

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

**REGRESIONES FUNCIONALES  
NEURO-WAVELET APLICADAS AL  
PRONÓSTICO DE CAPTURA DE  
ANCHOVETAS**

**ALEJANDRO ANDRÉS TRIGO GALDAMES**

TESIS DE GRADO  
MAGÍSTER EN INGENIERÍA INFORMÁTICA

SEPTIEMBRE, 2011

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

**REGRESIONES FUNCIONALES  
NEURO-WAVELET APLICADAS AL  
PRONÓSTICO DE CAPTURA DE  
ANCHOVETAS**

**ALEJANDRO ANDRÉS TRIGO GALDAMES**

**Profesor Guía: Nivaldo Rodríguez Arancibia**

**Programa: Magíster en Ingeniería Informática**

SEPTIEMBRE, 2011

## Contenido

---

Resumen . . . . .	VIII
Abstract . . . . .	VIII
<b>1. Introducción</b>	<b>1</b>
1.1. Descripción del Problema . . . . .	1
1.1.1. Solución Propuesta . . . . .	3
<b>2. Definición del Proyecto</b>	<b>4</b>
2.1. Objetivos . . . . .	4
2.1.1. Objetivos Generales . . . . .	4
2.1.2. Objetivos Específicos . . . . .	4
<b>3. Regresiones Lineales</b>	<b>5</b>
3.1. Introducción . . . . .	5
3.2. Regresión Lineal Simple . . . . .	5
3.3. Estimación de los coeficientes de regresión . . . . .	6
3.4. Regresión Lineal Múltiple . . . . .	7
3.5. Métodos de estimación de los coeficientes de regresión . . . . .	9
3.5.1. Descomposición en valores singulares . . . . .	9
3.5.2. Estimación Mediante Mínimos Cuadrados . . . . .	9
3.5.3. Factorización de Cholesky . . . . .	10
3.5.4. Aplicación de factorización QR . . . . .	11
3.5.5. Estimación mediante Algoritmos Genéticos . . . . .	12
3.6. Determinación del largo del Vector Regresión . . . . .	12
3.6.1. Coeficientes de determinación . . . . .	13

3.6.2.	Criterio $C_p$ Mallows . . . . .	13
3.6.3.	Selección paso a paso . . . . .	14
3.6.4.	Algoritmos Genéticos . . . . .	14
3.7.	Modelos de Autoregresión . . . . .	15
3.7.1.	Modelo de Medias Móviles . . . . .	16
3.7.2.	Modelo Autorregresivo Integrado de Media Móvil . . . . .	16
3.8.	Modelo Autorregresivo Neuro-Wavelet . . . . .	17
3.8.1.	Mínimos Cuadrados Separables . . . . .	17
<b>4.</b>	<b>Redes Neuronales</b>	<b>19</b>
4.1.	Introducción . . . . .	19
4.1.1.	Reseña Histórica . . . . .	20
4.1.2.	Red Neuronal Biológica . . . . .	22
4.1.3.	Red Neuronal Artificial (RNA) . . . . .	23
4.1.4.	Ventajas de una RNA . . . . .	28
4.1.5.	Características de una RNA . . . . .	29
4.1.6.	Clasificación . . . . .	30
4.1.7.	Perceptron . . . . .	35
4.1.8.	Red Neuro-Wavelet. (Red NW) . . . . .	35
4.2.	Algoritmos Genéticos . . . . .	37
4.2.1.	Codificación . . . . .	39
4.2.2.	Población . . . . .	41
4.2.3.	Función Objetivo . . . . .	43
4.2.4.	Función de Selección . . . . .	44
4.2.5.	Operadores de cruce . . . . .	45
4.2.6.	Operador de Mutación . . . . .	47
<b>5.</b>	<b>Discusión de Resultados</b>	<b>48</b>
5.1.	Indicadores de Desempeño . . . . .	48
5.2.	Técnicas de filtrado de datos . . . . .	51
5.2.1.	Promedio móvil tres . . . . .	51

5.2.2. Filtrado mediante convolución de parámetro $\alpha$ . . . . .	51
5.3. Regresión Funcional Neuro-Wavelet Evolutiva. NWE-AR. . . . .	51
5.3.1. Resultados Finales . . . . .	63
<b>Conclusiones</b>	<b>66</b>

## Lista de figuras

---

4.1. Red Neuronal Biológica . . . . .	22
4.2. Red Neuronal Artificial . . . . .	23
4.3. Modelo de una Red Neuronal . . . . .	25
4.4. Representación de una arquitectura feed-forward . . . . .	30
4.5. Representación de una arquitectura feedback . . . . .	31
4.6. Aprendizaje Supervisado . . . . .	32
4.7. Aprendizaje No Supervisado . . . . .	32
4.8. Aprendizaje Reforzado . . . . .	33
4.9. Clasificación de las RNA . . . . .	33
4.10. Clasificación de las RNA . . . . .	35
4.11. Representación de una Red Neuronal Wavelet . . . . .	40
4.12. Ejemplo de cromosoma real de una Red Neuro-Wavelet . . . . .	41
4.13. Operador de cruce en un punto . . . . .	46
4.14. Operador de cruce en dos puntos . . . . .	46
4.15. Operador de cruce Máscara de Cruce . . . . .	46
4.16. Operador de Mutación . . . . .	47
5.1. Gráfico fase de entrenamiento. PM3. . . . .	54
5.2. Gráfico RMSE. PM3. . . . .	55
5.3. Gráfico fase de validación. PM3. . . . .	55
5.4. Gráfico Error Absoluto en la fase de validación. PM3. . . . .	56
5.5. Gráfico dispersión en fase de validación. PM3. . . . .	57
5.6. Gráfico fase de entrenamiento. Convolución . . . . .	59
5.7. Gráfico RMSE. Convolución. . . . .	60

5.8. Gráfico fase de validación. Convolución . . . . .	60
5.9. Gráfico Error Absoluto fase de Validación. Convolución. . . . .	61
5.10. Gráfico dispersión en fase validación. Convolución. . . . .	62

## Lista de tablas

---

4.1. Codificación de topologías de Redes Neuronales . . . . .	42
5.1. Planificación . . . . .	48
5.2. Resumen de resultados con datos filtrados con PM3 . . . . .	53
5.3. Comparación de indicadores. Modelo AR-Neuro-Wavelet Evolutiva con PM3 y Modelo NF1 . . . . .	57
5.4. Resumen de resultados con datos filtrados con convolución . . . . .	58
5.5. Comparación de indicadores. Modelo AR-Neuro-Wavelet Evolutiva con Convolución y Modelo NF1 . . . . .	62
5.6. Comparación de indicadores. Modelo AR Neuro-Wavelet Evolutiva con PM3 y Modelo AR Neuro-Wavelet Evolutiva con Convolución . . .	63
5.7. Comparación de indicadores. Comparación con otros modelos predic- tivos . . . . .	64
5.8. Comparación de indicadores. Comparación con otros modelos predic- tivos considerando la Ganancia Obtenida . . . . .	65



## Resumen

Este documento trata sobre la aplicación de Regresiones Funcionales Neuro-Wavelet y Algoritmos Genéticos al pronóstico de captura de anchovetas en la zona norte de Chile. Para realizar el trabajo se presenta la investigación realizada sobre cada una de las materias expuestas anteriormente, además de la problemática que se aborda, la aplicación de Algoritmos Genéticos como algoritmo de aprendizaje, y la determinación de la topología de la regresión funcional codificada en cromosomas de Algoritmos Genéticos.

Los resultados obtenidos mediante el modelo de Regresiones Funcionales Neuro-Wavelet con Algoritmo Genético presenta una explicación de la varianza de hasta un 85 %, lo que mejora el resultado obtenido por el autor con un modelo Neuro-Wavelet Evolutivo, modelo que alcanza un 81 %.

**Palabras Claves:** Regresiones Funcionales; Red Neural Artificial; Red Neuro-Wavelet; Algoritmo Genético; Predicción de captura de anchovetas.

## Abstract

This document discusses the application of Functional Neuro-Wavelet Regression and Algorithm Genetic to the forecasting anchovy catch in northern of Chile. To make this document presents the research on each of the matters discussed above, in addition to the problems addressed, the application of genetic algorithms as learning algorithm, and the determination of the topology of the functional regression coded chromosomes Genetic Algorithms. The results obtained by the functional regression model Neuro-Wavelet, present an explanation of the variance of up to 85 %, which improves the result obtained by the author with a model Neuro-Wavelet evolutionary model that reaches 81 %.

**Keywords:** Funtional Regression; Artificial Neural Network; Neuro-Wavelet; Genetic Algorithms; Forecasting anchovy catch.

---

## Introducción

---

### 1.1 Descripción del Problema

Las costas de Chile y Perú se ven afectadas en gran parte de su extensión por corrientes marinas, en especial por la corriente de Humboldt. Esta corriente transporta hacia el norte, entre los 40°S y 45°S aproximadamente, masas de agua más frías, de baja salinidad y alto contenido de oxígeno disuelto.

Estas características que presenta la corriente de Humboldt, crean las mejores condiciones para la existencia de vida marina, convirtiéndolo en el ecosistema más rico del mundo.

Es en este ecosistema que se presentan dos eventos principales que hacen que sus condiciones varíen en lo que podría considerarse como condiciones óptimas: Por un lado se presenta el fenómeno de El Niño, el cual consiste en una invasión de aguas cálidas que avanzan de norte a sur. Además de alteraciones en el ecosistema, el fenómeno afecta a los recursos pelágicos de la zona, en especial, la sardina, anchoveta y plancton se profundizan y aumenta la mortandad de aves guaneras que no encuentran alimento en las costas. Por otro lado el fenómeno de la Niña, que consiste en la fase fría, en que la temperatura de las aguas desciende.

En la década del 50 comienza a desarrollarse en la zona norte de Chile y en gran parte de Perú, una intensa actividad pesquera enfocada principalmente en la extracción de la anchoveta, actividad que se extiende hasta nuestros días [1].

La actividad se concentra en el territorio chileno, especialmente en la I, II y XV región. En esta zona el promedio de captura por parte de Chile en el periodo comprendido

## CAPÍTULO 1. INTRODUCCIÓN

de 1995 a 2005, corresponde a los 5.5 millones de toneladas anuales. De este total el 42 % corresponde a la captura de anchovetas y sardinas [2].

El Gobierno de Chile viene realizando desde el año 1964 investigaciones en las zonas en las que se presenta el recurso pelágico. Estas investigaciones comprenden cruceros bio-oceanográficos estacionales en la zona norte, que han sido de vital importancia, ya que entregan información actualizada del comportamiento de la anchoveta y su reacción, vista del punto de vista de su abundancia, en respuesta a las fluctuaciones de las variables climáticas. Se ha observado que los efectos del fenómeno de el Niño o su contraparte, la Niña, no han sido siempre iguales, y depende en gran medida de la intensidad y duración del evento, de la época de ocurrencia, del periodo de máxima intensidad y en especial de la composición faunística presente en el ecosistema al momento de la ocurrencia del fenómeno [3] [4].

Por otro lado, entrega información acerca del comportamiento de la industria pesquera chilena, y el comportamiento de los niveles de captura de las distintas especies pelágicas presentes en la zona. Al comienzo de la recolección de datos, las capturas estaban enfocadas sobre la Anchoveta. A mediados de la década del 70 esta industria colapsa y los niveles de extracción se ven disminuidos casi totalmente, tras lo cual es reemplazada por la extracción de Sardina. Después de 1975 también aumentan las capturas de Jurel y Caballa, representando en el norte de Chile el 15 % y 4 % de los desembarques del período 1976-2002; en tanto la Anchoveta y Sardina representan el 34 % y 46 % respectivamente.

Además de la influencia de los cambios climáticos en el ecosistema marino de la zona, y que afecta en gran medida a la abundancia de la anchoveta, es importante destacar que los recursos de la zona geográfica marina y que habitan en la corriente de Humboldt, también son compartidos con los stock explotados por las pesqueras del Perú. También, se ha descrito una sucesión en la abundancia de anchoveta y sardina española, dependiendo de las condiciones oceanográficas. Actualmente, el área se encuentra bajo el régimen de alta abundancia de anchoveta y escasa abundancia de sardina española [5].

A la fecha, se desconocen los procesos y mecanismos del ambiente marino que expli-

## CAPÍTULO 1. INTRODUCCIÓN

can en forma directa la dinámica de las poblaciones de recursos pelágicos.

La recopilación de información de las fluctuaciones de abundancia de la anchoveta en la zona norte tiene como fin, ser la base desde la cual se dicten las decisiones gubernamentales respecto de la caza del recurso pelágico, es decir, tomar las medidas necesarias para asegurar la disponibilidad del recurso para el futuro.

Es aquí donde se vuelve importante la habilidad de predecir con cierto grado de exactitud los niveles de abundancia de los recursos pelágicos en la zona, con el fin de planificar la captura de los mismos y asegurar su disponibilidad.

Es tarea del gobierno, mediante leyes gubernamentales, controlar las actividades de las pesqueras que desempeñan sus labores en la zona, indicándoles los niveles de captura permitidos, así estos podrán planificar su esfuerzo de pesca de acuerdo a estas medidas.

Para tomar estas medidas, es necesario pronosticar el comportamiento de los niveles de población de los recursos pelágicos. Es ese el motivo de este proyecto: entregar una herramienta para la predicción de los niveles de captura de anchovetas en la zona norte.

### **1.1.1 Solución Propuesta**

La solución que se propone es la implementación de un modelo de Regresión Funcional Neuro-Wavelet Evolutivo que ayude a predecir los niveles de captura de Anchovetas en la zona Norte de Chile.

Para llevar a cabo esta tarea se investigará sobre cada uno de los componentes de la solución propuesta, esto es, Regresiones Funcionales, Redes Neuronales Artificiales Wavelet y Algoritmos Genéticos.

Finalmente se evaluará el rendimiento del modelo de predicción propuesto versus otros modelos existentes, en especial con un modelo Neuro-Wavelet evolutivo y un modelo Neuro-Wavelet Híbrido del autor, desarrollado y discutido en la memoria para obtener el título de Ingeniero Civil en Informática.

---

**Definición del Proyecto**

---

**2.1 Objetivos**

**2.1.1 Objetivos Generales**

Modelar e Implementar un modelo de predicción de captura de Anchovetas de la zona norte chilena utilizando Regresiones Funcionales Neuro-Wavelet Evolutivas.

**2.1.2 Objetivos Específicos**

- Comprender el funcionamiento de las Regresiones Funcionales.
- Comprender el funcionamiento de las arquitecturas de Redes Neuronales Wavelet aplicadas a Regresiones Funcionales.
- Comprender el funcionamiento de los Algoritmos Genéticos aplicados a Redes Neuronales y Regresiones Funcionales.
- Evaluar y contrastar el rendimiento del predictor propuesto con predictores existentes.

---

## Regresiones Lineales

---

### 3.1 Introducción

El objetivo de este punto es presentar los conceptos y metodología para extraer información de un conjunto de datos. En el caso de este proyecto, se busca investigar un método mediante el cual se extraigan las características principales que relacionan los datos con los cuales se trabajará.

El método a utilizar es Regresiones Funcionales Neuro-Wavelet Evolutivas. Para el análisis de este concepto se definirán primero lo correspondiente a Regresiones Lineales, tanto simple como múltiple.

En este punto se partirá desde la regresión funcional Simple Lineal, para a continuación avanzar hacia Regresiones Lineales Múltiples, además se explicarán algunos métodos para encontrar los valores del vector de regresión.

En puntos posteriores se detallará el funcionamiento de las Redes Neuronales Artificiales y Algoritmos Genéticos, de modo de desarrollar las bases para el diseño de un modelo de pronóstico utilizando Regresiones Funcionales Neuro-Wavelet Evolutivas, que es el objetivo de esta Tesis.

### 3.2 Regresión Lineal Simple

Se presenta la Regresión Lineal Simple como primer paso hacia la comprensión del modelo que se desea desarrollar.

Sea  $Y$  una variable aleatoria y  $x$  una variable controlable, lo que quiere decir, que los valores que toma  $x$  son fijados con anterioridad.

La fórmula que se presenta a continuación, representa el cálculo de  $Y$  para diferentes valores de  $x$ :

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i \text{ con } i = 1, 2, \dots, n \quad (3.1)$$

además,  $E(\epsilon_i) = 0$ ,  $var(\epsilon_i) = \sigma^2$  con  $i = 1, 2, \dots, n$ .

Este modelo representa la formulación lineal del problema de hallar la recta de regresión  $Y$  sobre  $x$ . Los parámetros  $\beta_0$ ,  $\beta_1$  reciben el nombre de coeficientes de regresión. El primero, representa la ordenada en el origen, y el segundo es la pendiente de la recta. De manera matricial, tenemos:

$$\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{bmatrix} \quad (3.2)$$

Una vez definida la matriz a utilizar, el problema se traslada a la identificación de los parámetros  $\beta_0$  y  $\beta_1$ . Para llevar a cabo esta operación existen varios métodos. El más utilizado es la estimación de los coeficientes mediante el método de Mínimos Cuadrados.

### 3.3 Estimación de los coeficientes de regresión

Utilizando los datos con los que se cuenta para el análisis, es posible realizar el cálculo de los coeficientes de regresión. Para esto, es necesario calcular los siguientes estadísticos:

$$\bar{x} = \left(\frac{1}{n}\right) \sum x_i \quad (3.3)$$

$$\bar{y} = \left(\frac{1}{n}\right) \sum y_i \quad (3.4)$$

$$s_x^2 = \left(\frac{1}{n}\right) \sum (x_i - \bar{x})^2 \quad (3.5)$$

$$s_y^2 = \left(\frac{1}{n}\right) \sum (y_i - \bar{y})^2 \quad (3.6)$$

$$s_{xy} = \left(\frac{1}{n}\right) \sum (x_i - \bar{x})(y_i - \bar{y}) \quad (3.7)$$

donde  $\bar{x}$ ,  $\bar{y}$ ,  $s_x^2$ ,  $s_y^2$ ,  $s_{xy}$ , son las medias, varianzas y covarianzas muestrales.

