

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

**PREDICCIÓN DE ACCIDENTES DE TRÁNSITO  
UTILIZANDO REDES NEURONALES ARTIFICIALES Y  
CUCKOO SEARCH**

**MAURICIO ROBERTO ROJAS GODOY**

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

ABRIL, 2015

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

**PREDICCIÓN DE ACCIDENTES DE TRÁNSITO  
UTILIZANDO REDES NEURONALES ARTIFICIALES Y  
CUCKOO SEARCH**

**MAURICIO ROBERTO ROJAS GODOY**

Profesor Guía: **Nibaldo Rodríguez Agurto**

Profesor Co-referente: **Ignacio Araya Zamorano**

Carrera: **Ingeniería Civil Informática**

ABRIL, 2015

*Dedicado a cada uno de los profesores que formaron parte de mi vida desde los inicios de mi educación, entregando conocimiento, valores y las herramientas necesarias para formarme como persona y luego como profesional. Los cuales merecen todo mi respeto y reconocimiento social ante su gran labor.*

*A mi familia los que me han apoyado en todo momento, siendo un pilar fundamental en mi formación profesional y me han enseñado el más hermoso valor, "la fraternidad hacia el que no conozco".*

*Espero con el tiempo retribuir de alguna forma a mi familia y a la sociedad en general por lo que me ha sido entregado, para que otros jóvenes al igual que yo puedan cumplir sus sueños.*

## Abstract

Over the past 10 years, diverse investigations have been developed related to traffic accidents using computational intelligence; nevertheless, the use of Artificial Neural Networks in prediction of traffic accidents is reduced.

From a series of traffic accidents data in the region of Valparaíso, a smoothed series by the Moving Average and an unsmoothed series were obtained. For both cases was used a model of Artificial Neural Network together with metaheuristics Cuckoo Search (CS) and Particle Swarm Optimization (PSO), obtaining as result four predictive models.

The results showed a significant improvement in the case of the smoothed series compared to the unsmoothed. To the smoothed series, the optimal topology found for CS was 5 input nodes, 8 hidden nodes and 1 output node, whereas for PSO was 13 input nodes, 10 hidden nodes and 1 output node.

Among the smoothed series, the superiority of CS over PSO in the Mean Absolute Percentage Error (MAPE) metric is evident, obtaining 12.11 % and 14.96 % respectively, whereas in the case of the other metrics corresponding to the Root Mean Square Error (RMSE), Correlation Coefficient (R) and Coefficient of Determination (R<sup>2</sup>), the difference was minimal.

**Keywords:** Artificial Neural Networks, Cuckoo Search, Particle Swarm Optimization, Moving Average, Traffic Accidents.

## Resumen

Durante los últimos 10 años se han desarrollado diversas investigaciones relacionadas con accidentes de tránsito utilizando Inteligencia Computacional, sin embargo el uso de Redes Neuronales Artificiales en predicción de accidentes de tránsito es reducido.

A partir de una serie de datos de accidentes de tránsito ocurridos en la región de Valparaíso, se obtuvo una serie suavizada por medio de la Media Móvil y una serie sin suavizar. Para ambos casos se utilizó un modelo de Red Neuronal Artificial junto a las metaheurísticas Cuckoo Search (CS) y Optimización por Enjambre de Partículas (PSO), obteniendo finalmente 4 modelos predictivos.

Los resultados mostraron una mejora sustantiva en el caso de la serie suavizada en comparación a la serie sin suavizar. Para el caso de la serie suavizada, la topología óptima hallada para CS fue de 5 nodos de entrada, 8 nodos ocultos y 1 nodo de salida, mientras que para PSO fue de 13 nodos de entrada, 10 nodos ocultos y 1 nodo de salida.

Entre la serie suavizada se evidencia una superioridad de CS sobre PSO en la métrica Porcentaje de Error Absoluto Medio (MAPE), obteniendo 12,11 % y 14,96 % respectivamente, mientras que en el caso de las otras métricas medidas correspondiente a la Raíz del Error Cuadrático Medio (RMSE), Coeficiente de Correlación (R) y Coeficiente de Determinación (R<sup>2</sup>), la diferencia fue mínima.

**Palabras Clave:** Redes Neuronales Artificiales, Cuckoo Search, Optimización por Enjambre de Partículas, Media Móvil, Accidentes de Tránsito.

# Índice

<b>Lista de Abreviaturas o Siglas</b>	<b>VII</b>
<b>1 Introducción</b>	<b>1</b>
<b>2 Objetivos</b>	<b>3</b>
2.1 Objetivo General . . . . .	3
2.2 Objetivos Específicos . . . . .	3
2.3 Planteamiento del Problema . . . . .	3
<b>3 Estado del Arte de Accidentes de Tránsito</b>	<b>4</b>
<b>4 Redes Neuronales Artificiales</b>	<b>6</b>
4.1 Reseña Histórica . . . . .	6
4.2 La Neurona Biológica . . . . .	7
4.3 La Neurona Artificial . . . . .	7
4.4 Operación de la Red . . . . .	9
4.5 Modelos de Redes Neuronales Artificiales . . . . .	9
4.5.1 Perceptrón . . . . .	10
4.5.2 Perceptrón Multicapa . . . . .	11
<b>5 Algoritmos de Optimización</b>	<b>13</b>
5.1 Optimización por Enjambre de Partículas . . . . .	13
5.1.1 Algoritmo PSO . . . . .	13
5.2 Cuckoo Search . . . . .	14
5.2.1 Comportamiento Reproductivo de Aves Cuckoo . . . . .	14
5.2.2 Lévy Flights . . . . .	14
5.2.3 Algoritmo Cuckoo Search . . . . .	15
<b>6 Suavizado de Datos</b>	<b>17</b>
<b>7 Métricas de Calidad</b>	<b>18</b>
7.1 Error Cuadrático Medio . . . . .	18
7.2 Raíz del Error Cuadrático Medio . . . . .	18
7.3 Porcentaje de Error Absoluto Medio . . . . .	18
7.4 Coeficiente de Determinación . . . . .	19
7.5 Coeficiente de Correlación . . . . .	19

