, el cual detalla que del 100 % de los datos estimados, un 5,7143 % tiene un error relativo por sobre al $\pm 20\%$.



(b) Correlación Pronóstico vs Deseado [TESTING] $R^2: 0.96459$



(c) Error Relativo

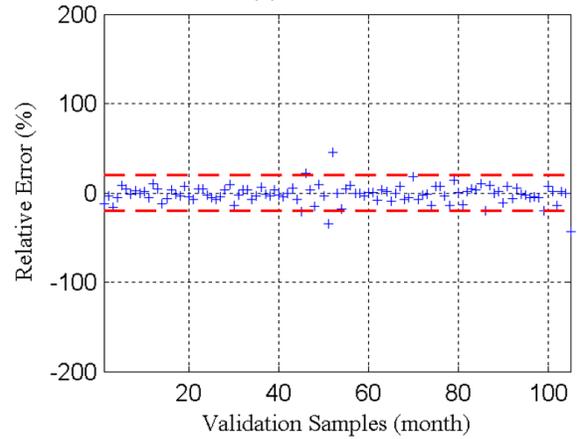


Figura 7: Gráficos de Resultados - Alta + Baja

(a) Gráfico Estimación vs Pronóstico (b) Gráfico Correlación (c) Error Relativo

7. Conclusión

En el presente proyecto de título, se desarrolló el estado del arte de los componentes fundamentales que se utilizarán en la confección del modelo predictivo de la alta y baja frecuencia de la captura mensual de anchovetas en el norte de Chile mediante la red neuronal sigmoideal con inteligencia de enjambre de partículas, cumpliendo así el primer objetivo específico del proyecto.

En base a la investigación realizada sobre las Redes Neuronales -el estudio de sus distintas topologías y condiciones internas- junto con las aplicaciones encontradas en la literatura, han dejado claro que las redes neuronales corresponden a un modelo computacional muy potente, capaz de resolver una gran cantidad de problemas, no obstante, dependiendo del problema que se desee resolver, se debe configurar una red que satisfaga dichos requerimientos.

En los últimos quince años, las técnicas de computación evolutiva se han consolidado como un conjunto de metaheurísticas efectivas y robustas en consecuencia de su alta aplicabilidad a la resolución de problemas en múltiples áreas. Ante esto y tomando en cuenta los numerosos y exitosos antecedentes que existen sobre la utilización de técnicas evolutivas para la resolución de problemas de predicción, el algoritmo PSO continuo surge como una muy buena alternativa para enfrentar estos problemas, ya que si es bien configurado puede llegar a superar los resultados de otras técnicas que generalmente son utilizadas para desarrollar este tipo de problemas, generando así un predictor de calidad.

Por medio de la investigación realizada, se logró diseñar un modelo predictivo que consiste en una Red Neuronal Sigmoideal combinada con algoritmos PSO. Además, para poder estimar correctamente los parámetros internos del predictor neuronal propuesto, se utilizó como fitness el Error Cuadrático Medio.

Al observar los resultados entregados por el modelo desarrollado, tanto para los datos de alta y baja frecuencia, y previo al análisis de éstos, se puede concluir que la Red Neuronal Sigmoideal en conjunto con el algoritmo PSO es una herramienta muy potente a la hora de realizar pronósticos, ya que la topología propuesta para uno y otro dato, generaron resultados con más de un 90 % de varianza explicada, siendo un 93,13 % de R^2 para el caso de alta frecuencia, en donde se utilizaron 12 nodos de entrada, 8 nodos ocultos, 20 partículas y 4000 iteraciones; para los datos de la baja frecuencia se alcanzó un 98,76 % de R^2 con 4 nodos de entrada, 8 nodos ocultos, 19 partículas y 3500 iteraciones, destacando que para ambos tipos de datos se utilizó un 80 % como entrenamiento. Con los resultados obtenidos, se procedió a sumar los datos estimados de la red, alta y baja frecuencia, donde también se lograron métricas con resultados bastante buenos, en donde el R^2 fue de un 96,46 % y un Mape Escalonado de 4,7932 %.