Con esta notación, las ecuaciones normales son:

$$X'X\beta = X'Y \iff \begin{bmatrix} n & n\bar{x} \\ n\bar{x} & \sum x_i^2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} n\bar{y} \\ \sum x_i y_i \end{bmatrix} \quad (3.8)$$

y como

$$(X'X)^{-1} = \frac{1}{n_x s^2} \begin{bmatrix} \frac{1}{n} \sum x_i^2 & -\bar{x} \\ -\bar{x} & 1 \end{bmatrix} \quad (3.9)$$

Resolviendo la ecuación matricial se obtiene que:

$$\widehat{\beta}_0 = \bar{y} - \widehat{\beta}_1 \bar{x} \quad (3.10)$$

$$\widehat{\beta}_1 = \frac{S_{xy}}{S_x} = \frac{s_{xy}}{s_x^2} \quad (3.11)$$

donde

$$S_{xy} = n s_{xy} S_x = n s_x^2 \quad (3.12)$$

### 3.4 Regresión Lineal Múltiple

En el punto anterior se definió la Regresión Lineal Simple, ahora es el turno de definir la Regresión Lineal Múltiple. Al igual que en la regresión lineal simple, se define un modelo lineal entre una variable aleatoria  $Y$  y un conjunto de  $k$  variables no aleatorias  $x_1, \dots, x_k$  explicativas o regresoras.

Si  $y_1, \dots, y_n$  son  $n$  observaciones independientes de  $Y$ , el modelo lineal de la regresión múltiple se define como:



$$y_i = \beta_0 + \beta_1 x_{i1} + \dots + \beta_k x_{ik} + \epsilon_i \text{ con } i = 1, 2, \dots, n \quad (3.13)$$

donde  $(x_{i1}, \dots, x_{ik})$  son los valores observados correspondientes a  $y_i$ .

En notación matricial, el modelo queda descrito por:

$$Y = X\beta + \epsilon \quad (3.14)$$

donde  $Y = (y_1, \dots, y_n)'$ ,  $\beta = (\beta_1, \dots, \beta_k)'$ ,  $\epsilon = (\epsilon_1, \dots, \epsilon_n)'$ . Y la matriz de diseño es:

$$X = \begin{bmatrix} 1 & x_{11} & \dots & x_{1k} \\ 1 & x_{21} & \dots & x_{2k} \\ \vdots & \vdots & & \vdots \\ 1 & x_{n1} & \dots & x_{nk} \end{bmatrix} \quad (3.15)$$

Se trata de calcular el ajuste de Mínimos Cuadrados a un hiperplano  $k$  dimensional, donde  $\beta_0$  es el punto de intersección del hiperplano con el eje  $y$  cuando  $x_1 = x_2 = \dots = x_k = 0$ .

Las ecuaciones normales son  $X'X\beta = X'Y$  donde

$$X'X = \begin{bmatrix} n & \sum x_{i1} & \sum x_{i2} & \dots & \sum x_{ik} \\ \sum x_{i1}^2 & \sum x_{i1}x_{i2} & \dots & \sum x_{i1}x_{ik} \\ & \sum x_{i2}^2 & \dots & \sum x_{i2}x_{ik} \\ & & \vdots & \dots \\ & & & \sum x_{ik}^2 \end{bmatrix} \quad (3.16)$$

y cuya solución son las estimaciones  $\widehat{\beta}_0, \widehat{\beta}_1, \dots, \widehat{\beta}_k$ .

Las predicciones de los valores de  $Y$  dadas las observaciones de las variables regresoras  $x_1, \dots, x_k$  son:

$$\widehat{Y} = X\widehat{\beta} \quad (3.17)$$

es decir,

$$\widehat{y}_i = \widehat{\beta}_0 + \widehat{\beta}_1 x_{i1} + \cdots + \widehat{\beta}_k x_{ik} \text{ con } i = 1, 2, \dots, n \quad (3.18)$$

### 3.5 Métodos de estimación de los coeficientes de regresión

Para el cálculo de los coeficientes de regresión se pueden utilizar diferentes métodos, por una parte al igual que en la regresión lineal simple, es posible hacer uso del método de Mínimos Cuadrados, por otra parte también se pueden utilizar Algoritmos Genéticos entre otras técnicas.

#### 3.5.1 Descomposición en valores singulares

Esta técnica tiene por objetivo disminuir el rango de la matriz  $X$ , para hacer esto se deben encontrar los valores propios [6] de la matriz para después encontrar sus valores singulares. Este proceso se define a continuación: Se tiene que  $X \in \mathcal{L}(\mathbb{R}^n, \mathbb{R}^m)$ ; entonces la matriz cuadrada de dimensión  $nA^T A$  es simétrica y semidefinida positiva; por tanto, sus valores propios  $\lambda_1, \dots, \lambda_n$  son no negativos y existen los valores  $\sigma_i = +\sqrt{\lambda_i}$  que son los valores singulares de la matriz  $X$ . Además, existen dos matrices ortogonales  $U$  y  $C$  de dimensión  $n$  y  $m$ , respectivamente, tales que:

$$U^T A V = U^{-1} A V = \Sigma \quad (3.19)$$

donde la matriz  $\Sigma \in \mathcal{L}(\mathbb{R}^n, \mathbb{R}^m)$  es de la forma

$$\Sigma = \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix} \quad (3.20)$$

con  $D = \text{diag}(\sigma_1, \dots, \sigma_r)$ , con  $\sigma_1 \geq \dots \geq \sigma_r \geq 0$  valores singulares de la matriz  $X$  no nulos y por tanto,  $r$  es el rango de  $X$ .

#### 3.5.2 Estimación Mediante Mínimos Cuadrados

Partiendo de la ecuación de regresión se busca estimar los  $\widehat{\beta}_0, \widehat{\beta}_1, \dots, \widehat{\beta}_k$  tal que

$$\widehat{y}_i = \widehat{\beta}_0 + \widehat{\beta}_1 x_{i1} + \cdots + \widehat{\beta}_k x_{ik} \text{ con } i = 1, 2, \dots, n \quad (3.21)$$

de modo que  $\widehat{y}_i$  estima  $E(y_i)$ .

Para estimar los coeficientes, se busca minimizar la suma de los cuadrados del error.

Con lo cual se tiene lo siguiente:

$$\sum_{i=1}^n \epsilon_i^2 = \sum_{i=1}^n (y_i - \widehat{y}_i)^2 = \sum_{i=1}^n (\widehat{y}_i - \widehat{\beta}_0 - \widehat{\beta}_1 x_{i1} - \cdots - \widehat{\beta}_k x_{ik})^2 \quad (3.22)$$

Los valores de  $\widehat{\beta} = (\widehat{\beta}_0, \widehat{\beta}_1, \dots, \widehat{\beta}_k)$  que minimiza la suma de cuadrados del error están dados por:

$$\widehat{\beta} = (X'X)^{-1}X'Y \quad (3.23)$$

### 3.5.3 Factorización de Cholesky

Implica la factorización de la matriz de variables de regresión de acuerdo a lo siguiente:

$$X = HH'; H \text{ matriz triangular inferior} \quad (3.24)$$

La existencia de la factorización de Cholesky puede ser demostrada a través de la factorización  $LU$ , esto es,  $X = LU$ . La matriz triangular superior  $U$  puede ser escrita como  $U = DU_1$ , donde  $D = \text{diag}(u_{11}, u_{22}, \dots, u_{nn})$  y  $U_1$  es una matriz triangular superior con unos en su diagonal. Luego,  $X = LDU_1$  y como  $X = X'$  se tiene que  $LDU_1 = U_1' S L'$  donde  $D = (U_1')^{-1} L D U_1 (L')^{-1}$ .

La matriz  $(U_1')^{-1} L$  es una matriz triangular inferior unitaria y la matriz  $U_1 (L')^{-1}$  es una matriz triangular superior unitaria, por lo tanto:

$$(U_1')^{-1} L = U_1 (L')^{-1} = 1 \quad (3.25)$$

En consecuencia,  $U_1 = L'$ , y  $X$  puede ser escrita como  $X = LDL'$  donde  $L$  es una matriz triangular inferior unitaria. Cuando  $X$  es definida positiva los elementos de la diagonal de  $D$  son positivas y se puede escribir  $D = D^{\frac{1}{2}} D^{\frac{1}{2}}$ , donde  $D^{\frac{1}{2}} =$

$diag(\sqrt{u_{11}}, \sqrt{u_{22}}, \dots, \sqrt{u_{mm}})$ . Así que,

$$x = LDL' = LD^{\frac{1}{2}}D^{\frac{1}{2}}L' = HH' \text{ con } H = D^{\frac{1}{2}} \quad (3.26)$$

### 3.5.4 Aplicación de factorización QR

Otra manera de usar la técnica de Mínimos Cuadrados para encontrar el vector  $\beta$  es la aplicación de la factorización QR. El objetivo de esta técnica es minimizar la siguiente ecuación:

$$\|X\beta - Y\|_2^2 \quad (3.27)$$

Ahora se define  $Q$  como una matriz ortogonal cuadrada de orden  $n$  y  $x$  un vector  $n$  dimensional, con lo cual se tiene que:

$$\|Q'x\|_2^2 = (Q'x)'(Q'x) = x'QQ'x = x'x = \|x\|_2^2 \quad (3.28)$$

luego,

$$\|X\beta - Y\|_2^2 = \|Q'X\beta - Q'Y\|_2^2 \quad (3.29)$$

Por otro lado,  $X$  puede ser factorizada como  $X = QR$  donde  $Q$  es una matriz ortogonal  $nxn$  y  $R$  es de orden  $nx(p + 1)$ , donde  $p$  es el número de variables de predicción,  $R$  es de la forma

$$R = \begin{bmatrix} R_1 \\ 0 \end{bmatrix} \quad (3.30)$$

donde  $R_1$  es una matriz triangular superior de orden  $(p + 1)x(p + 1)$ . Haciendo

$$Q'Y = \begin{bmatrix} c \\ d \end{bmatrix} \quad (3.31)$$

Luego la ecuación

$$\| Q'X\beta - Q'Y \|_2^2 = \left\| \begin{bmatrix} E_1 \\ 0 \end{bmatrix} - \begin{bmatrix} c \\ d \end{bmatrix} \right\|_2^2 = \left\| \begin{bmatrix} E_1\beta - c \\ -d \end{bmatrix} \right\|_2^2 = ((E_1\beta - c)' - d') \begin{bmatrix} R_1\beta - c \\ d \end{bmatrix} = \| R_1\beta - c \|_2^2 + \| d \|_2^2 \quad (3.32)$$

En consecuencia  $\| X\beta - Y \|_2^2$ , será mínimo con respecto a  $\beta$  cuando  $\beta$  es tal que:

$$R_1\beta = c \quad (3.33)$$

y la suma de cuadrados residual estará dada por

$$SCE = \| d \|_2^2 \quad (3.34)$$

Finalmente la matriz  $(X'X)^{-1}$  puede ser calculada como  $(R'R)^{-1} = R^{-1}(R')^{-1}$

### 3.5.5 Estimación mediante Algoritmos Genéticos

Comprende la codificación de los coeficientes mediante cromosomas. Se parte con una población de individuos (cromosomas), los cuales se prueban en la regresión y reciben una calificación de acuerdo al grado de minimización de alguna función objetivo. En el caso de las regresiones lineales múltiples, la función a minimizar podrá ser la función de mínimos cuadrados.

De acuerdo a la teoría sobre Algoritmos Genéticos, los individuos que mejor se adapten al problema, tendrán más posibilidades de pasar a una siguiente generación o de ser elegido como padre de un nuevo individuo de la siguiente generación.

El funcionamiento de los Algoritmos Genéticos se explica en profundidad en puntos posteriores.

### 3.6 Determinación del largo del Vector Regresión

Este punto tiene relación con la cantidad de variables a considerar para incorporar al modelo de regresión. La decisión sobre el número de variables a considerar no es un asunto trivial. Si se considera un número demasiado pequeño de variables es

posible que la potencia del modelo se vea reducida y que las estimaciones obtenidas sean sesgadas, tanto de los coeficientes de regresión, como de las predicciones. Este sesgo se produce debido a que el error calculado con los datos observados pueden contener efectos no aleatorios de las variables desechadas.

Un número muy grande de variables explicativas complica la utilidad práctica del modelo y, aunque mejora el ajuste aparente, aumenta la varianza de los estimadores de los parámetros [7]. Algunas técnicas para comparar modelos de regresión se presentan a continuación.

### 3.6.1 Coeficientes de determinación

El coeficiente de determinación  $R^2$  [23], mide el porcentaje de variabilidad presente en los datos que puede ser explicado mediante el modelo. En el capítulo de discusión de resultados se entregará más información acerca de la forma de cálculo del indicador.

El valor de  $R^2$  varía entre 0 y 1. Un valor cercano a 1 significa que el modelo explica la mayor parte de la variabilidad existente en los datos, un valor cercano a 0 significa lo contrario.

Para el caso de determinar el largo del Vector Regresión, se calculan los coeficientes de determinación de todos los modelos posibles con la combinación de cualquier número de variables explicativas. El objetivo es reconocer el modelo con mayor coeficiente. El cálculo se complica si el número de variables es considerable [7].

### 3.6.2 Criterio $C_p$ Mallows

Con este criterio se debe fijar en primera instancia un número  $p$  de parámetros, incluido el término independiente, aunque con posterioridad se podrá variar. Se trata de hallar el mejor modelo con  $p$  variables explicativas, incluida la constante, utilizando el estadístico de Mallows [7].

$$C_p = \frac{SCR_p}{\hat{\sigma}^2} - (n - 2p) \quad (3.35)$$

donde  $SCR_p$  es la suma de cuadrados residual del modelo particular y  $\widehat{\sigma}^2$  un estimador de la varianza del modelo que acostumbra a ser el error cuadrático medio del modelo completo (ECM).

Para el modelo completo  $p = k + 1$ , el estadístico de Mallows es:

$$C_{k+1} = \frac{SCR}{ECM} - (n - 2(k + 1)) = n - (k + 1) - (n - 2(k + 1)) = k + 1 \quad (3.36)$$

También para todo modelo no completo se puede demostrar que aproximadamente  $E(C_p) = p$ , si el modelo es adecuado. En consecuencia parece recomendable elegir los conjuntos para los que  $C_p$  sea aproximadamente  $p$ .

### 3.6.3 Selección paso a paso

El proceso se puede realizar hacia delante o hacia atrás, seleccionando las variables una a una e incorporándolas desde el modelo inicial o eliminándolas desde el modelo completo en función de su contribución al modelo.

La contribución al modelo puede ser medida comparando algún indicador en particular, como por ejemplo el coeficiente de determinación  $R^2$ , al evaluar el desempeño del modelo una vez que se le agrega o quita alguna variable.

Es el método más utilizado por su facilidad de computación [7]. Presenta el inconveniente de que puede conducir a modelos distintos y no necesariamente óptimos.

### 3.6.4 Algoritmos Genéticos

Al igual que en el caso de la codificación de los coeficientes de regresión, en este caso lo que se codifica es la presencia o no de las variables dentro de la función de regresión.

La codificación se realiza utilizando cromosomas binarios, en los que un 0 en la posición  $i$  del cromosoma representa que la variable  $i$  del modelo no será considerada en el Vector Regresión, y en el caso de que en la posición  $j$  exista un 1 significará que la variable  $j$  será considerada en el vector.

Para encontrar la mejor combinación del Vector de Regresión se utilizan operadores

genéticos, los cuales se discuten en profundidad en el capítulo de Redes Neuronales.

### 3.7 Modelos de Autoregresión

Se define un modelo como autorregresivo si la variable endógena de un período  $t$  es explicada por las observaciones de ella misma correspondientes a períodos anteriores añadiéndose un término de error. Este es el caso del modelo a ocupar en este trabajo, donde el nivel de captura de un mes  $t$  será pronosticado a través de los valores de los meses anteriores a  $t$ .

Los modelos autorregresivos se abrevian con la palabra *AR* tras la que se indica el orden del modelo: *AR(1)*, *AR(2)*, etc. El orden del modelo expresa el número de observaciones retrasadas de la serie temporal analizada que intervienen en la ecuación. Así, por ejemplo, un modelo *AR(1)* tendrá la siguiente expresión:

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \varepsilon_t \quad (3.37)$$

El término de error de los modelos de este tipo se denomina generalmente ruido blanco cuando cumple las siguientes condiciones:

- Media nula
- Varianza constante
- Covarianza nula entre errores correspondientes a observaciones diferentes.

La expresión general de un modelo de autorregresión para un *AR(p)* es la siguiente:

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + \varepsilon_t \quad (3.38)$$

de forma abreviada

$$\phi_p(L)Y_t = \phi_0 + \varepsilon_t \quad (3.39)$$

donde  $\phi_p(L)$  se conoce como operador polinomial de retardos, y esta definido por

$$\phi_p(L)Y_t = 1 - \phi_1 L - \phi_2 L^2 - \dots - \phi_p L^p \quad (3.40)$$



y donde a su vez, el término  $L$  es lo que se conoce como operador de retardo, tal que aplicado al valor de una variable en  $t$  dé como resultado el valor de esa misma variable en  $t - 1$ , esto es

$$LY_t = Y_{t-1} \quad (3.41)$$

y aplicado sucesivamente  $p$  veces retarda el valor en  $p$  periodos

$$L^p Y_t = Y_{t-p} \quad (3.42)$$

### 3.7.1 Modelo de Medias Móviles

Un modelo de medias móviles es aquel que explica el valor de una determinada variable en un periodo  $t$  en función de un término independiente y una sucesión de errores correspondientes a periodos precedentes, ponderados convenientemente. Estos modelos se denotan normalmente con las siglas  $MA$ , seguidos, como en el caso de los modelos autorregresivos, del orden entre paréntesis. Así, un modelo con  $q$  términos de error  $MA(q)$  responderá a la siguiente expresión:

$$Y_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_q \varepsilon_{t-q} \quad (3.43)$$

que también puede abreviarse utilizando el polinomio de retardos:

$$Y_t = \theta_q(L)\varepsilon_t + \mu \quad (3.44)$$

donde  $\theta_q$  representa el factor de ponderación del término de error correspondiente al periodo  $t - q$ .

### 3.7.2 Modelo Autorregresivo Integrado de Media Móvil

Estos modelos se denotan generalmente con las siglas  $ARIMA$ , y responden a una combinación de los modelos  $MA(q)$  y  $AR(p)$ . Cuando se especifica un modelo  $ARIMA$  se debe acompañar de las del orden de los modelos que lo componen y además de la

cantidad de diferenciaciones que se han necesitado para su composición, en general,  $ARIMA(p, d, q)$ , es la combinación de los modelos  $MA(q)$  y  $AR(p)$  con  $d$  series de diferenciaciones en la parte regular.