<b>8 Datos</b>	<b>20</b>
<b>9 Resultados</b>	<b>22</b>
9.1 Red neuronal con CS . . . . .	22
9.2 Red neuronal con PSO . . . . .	23
9.3 Discusión de Resultados . . . . .	24
<b>10 Conclusión</b>	<b>26</b>

## Lista de Figuras

Figura 4.1	Red Neuronal Artificial. . . . .	8
Figura 4.2	Perceptrón Multicapa. . . . .	11
Figura 8.1	Cantidad de accidentes de tránsito (a) Serie no suavizada (b) Serie suavizada. . . . .	20
Figura 8.2	Autocorrelación de accidentes de tránsito (a) Serie no suavizada (b) Serie suavizada. . . . .	20
Figura 8.3	Estrategia de resolución con suavizado y sin suavizado. . . . .	21
Figura 9.1	Obtención de parámetros de CS. . . . .	23
Figura 9.2	Valor observado vs Valor estimado utilizando CS . . . . .	24
Figura 9.3	Valor observado vs Valor estimado utilizando PSO . . . . .	25

## **Lista de Abreviaturas o Siglas**

BPNN: Back Propagation Neural Network.

CS: Cuckoo Search.

IC: Inteligencia Computacional.

MA: Movil Average.

MAPE: Porcentaje de Error Absoluto Medio.

MLP: Multi Layer Perceptron.

MSE: Error Cuadrático Medio.

PSO: Particle Swarm Optimization.

PUCV: Pontificia Universidad Católica de Valparaíso.

R: Coeficiente de Correlación.

R2: Coeficiente de Determinación.

RMSE: Raíz del Error Cuadrático Medio.

RNA: Red Neuronal Artificial.

# 1 Introducción

Las lesiones causadas por el tránsito son la octava causa de muerte a nivel mundial, y la primera entre los jóvenes de 15 a 29 años. Las tendencias actuales indican que si no se toman medidas urgentes, para el año 2030 los accidentes de tránsito se convertirán en la quinta causa de muerte en el Mundo [1].

El año 2010, gobiernos de todo el mundo proclamaron el Decenio de Acción para la Seguridad Vial (2011-2020), cuyo objetivo consiste en estabilizar, y después reducir, la tendencia al aumento de las muertes por accidentes de tránsito, con lo que se calcula se salvarían 5 millones de vidas en esos 10 años.

De los ciento noventa y cuatro países del mundo, ochenta y ocho en los que viven cerca de 1.600 millones de personas, han logrado reducir el número de muertos en sus carreteras entre los años 2007 y 2011 gracias a las medidas adoptadas en el Decenio de Acción para la Seguridad Vial, lo que demuestra que se conseguirán salvar muchas más vidas si los países adoptan nuevas medidas. Sin embargo, en el mismo periodo hubo en 87 países en los cuales no se adoptaron estas medidas, en los que se evidenció un aumento en el número de muertes por accidentes de tránsito [1].

De acuerdo a cifras del año 2013, los países de ingresos medios son los que tienen las mayores tasas de mortalidad por accidentes de tránsito, alcanzando un 20,1 por 100.000 habitantes, en comparación de 8,7 en los de ingresos elevados y 18,3 en los de ingresos bajos [1].

La situación en Chile durante el año 2012, según las cifras de la Comisión Nacional de Seguridad de Tránsito, fue de 8,75 por cada 100.000 habitantes, presentando una tasa de 22,8 vehículos cada 100 habitantes. Durante ese año se produjeron 61.791 siniestros dejando 53.225 lesionados, y entre ellos 1.523 fallecidos [2].

Las tres regiones de Chile que presentaron la mayor cantidad de siniestros ocasionados por accidentes de tránsito durante el año 2012, corresponden a la Región Metropolitana (23.240 siniestros y 15.433 fallecidos), la Región de Valparaíso (7.666 siniestros y 6.577 fallecidos) y por último la Región del Bío-Bío (7.337 siniestros y 9.121 fallecidos) [3].

Durante los últimos 10 años se han desarrollado diversas investigaciones relacionadas a accidentes de tránsito en distintas ciudades del Mundo utilizando técnicas de Inteligencia Computacional (IC), permitiendo visualizar variables y patrones significativos que ayudan a identificar de mejor manera las condiciones en las que ocurren los accidentes de tránsito, considerando que cada siniestro implica la conjugación de un gran número de variables, que determinan la cantidad de accidentes [4, 5, 6, 7, 8, 9, 10, 11].

Una de estas técnicas de IC son las RNA, que consisten en un algoritmo que emula el comportamiento del sistema nervioso animal. Este tipo de técnicas ha demostrado tener buenos resultados tanto en la predicción como la clasificación de series de datos.

El uso de RNA en predicción de accidentes de tránsito es reducido, es por esto que este

proyecto busca contribuir con un modelo predictivo que sea un aporte al estudio de la seguridad vial, y así poder controlar y reducir la tasa de accidentes de tránsito.