Otro punto importante que se debe destacar, es el tiempo de ejecución empleado por el predictor neuronal propuesto a la hora de encontrar la mejor configuración, considerando su aplicación para los datos de alta y baja frecuencia. Basándose en lo explicado anteriormente, se logró percibir que el predictor neuronal aplicado a los datos de alta frecuencia, presenta un tiempo de procesamiento de 20 minutos con 10 segundos. Por su parte, el predictor neuronal aplicado a los datos de baja frecuencia, presenta un tiempo de procesamiento de 10 minutos con 47 segundos, disminuyendo en cierta cantidad el tiempo de ejecución en comparación con los datos de alta frecuencia. Posteriormente, al realizar la suma de las salidas de los datos analizados por el modelo propuesto, el tiempo de ejecución correspondiente a esta etapa tarda 10 segundos. Finalmente, al sumar todos los tiempos de ejecución queda un total de 31 minutos con 7 segundos, lo cual en comparación al pronóstico realizado por el modelo propuesto correspondiente a 1 mes, es un tiempo considerablemente pequeño.

Además, se debe considerar la ganancia que alcanza el predictor neuronal mostrado en el presente proyecto de título por sobre los resultados obtenidos en investigaciones anteriores, donde en comparación con el proyecto de título de Carlos Cosming [14] se obtiene una ganancia de un 1,83 %,

cabe destacar que el mejor resultado que obtiene es aplicando un Vector de Soporte Autorregresivo en conjunto del algoritmo LAPSO [27], donde el resultado corresponde a un 94,63 % de varianza explicada utilizando una topología de 10 meses de desfase, un tamaño de muestra equivalente a un 70 % de los datos, 20 iteraciones y un tamaño del enjambre igual a 20, un factor cognitivo de 1,8 y factor social de 2,4. Del mismo modo, contrastando el resultado obtenido por Francesco Villarroel [12], correspondiente a un 94,94 % de varianza explicada, versus un 96,46 % obtenido por la Red Neuronal Sigmoidal combinada con PSO continuo presentada en el presente proyecto de título, se observa una ganancia de un 1,52 %. Cabe destacar, que en dicha investigación el mejor resultado obtenido resulta de la aplicación de una Red Neuronal Multiplicativa combinada el PSO con pesos de inercia, utilizando una topología de 3 nodos de entradas, 6 nodos ocultos, 20 partículas y 200 iteraciones.

De esta manera, el valor del coeficiente de determinación, el tiempo de ejecución empleado por el modelo propuesto y la ganancia que éste obtiene, cumplen un papel muy importante para el modelo propuesto, ya que muestran que la Red Neuronal Sigmoidal más el algoritmo PSO es un modelo muy potente para el pronóstico de datos.

Finalmente, se puede concluir que la Red Neuronal Sigmoidal en conjunto con el algoritmo PSO continuo, es una muy buena opción para apoyar la toma de decisiones y planificaciones de las futuras cuotas de capturas de especies pelágicas, ya que muestra una alta capacidad de aprendizaje sobre los registros, que en este caso corresponden a la anchoveta que habita en el norte de Chile, a pesar de la separación de los datos y de su alta variabilidad.

Es importante destacar que este trabajo de título, desarrollado durante el año 2013, fue financiado económicamente por el proyecto FONDECYT Regular 1131105 "HARMONIC COEFFICIENTS FORECASTING OF NON-STATIONARY TIME SERIES TO SUPPORT THE MANAGEMENT OF THE SMALL PELAGIC FISHING IN CHILE".