### 3.8 Modelo Autorregresivo Neuro-Wavelet

A continuación se presenta la ecuación que se propone para esta Tesis.

$$\widehat{y}(t + 1) = \sum_{i=0}^n [y(t - i) * \sum_{k=1}^m (w_k * \sum_{l=1}^p 2^{\frac{-a}{2}} h(2^{-a} x_l - b))] \quad (3.45)$$

donde  $\widehat{y}(t + 1)$  es la salida estimada por el modelo RNW-AR,  $n$  es el número de variables regresoras utilizadas,  $m$  es el número de neuronas ocultas de la red neuronal  $i$ .  $p$  representa el número de neuronas en la capa de entrada de la red  $i$ , y finalmente,  $a$  y  $b$ , son los parámetros de amplitud y dilatación de la función Wavelet de la neurona oculta  $k$  de la red neuronal  $i$ .

La arquitectura y parámetros de cada una de las redes neuronales presentes en la Regresión Funcional será determinada mediante la utilización de Algoritmos Genéticos.

La codificación de todas estas características de la Regresión Funcional será realizada mediante cromosomas reales, a los cuales se le aplicarán operadores genéticos.

En el siguiente capítulo se profundizará en lo referente a redes neuronales wavelet, como también a algoritmos genéticos y su aplicación a redes neuronales wavelet y modelos autorregresivos.

#### 3.8.1 Mínimos Cuadrados Separables

Esta técnica se utiliza para resolver el problema de mínimos cuadrados en funciones que poseen tanto una parte no lineal como una parte lineal, como es el caso del modelo de autorregresión que se propone.

Se considera que se tiene una variable  $y_i \in \mathbb{R}^p (i = 0, \dots, n)$ , el problema está en encontrar el vector de parámetros  $\theta_n \in \mathbb{R}^{\widehat{n}}$ , y  $\theta_l \in \mathbb{R}^{\widehat{l}}$  que minimice la siguiente función

[8]:

$$F(\theta_m, \theta_l) = \sum_{t=1}^n |y_t - \Phi_t(\theta_n)\theta_l|^2 \quad (3.46)$$

En la expresión anterior  $\Phi_t(\theta_n)\theta_l$ , donde  $\Phi_t(\theta_n) \in \mathbb{R}^{\widehat{p} \times \widehat{l}}$  es una matriz no lineal que es función de  $\theta_n$ , es el modelo de datos donde se asume que no hay parámetros comunes entre  $\theta_n$  y  $\theta_l$ .

Este tipo de modelo de datos toma una forma especial: por una parte es una función no-lineal de ciertos parámetros, y también es una función lineal del resto de los parámetros.

Si  $\theta_n$  es conocido, entonces el correspondiente  $\theta_l$  que minimiza  $F$  está dado por  $\Phi_t(\theta_n)^+ Y$ , donde  $M^+$  denota la pseudoinversa de una matriz  $M$ .

Entonces el problema se divide en dos pasos.

- Primero se encuentra el vector  $\theta_n$  que minimiza la función  $F(\theta_n) = |Y - \Phi_t(\theta_n)\Phi_t(\theta_n)^+ Y|$
- En el segundo paso se calcula  $\theta_l$  para  $\Phi_t(\theta_n)^+ Y$ .

---

## Redes Neuronales

---

### 4.1 Introducción

Las Redes Neuronales Artificiales fueron originalmente una simulación abstracta de los sistemas nerviosos biológicos.

El primer modelo de una red neuronal fue propuesto por McCulloch and Pitts en 1943. En este, las neuronas fueron presentadas como modelos de neuronas biológicas y como componentes de circuitos que pueden realizar tareas computacionales [9].

Durante este capítulo se expondrán algunas de las características, ventajas y aplicaciones de las RNA, en especial de las aplicaciones y características que sirven para utilizar las RNAs como herramientas para la predicción.

Es este punto el que tiene relación con este trabajo y en el cual se profundizará más a lo largo del capítulo.

Quizás, la característica más importante de una RNA es su capacidad de aprendizaje, lo cual logra principalmente a través de la presentación de ejemplos del ámbito donde se aplicará.

Es por esta razón que este tipo de redes recibe el nombre de RNA, haciendo una analogía a las redes neuronales biológicas (RNB), de las cuales recibe su nombre, y es debido a que las RNB tienen la capacidad de aprender de situaciones anteriores y de aplicar este conocimiento adquirido para tomar decisiones o elegir entre distintas acciones a seguir.

Para emular este comportamiento que posee el cerebro humano, se utiliza un estilo de computación denominado en paralelo, mediante lo cual una RNA puede apren-

der de los ejemplos que recibe y tomar decisiones de acuerdo a la tarea que tenga encomendada.

En general se puede considerar a una RNA como un "modelo simplificado del cerebro humano"[9].

Resulta natural pensar que la búsqueda de arquitecturas para modelar la solución a un problema busque alguna similitud con algo que ya existe y se aplique a problemas similares. Es por esta razón que no debe parecer extraño que el modelo matemático que se está analizando tome las similitudes y características del comportamiento del cerebro humano.

El cerebro humano puede resolver problemas apelando a los conocimientos adquiridos durante su vida, y aplicando ciertas reglas inferidas desde su propia experiencia. La aplicación de este conocimiento se hace a través de la red neuronal que compone el cerebro. Esta red está compuesta por gran cantidad de pequeñas neuronas, las que al trabajar paralelo y conjuntamente pueden resolver los problemas que se le presenten.

Es este comportamiento paralelo lo que trata de emular una RNA, a través de pequeños componentes denominados neuronas artificiales.

En general una RNA posee las siguientes características:

- Está compuesta por un gran número de elementos simples que procesan de modo similar a las neuronas.
- Un gran número de conexiones con pesos entre los elementos. Estos pesos sirven para codificar el conocimiento de la red.
- Control distribuido altamente paralelo.
- Un énfasis en el aprendizaje automático de representaciones internas.

### 4.1.1 Reseña Histórica

La historia de las RNA parte en la década del 40 destacan los trabajos realizados por Warren McCulloch y Walter Pitts [10], que realizaron el primer modelo matemático de redes neuronales artificiales. El modelo que presentaron estaba basado en la idea de que las neuronas funcionaban mediante impulsos binarios, además el modelo

presentaba la atrayente capacidad del aprendizaje.

Siete años más tarde, Donald Hebb, en su libro *Organization of Behavior*, presenta un procedimiento matemático para el aprendizaje que hoy lleva su nombre, aprendizaje Hebbiano [11].

En 1951, Marvin Minsky obtuvo los primeros resultados prácticos en RNA, trabajando con Edmons, diseñaron una máquina con 40 neuronas con aprendizaje Hebbiano. Albert Uttley, en 1955, aplicó en sus trabajos en RNA la teoría de Shannon, mediante la cual se ajustaban los parámetros de entrada.

En 1957, Frank Rosenblatt crea el perceptron, generalizando el modelo de células de McCulloch-Pitts añadiéndole aprendizaje.

En 1959, Bernard Widrow creó el Adaline, *Adaptative Linear Element*, el cual es similar al perceptron. Esta RNA ajusta los pesos entre los niveles de entrada y salida de acuerdo al error entre el valor de salida obtenido y el esperado.

James Anderson en sus trabajos de 1968 y 1973 publicó un modelo de memoria basado en la asociación de activaciones de la sinapsis de una neurona, además realizó un modelo de asociación lineal, siguiendo el planteamiento de Hebb. Creó el *Brain-State-in-Box* que consistía en un nuevo método de corrección de error, y que sustituía la función umbral lineal por una de rampa.

John Hopfield en 1982 y 1984, presentó una variación del asociador lineal basado en la función de energía de Lyapunov para ecuaciones no lineales. Hopfield demuestra que se puede construir una función de energía que describa la actividad de una Red Neuronal monocapa, en tiempo discreto, y que esta ecuación puede ir disipando y la red converger a un valor mínimo local, lo que hizo resurgir el interés por aplicar las RNA a problemas difíciles que los computadores convencionales no podían resolver. Kunihiko Fukushima entre 1969 y 1982, desarrolló el neocognitio, un modelo de red neuronal para el reconocimiento de patrones visuales.

El MIT Lincoln Laboratory publica en 1988, un compendio del conocimiento neuronal hasta la fecha, denominado DARPA, *Neural Networks Study*.

En 1984, sale a la luz el algoritmo de aprendizaje de la máquina de Boltzmann, creado por el Dr. Terence Sejnowski. Junto con Geoff Hinton (1986), realizan la primera red

neuronal artificial de tres niveles que reconoce un algoritmo de aprendizaje [12].

McClelland y Rumelhart popularizaron en 1986 el algoritmo Backpropagation en el libro "Parallel Distributed Processing: Explorations in the Microstructure of Cognition", el cual fue redescubierto simultáneamente por una serie de investigadores.

En la actualidad el crecimiento más importante se sitúa en el entorno de las aplicaciones, en especial los campos relacionados con la industria y la empresa.

Los sectores industriales y de empresa en los que se utilizan redes neuronales artificiales se encuentran: Negocio y Marketing, Banca, Transporte, Energía, entre otros sectores.

#### 4.1.2 Red Neuronal Biológica

La cantidad de neuronas presentes en el cerebro humano es cercana a los cien mil millones, las neuronas al conectarse entre ellas, generan más de mil sinapsis por término medio a su entrada y salida. De este modo, sus características hacen de esta red un sistema cuya complejidad es muy superior a cualquier otro producto del ser humano[13].

Para llevar a cabo la comunicación entre neuronas se utilizan neurotransmisores, los cuales son liberados por la neurona emisora y captados por la neurona receptora.

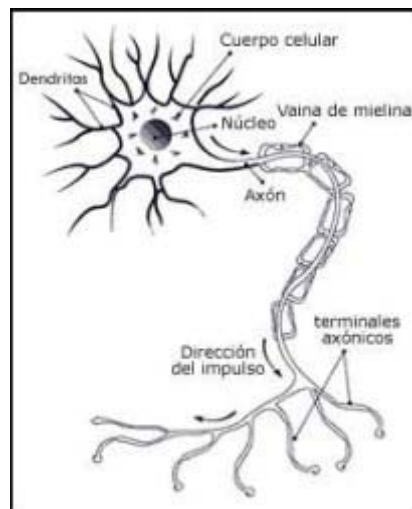


Figura 4.1: Red Neuronal Biológica

En la Figura 4.1 se muestra la constitución de una neurona biológica. En ella se

aprecian las dendritas, las cuales cumplen la función de entrada de los impulsos que llegan de otras neuronas a través de un Axón. El contacto de cada axón con una dendrita se realiza a través de la sinapsis. Tanto el axón como las dendritas transmiten la señal en una única dirección.

#### 4.1.3 Red Neuronal Artificial (RNA)

Como ya se ha visto en puntos anteriores, una RNA busca imitar el comportamiento del cerebro humano en especial lo relacionado a la RNB.

Sin embargo, pueden considerarse solo como un modelo matemático, que simplifica el comportamiento de una red neuronal biológica. Al igual que una RNB, su estructura está compuesta de pequeños elementos que reciben el nombre de neuronas.

En general, se considerará una RNA como un producto software que está basado en modelos que imitan la estructura del cerebro humano.

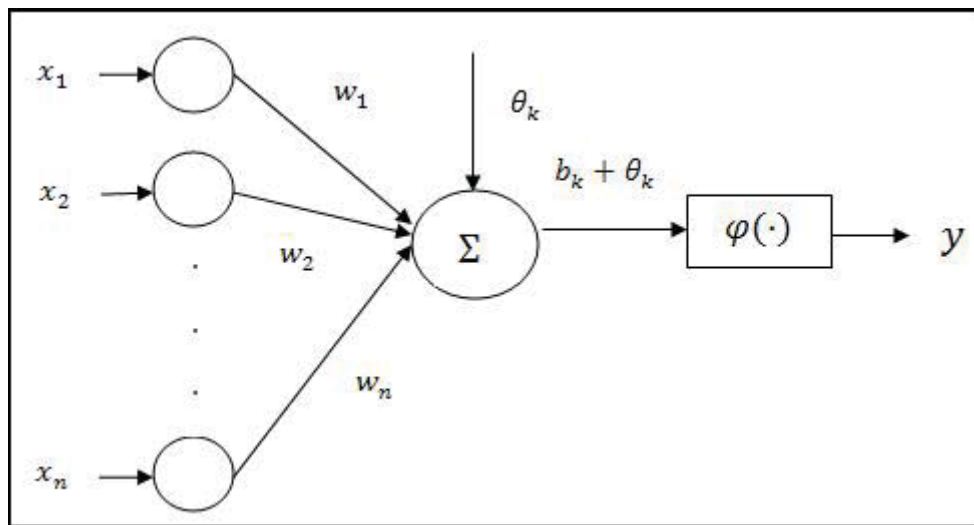


Figura 4.2: Red Neuronal Artificial

Una neurona artificial está compuesta de al menos tres elementos: una serie de conexiones de entrada, cada una con un peso que puede tomar valores negativos o positivos; un estado interno de la neurona que puede ser alterado por los valores de entrada que han sido linealmente combinados con los pesos de las conexiones; y una función de activación que determina el estado de activación de la neurona una vez



evaluadas las entradas a la misma, este estado es el propagado hacia el resto de la red.

La Figura 4.2 muestra la estructura básica de una neurona artificial. Matemáticamente la salida de una neurona artificial estará determinada por los datos de entrada, la función de activación y los pesos de las conexiones de la neurona. Si se considera el estado interno de la neurona  $k$  como  $b_k$ , el vector de entrada de la neurona como  $x_k = [x_{k,1}, x_{k,2}, \dots, x_{k,n}]$ , el vector de pesos de las conexiones  $w_k = [w_{k,1}, w_{k,2}, \dots, w_{k,n}]$ , la función de activación  $\psi_k$ , entonces la salida  $y_k$  de la neurona estará determinada por la Ecuación 4.1.

$$b_k = \sum_{i=1}^n x_{k,i} * w_{k,i} y_k = \psi(b_k + \theta_k) \quad (4.1)$$

El conjunto de neuronas artificiales conectadas es lo que se denomina red neuronal artificial. La Figura 4.3 presenta una RNA básica con nueve neuronas. Las neuronas dentro de una RNA presentan conexiones entre ellas lo que determina su topología. Comunmente las neuronas se organizan en capas que se conectan entre si. Las neuronas que componen una capa aparecen conectadas con neuronas de la siguiente. Principalmente su funcionamiento comprende la inserción de datos por medio de la capa de entrada, los que pasan a la capa oculta (pueden ser cero o más), y salen por la capa de salida.

En el caso de este proyecto, existirán 3 capas dentro de la red, una capa de entrada en la cual se insertarán los datos dentro de la red, una capa oculta, y una capa de salida.

Haciendo la analogía con la neurona biológica, la entrada de una neurona puede asimilarse al funcionamiento de las dendritas en la neurona biológica, los pesos con la sinapsis, y la salida de la neurona artificial con el Axón de la neurona biológica.

En los siguientes párrafos se detallan las funciones presentes en una neurona [9].

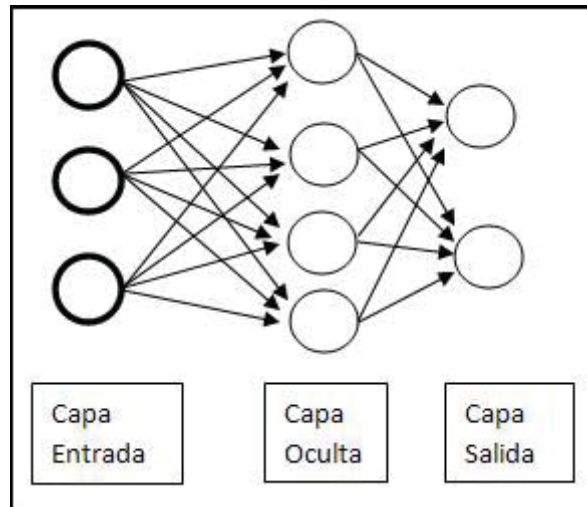


Figura 4.3: Modelo de una Red Neuronal

#### 4.1.3.1 Función de entrada

La función de entrada tiene por objetivo agrupar los datos de manera de utilizarlos en la función de activación. Los datos agrupados por la función de entrada corresponden a los  $n$  datos que recibe la neurona de las neuronas conectadas a ella. Generalmente la aplicación de la función de entrada al vector de datos corresponde con alguna de las siguientes operaciones:

- I. **Sumatoria de las entradas pesadas:** Corresponde a la suma de todos los valores de las entradas a la neurona multiplicados por sus correspondientes pesos.

$$\sum_i (n_{i,j} w_{i,j}), \text{ con } j = 1, 2, 3, \dots, n; i \text{ número de neurona} \quad (4.2)$$

- II. **Productoria de las entradas pesadas:** En este caso es el producto de todos los valores de las entradas a la neurona multiplicados por los correspondientes pesos.

$$\prod_i (n_{i,j} w_{i,j}), \text{ con } j = 1, 2, 3, \dots, n; i \text{ número de neurona} \quad (4.3)$$

- III. **Máximo de las entradas:** El resultado de la aplicación de esta función de entrada

es el máximo de todos los valores de entrada previamente multiplicado por su correspondiente peso.

$$\text{Max}_i(n_{i,j}w_{i,j}), \text{ con } j = 1, 2, 3, \dots, n; i \text{ número de neurona} \quad (4.4)$$

#### 4.1.3.2 Función de activación

Esta función determina si una neurona está activa o inactiva, es decir, determina el estado de excitación de la neurona. Existen casos en que la neurona puede lograr un nivel de activación, lo que implica que tome un valor dentro de un intervalo específico.

La función de activación toma los valores de la entrada global y lo resta del valor del umbral,  $\theta$ , para luego transformar este resultado en un valor que representa su estado de activación. El rango que puede tomar este valor es de  $[0, 1]$  o de  $[-1, 1]$ . Estos rangos representan el estado de la neurona cuando está totalmente activa (1) o totalmente inactiva (0 o  $-1$ ).