En esta investigación se desarrollará y evaluará cuatro modelos de predicción utilizando RNA de tipo Multilayer Perceptron (MLP), junto con el algoritmo de aprendizaje Backpropagation (BP), y utilizando los algoritmos de optimización Cuckoo Search (CS) [12, 13, 14, 15] y Optimización por Enjambre de Partículas (PSO) [16, 17], tanto para la serie de datos suavizada por medio de la técnica de suavizado Media Móvil (MA), como con la serie de datos sin suavizar. Este modelo permitirá predecir tanto la cantidad de siniestros, mas no de fallecidos, ocurridos en la Región de Valparaíso entre los años 2003 y 2012.

En los siguientes capítulos, se presentará el Marco General del Proyecto, presentando el objetivo general y los objetivos específicos, el problema a abordar y el alcance de este. Posteriormente se describirá el estado del arte de las técnicas de IC, aplicadas a los accidentes de tránsito en distintas ciudades del Mundo. Luego se presentará el marco teórico, describiendo la arquitectura de las RNA y el algoritmo de optimización CS y PSO, la técnica de suavizado MA, las métricas a utilizar, la estrategia de resolución del problema, los datos a utilizar, seguido de los resultados obtenidos luego de aplicar esta técnica. Finalmente se presentará la conclusión obtenida en este proyecto.

## 2 Objetivos

A continuación se presentarán los objetivos del proyecto y el problema a tratar.

### 2.1 Objetivo General

Desarrollar un modelo para la predicción de accidentes de tránsito ocurridos en la región de Valparaíso, utilizando una Red Neuronal Artificial con el algoritmo de optimización Cuckoo Search.

### 2.2 Objetivos Específicos

- a) Explicar la estructura de una Red Neuronal Artificial y el algoritmo Cuckoo Search.
- b) Diseñar la topología de una Red Neuronal Artificial y estimar los pesos utilizando Cuckoo Search.
- c) Evaluar y comparar el rendimiento del modelo propuesto con el algoritmo Optimización por Enjambre de Partículas.

### 2.3 Planteamiento del Problema

Las Redes Neuronales Artificiales en predicción, se utilizan desde los años 90, sin embargo, sólo desde la última década se empezaron a utilizar en problemas de predicción y clasificación de accidentes de tránsito en distintas ciudades del Mundo [8].

El departamento de investigación de la Escuela de Ingeniería en Transporte de la Pontificia Universidad Católica de Valparaíso, ha utilizado minería de datos para el análisis de accidentes de tránsito, obteniendo buenos resultados [18], sin embargo se han encontrado escasos trabajos de predicción de los accidentes en Chile, más sí de la clasificación de éstos [5, 6, 7].

En base a lo anterior, surge la idea de la aplicación de técnicas de predicción de Accidentes de Tránsito, con el fin de construir un modelo que pronostique la cantidad de accidentes ocurridos y comparar las metaheurísticas CS y PSO.

### 3 Estado del Arte de Accidentes de Tránsito

Desde hace más de una década se han estudiado y analizado accidentes de tránsito mediante minería de datos en todo el Mundo, algunos de ellos se detallan a continuación.

En 1997, Dia y Rose [9] utilizaron datos reales para comparar la técnica de red neuronal Perceptrón Multicapa (MLP), con un modelo de detección de accidentes en la autopista de Melbourne. Los resultados mostraron que el modelo de la red neuronal podía detectar más rápidamente los accidentes.

En 1999, Yang [10] utilizó redes neuronales para detectar patrones seguros de conducción que tuviesen menos probabilidad de causar la muerte y lesiones cuando se produce un accidente de tráfico. Concluyeron que mediante el control de una sola variable (velocidad de conducción o condiciones de luz), podían reducir las muertes y lesiones hasta en un 40%. En el mismo año Mussone *et al.* [11], utilizaron redes neuronales para analizar accidentes ocurridos en intersecciones en Milán, Italia. Eligieron las técnicas de feedforward (MLP) con Back Propagation (BP). Los resultados mostraron que el mayor índice de accidentes de atropellos en peatones era en intersecciones no señalizadas y durante la noche.

En el año 2002, Bedard *et al.* [19], aplicaron una regresión logística, encontraron que la prevención de fatalidades radicaba en el mayor uso del cinturón de seguridad, en la reducción de velocidad e impactos del lado del conductor. En el mismo año, Ossiander y Cummings [20], usaron modelos de regresión de Poisson, concluyendo que al aumentar el límite de velocidad, aumentaba el índice de accidentes mortales en las carreteras del estado de Washington.

Por otra parte, Abdelwahab y Abdel en 2003 [21], estudiaron los accidentes de tránsito en Florida. El análisis se centró en los accidentes de vehículos producidos en intersecciones con semáforos. Se comparó el desempeño de las técnicas de Perceptron Multicapa (MLP) con Fuzzy ARTMAP, concluyendo que la precisión de clasificación de MLP era superior. En el mismo año, Sohn y Lee [22], aplicaron clustering para mejorar la precisión de clasificadores individuales para dos categorías de gravedad (lesiones corporales y daños a la propiedad). Para ello utilizaron redes neuronales y árboles de decisión, concluyeron que los algoritmos de clustering trabajan mejor para clasificar accidentes de tránsito. En el mismo año 2003 Chong, Abraham y Paprzycki [23], utilizaron redes neuronales, árboles de decisión y un modelo híbrido, para construir modelos que pudiesen predecir la gravedad de las lesiones. Además incluyeron una breve aplicación con Maquinas de Soporte Vectorial (SVM). Concluyeron que para los casos en que las personas resultan ilesas o posiblemente lesionadas, tenía un mejor resultado el modelo híbrido que las redes neuronales, para el caso no lesionado y posiblemente lesionado se pudo modelar mejor haciendo uso de árboles de decisión. La técnica de SVM no obtuvo muy buenos resultados, el mayor porcentaje de exactitud obtenido fue de un 60,3%.