Referencias

- [1] J. Cañon. Distribución de la anchoveta (*engraulis ringens*) en el norte de Chile en relación a determinadas condiciones oceanográficas. *Serie Invest. Pesq., IFOP (Chile)*, 30:1–122, 1978.
- [2] A. González E. Yáñez and M.A. Barbieri. Estructura térmica superficial del mar, asociada a la distribución espacio-temporal de sardina y anchoveta en la zona norte de Chile entre 1987 y 1992. *Invest. Mar., Valparaíso*, 23:123–147, 1995.
- [3] Subsecretaría de Pesca y Acuicultura. *Cuenta Pública Año 2012*. Gobierno de Chile, 2012.
- [4] Subsecretaría de Pesca y Acuicultura. *Informe Sectorial de Pesca y Acuicultura* . 2013.
- [5] Kennedy, J. y Eberhart, R.,. Particle Swarm Optimization. In *Proceedings of the IEEE International Conference on Neural Networks.*, 1995.
- [6] V. Hilera González, J.; Martínez Hernández. *Redes neuronales artificiales: fundamentos, modelos y aplicaciones*. 1995.
- [7] Riccardo Poli. Analysis of the publications on the applications of particle swarm optimisation. *Journal of Artificial Evolution and Applications.*, pages 1–10, 2008.
- [8] R. Eberhart and X. Hu. Human tremor analysis using particle swarm optimization. In *Proceedings of the 1999 Congress on Evolutionary Computation, 1999. CEC 99.*, 1999.
- [9] W.-G. Zhou H.-Y. Lv and C.-G. Zhou. A discrete particle swarm optimization algorithm for phylogenetic tree reconstruction. In *Proceedings of 2004 International Conference on Machine Learning and Cybernetics.*, 4:2650–2654, 2004.
- [10] Basterrechea R. P. J. Jesús, M. A. C. J. V. Aplicación de PSO y GAs a la síntesis de agrupaciones lineales de antenas. 2005.
- [11] P. Quezada. Redes RRBF con aprendizaje Híbrido para la predicción de captura de Anchovetas. *Memoria de Título PUCV*, 2010.
- [12] F. Villarroel. Redes Neuronales Multiplicativas Para Pronóstico de Captura de Anchoveta en la Zona Norte de Chile. *Memoria de Título PUCV*, 2010.
- [13] J. Osorio. Vector de soporte autoregresivo para Pronosticar capturas de anchoveta En la zona centro - sur de Chile. *Memoria de Título PUCV*, 2010.
- [14] C. Cosming. Pronóstico de captura de Anchovetas de la zona norte de Chile usando vector de soporte Autoregresivo. *Memoria de Título PUCV*, 2010.
- [15] Michael Y. Hu G. Zhang, B. Eddy Patuwo. Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting, Elsevier*, 30(1):35–62, 1997.
- [16] L. Isasi, V. y Galván. *Redes Neuronales Artificiales. Un enfoque práctico*. Pearson Prentice Hall. 2004.
- [17] A.B. Kock and T. Terasvirta. Forecasting performance of three automated modelling techniques during the economic crisis 2007-2009. *Technical report, CREATES Research Papers, School of Economics and Management, University of Aarhus*, 2011.

- [18] S. Zaim. G. Ogcu, O. Demirel. Forecasting Electricity Consumption with Neural Networks and Support Vector Regression. *Procedia - Social and Behavioral Sciences*, 58:1576–1585, 2012.
- [19] Al-Humidan S. Alarifi A.S.N., Alarifi N.S.N. Earthquakes magnitude predication using artificial neural network in northern Red Sea area. *Journal of King Saud University - Science*, 24:301–313, 2012.
- [20] Mohapatra S. Beura-S. Pradhan G. Tripathy, A. Weather Forecasting using ANN and PSO. *International Journal Scientific Engineering Research*, 2:1–5, 2011.
- [21] Rastegar H. Kojabadi H. M. Monfared, M. A new strategy for wind speed forecasting using artificial intelligent methods. *Renewable Energy*. 34(3):845–848, 2009.
- [22] Oktay AB. Kurt A. Forecasting air pollutant indicator levels with geographic models 3 days in advance using neural networks. 37:7986–7992, 2010.
- [23] P. C. Pandey R. Barman, B. Prasad Kumar and S. K. Dube. "Tsunami travel time prediction using neural networks". *Geophysical Research Letters*, 33(6), 2006.
- [24] Eberhart R. y Shi Y. Kennedy, J. *Swarm Intelligence*. The Morgan Kaufmann Series in Artificial Intelligence, Morgan Kaufmann., 2001.
- [25] Guo H. y Liu. H. Abraham, A. âSwarm Intelligence: Foundations, Perspectives and Applicationsâ. *Studies in Computational Intelligence (SCI)*, Springer-Verlag.
- [26] J. Tang and X. Zhao. A hybrid particle swarm optimization with adaptive local search. 5(4):411–418, 2010.
- [27] Knowles J. Hongmei L. Yizeng L. Yisu, J. and D. B. Kell. The landscape adaptive particle swarm optimizer. 8:295–304, 2008.
- [28] El-Dessouki N. Mervat A. N. Ghali, N. I. and L. Bakrawi. Exponential particle swarm optimization approach for improving data clustering. *International Journal of Electrical, Computer, and Systems Engineering*, 3:208–212, 2009.
- [29] Grimaccia-F. Mussetta M. Pirinoli P. Zich R. E. Grimaldi, E. A. Genetical swarm optimization: a new hybrid evolutionary algorithm for electromagnetic applications. *In Applied Electromagnetics and Communications, 2005. ICECom 2005. 18th International Conference on (pp. 1-4)*. IEEE., 2005.
- [30] Jiao Lin-Shen G. Yu. Shen, Jiang. Hybridized Particle Swarm Algorithm for Adaptative Structure Training of Multilayer Feed-Forward Neural Network: QSAR Studies of Bioactivity of Organic Compounds. *Wiley InterScience*., 2004.
- [31] G. C. Anagnostopoulos R. Xu and D. C. Wunsch. Multiclass cancer classification using semisupervised ellipsoid artmap and particle swarm optimization with gene expression data. *In IEEE/ACM Transactions on Computational Biology and Bioinformatics*., pages 65–77, 2007.
- [32] X. Cai R. Xu and D. Wunsch. Gene expression data for dlbc cancer survival prediction with a combination of machine learning technologies. *In Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference*., pages 894–897, 2005.
- [33] S.-H. Doong. Protein homology modeling with heuristic search for sequence alignment. *In 40th Annual Hawaii International Conference on System Sciences*., pages 128–128, 2007.