La función de activación es una función de entrada global menos el umbral. A continuación se detallan algunas de las funciones de activación más utilizadas al momento de implementar una RNA:

- i. **Función Lineal:** Los valores de salidas obtenidos por medio de esta función de activación serán  $a * (gin_i - \theta_i)$ , cuando el argumento de  $(gin_i - \theta_i)$  está comprendido dentro del rango  $(-1/a, 1/a)$ . Por encima o por debajo de este rango se fija la salida en 1 o  $-1$ , respectivamente. Cuando  $a = 1$ , la salida es igual a la entrada.

$$f(x) = \begin{cases} -1 & x \leq -1/a \\ a * x & -1/a < x < 1/a \\ 1 & x \geq 1/a \end{cases} \text{ con, } a > 0. \quad (4.5)$$

- ii. **Función sigmoidea:** Los valores de salida de esta función están comprendidos entre 0 y 1. El modificar el valor de  $g$  tiene influencia en la pendiente de la

curva.

$$f(x) = \frac{1}{e^{gx} + e^{-gx}} \text{ Con, } x = gin_i - \theta_i. \quad (4.6)$$

III. **Función tangente hiperbólica:** Los valores de salida de la función tangente hiperbólica están comprendidos en el rango de  $-1$  a  $1$ . Al igual que en la función sigmoidea, al modificar el valor de  $g$  se ve afectada la pendiente de la curva.

$$f(x) = \frac{e^{gx} - e^{-gx}}{e^{gx} + e^{-gx}} \text{ Con, } x = gin_i - \theta_i. \quad (4.7)$$

#### 4.1.3.3 Función de salida

Finalmente, para que una neurona esté completa y pueda realizar la tarea para la que fue creada, necesita de una función de salida.

El valor entregado por esta función corresponde al valor que se transfiere a las neuronas a las que estuviese vinculada. Toma como parámetros el resultado de la función de activación y lo compara con un determinado umbral, en el caso de que la diferencia entre ambos sea negativa, no se pasa ningún valor a la neurona siguiente. En el caso que el resultado sea positivo el valor que la neurona entrega a las neuronas siguientes estará comprendido en el rango  $[0, 1]$  o  $[-1, 1]$ , sin embargo también puede ser binario  $\{0, 1\}$  o  $\{-1, 1\}$ , en este último caso, y al igual que en la función de activación, se entenderá que un  $1$  significará que la neurona está activa, y un  $0$  o un  $-1$  que la neurona se encuentra inactiva.

En algunos casos los valores entregados por esta función no necesariamente pertenecen a alguno de estos rangos.

Las funciones de salida más comúnmente utilizadas son:

- I. **Ninguna:** Es denominada también función de identidad. La salida entregada corresponde a la misma que la entrada.
- II. **Binaria:** La salida de esta función será

$$f(x) = \begin{cases} 1 & \text{si } activacion_i \geq \xi_i \\ 0 & \text{de lo contrario} \end{cases} \text{ donde } \xi_i \text{ es el umbral.} \quad (4.8)$$

### 4.1.4 Ventajas de una RNA

Al estar basado en el comportamiento del cerebro humano, una RNA debiera poseer algunas de las características más importantes de éste. Es por ejemplo el caso de la capacidad de aprendizaje, de generalizar a partir de la experiencia de casos anteriores a nuevos casos, entre otras características. A continuación se explican algunas de estas ventajas [9].

**Aprendizaje Adaptativo.** Es quizás la característica más importante de una red neuronal, al utilizar una RNA no es necesario tener modelos a priori o especificar alguna función de probabilidad para aplicar al problema, ya que la red puede generar su propio modelo, debido a la propiedad de aprendizaje adaptativo.

Para que la RNA pueda aprender y adaptarse al problema es que se entrena con ejemplos ilustrativos del problema en que se aplique.

**Auto-organización.** Tiene relación con la aplicación del aprendizaje adaptativo de la red, en este caso la auto-organización consiste en la modificación de la RNA completa para llevar a cabo un objetivo específico, en el caso anterior la modificación se producía en la organización de la información que se recibe durante el aprendizaje y/o la operación.

**Tolerancia a Fallos.** Una RNA posee una capacidad inherente a fallos, ya que la falla de un número no mayor de neuronas de la red no provoca una caída repentina de ésta, aun cuando el comportamiento del sistema si se vea afectado.

Se puede ver la tolerancia a fallos desde dos puntos:

Primero, la capacidad de la red de reconocer patrones aun en la presencia de ruido o distorsiones, denominado tolerancia a fallos con respecto a los datos.

Segundo, aun cuando se destruya parte de la red, esta puede seguir funcionando.

**Operación en tiempo real.** Tiene relación con la capacidad de las RNA de realizar sus tareas en forma paralela, lo que aumenta sus aplicaciones en las áreas que se desempeñan en tiempo real.

#### 4.1.5 Características de una RNA

El primer modelo de una RNA fue diseñado por McCulloch-Pitts en 1943 [10]. Este modelo constaba de un conjunto de neuronas y cada una de éstas constaba de dos estados discretos. Asociados a cada uno de estos estados se conseguía un output que se transmitía a través de la estructura de conexión que poseía la neurona con las otras neuronas presentes en la red. El modelo tenía como limitación el hecho de solo poder computar funciones booleanas .

El modelo de McCulloch-Pitts posee las siguientes hipótesis:

- El estado de una neurona en el tiempo  $t + 1$  depende solamente del estado de la neurona en el periodo anterior,  $t$ .
- Una neurona estará activa o no si supera un umbral,  $\theta$ .
- Y por último, se asume la sincronía entre input-output aun conociendo la asincrona en la neurona biológica.

La formalización de este modelo consiste en primer lugar, en definir el estado de la entrada  $x_t$  y en segundo lugar, la salida en el momento  $y_t$ . La expresión que describe su funcionamiento es:

$$y_i^{t+1} = f\left(\sum_{j=1}^m w_{i,j}x_j^t - \theta_i\right) = f(a) \quad (4.9)$$

$$f(a) = \begin{cases} 1 & a \geq 0 \\ 0 & \text{en otro caso} \end{cases} \quad (4.10)$$

donde la función  $f(\cdot)$ , esta descrita por  $n + 1$  parámetros,  $n$  -pesos  $(w_1, w_2, \dots, w_n)$  y el umbral o tendencia  $(\theta)$ .

Para el funcionamiento de una red neuronal se necesitan tres elementos: la estructura de nodos, la topología de la red y el algoritmo de aprendizaje. Estos elementos permiten la extracción de la estructura estadística presente en los datos, y además en el caso del algoritmo de aprendizaje, es utilizado para estimar los pesos o parámetros de la red.

### 4.1.6 Clasificación

#### 4.1.6.1 Según su arquitectura

De acuerdo a esta clasificación se tienen tres tipos de RNA: redes feed-forward, redes feedback y redes recurrentes o retroalimentadas. Principalmente las diferencias entre cada uno de los tipos de redes de esta clasificación tienen relación con las direcciones que pueden tomar las conexiones entre las neuronas de las distintas capas y entre neuronas de la misma capa.

- I. **Redes feed-forward.** En este tipo de redes solo se permiten conexiones entre capas sucesivas. Además la información dentro de la red se propaga en una sola dirección, que va desde la capa de entrada, pasando por las capas ocultas, hasta la capa de salida.(Figura 4.4)

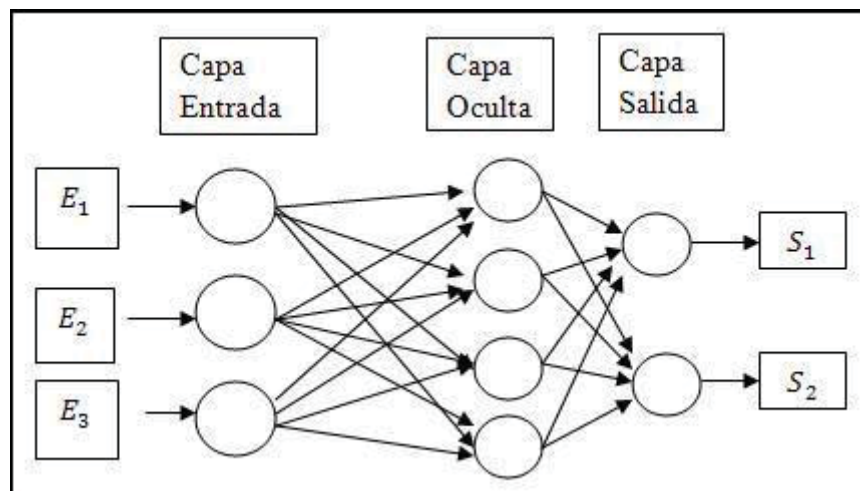


Figura 4.4: Representación de una arquitectura feed-forward

- II. **Redes feedback.** A diferencia del tipo de redes anterior, aquí si se permiten las conexiones hacia atrás, y además se permiten conexiones laterales, de esta manera el flujo de información de la red puede ir en cualquier dirección y tener la capacidad de que la salida de una capa de un nivel mayor, entendiéndose por mayor una capa cercana a la capa de salida, sea la información de entrada de una neurona de una capa de nivel menor, o de alguna neurona de la misma capa.(Figura 4.5)

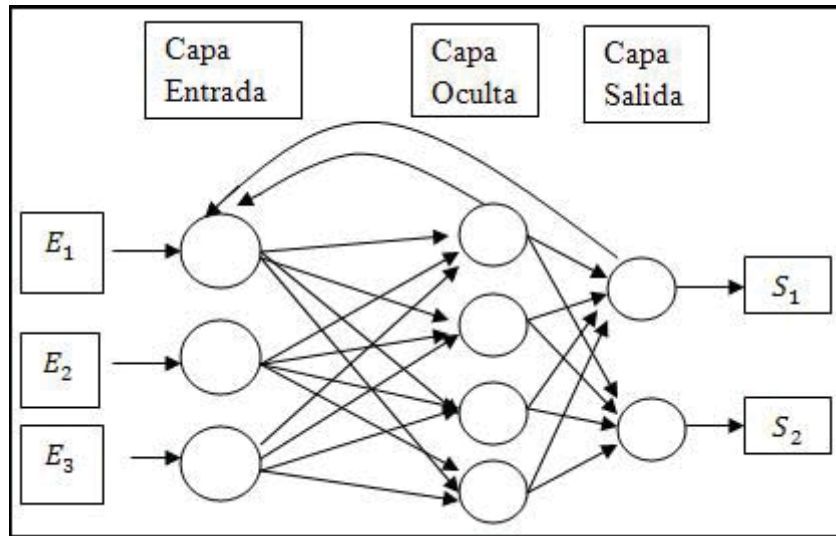


Figura 4.5: Representación de una arquitectura feedback

III. **Redes Recurrentes o retroalimentadas.** Se presenta como una combinación de los tipos de redes descritos anteriormente, permite tanto las conexiones hacia atrás como hacia adelante y lateralmente, además permite la realimentación de una neurona, lo que significa que una neurona tenga una conexión hacia ella misma.

#### 4.1.6.2 Según la forma de aprendizaje

Desde este punto de vista una RNA se puede clasificar en cuatro formas de aprendizaje: supervisado, no supervisado, aprendizaje reforzado e híbrido.

- I. **Aprendizaje supervisado:** El aprendizaje supervisado (Figura 4.6), consiste en un modelo neuronal en el que se pueden estimar las relaciones entre los inputs y outputs sin la necesidad de proponer una forma funcional a priori. Principalmente en su funcionamiento se compara el output obtenido con el output deseado del modelo neuronal, a modo de encontrar el error de salida, que es calculado comparando ambos outputs.  
Este tipo de aprendizaje será el utilizado en este proyecto.
- II. **Aprendizaje no supervisado o autoorganizado:** Es aquel en el que no se posee un output deseado para comparar el output del modelo neuronal, con la cual



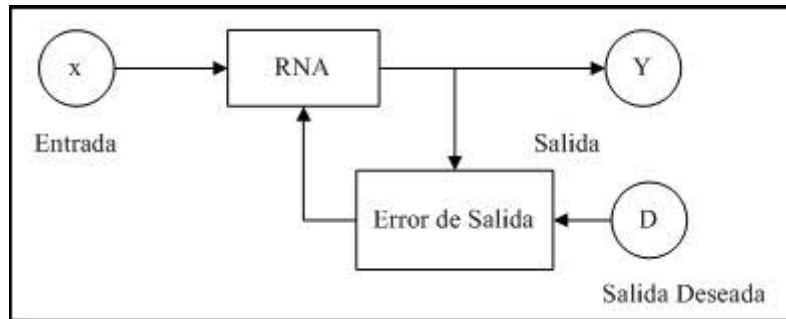


Figura 4.6: Aprendizaje Supervisado

no poseen influencia externa para ajustar los pesos de las conexiones entre neuronas (Figura 4.7). Debido a sus características este tipo de aprendizaje es utilizado para descubrir regularidades presentes en los datos, extraer rasgos o agrupar patrones según su similitud.

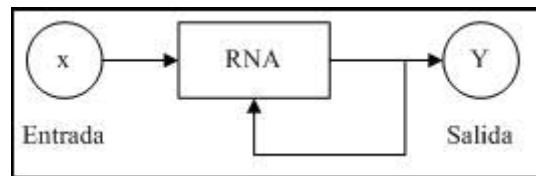


Figura 4.7: Aprendizaje No Supervisado

- III. **Aprendizaje reforzado:** El aprendizaje reforzado (Figura 4.8) consiste en una media de los dos aprendizajes descritos anteriormente. Esto significa que se cuenta con información acerca del error cometido calculado de forma global y no para cada uno de los outputs, pero no se cuenta con el output deseado. Este método se basa en la idea de premio-castigo, con lo cual se refuerza toda acción que pueda significar una mejora del modelo mediante la definición de una señal crítica.
- IV. **Aprendizaje Híbrido:** Finalmente se tiene el aprendizaje Híbrido, el cual posee los dos primeros tipos de aprendizaje antes explicados, el supervisado y el no supervisado, de manera que coexisten en distintas capas de neuronas.

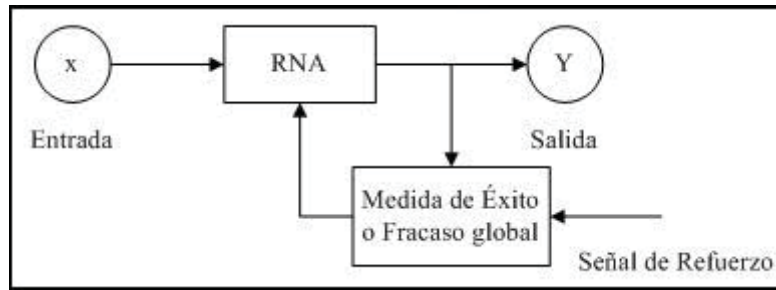


Figura 4.8: Aprendizaje Reforzado

#### 4.1.6.3 Según el área de aplicación

Esta clasificación hace énfasis en el campo de acción de la RNA, en especial se refiere al tipo de problema que se desea resolver utilizando esta tecnología.

A continuación se presentan distribuidas las Redes neuronales respecto a seis grandes grupos de aplicación [13].(Figura 4.9)

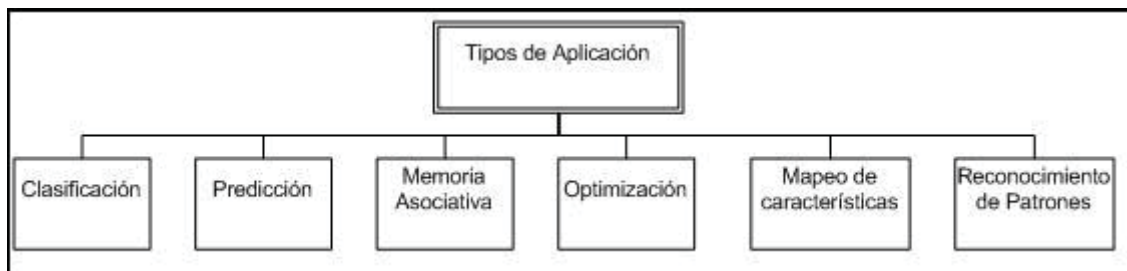


Figura 4.9: Clasificación de las RNA

1. **Clasificación:** Se pueden dar dos tipos de aplicación de clasificación utilizando RNA. Estos son:
  - a) **Clasificación no supervisada:** Para esta aplicación los pesos sinópticos de la red son entrenados mediante la técnica de aprendizaje no supervisado.
  - b) **Clasificación supervisada:** Adopta criterios de interpolación o aproximación. Se utilizan en muchas aplicaciones, por ejemplo, el reconocimiento de voz. En esto casos, los datos de entrenamiento consisten en pares de patrones de entrada y salida.

- II. **Predicción:** Su funcionamiento es tal que una vez entrenada la red con datos de ejemplos, estará capacitada, con un grado de error, para predecir el comportamiento futuro de los datos con los cuales se entrenó.

En este grupo de aplicaciones es que está inserto este trabajo, ya que una vez entrenada la red con la base de datos disponibles sobre los niveles de captura de anchovetas, la red podrá predecir los niveles futuros de captura.

- III. **Memoria Asociativa:** Corresponde a dos tipos de asociaciones: **autoasociación** y **heteroasociación**. El primer caso consiste en recuperar un patrón enteramente, dada una información parcial del patrón deseado. Su funcionamiento consiste en asociar la información de entrada con el ejemplar más parecido de los almacenados conocidos por la red.

La heteroasociación consiste en recuperar un conjunto de patrones B, dado un patrón de ese conjunto.

Los pesos en estos tipos de redes son a menudo predeterminados basados en la regla de Hebb. Normalmente, la autocorrelación del conjunto de patrones almacenado determina los pesos en las redes autoasociativas. Por otro lado, la correlación cruzada de muchas parejas de patrones se usa para determinar los pesos de la red de heteroasociación.

- IV. **Optimización:** Consisten en la resolución de problemas de optimización combinatoria que realizándolos de manera convencional tomará mucho tiempo de computación.

- v. **Mapeo de características:** Consiste en la simulación de la capacidad del cerebro humano de crear mapas topológicos de las informaciones recibidas del exterior.

- VI. **Reconocimiento de Patrones:** En este tipo de aplicaciones, la RNA debe inducir la característica saliente del estímulo de entrada y detectar la regularidad de los datos.

#### 4.1.7 Perceptron

Es uno de los modelos de redes neuronales artificiales más usado. Utiliza una arquitectura feed-forward y aprendizaje supervisado. Poseen en la red solo una capa oculta, y puede aproximar hasta el nivel deseado dentro de un conjunto compacto cualquier función continua [14] (Figura 4.10).