Por otra parte, miembros de la Escuela de Ingeniería en Transportes de la PUCV han trabajado en el análisis de accidentes de tránsito en las principales regiones de Chile mediante técnicas estadísticas y de minería de datos. En el 2006 Montt *et al.* [24], analizaron los siniestros me-

diante estadísticas descriptivas y árboles de decisión, sin embargo, no se utilizó una metodología que ameritase un trabajo con gran cantidad de datos. Más tarde, Montt, Zúñiga y Chacón en el 2009 [25], trabajaron con técnicas de minería de datos basadas principalmente en estructuras de Redes Bayesianas. Procedieron a aplicar algoritmos de aprendizaje paramétrico y propagación de evidencia. Obtuvieron que el riesgo de salir afectado en un accidente de tránsito era de un 65 %, y que el tipo de atropello era el accidente con mayor probabilidad de ser afectado con un 99,52 %. Además concluyeron que la mayor posibilidad de salir ileso era la colisión y que en zonas rurales existía mayor probabilidad de salir afectado con un 79 %, y el 21 % restante correspondía el riesgo de salir afectado en una zona urbana. Finalmente, el porcentaje de exactitud del modelo general que consistía en el estado (lesionado o ileso) en el cual resultan las personas involucradas de tránsito, fue de un 78 %.

En el mismo año Montt, Musso y Chacón [26], utilizaron métodos de agrupamiento e índices de Calinski y Harabasz para encontrar agrupaciones que representaran mejor la información. Lograron caracterizar los distintos tipos de accidentes, como choques con objeto, volcadura, caídas y los atropellos, encontrando características particulares en cada uno de ellos.

Montt *et al.* [7] utilizó técnicas de clasificación basadas en redes neuronales artificiales y los algoritmos de entrenamiento LDWPSO, QPSO y LDWQPSO, con diferentes topologías de red y tamaños de muestra para clasificar el grado de severidad de las lesiones con que resultan las personas en accidentes de tránsito en la Región de Valparaíso, Chile. E. Cárdenas [6] presentó un modelo de clasificación entre personas lesionadas o ilesas en accidentes de tránsito de la Región Metropolitana, utilizando redes neuronales artificiales en conjunto con algoritmos evolutivos (PSO). Los mejores resultados se obtuvieron con la variante Optimizador por Enjambre de Partículas con Adaptación Dinámica (DAPSO), la cual arrojó una exactitud del 82,36 %. Por último F. Castro [5] realizó una clasificación utilizando máquinas de soporte vectorial con algoritmos de optimización de enjambre de partículas, para clasificar el grado de severidad (lesionado o ileso) en el cual resultan las personas involucradas en accidentes de tránsito de la Región de Valparaíso. El análisis de resultados indica que la Optimización por Enjambre de Partículas (PSO) con factor de inercia lineal obtuvo un mejor rendimiento, alcanzando un 82,04 %.

En síntesis, las Redes Neuronales, Árboles de Decisión, Redes Bayesianas, máquinas de soporte vectorial y Clustering son técnicas predominantes dentro de los estudios realizados referentes a los accidentes de tránsito. Sin embargo, sólo clasifican los accidentes de tránsito, por lo cual en este trabajo, con técnicas de inteligencia computacional (IC) se obtendrán predicciones, para poder contribuir con un conjunto de modelos que sean un aporte al estudio de la seguridad.

## 4 Redes Neuronales Artificiales

A continuación se presentan las características de las **RNA**, un algoritmo de aprendizaje y procesamiento automático inspirado en la forma en que funciona el sistema nervioso animal.

### 4.1 Reseña Histórica

La primera implementación de un **Red Neuronal Artificial (RNA)**, fue un dispositivo hidráulico descrito por Russell en 1913, pero no fue sino hasta la década de los 40 en que las RNA tomaron fuerza y evolucionaron hasta nuestros días.

Warren McCulloch y Walter Pitts fueron los primeros en modelar matemáticamente una RNA, en el modelo se plantea la idea de que las neuronas operan mediante impulsos binarios. El factor clave del modelo es la capacidad de aprendizaje.

Marvin Minsky en 1951 obtuvo los primeros resultados prácticos de las RNA's, logrando modelar el comportamiento de una rata buscando comida en un laberinto.

Frank Rosenblatt en 1957 creó el modelo llamado Perceptrón, el cual tomó como base el modelo de McCulloch, al que le añadió aprendizaje [27]. Este modelo ajusta sus pesos dependiendo del error entre la salida deseada y la salida obtenida. Algunos de sus trabajos están en su libro "*Principles of Neurodynamics*" [28].

Bernard Widrow en 1959 [29] basándose en el modelo Perceptrón creó una RNA llamada ADELIN (Adaptative Linear Element). Tanto ADELIN como Perceptrón tienen el problema que sólo pueden resolver problemas linealmente separables.

Steinbuch a principios de los años sesenta desarrolló métodos de codificación de información en RNA. Sus redes se aplicaron en control de múltiples procesos de producción, diagnóstico de fallos en maquinarias y reconocimiento de escritura a mano distorsionada.

Marvin Minsky y Seymour Paper en 1959 [30] probaron matemáticamente que el Perceptrón no era capaz de resolver problemas relativamente fáciles, como el aprendizaje de una función no lineal, frenando el crecimiento de las investigaciones sobre RNA.