- [34] H.-L. Huang S.-F. Hwang B.-F. Liu, H.-M. Chen and S.-Y. Ho. Flexible protein-ligand docking using particle swarm optimization. *In IEEE Congress on Evolutionary Computation.*, 1:251–258, 2005.
- [35] L. Messerschmidt and A. Engelbrecht. Learning to play games using a pso-based competitive learning approach. *IEEE Transactions on Evolutionary Computation.*, 8:280–288, 2004.
- [36] J. Conradie and A. Engelbrecht. Training bao game-playing agents using coevolutionary particle swarm optimization. *In 2006 IEEE Symposium on Computational Intelligence and Games.*, pages 67–74, 2006.
- [37] S. Ragab A. Abdelbar and S. Mitri. Co-evolutionary particle swarm optimization applied to the 7/spl times/7 seega game. *In 2004 IEEE International Joint Conference on Neural Networks.*, 2004.
- [38] S. Kinawy A. Abdelbar, O. Soliman and H. Sayed. An evolved seega player capable of strong novice-level play. *In 2005 IEEE International Joint Conference on Neural Networks.*, 1:332–336, 2005.
- [39] T. Blackwell and P. Bentley. Don't push me! collision-avoiding swarms. *In Proceedings of the 2002 Congress on Evolutionary Computation, 2002. CEC '02.*, pages 1691–1696, 2002.
- [40] T. Blackwell and P. Bentley. Improvised music with swarms. *In Proceedings of the 2002 Congress on Evolutionary Computation, 2002. CEC '02.*, pages 1462–1467, 2002.
- [41] Z. Chen Y. Song and Z. Yuan. New chaotic pso-based neural network predictive control for nonlinear process. *IEEE Transactions on Neural Networks.*, 18:595–601, 2007.
- [42] N. Wang Y. Du and J. Zhang. An optimum design method based on pso algorithm for neuron controllers. *In Fifth World Congress on Intelligent Control and Automation, 2004. WCICA 2004.*, 3:2617–2621, 2004.
- [43] C.-F. Juang. A hybrid of genetic algorithm and particle swarm optimization for recurrent network design. *IEEE Transactions on Systems, Man and Cybernetics, Part B.*, 34:997–1006, 2004.
- [44] C.-F. Juang and Y.-C. Liou. On the hybrid of genetic algorithm and particle swarm optimization for evolving recurrent neural network. *In 2004 IEEE International Joint Conference on Neural Networks.*, 3:2285–2289, 2004.
- [45] H. Gao C. Zhou, L. Gao and C. Peng. Pattern classification and prediction of water quality by neural network with particle swarm optimization. *In The Sixth World Congress on Intelligent Control and Automation, 2006. WCICA 2006.*, pages 2864–2868, 2006.
- [46] G. Venayagamoorthy X. Cai, N. Zhang and I. W. D.C. Time series prediction with recurrent neural networks using a hybrid pso-ea algorithm. *In 2004 IEEE International Joint Conference on Neural Networks.*, 2:1647–1652, 2004.
- [47] Z. Ye Y. Yang, R. Chen and Z. Liu. Fdtd time series extrapolation by the least squares support vector machine method with the particle swarm optimization technique. *In Microwave Conference Proceedings, 2005. APMC 2005. Asia-Pacific Conference Proceedings.*, pages 3–3, 2005.

- [48] Y. Chen J. Zhao, L. Jia and X. Wang. Urban traffic flow forecasting model of double rbf neural network based on pso. *In Sixth International Conference on Intelligent Systems Design and Applications, 2006. ISDA '06.*, pages 892–896, 2006.