Las redes de perceptron multicapa, habitualmente son entrenadas con el algoritmo de aprendizaje Back-propagation, uno de los algoritmos con más importancia histórica en el desarrollo de las redes neuronales [15].

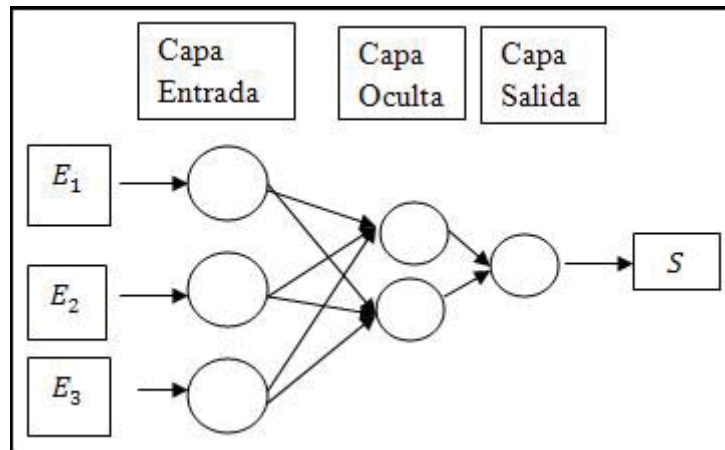


Figura 4.10: Clasificación de las RNA

#### 4.1.8 Red Neuro-Wavelet. (Red NW)

Una Red Neuro-Wavelet (RNW) es una red neuronal artificial cuya función de activación corresponde a una transformada Wavelet.

La teoría Wavelet está basada en la teoría de conjuntos que permiten representar una señal o un campo tanto en escala como espacio y posibles direcciones [16].

Estas funciones se emplean como funciones de análisis de modo que examinan la señal de interés para obtener sus características de espacio, tamaño y dirección.

Principalmente las Wavelet son generadas a partir de una Wavelet madre, a la cual se le aplican dos variables, de escala ( $a$ ) cuya función es la de permitir dilataciones y

contracciones de la señal, y una variable de traslación ( $b$ ), la cual permite mover la señal en el tiempo.

La transformada Wavelet es una herramienta matemática que permite resolver problemas que la Transformada de Fourier por ventanas (TFV)[17], no puede, como es el caso del análisis de señales tanto en tiempo como en frecuencia [18].

A diferencia de la TFV en la TW el tamaño de las ventanas puede ser adaptado a las frecuencias, al contrario de las TFV en que el tamaño de las ventanas queda fijo para todas las frecuencias analizadas, con las mismas resoluciones de tiempo y frecuencia. Como una definición general, se considera la transformada Wavelet de una función  $f(t)$  a la descomposición de  $f(t)$  en un conjunto de funciones  $\psi_{a,b}(t)$ , que forman una base. Este conjunto de funciones que se pueden obtener de esta base son las denominadas wavelets.

La transformada wavelet se define por la siguiente ecuación [16]:

$$W_f(a, b) = \int f(t)\overline{\psi_{a,b}(t)}dt \quad (4.11)$$

La wavelet madre se define como:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \quad (4.12)$$

donde  $a$  corresponde al factor de escala y  $b$  es el factor de traslación [19].

Para efectos de aplicar la transformada wavelet como función de activación en una RNA, se utiliza la transformada wavelet discreta, cuya descripción aparece en la Ecuación 4.13.

$$h_{a,b}(t) = 2^{-\frac{a}{2}}h(2^{-a}t - b) \quad (4.13)$$

Dentro de las wavelet más utilizadas se encuentran las siguientes [18]:

1. **Wavelet de Haar.** Es la más simple y antigua de las wavelet. Se describe mediante la siguiente ecuación:

$$h(x) = \begin{cases} 1 & 0 < x < \frac{1}{2} \\ -1 & \frac{1}{2} < x < 1 \\ 0 & \text{otro valor} \end{cases} \quad (4.14)$$

- ii. **Mexican Hat.** Recibe su nombre por la forma de su gráfica. Corresponde a la segunda derivada de la función de densidad de probabilidades Gaussiana, tiene la característica de ser simétrica, lo que le permite examinar las señales de un modo simétrico y lineal en la fase. Esta definida por:

$$mexh(x) = \frac{2(1-x^2)e^{-\frac{x^2}{2}}}{\Pi^{\frac{1}{4}}\sqrt{3}} \quad (4.15)$$

- iii. **Daubechies.** Este tipo de wavelet puede presentar diversos valores para definir su orden, el cual depende del número de momentos de desvanecimientos que se deseen. El orden de la wavelet será un número entero positivo que denota el número de coeficientes de filtro que posee la wavelet.

Otra particularidad de esta wavelet es la de realizar transformaciones continuas y discretas con ella.

- iv. **Gaussiana.** Se define como la primera derivada de la función de densidad de probabilidad Gaussiana. Esta definida por:

$$gaus(x, n) = C_n \frac{d}{dx}(e^{-x^2}, n) \quad (4.16)$$

donde  $C_n$  es una constante determinada por  $2 - normdegauss(x, n) = 1$ .

Con esta wavelet solo se puede realizar la transformada continua, y además puede ser simétrica o asimétrica dependiendo del valor que toma  $n$ .

## 4.2 Algoritmos Genéticos

Llevan su nombre debido a que están basados en el proceso genético de los individuos vivos, y se enfocan en el hecho de que si existe una población de individuos, será el individuo más apto el que sobreviva.

## CAPÍTULO 4. REDES NEURONALES

Se basa en los principios de selección y supervivencia natural postulados por Darwin (1859).

Los principios básicos del funcionamiento de los AG fueron establecidos por Holland en su trabajo *Adaptation in Natural and Artificial System* [20].

En la naturaleza los individuos de una población compiten por recursos que aseguren su supervivencia, así como también, pueden competir por encontrar compañero que asegure la continuidad de los genes y tener un mayor número de descendientes, lo cual trae la consecuencia de que si un individuo es elegido para procrearse, sus genes podrán estar presentes en una mayor cantidad de individuos en las generaciones posteriores.

Al elegirse individuos aptos se puede dar el caso en que existan superindividuos en las generaciones posteriores, que se adapten de mejor manera que sus ancestros, con lo cual las especies evolucionan logrando poseer características que se adaptan mejor a su entorno.

Los AG están basados en el comportamiento descrito en el párrafo anterior, haciendo una analogía directa de este comportamiento natural.

En primer lugar los AG trabajan con una población de individuos en que cada uno representa una solución factible para un problema dado. Según el grado en que el individuo responde como solución del problema, se le asigna un valor o puntuación. Este valor aumenta o disminuye de acuerdo al grado de adaptación del individuo al problema, y es utilizado como base para la elección de los individuos que serán utilizados para mezclarse y producir la siguiente generación.

Mientras mayor sea el valor de adaptación que posea el individuo, mayor será la probabilidad de que el individuo sea elegido como padre y mezclado su material genético con otro individuo seleccionado de la misma manera.

Al mezclar los padres se producen individuos que comparten algunas características de sus progenitores, y asegura que el material genético de sus padres no se pierda y se propague en las sucesivas generaciones.

Es así como se genera una nueva población de individuos que reemplaza a la anterior y verifica la interesante propiedad de que contiene una mayor proporción de buenas

características, y que en conjunto los individuos se adaptan de mejor manera como solución del problema que generaciones anteriores.

A medida que avanzan las generaciones se asegura la presencia de mejores individuos en la población y además de que sean estos los elegidos para el cruce, con lo cual se exploran las áreas más prometedoras del espacio de búsqueda.

Es importante destacar que si el AG ha sido bien diseñado, la población convergerá a la solución óptima del problema.

El poder de los AG radica en el hecho de que es una técnica robusta, y pueden ser aplicados a una gran variedad de problemas de diferentes áreas. Si bien no se garantiza que los AG encuentren una solución óptima, existe evidencia empírica de que se encuentran soluciones de nivel aceptable, en un tiempo competitivo en comparación con el resto de algoritmos de optimización combinatoria.

En el caso de técnicas especializadas en la resolución de un determinado problema, es probable que superen el desempeño del AG, es por esto que su aplicación está destinada a problemas en los cuales no existan técnicas especializadas.

Los AG pueden ser aplicados a las redes neuronales tanto para encontrar la arquitectura de la red como también para encontrar los pesos adecuados en las conexiones entre neuronas. Este procedimiento se logra mediante la codificación del problema mediante un cromosoma, el cual representa una solución factible para el problema dado.

### 4.2.1 Codificación

Los AG pueden ser utilizados para realizar la codificación de los pesos de las conexiones de una RNA, en estos casos, los pesos se codifican en el cromosoma representado por un arreglo. En el caso de una Red Neuronal Wavelet, la codificación resulta distinta de la de una Red Neuronal Tradicional, ya que posee dos parámetros en vez de pesos, que corresponden a la escala y traslación de la Wavelet Madre, los cuales se representan como pesos dentro de la red.

En experimentos realizados comparando este tipo de implementación con el algorit-



mo back-propagation, se obtuvieron resultados que indicaban un mejor desempeño por parte de los AG, y además que convergían a una mejor solución [21].

Para el caso de esta tesis, se utilizó una codificación con parámetros reales. Para la codificación se considera una red como la que aparece en la Figura 4.11

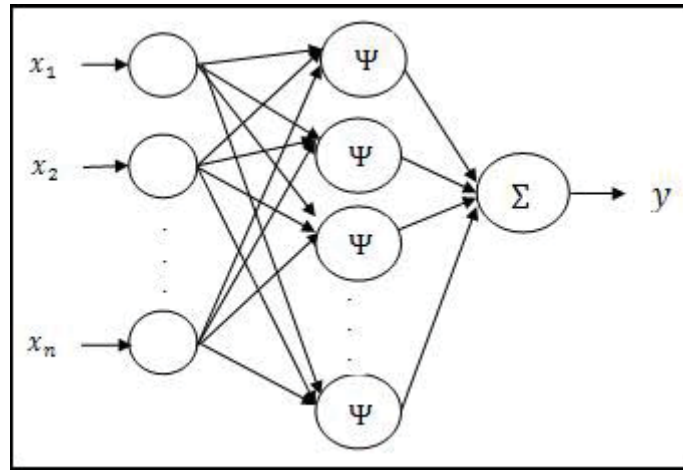


Figura 4.11: Representación de una Red Neuronal Wavelet

En una RNA Wavelet por cada nodo oculto presente en la red, existen dos parámetros, uno de escala y uno de traslación, es decir, tendremos  $2n$  parámetros entre la capa de entrada y la oculta, si existen  $n$  neuronas en la capa de oculta. Además existen otros parámetros que representan a los pesos en las conexiones entre las neuronas presentes en la capa oculta de la red y la neurona de salida de la misma.

Tomando en cuenta todos los parámetros mencionados anteriormente, la codificación de los parámetros de una red Neuro-Wavelet, estará compuesto por  $3n$  parámetros, donde  $n$  representa la cantidad de neuronas presentes en la capa oculta de la red [22].

Un ejemplo de la codificación real, se presenta en la Figura 4.12: considera los parámetros de escala ( $a_i$ ), y traslación ( $b_i$ ) como valores reales dentro del cromosoma, además de los pesos presentes en la capa de salida ( $w_i$ ). Con este estilo de codificación el cromosoma presenta una cantidad menor de genes en comparación con una codificación binaria.

En cuanto a la codificación de la arquitectura de la red, se pueden utilizar dos tipos de codificación:

- Codificar la estructura de la red mediante un cromosoma.

$a_1$	...	$a_n$	$b_1$	...	$b_n$	$w_1$	...	$w_n$
-------	-----	-------	-------	-----	-------	-------	-----	-------

Figura 4.12: Ejemplo de cromosoma real de una Red Neuro-Wavelet

- Utilizar una codificación gramatical de la estructura de la red.

El primero utiliza un arreglo de binarios, en el que se van agrupando los valores que representan la existencia de conexión de las neuronas en la capa de entrada con las neuronas de la capa oculta.

El segundo nace debido a la problemática de la codificación de la red en un cromosoma, cuando la red se hace muy grande. La solución fue presentar la red codificada en una gramática, mediante la cual se generan los individuos. Esta gramática contiene reglas que representan a los individuos que son posibles soluciones del problema [21].

Dentro del ámbito del proyecto, se considera la codificación en un arreglo de la arquitectura de cada una de las redes presentes en la regresión funcional.

La Tabla 4.1 presenta ejemplos de codificación de arquitectura de cuatro topologías de red. Al analizar la primera red presentada en la tabla, se observa una codificación en el arreglo  $[0, 1, 0, 0, 0, 1]$ , las dos primeras posiciones del arreglo representan las conexiones entre la primera neurona de la capa de entrada con las neuronas de la capa oculta, la tercera y cuarta posición corresponden a las conexiones de la segunda neurona de entrada con las neuronas de la capa oculta, finalmente las últimas dos posiciones del arreglo representan las conexiones de las neuronas de la capa oculta con la neurona de salida.

#### 4.2.2 Población

Un aspecto importante en la aplicación de AG, es la definición de dos elementos: **tamaño de la población** y **población inicial**.

El **tamaño de la población** es uno de los puntos claves a la hora de modelar el problema y adaptarlo. Desde un primer acercamiento se puede ver que si se elige una

Tabla 4.1: Codificación de topologías de Redes Neuronales

Codificación	Topología
[0, 1, 0, 0, 0, 1]	
[1, 0, 0, 0, 1, 0]	
[1, 0, 1, 1, 1, 1]	
[1, 1, 0, 0, 1, 1]	

población pequeña puede ocurrir que no se cubra bien el espacio de solución y que con poblaciones grandes puede traer problemas en el excesivo costo de computación. La **población inicial** se realiza al azar generando ristas en que cada gen puede contener los posibles valores del alfabeto con una probabilidad uniforme.

Existen trabajos en que la creación de la población inicial se realiza utilizando técnicas heurísticas o de optimización local, lo cual puede acelerar la convergencia del AG, sin embargo la rápida convergencia del algoritmo puede indicar que esta sea hacia óptimos locales.

Los AG basan su funcionamiento en la evolución de los individuos dentro de la población y en la generación de nuevos individuos a partir de los mejores de una generación anterior, para lo cual es necesario identificar la función que será utilizada para evaluar el desempeño de los AG en su aplicación en la regresión funcional neuro-wavelet.

### 4.2.3 Función Objetivo

Tan importante como la determinación de la codificación del problema utilizado es la elección de la función objetivo adecuada al ámbito del problema. Lo que se busca es construir funciones objetivos que verifiquen que si dos individuos están cercanos en el espacio de búsqueda, el valor entregado por sus funciones objetivo sean similares.

El tipo de función objetivo elegida puede tener gran influencia en la rapidez de convergencia del algoritmo, lo que se denomina convergencia prematura. Esto sucede cuando el algoritmo converge a óptimos locales, así como también puede suceder el caso contrario, en el que el algoritmo presenta una lenta convergencia.

La regla general para construir una buena función objetivo es que esta represente el valor del individuo de una manera real, debido a que, en algunas aplicaciones como las de optimización combinatoria, donde existen gran cantidad de restricciones, buena parte de los individuos representativos del espacio de búsqueda no son válidos.

La función de evaluación que se utilizará en el proyecto corresponde al Error Cuadrático Medio, este valor responde a la siguiente fórmula:

$$\frac{1}{n} * \sum_{i=0}^n (\widehat{y}_i - y_i)^2 \quad (4.17)$$

donde  $\widehat{y}_i$  representa la  $i$ -ésima salida de la regresión funcional, e  $y_i$  representa el valor observado real para el periodo  $i$ .

Esta fórmula permite medir la variación media experimentada por la regresión funcional, y mediante su uso se compara el valor retornado por la regresión funcional y

el valor esperado como salida.

Es esta función la que se pretende minimizar utilizando AG, y es por esto que se la ha elegido como la función de evaluación de cada uno de los cromosomas.

#### 4.2.4 Función de Selección

Una vez aplicados cada uno de los individuos pertenecientes a la población, se les calcula el valor de su función objetivo y se procede a la fase reproductiva para generar la nueva población de individuos que formará parte de la nueva generación, el proceso de selección de padres puede ser llevado a cabo de distintos modos. A continuación se presentan dos modos que fueron utilizados en este trabajo. Ambos métodos comprenden el hecho de que si el individuo está bien adaptado al problema sus posibilidades de ser elegido para la próxima generación son mayores a las de un individuo con menor valor en su función objetivo.

##### 4.2.4.1 Función de selección proporcional a la función Objetivo

Es la más utilizada, su funcionamiento consiste en que cada individuo de una generación tiene una probabilidad de ser seleccionado proporcional al valor de su función objetivo.

$$p_{i,k}^{prop} = \frac{g(I_t^i)}{\sum_{j=1}^{\lambda} g(I_t^j)} \quad (4.18)$$

donde  $p_{i,k}^{prop}$  representa la probabilidad de que el individuo  $I_t^i$  sea seleccionado como padre.

##### 4.2.4.2 Función de selección proporcional al rango del individuo

Tiene como fin evitar la rápida convergencia proveniente de los superindividuos, que surge al aplicarse la función de selección anterior. Su funcionamiento es el siguiente: si denotamos por  $rango(g(I_t^i))$  al rango de la función objetivo del individuo  $I_t^i$

cuando los individuos de la población han sido ordenados de menor a mayor (es decir el peor individuo tiene rango 1 y el mejor  $\lambda$ , siendo  $\lambda$  el tamaño de la población), y sea  $p_{j,t}^{rango}$  la probabilidad de que el individuo  $I_t^j$  sea seleccionado como padre cuando la selección se efectúa proporcionalmente al rango del individuo, se tiene que:

$$p_{j,t}^{rango} = \frac{rango(g(I_t^j))}{(\lambda(\lambda + 1)/2)} \quad (4.19)$$

#### 4.2.5 Operadores de cruce

Una vez elegidos los padres mediante la función de selección, se procede a cruzar a los individuos de modo de generar los individuos pertenecientes a la próxima generación, para esto se les aplican operadores de cruce. Dentro de los operadores de cruce utilizados en este proyecto se cuentan:

##### 4.2.5.1 Operador de Cruce en uno o más puntos

El operador de cruce toma dos padres seleccionados y corta sus ristas en uno o varios puntos al azar. En el primer caso (Figura 4.13), se generarán cuatro subristras, dos iniciales y dos finales. A continuación se mezclan las ristas con lo cual se generan dos individuos, los cuales heredan genes de cada uno de los padres. Este operador básico es conocido como operador de cruce basado en un punto.