Paul Werbos en 1974 desarrolló la idea básica del algoritmo de aprendizaje "Back Propagation". Mientras que Stephen Grossberg en 1977 plantea la Teoría de Resonancia Adaptada, red que puede simular memoria a corto y largo plazo.

John Hopfield en 1985 causa el resurgimiento de las investigaciones asociadas a las RNA [31].

Desde 1986 en adelante el panorama de las RNA's ha sido alentador, cada vez son más las investigaciones realizadas y diversas ramas de investigación han visto su potencial.

## 4.2 La Neurona Biológica

El Sistema Nervioso es un sistema complejo, formado por unidades individuales denominadas neuronas, unidas entre sí por una malla de fibras nerviosas. Se estima que el número de neuronas es de  $10^{11}$  y las interconexiones entre ellas son del orden de  $10^{15}$ .

La transmisión de un lado a otro de las uniones entre neuronas, llamadas sinapsis, son de naturaleza química. La cantidad de energía transferida depende de la cantidad química aportada por el axón y recibida por las dendritas. La intensidad sináptica es la que resulta modificada cuando decimos que el cerebro aprende. Las sinapsis, combinadas con el proceso de información de la neurona, forman el mecanismo básico de la memoria.

Una neurona recibe potenciales excitatorios e inhibitorios procedentes de diversas sinapsis. La neurona combina el efecto de estos potenciales en forma de sumas y restas, desencadenando un impulso nervioso sólo si la combinación supera cierto umbral.

La intensidad de una sinapsis no está representada por una cantidad fija, se piensa que esta característica constituye en buena medida el aprendizaje.

El cerebro de un ser vivo se modela durante su desarrollo, adquiriendo una serie de cualidades por la influencia de información que recibe de su entorno. Existen diversas formas de modelar el Sistema Nervioso: establecer nuevas conexiones, eliminar algunas ya existentes, modificar las intensidades sinápticas o eliminar neuronas. La modificación de las intensidades sinápticas es la característica principal que utilizarán las **Redes Neuronales Artificiales** para adquirir el aprendizaje [32].

## 4.3 La Neurona Artificial

La neurona artificial fue diseñada para imitar las características del funcionamiento básico de una neurona biológica. Básicamente se aplica un conjunto de entradas a la neurona, cada una de las cuales puede representar la salida de otra neurona. Cada entrada se multiplica por su peso o ponderación, análogo al grado de conexión de la sinapsis [33].

La neurona artificial posee un estado interno, el que se conoce como nivel de activación. El nivel de activación está determinado por las entradas y por los pesos sinápticos que posea, si bien las entradas son externas a la neurona, los pesos pueden cambiar para así poder adaptar el nivel de activación.

Como se desea calcular el nivel de activación, se hace necesario primero calcular la entrada total de la célula  $E_i$ . Este valor se calcula mediante la fórmula 4.1, en donde se suma cada entrada  $X = [x_1, x_2, \dots, x_j, \dots, x_n]$ , que representan las señales de la sinapsis en el caso de una neurona biológica, multiplicada por un peso sináptico  $W = [w_{i1}, w_{i2}, \dots, w_{ij}, \dots, w_{in}]$ , que representa la fuerza de conexión sináptica.

$$E = (X^T \times W) = x_1 \times w_1 + \dots + x_n \times w_n = \sum_{i=1}^n w_i \times x_i \quad (4.1)$$

El aprendizaje en las **Redes Neuronales Artificiales (RNA's)** es visto como el proceso de ajuste de parámetros libres. Se parte de un conjunto de pesos sinápticos aleatorios, con el fin de buscar un conjunto de pesos que permitan a la red desarrollar correctamente una tarea en particular. El proceso de aprendizaje es un proceso iterativo, en el cual se va refinando la solución hasta alcanzar un nivel de operación que de conformidad.

Una gran cantidad de métodos de entrenamiento utilizados en las redes neuronales "feed forward"(con conexión hacia adelante) consisten en proponer una función de error que mida el rendimiento de la red en función de los pesos sinápticos. El objetivo que cumple entrenar la red es obtener un conjunto de pesos sinápticos que minimizan (o maximizan) la función.

Uno de los métodos de entrenamiento más utilizados en las redes es el **Backpropagation** (retropropagación). Es un método de aprendizaje supervisado, en el que se aplica un patrón de entrada, el cual se propaga por las distintas capas que componen la red hasta producir la salida de la misma. La salida se compara con la salida deseada y se calcula el error cometido por cada neurona de salida. Estos errores se transmiten hacia atrás, partiendo de la capa de salida, hacia todas las neuronas de las capas intermedias. Cada neurona recibe un error que es proporcional a su contribución sobre el error total de la red. Basándose en el error recibido, se ajustan los errores de los pesos sinápticos de cada neurona. Por lo general una forma de entrenar redes neuronales es mediante la raíz del error cuadrático medio, especialmente para fines predictivos y está dada por la ecuación 4.2:

$$E = \sqrt{\frac{1}{N} \times \sum_{i=1}^N (d_i(n) - y_i(n))^2} \quad (4.2)$$

Donde  $N$  es el número de patrones para entrenar la red,  $d_i$  es el valor deseado para el  $i$  –ésimo patrón de entrenamiento e  $y_i$  es el valor que entrega la red para el  $i$ –ésimo patrón de entrenamiento.