Habitualmente, este operador no es aplicado a todos los individuos seleccionados, sino que se aplica de forma aleatoria con una probabilidad ente 0.5 y 1.0. En los casos en que el operador no se aplique, la descendencia se obtiene mediante la replicación de los padres en la generación siguiente.

En el caso de que los puntos de cruces sean más de uno, variará la cantidad de subristras generadas a raíz de los puntos de corte. La Figura 4.14 presenta el caso en que el operador de cruce se aplica en dos puntos.

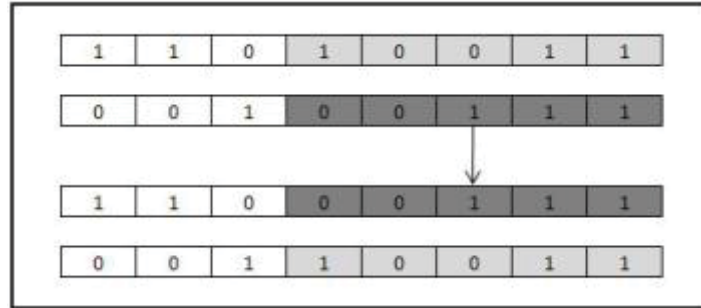


Figura 4.13: Operador de cruce en un punto

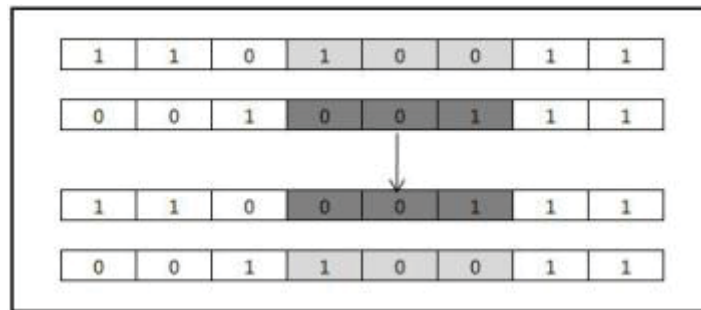


Figura 4.14: Operador de cruce en dos puntos

#### 4.2.5.2 Operador de cruce utilizando máscara de cruce

Consiste en la aplicación de una máscara de cruce (Figura 4.15) generada aleatoriamente, mediante la cual se generan los hijos, copiando los correspondientes genes de los padres de acuerdo a esta máscara. Cuando exista un 1 en la máscara de cruce el gen copiado será el del primer padre, en caso contrario será utilizado el gen del segundo padre.

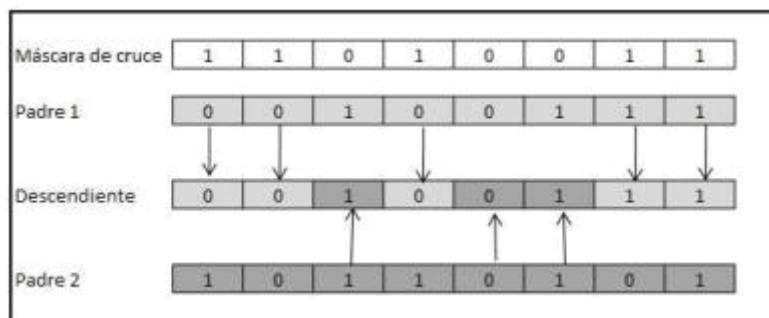


Figura 4.15: Operador de cruce Máscara de Cruce

### 4.2.6 Operador de Mutación

Finalmente, a cada individuo de la nueva generación se le aplica, con cierta probabilidad, el operador de mutación (Figura 4.16). El procedimiento consiste en modificar de manera aleatoria, con una probabilidad pequeña, cada gen del cromosoma. Este operador asegura que ningún punto del espacio de búsqueda posea probabilidad cero de ser examinado, y es vital para la convergencia del AG.

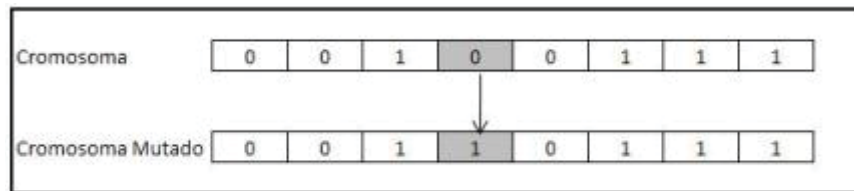


Figura 4.16: Operador de Mutación

La mutación es considerada como un operador básico que proporciona un cierto grado de aleatoriedad en la vecindad de los individuos de la población. Aunque se admite que el operador de cruce es más importante al momento de efectuar la búsqueda a través del espacio de posibles soluciones, el operador de mutación va adquiriendo importancia a medida que la población de individuos va convergiendo.



---

**Discusión de Resultados**

---

En este trabajo se implementará y evaluará la Transformada Wavelet Sombrero Mexicano como función de activación de las neuronas de la capa oculta de las redes neuronales a utilizar. Para efecto de evaluar el desempeño del modelo predictivo y poder contrastar los resultados obtenidos contra otros modelos, es que se definen a continuación los indicadores a utilizar.

La planificación considerada para realizar esta investigación se detalla en la siguiente tabla:

**Tabla 5.1: Planificación**

Tarea	Duración
Estudio Estado del Arte	Marzo 2009 - Junio 2009
Implementación aplicación de pronóstico	Mayo 2009 - Agosto 2009
Ejecución de pruebas	Septiembre 2009 - Octubre 2009
Evaluación del modelo y comparación de resultados	Noviembre 2009

### 5.1 Indicadores de Desempeño

Existen diversas métricas a considerar al momento de evaluar un modelo de pronóstico [23]. La más utilizada es el cálculo del coeficiente de determinación  $R^2$ , el cual mide el porcentaje de variabilidad presente en los datos que puede ser explicado

mediante el modelo. La fórmula es la siguiente:

$$R^2 = \frac{\sum_{i=0}^n SCE}{\sum_{i=0}^n SCT} \quad (5.1)$$

donde

$$SCE = \sum_{i=0}^n (\widehat{y} - y_i)^2 \quad (5.2)$$

representa la suma de los cuadrados del error, y

$$SCT = \sum_{i=0}^n (y_i - \bar{y})^2 \quad (5.3)$$

que representa la variación de los datos con respecto a su media, y que define la suma de los cuadrados totales.

El valor de  $R^2$  varía entre 0 y 1. Un valor cercano a 1 significa que el modelo explica la mayor parte de la variabilidad existente en los datos.

El Error Cuadrático Medio (ECM) corresponde a  $SCE/n$ , donde el término  $SCE$  fue anteriormente explicado.

Principalmente, este índice sirve para comparar los resultados obtenidos con los resultados obtenidos por otros autores en un ámbito similar o igual.

Otras medidas de evaluación son: La raíz del error cuadrático medio,

$$RMSE = \sqrt{\frac{\sum_{i=0}^N (y_i - \widehat{y}_i)^2}{N}} = \sqrt{ECM} \quad (5.4)$$

El error medio absoluto,

$$MAE = \frac{\sum_{i=0}^N |y_i - \widehat{y}_i|}{N} \quad (5.5)$$

donde  $y_i$  es el dato observado e  $\widehat{y}_i$  es el valor estimado de salida.

Otro indicador es el porcentaje del error en la predicción, este indicador permite comparar diferentes modelos de pronósticos aplicados a los mismos datos. Su fórmula es la siguiente:

$$SEP = \frac{100}{\bar{y}} * RMSE \quad (5.6)$$

donde  $\bar{y}$  es el promedio de los datos observados en la fase de validación.

El coeficiente de Eficiencia  $E_2$  y la Varianza relativa promedio  $ARV$  son usados para saber cómo el modelo explica la varianza total de los datos y representa la proporción de la variación de los datos observados considerados por el modelo. Sus fórmulas son:

$$E_2 = 1,0 - \frac{\sum_{i=0}^N |y_i - \widehat{y}_i|^2}{\sum_{i=0}^N |y_i - \bar{y}|^2} \quad (5.7)$$

$$ARV = \frac{\sum_{i=0}^N (y_i - \widehat{y}_i)^2}{\sum_{i=0}^N (y_i - \bar{y})^2} = 1,0 - E_2 \quad (5.8)$$

Otra forma de medir el desempeño del modelo es el índice de Persistencia.

$$PI = 1 - \frac{\sum_{i=0}^N (y_i - \widehat{y}_i)^2}{\sum_{i=0}^N (y_i - y_{t-L})^2} \quad (5.9)$$

donde,  $y_{t-L}$  es el dato observado al tiempo  $t - L$ , donde  $L$  es una valor determinado como un plazo de tiempo. En las aplicaciones referidas a pronóstico de captura de anchovetas, el valor  $L$  se iguala a 1. Un valor de  $PI$  de uno refleja un ajuste perfecto entre los datos pronosticados y los observados. Un valor de cero demuestra que el modelo no sirve para realizar pronóstico sobre el ámbito de análisis.

Finalmente el criterio de información Bayesiano y el criterio de información de Akaike, los cuales permiten comparar los valores obtenidos por un modelo, con los obtenidos por otros modelos de pronósticos.

$$AIC = \log(ECM) + \frac{2m}{N} \quad (5.10)$$

$$BIC = \log(ECM) + \frac{m * \log(N)}{N} \quad (5.11)$$

donde  $m$  es el número de parámetros del modelo. En las fórmulas anteriores, el primer término mide la capacidad del modelo para predecir de manera correcta los valores, y el segundo término penaliza la cantidad de parámetros del modelo.

## 5.2 Técnicas de filtrado de datos

Las técnicas de filtrado de datos son utilizadas para eliminar parte del ruido existente en la información recopilada.

Para el caso de este proyecto, se consideraron las técnicas de filtrado Promedio móvil tres y Convolución de parámetro  $\alpha$ , las cuales se describen a continuación.

### 5.2.1 Promedio móvil tres

El PM3 consiste en obtener el promedio de tres periodos consecutivos y considerar este promedio como el dato filtrado. Se le denomina promedio móvil debido a que los tres periodos que se consideran para el promedio varían a medida que se van obteniendo los datos.

### 5.2.2 Filtrado mediante convolución de parámetro $\alpha$

El filtrado mediante convolución considera la siguiente ecuación:

$$y'_t = \alpha y_t + (1 - \alpha) * y'_{t-1} \quad (5.12)$$

donde  $y'_t$  representa el dato filtrado en el tiempo  $t$ ,  $y_t$  el dato sin filtrar,  $\alpha$  representa el coeficiente de ruido aplicado a los datos, y finalmente  $y'_{t-1}$  es el dato filtrado del periodo anterior a  $t$ .

## 5.3 Regresión Funcional Neuro-Wavelet Evolutiva. NWE-AR.

Para obtener un modelo de predicción para el problema propuesto se utilizó información obtenida de tres fuentes principales: por un lado las estadísticas anuales de pesca del Servicio Nacional de Pesca [3], por otro, Servicio Nacional de Agrícola y Granadero de Chile [4] y finalmente, datos obetenidos del Instituto Peruano del Mar ([www.imarpe.gob.pe](http://www.imarpe.gob.pe)).

Esta información contiene los niveles de captura de anchoveta en la zona norte de

## CAPÍTULO 5. DISCUSIÓN DE RESULTADOS

Chile. Estos datos comprenden el periodo entre Enero de 1963 a Diciembre de 2005. Del conjunto de datos de los niveles de captura de anchovetas se obtuvieron dos subconjuntos, uno de ellos fue utilizado para el entrenamiento de las regresiones funcionales, que forman parte del modelo predictor, y el segundo conjunto fue utilizado para la validación del modelo encontrado.

Para el primer subconjunto de datos se eligieron, de manera aleatoria, el 60 % del total, lo que corresponde a 286 datos. Para el segundo subconjunto, se eligieron datos correspondiente a los 96 últimos del total, lo que corresponde a un 20 % de los datos totales.

Para el modelo propuesto se consideró una función wavelet Mexican Hat, de amplio uso como función de activación de RNA. La arquitectura de cada una de las redes neuronales presentes en la regresión funcional, estará compuesta por una capa de entrada, una capa oculta y una capa de salida con una neurona.

El desfase de cada RNA presente en la Autorregresión será de un máximo de ocho meses, aún cuando este valor podrá verse reducido, ya que será el algoritmo genético el que decida la arquitectura final de cada red neuronal.

Para el cálculo de la cantidad de neuronas ocultas también se utilizará el algoritmo genético, tomando como máximo 10 neuronas.

El desfase de la regresión también será de un máximo de ocho meses.

La función objetivo utilizada es el ECM. De este modo si un individuo tiene un menor ECM tendrá mayor probabilidad de ser elegido como padre de una nueva generación.

La función de selección utilizada es la "Función de selección proporcional al rango del individuo". Esta función de selección permite una búsqueda del mínimo "más suave" en comparación con la aplicación de la "Función de selección proporcional a la función objetivo".

La segunda función de selección lleva al algoritmo a converger tempranamente hacia mínimos locales lo cual fundamenta la elección de la primera función de selección.

El operador de cruce seleccionado fue el operador "Máscara de Cruce proporcional a la función objetivo". Este operador de cruce garantiza una mayor probabilidad de

que los genes de un buen individuo sean elegidos para conformar el individuo de la próxima generación.

En una primera instancia el tamaño de la población fue dejado fijo con 100, 200 y 300 individuos en la población. Con este tamaño de población se realizaron 30 pruebas. En cada una de las pruebas se utilizaron datos filtrados con las dos técnicas de filtro descritas anteriormente en este documento: promedio móvil tres (PM3) y convolución de parámetro  $\alpha$ .

En el caso del filtrado por convolución aplicado a los datos se utilizó un coeficiente de ruido ( $\alpha$ ) igual a 1/3.

En la Tabla 5.2 se presenta el resumen de los resultados obtenidos con el conjunto de datos filtrados con PM3, se presenta la media y varianza del indicador  $R^2$  tanto para la fase de entrenamiento como para la fase de validación del modelo.

**Tabla 5.2: Resumen de resultados con datos filtrados con PM3**

Tam. Pob.	Entrenamiento			Validación		
	MEDIA $R^2$	MAX $R^2$	VAR $R^2$	MEDIA $R^2$	MAX $R^2$	VAR $R^2$
100	0.72	0.83	0.01083	0.65	0.85	0.02315
200	0.83	0.86	0.00045	0.79	0.85	0.00141
300	0.83	0.87	0.00631	0.78	0.84	0.01378

De los resultados se resaltan el obtenido con un individuo con  $R^2$  de 0.87 en la fase de entrenamiento y además el máximo  $R^2$  de 0.85 en validación.

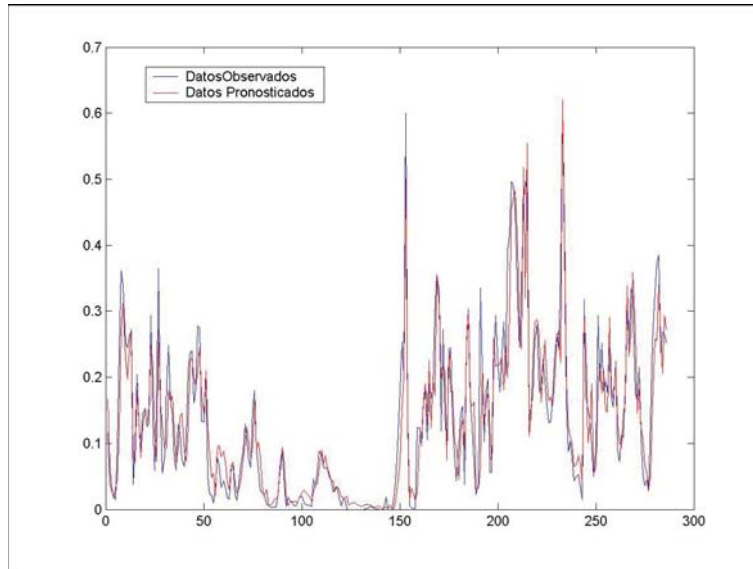
Estos resultados superan al mejor modelo encontrado con una red neuronal wavelet con entrenamiento con algoritmo genético realizado por el autor, modelo que con la misma cantidad de individuos en la población logra un  $R^2$  de 0.82 en la fase de validación.

Para el análisis posterior se seleccionó el modelo que presenta un mejor  $R^2$  en la fase de validación, es por esto que se utilizará el modelo obtenido con una población de 200 individuos, cuyo  $R^2$  fue de 0.85 en la etapa de validación y también en la fase de entrenamiento.

Para una mejor apreciación de los resultados obtenidos por el modelo analizado en párrafos anteriores, se presenta el gráfico que muestra el comportamiento de los

## CAPÍTULO 5. DISCUSIÓN DE RESULTADOS

datos pronosticados comparándolo con los datos observados (Figura 5.1). El gráfico muestra el comportamiento de las curvas de datos obtenidos y esperados durante la fase de entrenamiento del modelo. Al observar ambas curvas se aprecia un acople en los segmentos en que la variación de los datos no es elevada, en cambio cuando la curva de datos observados sufre grandes inflexiones, el modelo encontrado con filtro PM3 no logra alcanzar estos picos altos y bajos de la curva de datos observados.



**Figura 5.1: Gráfico fase de entrenamiento. PM3.**

Además de la falta de acople entre ambas curvas en los momentos de alta inflexión presentes en la curva de datos observados, el análisis de los resultados en la fase de entrenamiento hacen evidente el desfase presente en la curva entregada por el modelo en comparación con la curva de datos observados.

La evolución del indicador  $RMSE$  a lo largo de las generaciones del Algoritmo Genético se ve reflejado en el gráfico de la Figura 5.2. Al analizar este gráfico se puede notar que el algoritmo logra su convergencia aproximadamente en la generación número 650.

En la etapa de validación del modelo se vuelven a encontrar las mismas conclusiones obtenidas al analizar el gráfico de la fase de entrenamiento. En este caso el gráfico de la Figura 5.3 muestra las curvas de los datos entregados por el modelo y la curva de los datos observados durante la fase de validación.