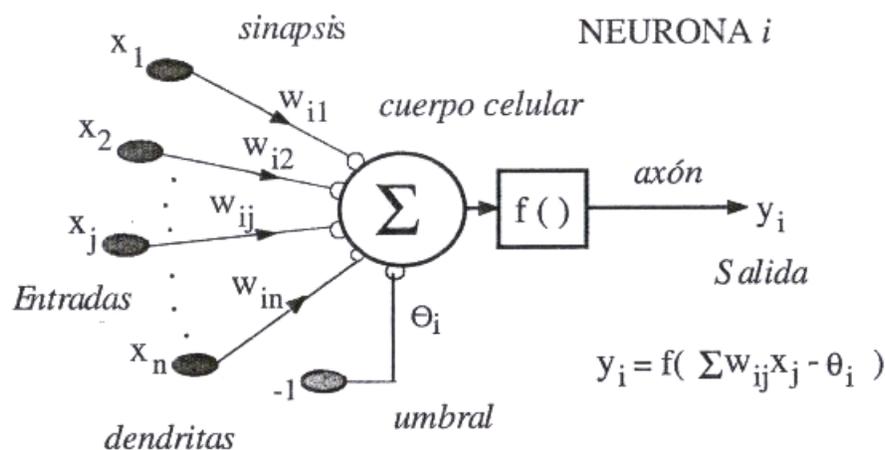


Figura 4.1: Red Neuronal Artificial.

El método de optimización proporciona una regla de actualización de los pesos que en función de los patrones de entrada modifica iterativamente los pesos hasta alcanzar el punto óptimo de la red neuronal. Una forma de obtener mejores resultados en la determinación de estos pesos, es mediante la utilización de algoritmos de optimización, como por ejemplo, Algoritmos Genéticos (GA), Optimización por Enjambre de Partículas (PSO) o Cuckoo Search (CS), los dos últimos son el que serán utilizados en éste proyecto.

La validación de los modelos propuestos se realizará mediante la técnica de Validación Cruzada. Este método consiste en dividir los datos en dos partes, una para pruebas y otra para entrenamiento de la red neuronal.

## **4.4 Operación de la Red**

Las Redes Neuronales Artificiales (RNA) son sistemas entrenables, capaces de realizar un determinado tipo de procesamiento a partir de un conjunto de ejemplos denominados patrones de entrenamiento.

En una RNA se distinguen dos fases de operación:

1. Fase de aprendizaje o entrenamiento: En esta fase se presentan a la red un conjunto de patrones de entrenamiento de forma iterativa. La red adaptará los pesos de las conexiones de forma que la respuesta a cada uno de los patrones tenga un error cada vez menor. En general el entrenamiento concluye cuando se alcanza un error determinado.

La actualización de los pesos se realiza aplicando la regla de aprendizaje seleccionada para el modelo de red considerado. En general, dicha regla determina la modificación de los pesos en función del error cometido por la red para cada entrada presentada.

Interesa más una buena generalización que un error pequeño en el entrenamiento, pues ello indicará que la red ha capturado correctamente las regularidades en los datos. Si se intenta alcanzar un error pequeño en la fase de entrenamiento, se puede provocar que la red memorice los patrones de entrenamiento, con lo que sólo dará una respuesta muy buena para estos, pero mala para cualquier otro patrón.

2. Fase de recuerdo o ejecución: Los pesos de las conexiones permanecen fijos (excepto en algunos modelos de red) y la red generará una salida para cada entrada que se le presente. En este momento la red está preparada para procesar datos.

## **4.5 Modelos de Redes Neuronales Artificiales**

A continuación se describirá un tipo de RNA llamado "Perceptrón", que posteriormente dió origen a Multi Layer Perceptron (MLP). Este último es utilizado comúnmente en problemas de predicción y/o clasificación.

### 4.5.1 Perceptrón

Este modelo fue propuesto por Rosenblatt en 1958, en su momento causó gran interés por su capacidad para reconocer patrones sencillos.

El Perceptrón es un modelo unidireccional formado por dos capas de neuronas, la capa de entrada y de salida. Se diseñó para trabajar con patrones de entrada y salida del tipo binario (0,1 o -1,1). La capa de entrada no realiza procesamiento, se limita a pasar la entrada de la red a la siguiente capa [32].

Para obtener la salida de la red, primero se calcula la activación de la célula de salida mediante la suma ponderada por los pesos de todas las entradas: [34]

$$y = \sum_{i=1}^n w_i \times x_i + \theta \quad (4.3)$$

$$F(y) = \begin{cases} 1, & \text{si } y > 0 \\ -1, & \text{en otro caso} \end{cases} \quad (4.4)$$

Este ejemplo es para dos dimensiones, para el caso de  $n$  dimensiones, se podría expresar de la siguiente forma: [34]

Dados conjuntos de puntos en  $R^n$ :  $A = (a_1, \dots, a_{na})$  y  $B = (b_1, \dots, b_{nb})$ , obtener el hiperplano de tal forma que:

$$\forall \vec{a} \in A : w_1 \times a_1 + \dots + w_n \times a_n + \theta > 0 \quad (4.5)$$

$$\forall \vec{b} \in B : w_1 \times b_1 + \dots + w_n \times b_n + \theta < 0 \quad (4.6)$$

Entonces para la red se requerirían encontrar los valores que hagan:

$$\forall \vec{a} \in A : y(\vec{a}) = 1 \quad (4.7)$$

$$\forall \vec{b} \in B : y(\vec{b}) = -1 \quad (4.8)$$

Con lo cual ya no es una recta la que separa los puntos, sino un hiperplano. Para el modelo Perceptrón se utilizan iteraciones de modo que los pesos de las conexiones se modifiquen paulatinamente.

En 1969 Minsky y Papert demostraron que el Perceptrón no puede clasificar patrones de entrada que no sean linealmente separables, un conjunto de patrones de entrada se considera linealmente separable, cuando en una representación geométrica el espacio  $n$ -dimensional es posible dividirlo por un hiperplano. Para solucionar este problema Minsky y Papert propusieron introducir una capa adicional de neuronas entre la capa de entrada y la capa de salida [32].

## 4.5.2 Perceptrón Multicapa

La RNA **Perceptrón Multicapa** o **Multi Layer Perceptron (MLP)** surge como consecuencia de las limitantes presentadas por el Perceptrón simple en lo relacionado al problema de separabilidad no lineal. MLP es una aproximación universal, dado que cualquier función continua puede ser aproximada mediante él, con al menos una capa oculta de neuronas, aún cuando no implica que sea necesariamente el mejor aproximador universal. Las redes MLP es una de las más utilizadas en la resolución de problemas, debido a su característica de ser un aproximador universal, su fácil uso y su fácil aplicabilidad.

La arquitectura de la red MLP agrupa las neuronas en  $C$  capas, normalmente cada neurona de la capa  $t$  está conectada con todas las neuronas de la capa  $t + 1$ , por esta razón se conocen como redes feed forward (conectadas hacia adelante), además estas conexiones tienen un determinado peso que se va ajustando en la etapa de entrenamiento.

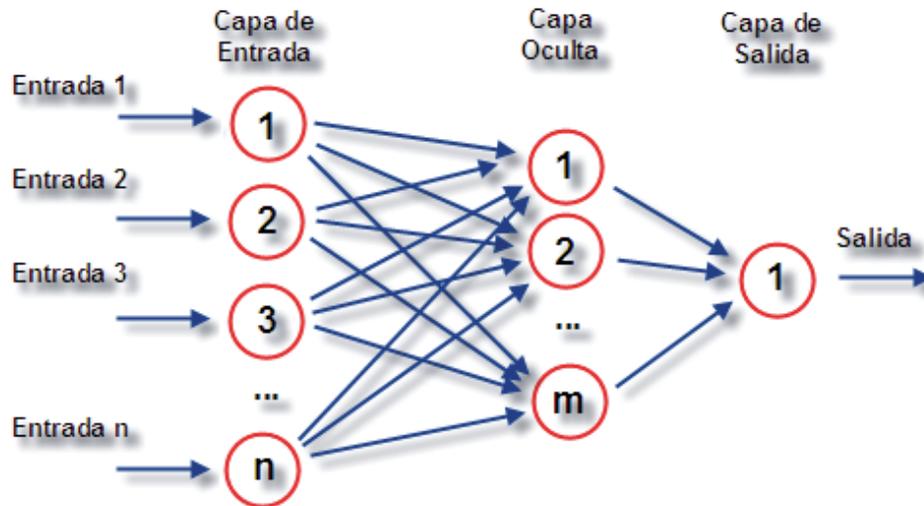


Figura 4.2: Perceptrón Multicapa.

Sea un MLP con  $C$  capas,  $C - 2$  capas ocultas y  $n_c$  neuronas en la capa  $c$ , para  $c = 1, 2, \dots, C$ . Sea  $W^c = (w_{ij}^c)$  la matriz de pesos asociada a las conexiones de la capa  $c$  a la capa  $c + 1$ , para  $c = 1, 2, \dots, C - 1$ , donde  $w_{ij}^c$  representa el peso de la conexión de la neurona  $i$  de la capa  $c$ , a la neurona  $j$  de la capa  $c + 1$ , se denota  $a_i^c$  a la activación de la neurona  $i$  de la capa  $c$ . Estas activaciones se calculan de la siguiente manera:

- a) **Activación de las neuronas de la capa de entrada ( $a_i^1$ ).** Las neuronas de la capa de entrada se encargan de transmitir hacia la red las señales recibidas del exterior:

$$a_i^1 = x_i, i = 1, 2, \dots, n_1 \quad (4.9)$$

Donde  $X = (x_1, x_2, \dots, x_{n_1})$  representa el vector de entrada de la red.

- b) **Activación de las neuronas de la capa oculta**  $c(a_i^c)$ . Las neuronas ocultas de la red procesan información recibida, aplicando la función de activación  $f$  a la suma de los productos de las activaciones que recibe por sus correspondientes pesos:

$$a_i^c = f\left(\sum_{j=1}^{n_{c-1}} w_{ji}^{c-1} a_j^{c-1}\right), \quad i = 1, 2, \dots, n_c \quad y \quad c = 2, 3, \dots, c-1 \quad (4.10)$$

- c) **Activación de las neuronas de la capa de salida**  $(a_i^c)$ . Al igual que en el caso anterior, la activación de las neuronas viene dada por la función de activación  $f$  aplicada a la suma de los productos de las entradas que recibe por sus correspondientes pesos:

$$y_i = a_i^c = f\left(\sum_{j=1}^{n_{c-1}} w_{ji}^{c-1} a_j^{c-1}\right), \quad i = 1, 2, \dots, n_c \quad (4.11)$$

Donde  $Y = (y_1, y_2, \dots, y_{n_c})$  es el vector de salida de la red.

La función  $f$  es llamada función de activación y es comúnmente utilizada la función Sigmoidal, la que presenta una forma similar a una  $S$ . La función Sigmoidal refleja el comportamiento de muchos procesos naturales en que al inicio presenta bajos niveles hasta llegar a un clímax y se encuentra acotada al intervalo  $[0,1]$ .