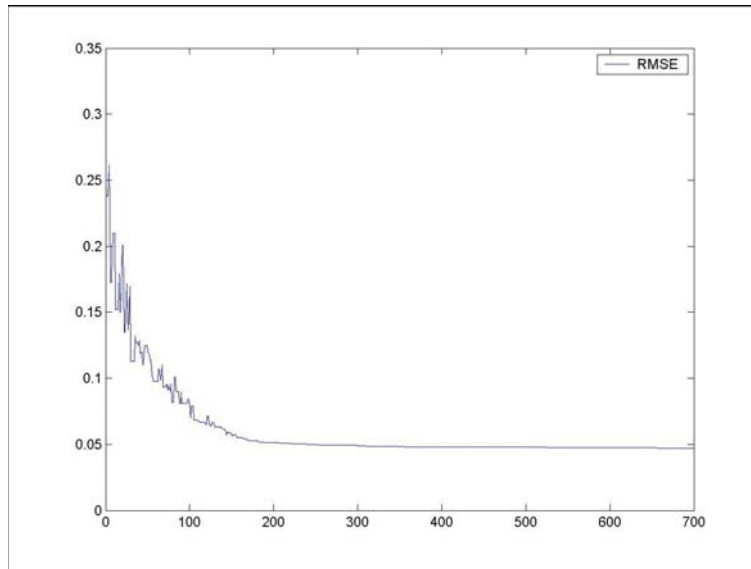


Figura 5.2: Gráfico RMSE. PM3.

Al igual que en la etapa anterior, al realizar un análisis sobre los resultados obtenidos, se concluye que el modelo es incapaz de igualar los picos altos y bajos de la curva de datos observados, aun cuando presenta cierto acople entre las curvas en los segmentos en que no existe una mayor variación. El desfase en este caso también se hace evidente.

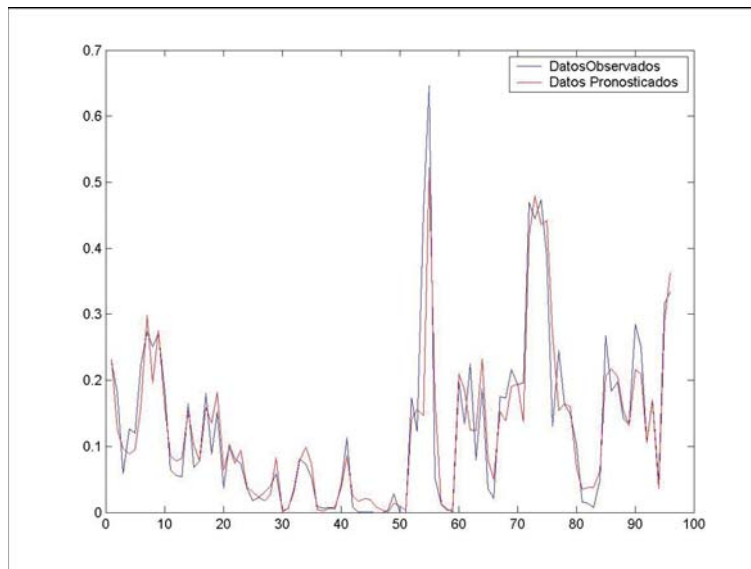
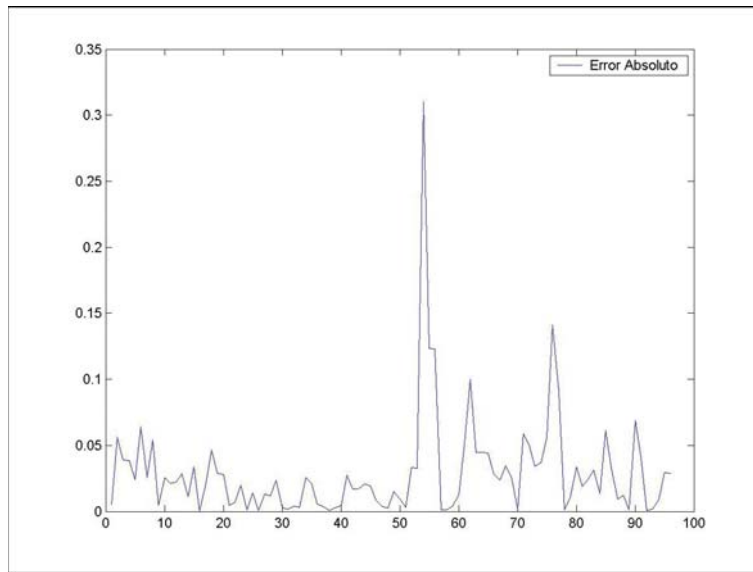


Figura 5.3: Gráfico fase de validación. PM3.

A modo de demostrar el análisis presentado en los gráficos de la fase de entrena-



miento y de validación del modelo encontrado, se muestra el gráfico que representa la variación del error absoluto del modelo (Figura 5.4).



**Figura 5.4: Gráfico Error Absoluto en la fase de validación. PM3.**

Este gráfico presenta el error absoluto entre el dato pronosticado por el modelo y el dato observado durante la fase de validación.

Del análisis de este gráfico se comprueba que durante los periodos en que la curva de datos observados presenta valores relativamente cercanos al promedio del total de los datos, las salidas del modelo presentan un comportamiento acorde con los datos esperados. En los periodos en que la curva de datos observados alcanza valores altos en comparación con el promedio de los datos observados, la curva que aparece en el gráfico del error absoluto presenta picos relativamente altos comparados con el resto de la curva. Estos picos altos de la curva del Error Absoluto, coinciden con los picos altos de la curva de datos observados, de este modo se comprueba el análisis realizado con los gráficos de la fase de entrenamiento y validación del modelo.

La Figura 5.5 muestra el gráfico de dispersión de la fase de validación del modelo, además se presenta dentro del gráfico la ecuación de la recta de regresión junto con el  $R$  de la muestra.

La Tabla 5.3 presenta una comparación entre el modelo NF1 (del inglés Naive Forecasting 1) y el modelo propuesto utilizando los datos filtrados con PM3.

## CAPÍTULO 5. DISCUSIÓN DE RESULTADOS

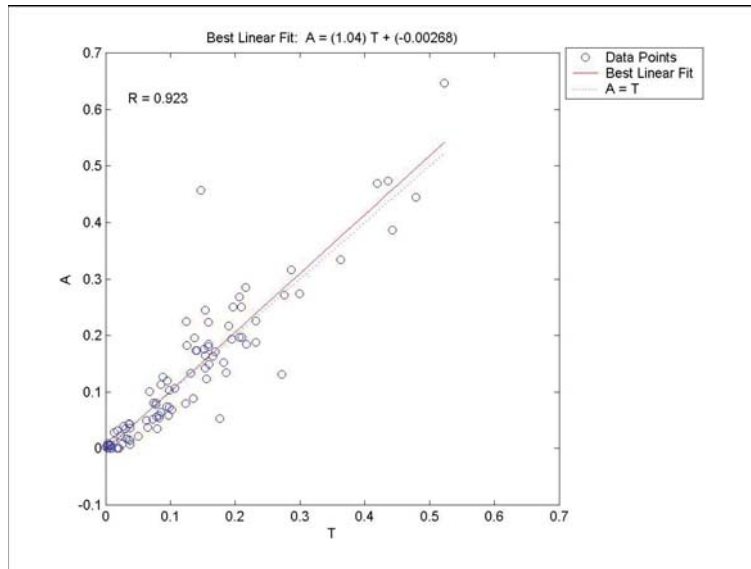


Figura 5.5: Gráfico dispersión en fase de validación. PM3.

El modelo NF1 funciona tomando como dato pronosticado para un mes, el dato inmediatamente anterior al mes que se desea pronosticar. En otras palabras, lo que sucedió en el mes  $t$  será lo que el modelo pronosticará que suceda en el mes  $t + 1$ .

Tabla 5.3: Comparación de indicadores. Modelo AR-Neuro-Wavelet Evolutiva con PM3 y Modelo NF1

Indicador	NWE-RA	NF1	Ganancia (%)
$R$	0.92	0.50	84
$R^2$	0.85	0.25	240
MSE	0.0024	0.0221	-88.93
RMSE	0.0495	0.1487	-66.74
MAE	0.0295	0.0738	-60.03
PI	0.89	0	-
SEP	38.98	36.05	8.13
ARV	0.15	0.86	-82.56
$E_2$	0.85	0.14	507.14
AIC	-0.5281	-1.6347	-67.69
BIC	-0.5466	-1.6348	-66.56

Los resultados obtenidos por el modelo propuesto logra mejores resultados en todos los indicadores respecto al modelo NF1. En el indicador  $R$ , que es el coeficiente de Pearson, el modelo propuesto obtiene 0.92, que es un 84% mejor que el indicador

## CAPÍTULO 5. DISCUSIÓN DE RESULTADOS

obtenido con el modelo NF1. En el indicador  $R^2$  el modelo propuesto obtiene un 0.85, lo que significa un 240 % mejor, que el 0.25 obtenido por el modelo NF1. En el indicador  $MSE$  el modelo propuesto también obtiene un resultado superior, en un 88.93 %, así sucede con el  $RMSE$ , con un 66.74 %, el índice  $MAE$ , con un 60.03 %. Solo en los indicadores  $AIC$  y  $BIC$  el modelo se ve superado por el modelo NF1 debido exclusivamente a la gran cantidad de parámetros (100) que presenta el modelo con un 67.69 % en el  $AIC$  y un 66.56 % en el  $BIC$ .

De los valores de los indicadores del modelo propuesto destaca un  $PI$  de 0.89, el que representa un acoplamiento de los datos pronosticados por el modelo de un 89 %.

Como ya se mencionó en puntos anteriores, además de utilizar datos filtrados con el promedio móvil 3, se utilizaron datos filtrados a través de la técnica de convolución.

Al igual que con PM3 se realizaron conjuntos de 30 pruebas con 100, 200 y 300 individuos integrantes de la población. Se presenta a continuación una tabla con el resumen de los resultados obtenidos al aplicar este tipo de filtro sobre los datos. En esta tabla se expone la media y varianza del coeficiente de determinación, además del mejor resultado obtenido tanto en la fase de entrenamiento como en la fase de validación.

La Tabla 5.4, presenta los datos obtenidos en las diferentes pruebas.

**Tabla 5.4: Resumen de resultados con datos filtrados con convolución**

Tam. Pob.	Entrenamiento			Validación		
	MEDIA $R^2$	MAX $R^2$	VAR $R^2$	MEDIA $R^2$	MAX $R^2$	VAR $R^2$
100	0.78	0.86	0.00375	0.74	0.84	0.00854
200	0.85	0.87	0.00012	0.82	0.85	0.00047
300	0.86	0.87	0.00004	0.83	0.85	0.00014

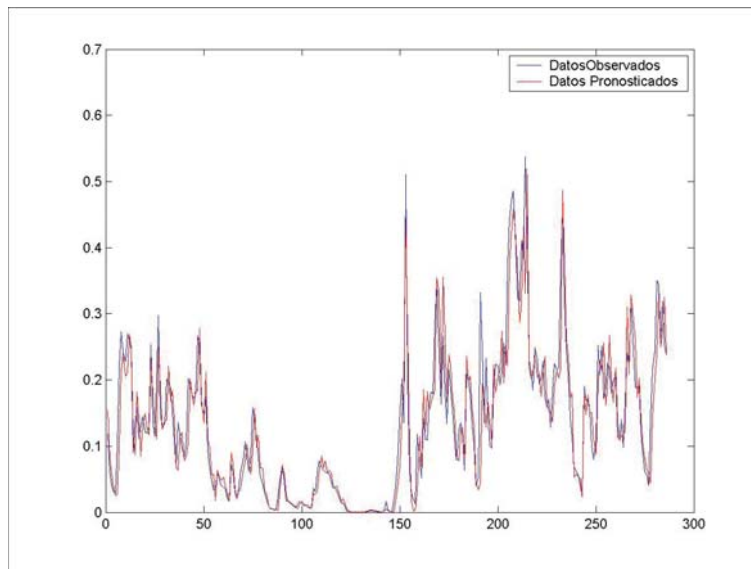
De los datos presentados en la tabla destaca el máximo alcanzado en la fase de entrenamiento del modelo por el indicador  $R^2$  con un 0.87. Otro punto a destacar, es el hecho de que los modelos encontrados con los distintos grupos de datos, filtrados con las dos técnicas utilizadas en este proyecto, tienen el mismo valor como máximo en el indicador  $R^2$  durante la fase de validación del modelo, con un 0.85.

## CAPÍTULO 5. DISCUSIÓN DE RESULTADOS

De la tabla también se concluye que la variación de los resultados en los indicadores de rendimiento se vuelven más estables al aumentar la cantidad de individuos en la población. Además esta estabilidad en los indicadores se logra más rápido en los modelos que utilizan datos filtrados mediante convolución que con PM3.

De los datos obtenidos, se seleccionó para el análisis posterior al individuo cuyo  $R^2$  alcanzó el valor de 0.86 en su fase de entrenamiento y de 0.85 en su fase de validación. Este resultado es superior al obtenido con el modelo Neuro-Wavelet evolutivo del autor, con el mismo conjunto de datos, ya que este modelo solo obtenía un valor de 0.81 en el indicador  $R^2$  en la fase de validación.

El gráfico de la Figura 5.6 presenta los resultados obtenidos por el mejor individuo. Se observa en el gráfico un desfase de la curva de pronóstico respecto a la curva con los datos observados. Además se observa que el modelo no es capaz de pronosticar los picos altos de la curva de datos observados, lo que se ve más claramente entre el dato 141 y 150 del gráfico.

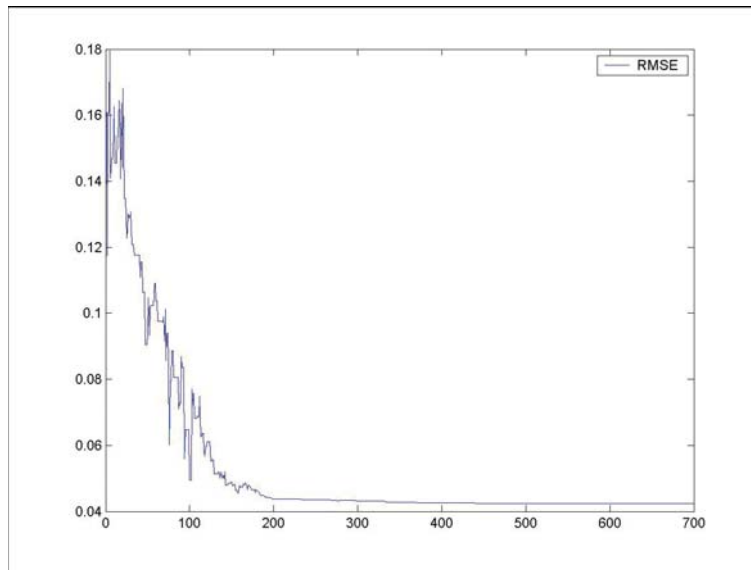


**Figura 5.6: Gráfico fase de entrenamiento. Convolución**

Al igual que al realizar el análisis a los resultados obtenidos con filtro PM3, a continuación se presenta el gráfico con la evolución del indicador  $RMSE$  a lo largo de las generaciones del Algoritmo genético, Figura 5.7. En este caso, el gráfico presen-

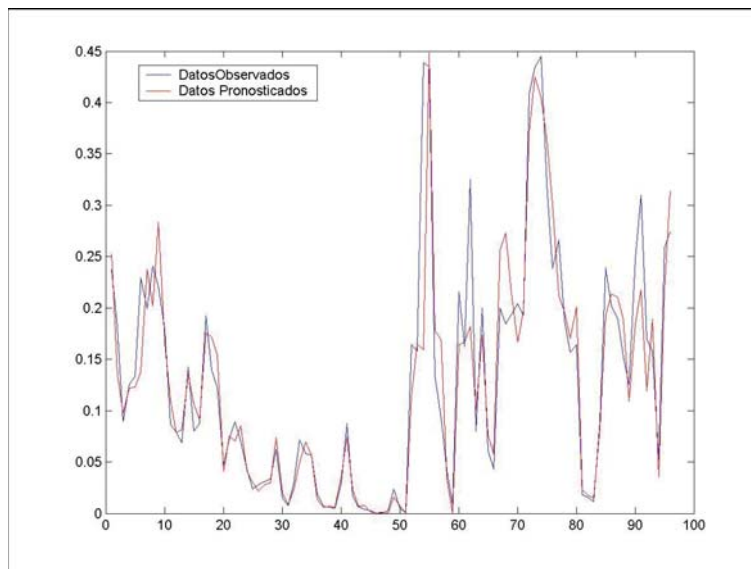
## CAPÍTULO 5. DISCUSIÓN DE RESULTADOS

tado muestra que el Algoritmo genético logra la convergencia cerca de la generación número 400 y la generación número 500.



**Figura 5.7: Gráfico RMSE. Convolución.**

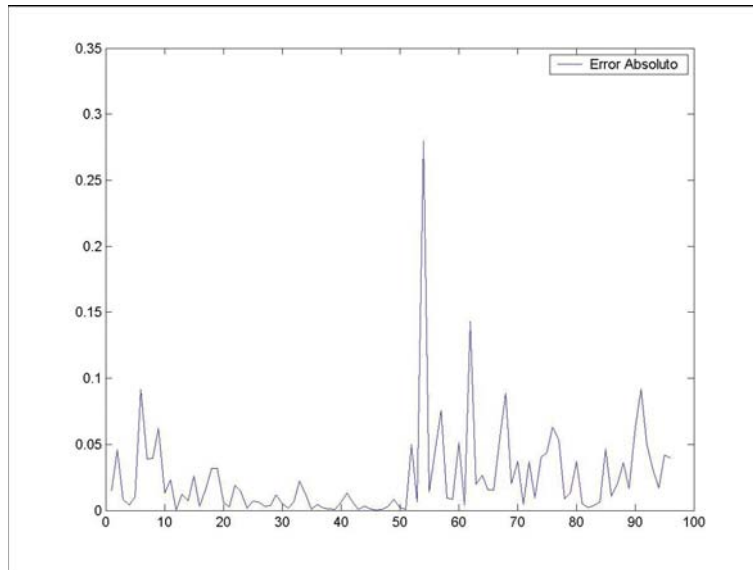
El gráfico de la Figura 5.8 presenta los datos obtenidos por el mejor individuo durante la fase de validación del modelo. Al igual que durante la fase de entrenamiento, el modelo no es capaz de llegar a los picos altos de la curva de datos observados, así como también existe un desfase entre el dato pronosticado y el dato observado.



**Figura 5.8: Gráfico fase de validación. Convolución**

## CAPÍTULO 5. DISCUSIÓN DE RESULTADOS

Para mostrar el comportamiento de las curvas más claramente la Figura 5.9 muestra el error absoluto entre el dato pronosticado y el dato observado. Este gráfico muestra el desempeño del modelo visto desde el punto de vista del error absoluto, tomando en cuenta este indicador, se nota claramente como el mayor error obtenido por el modelo coincide con los picos que toma la curva de los datos observados.



**Figura 5.9: Gráfico Error Absoluto fase de Validación. Convolución.**

La Figura 5.10 muestra el gráfico de dispersión de la fase de validación del modelo encontrado con filtro mediante convolución, junto con esto se presenta la ecuación de la recta de regresión junto con el  $R^2$  de la muestra. Se aprecia además que la recta logra aproximar la mayoría de los datos de la muestra, sin embargo, presenta datos relativamente distantes de la recta.

En la Tabla 5.5 se comparan los indicadores del modelo obtenido con los indicadores del Modelo NF1.

Al igual que en el caso del modelo obtenido con los datos filtrados con PM3, en el caso en que se aplica convolución, se obtienen mejores resultados que con el modelo NF1 en todos los indicadores, salvo en el  $AIC$  y el  $BIC$  en los que la cantidad de parámetros a determinar por el modelo penaliza su valor. Se resalta el resultado obtenido en el indicador  $R$  y  $R^2$ , donde la ganancia es de un 84 % y 240 % respectivamente.

Resultados satisfactorios también se obtienen con  $MSE$ , con una ganancia de 91.18 %,

CAPÍTULO 5. DISCUSIÓN DE RESULTADOS

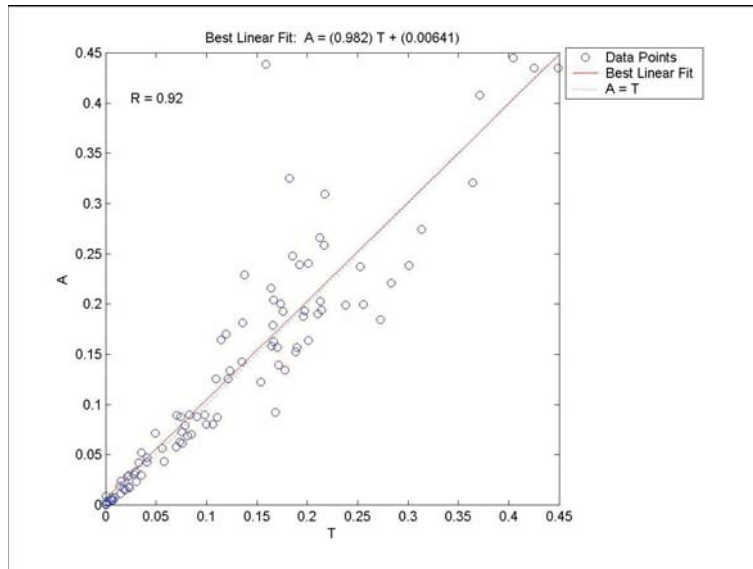


Figura 5.10: Gráfico dispersión en fase validación. Convolución.

Tabla 5.5: Comparación de indicadores. Modelo AR-Neuro-Wavelet Evolutiva con Convolución y Modelo NF1

Indicador	NWE-RA	NF1	Ganancia (%)
$R$	0.92	0.50	84
$R^2$	0.85	0.25	240
MSE	0.00195	0.0221	-91.18
RMSE	0.04418	0.1487	-70.29
MAE	0.0252	0.0738	-65.89
PI	0.91	0	-
SEP	34.03	36.05	-5.60
ARV	0.15	0.86	-82.55
$E_2$	0.85	0.14	507.14
AIC	-0.6261	-1.6347	-60.70
BIC	-0.6446	-1.6348	-60.57

$RMSE$  con un 70.29% de ganancia,  $MAE$ ,  $ARV$  y  $E_2$  con un 65.89%, 82.55% y 507.14% respectivamente de ganancia. Dentro de los datos destacado que muestra la tabla está el  $PI$  el cual alcanza un valor de 0.91 lo cual significa un mayor acoplamiento de la curva de datos observados y la curva de datos pronosticados.

La Tabla 5.6 presenta los resultados obtenidos por ambos modelos encontrados. Se destaca que los resultados obtenidos mediante los datos filtrados con convolución son similares que los resultados obtenidos mediante  $PM3$ .

**Tabla 5.6: Comparación de indicadores. Modelo AR Neuro-Wavelet Evolutiva con PM3 y Modelo AR Neuro-Wavelet Evolutiva con Convolución**

Indicador	NWE-AR-PM3	NWE-AR-Convolución
$R$	0.92	0.92
$R^2$	0.85	0.85
MSE	0.00245	0.00195
RMSE	0.04946	0.044183
MAE	0.0295	0.02517
SEP	38.98	34.03
PI	0.89	0.91
ARV	0.15	0.15
$E_2$	0.85	0.85
AIC	-0.52814	-0.6261
BIC	-0.54661	-0.6446

Del análisis de la tabla se concluye que ambos modelos encontrados con las distintas técnicas de filtrado, logran iguales resultados en los indicadores  $R$ ,  $R^2$ ,  $ARV$  y  $E_2$ . En el resto de los indicadores existen leves diferencias que benefician al modelo obtenido con el conjunto de datos filtrados con convolución, aun cuando resalta el indicador  $PI$  donde el modelo con convolución logra un 0.91 contra un 0.89 del modelo obtenido con datos filtrados con PM3. De los resultados obtenidos con ambas técnicas de filtrado se puede concluir que ambos no alcanzan los picos altos de la curva, además, presentan un desfase en cuanto al dato observado y el pronosticado.

### 5.3.1 Resultados Finales

En esta sección se presentarán algunos indicadores de desempeño de otros modelos de pronóstico, de modo de compararlos con el modelo propuesto.

La Tabla 5.7 presenta los indicadores obtenidos por otros modelos aplicados al pronóstico de captura de anchovetas vistos en [23], documento del año 2007, junto con los datos de los mejores modelos encontrados en este trabajo durante la etapa de validación, esto es el mejor modelo utilizando PM3 y el mejor modelo utilizando filtro mediante convolución. También se incluye en la tabla los indicadores del modelo NF1.



CAPÍTULO 5. DISCUSIÓN DE RESULTADOS

Al analizar la tabla se nota que los modelos encontrados a lo largo de este trabajo

**Tabla 5.7: Comparación de indicadores. Comparación con otros modelos predictivos**

Indicador	NF1	6-15s-15s-11	7-10s-10s-11	AR-NWE(PM3)	AR-NWE (Conv)
$R$	0.50	0.91	0.89	0.92	0.92
$R^2$	0.25	0.82	0.79	0.85	0.85
RMSE	0.1487	0.030	0.031	0.0494	0.04418
MAE	0.0738	0.019	0.021	0.0295	0.0258
PI	0	0.7732	0.7629	0.89	0.91
SEP	36.05	52.1075	51.1724	38.98	34.03
ARV	0.86	0.1887	0.2276	0.15	0.14
$E_2$	0.14	0.8117	0.7724	0.85	0.85
AIC	-1.6247	15.7005	12.6097	-0.5281	-0.6261
BIC	-1.6348	15.6709	12.6018	-0.5466	-0.6446
m	1	330	180	100	100

obtienen mejores resultados en la mayoría de los indicadores presentados, además la cantidad de parámetros que se necesitan definir en los modelos propuestos son inferiores a la cantidad de parámetros de los otros modelos de la tabla (100 contra 330 y 180 de los otros modelos).

Considerando el mejor modelo entre el modelo AR-NWE-PM3 y AR-NWE-Convención, se confeccionó la Tabla 5.8 la cual contiene expresado el porcentaje de mejora del modelo propuesto en comparación con el mejor indicador entre los modelos presentes en [23].

El modelo propuesto obtiene mejores resultados en los indicadores  $R$  con un 0.92 que representa un 1.1 % de mejora con respecto al mejor indicador entre los modelos presentados en la tabla.

También se obtienen mejoras: en  $R^2$  con un 3.65 % de mejora, en el índice de persistencia  $PI$  con un 18.18 % de ganancia, además se mejora los indicadores  $ARV$  y  $E_2$  con una mejora de un 20.51 % y 4.72 % respectivamente. Como ya se ha señalado con anterioridad, los valores en los indicadores  $AIC$  y  $BIC$ , reciben un efecto negativo a medida que la cantidad de parámetros a definir aumenta, lo que se ve reflejado en

**Tabla 5.8: Comparación de indicadores. Comparación con otros modelos predictivos considerando la Ganancia Obtenida**

Indicador	6-15s-15s-11	7-10s-10s-11	AR-NWE ( <i>Conv</i> )	Ganancia (%)
$R$	0.91	0.89	0.92	1.1
$R^2$	0.82	0.79	0.85	3.65
RMSE	0.030	0.031	0.04418	47.57
MAE	0.019	0.021	0.0251	32.11
PI	0.7732	0.7629	0.91	18.18
SEP	52.1075	51.1724	34.03	-33.5
ARV	0.1887	0.2276	0.15	-20.51
$E_2$	0.8117	0.7724	0.85	4.72
AIC	15.7005	12.6097	-0.6261	-104.97
BIC	15.6709	12.6018	-0.6446	-105.12
m	330	180	100	-86.67

los valores de los indicadores presentes en la tabla, en la cual los indicadores de los modelos con el que se compara el modelo propuesto presentan un  $BIC$  y  $AIC$  elevados en comparación a los mismos indicadores del modelo propuesto, es así como el modelo propuesto obtiene una mejora considerable de estos indicadores con un -0.6446 y -0.6261 en el indicador en el  $BIC$  y  $AIC$  comparado con un 12.6018 y -12.6097 de los otros modelos, respectivamente.

En los indicadores que el modelo propuesto no mejora los resultados planteados por los otros modelos con los cuales se comparó, se encuentran el  $RMSE$  el cual presenta un 0.04418 en el modelo propuesto y un 0.031 en el mejor resultado de los otros modelos. Caso similar sucede con el  $MAE$  donde el indicador del modelo propuesto logra un 0.0251 contra un 0.019 del mejor resultado de los modelos comparativos.

## Conclusiones

---

A lo largo de este documento se presentó la investigación, desarrollo e implementación de un modelo de pronóstico para la captura de anchovetas, este modelo está conformado por una Regresión funcional en la que cada coeficiente está conformado por una Red Neuronal de tres capas, una de entrada, una oculta y una de salida, con función de activación Wavelet en la capa oculta.

El algoritmo de aprendizaje aplicado a la red consistió en Algoritmos Genéticos, los cuales eran utilizados para modificar los parámetros y encontrar la topología de la red. Los modelos resultados de la aplicación de este tipo de algoritmos, presentan una alta variabilidad en los indicadores que miden su desempeño. Además cuentan con un gran gasto computacional para su cálculo.

Para la aplicación de algoritmo genético se realizaron 30 pruebas con 100, 200, y 300 individuos en la población, cada una de estas poblaciones utilizaba 700 generaciones para la obtención del modelo, se destaca que a medida que la cantidad de individuos en la población aumenta, la variabilidad de los resultados en los indicadores de desempeño de los modelos disminuye.

Los datos utilizados en las fase de entrenamiento y validación de los modelos fue realizada con datos filtrados con dos técnicas, Promedio Móvil Tres y mediante Convolución. Los resultados obtenidos por los indicadores de desempeño al aplicar los datos filtrados mediante Promedio Móvil Tres son inferiores a los resultados obtenidos mediante el filtrado por convolución, salvo en el indicador  $R$ ,  $R^2$ ,  $ARV$  y  $E_2$  donde los valores fueron idénticos.

De esto último se concluye que si el aprendizaje aplicado a una Regresión funcional Neuro-Wavelet de tres capas para el pronóstico de captura de anchovetas, consiste en Algoritmos Genéticos se obtienen mejores resultados si el conjunto de datos utilizado

## CAPÍTULO 5. CONCLUSIONES

es filtrado mediante Convolución.

De los resultados obtenidos mediante el mejor modelo de pronóstico, se puede decir que su resultado supera el obtenido mediante un modelo Neuro-Wavelet evolutivo, destacando el hecho de que se necesita un tamaño menor en la población de algoritmos genéticos para lograr mejores resultados.

Como mejor modelo de pronóstico se consideró el obtenido con el conjunto de datos filtrados mediante Convolución, debido a que obtiene mejores resultados en la mayoría de los indicadores de desempeño, e iguala los restantes, como el  $R^2$  y  $R$ .

Al comparar los indicadores de este modelo contra otros modelos de pronósticos aplicados a la captura de anchovetas [23], obtiene mejores indicadores que los otros modelos, con la excepción de los indicadores  $RMSE$  y  $MAE$ , los cuales corresponden a la raíz del error cuadrático medio y al error absoluto, respectivamente.

Se destaca que los indicadores de desempeño del modelo obtenido son superiores en el caso del  $R^2$  con un resultado un 3.65 % superior al mejor de entre los otros modelos, además de una mejora de un 1.1 % en el indicador  $R$ , un 4.72 % de mejora en el coeficiente de eficiencia  $E_2$  entre otros indicadores. Junto a esto se destaca el hecho de que el modelo encontrado solo utiliza 100 parámetros contra los 180 parámetros del mejor de los otros modelos. Este datos se ve reflejado en que el modelo obtiene mejores indicadores en el  $AIC$  y  $BIC$  con un  $-0,6446$  y  $-0,6261$  contra un  $12,6097$  y  $12,6018$ , respectivamente, del mejor de los otros modelos.

En los casos de los beneficios obtenidos por el autor al realizar este trabajo, se destaca la nueva visión incorporada al ámbito profesional mediante la obtención de herramientas que mejoren la capacidad de análisis que pueden ser aplicadas a otras áreas, como por ejemplo, Inteligencia de Negocios, Minería, Banca, entre otras.

Dentro del trabajo futuro se considera la elaboración de un modelo de pronóstico aplicado a la captura de anchovetas utilizando **Regresiones Funcionales Neuro-Wavelet Híbridas**, de modo de mejorar y comparar los resultados obtenidos a lo largo de este trabajo.

Tras el trabajo realizado se puede concluir que los objetivos específicos planteados en

## CAPÍTULO 5. CONCLUSIONES

el comienzo de este documento se pueden dar por cumplidos. Estos objetivos fueron los siguientes:

- Comprender el funcionamiento de las Regresiones Funcionales.
- Comprender el funcionamiento de las arquitecturas de Redes Neuronales Wavelet aplicadas a Regresiones Funcionales.
- Comprender el funcionamiento de los Algoritmos genéticos aplicados a Redes Neuronales y Regresiones Funcionales.
- Evaluar y contrastar el rendimiento del predictor propuesto con predictores existentes.

Una vez dado por cumplidos los objetivos específicos se concluye que el objetivo general planteado también al comienzo del documento, se puede dar por cumplido. Este objetivo general consistía en lo siguiente:

*Modelar e Implementar un modelo de predicción de captura Anchovetas de la zona norte chilena utilizando Regresiones Funcionales Neuro-Wavelet Evolutivas.*

## Referencias bibliográficas

---

- [1] Cañon, J. El Niño 1997-1998: Sus efectos en el sector pesquero industrial de la zona norte de Chile. El Niño-La Niña 1997-2000. Sus Efectos en Chile. Comité Oceanográfico Nacional, Chile (2004), pp 137-251.
- [2] Plaza, F. & Rodriguez, N. Non linear forecaste of anchovy carches in northern Chile: A multivariate approach. Globec International Newsletter, 75 (2008).
- [3] SERNAPESCA, 1978-2004. Anuarios Estadísticos de Pesca. Servicio Nacional de Pesca. Ministerio de Economía, Fomento y Reconstrucción, Chile.
- [4] SAG, 1963-1977. Anuarios Estadísticos de Pesca. Servicio Agrícola y Ganadero, Ministerio de Agricultura, Chile.
- [5] Ruiz de Giorgio, J. Resumen del análisis de los recursos pesqueros de importancia económica sobre la base de la información publicada por la Subsecretaría de pesca.
- [6] Grau, M. Cálculo Numérico, Edicions UPC, España (2001)
- [7] Carmona, F. Modelos Lineales, Barcelona, España (2003).
- [8] Bruls, J. Linear and non-linear system identification using separable least-squares (1997).
- [9] Matich, D.J. Redes Neuronales: Conceptos Básicos y Aplicaciones, Rosario (2001).
- [10] McCulloch, W. A logical calculus of the ideas immanent in nervous activity, Vol. 7, (1943).

## REFERENCIAS BIBLIOGRÁFICAS

- [11] Hebb, D. Organization of Behaviour, Nueva York: John Wiley & Sons (1949)
- [12] Sejnowski, T. Higher-order boltzmann machines, Nueva York: AOP Conference Proceedings, Neural Networks for computing (1986).
- [13] Torra, S. Siniestralidad en seguros de consumo anual de las entidades de previsión social, La Perspectiva probabilística y econométrica. Propuesta de un modelo econométrico neuronal, Universitat de Barcelona, Barcelona, España (2004).
- [14] Hornik, K. Multilayer Feedforward Networks are Universal Approximators (1992).
- [15] Kröse, B. & Van der Smagt, P. An introduction to Neural Networks, Eighth edition (1996).
- [16] Introducción a la Teoría de Wavelets, Universitat de Barselona, Barcelona, España (2005).
- [17] Papoulis, A. Sistemas digitales y analógicos, transformadas de Fourier, estimación espectral, Marcombo, Boixareu Editores (1986).
- [18] Hernandez, M. Algoritmos de comprensión de imagenes sin movimiento para comunicaciones móviles utilizando teoria de wavelets, Universidad de las Américas Puebla (2004).
- [19] Yuehui, C et al. A local Linear Wavelet Neural Network, Jinan University, China (2004).
- [20] Holland, J. Adaptation in Natural and Artificial System, University of Michigan Press (1975).
- [21] Mitchell, M. An Introduction to Genetic Algorithms, London: A bradford Book The MIT Press (1999).

## REFERENCIAS BIBLIOGRÁFICAS

- [22] Garcia, H & Ramírez, J. Método basado en Redes Neuronales Wavelet para eliminar ruido en espectros estelares, Segundo Congreso Español de Informática (2004).
- [23] Gutierrez-Estrada et al. Monthly catch forecasting of anchovy *Engraulis ringens* in the north area of Chile: Non-linear univariate approach, *Fisheries Research* 86 (2007), pp 188-200.