La función Sigmoidal se calcula de acuerdo a la siguiente ecuación:

$$y = \left[ \sum_{k=1}^m w_k h_k \right] \quad (4.12)$$

$$h_k = \frac{1}{1 + e^{-z}} \quad (4.13)$$

$$z = \sum_{j=1}^n x_j v_{ji} \quad (4.14)$$

Donde  $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]$  son los parámetros de entrada,  $v_k = [v_{k1}, v_{k2}, \dots, v_{kn}]$  son los pesos ocultos,  $w_k$  son los pesos de salida y  $H$  la cantidad de nodos ocultos.

## 5 Algoritmos de Optimización

Modernos algoritmos de optimización han sido desarrollados para realizar búsquedas globales, por ejemplo, Algoritmos Genéticos [35] y Optimización de Enjambre de Partículas (PSO) y Cuckoo Search (CS) [16, 17]. La eficiencia de estos algoritmos ha sido atribuida a que imitan las mejores características de la naturaleza, que han evolucionado por la selección natural durante millones de años.

A continuación se presentan dos algoritmos de optimización que se utilizarán en este trabajo, las metaheurísticas PSO y CS.

### 5.1 Optimización por Enjambre de Partículas

La Optimización por Enjambre de Partículas (PSO) nace al igual que otras metaheurísticas de la imitación de comportamientos de procesos naturales y busca imitar comportamientos sociales de un colectivo a partir de la interacción de sus individuos y estos con su entorno.

Básicamente PSO consiste en un algoritmo iterativo basado en una población de individuos denominada enjambre, en la que cada individuo o partícula, que representa una posible solución dentro del espacio de soluciones multidimensional, sobrevuela junto con el resto de sus congéneres el espacio de decisión en busca de una solución óptima, ajustando su posición de acuerdo a su propia experiencia y la de sus vecinos.

#### 5.1.1 Algoritmo PSO

Una partícula se define como un vector formado por la posición, velocidad y memoria. Sea  $i = 1, 2, \dots, n$  la  $i$ -ésima partícula y  $d = 1, 2, \dots, D$  las dimensiones (número de variables). Dado un espacio de búsqueda  $D$ -dimensional, cada partícula  $i$  del enjambre conoce su posición actual  $X_i = (x_{i1}, x_{i2}, \dots, x_{id})$ , la velocidad  $V_i = (v_{i1}, v_{i2}, \dots, v_{id})$  con la cual ha llegado a dicha posición y la mejor posición  $P_i = (p_{i1}, p_{i2}, \dots, p_{id})$  en la que se ha encontrado, denominada mejor personal  $pbest$ . Además todas las partículas conocen la mejor posición encontrada dentro del enjambre, la que se llama mejor global  $gbest$ .

En PSO convencional la velocidad y posición se actualizan de acuerdo a las siguientes ecuaciones:

$$v_{id}(t+1) = v_{id}(t) + c_1 \cdot r_1 \cdot (p_{id}(t) - x_{id}(t)) + c_2 \cdot r_2 \cdot (p_{gd}(t) - x_{id}(t)) \quad (5.1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (5.2)$$

Donde  $c_1$  es el parámetro cognitivo que indica la influencia máxima de la mejor experiencia

individual de la partícula y  $c_2$  es el parámetro social que indica la influencia máxima de la información social. Los parámetros  $r_1$  y  $r_2$  son dos números aleatorios que varían entre  $[0, 1]$ , cuyo objetivo es emular el comportamiento estocástico y un tanto impredecible que exhibe la población del enjambre.

A continuación se presenta el pseudocódigo de PSO:

**Data:** Función objetivo  $f(x)$ , donde  $x = (x_1, \dots, x_d)^T$

**Result:** Resultados post procesados

Inicializar las posiciones y velocidades de las partículas de forma aleatoria;

**while** ( $(t < \text{generación máxima})$  **or** ( $\text{criterio de término}$ )) **do**

    Evaluar la calidad/fitness de cada partícula;

    Comparar la calidad/fitness de cada partícula con la calidad/fitness de su mejor posición personal. Si la calidad/fitness actual es mejor que el histórico, se debe actualizar la mejor posición por la nueva obtenida.;

    Identificar la partícula con la mejor calidad/fitness hasta el momento y almacenar su índice;

    Ajustar la posición actual y la velocidad de cada partícula;

**return** Resultados post procesados;

**Algoritmo 5.1:** Optimización por Enjambre de Partículas (PSO)

## 5.2 Cuckoo Search

Recientemente se ha desarrollado un nuevo algoritmo de optimización, llamado Cuckoo Search (CS) [12]. Varios estudios muestran que CS posee un magnífico desempeño, superando incluso a PSO [12, 13, 14, 15].

A continuación se presenta el comportamiento reproductivo que realizan algunas especies de aves Cuckoo y por medio de esto se logre comprender mejor el algoritmo CS, luego se presenta el estilo de vuelo llamado Lévy Flights, para finalmente adentrarse en el algoritmo CS.

### 5.2.1 Comportamiento Reproductivo de Aves Cuckoo

Algunas especies de aves Cuckoo presentan una agresiva estrategia de reproducción, ponen sus huevos en nidos de otras aves, e incluso pueden remover los huevos del ave propietaria del nido, para que sus crías tengan una mayor probabilidad de incubación [36].

Si el ave propietaria del nido descubre que el huevo no es suyo, puede lanzarlo fuera, o simplemente abandonar el nido y construir uno nuevo en otro lugar. Algunas especies Cuckoo han evolucionado de una increíble manera, pueden cambiar el color y características de sus huevos para imitar los huevos del ave propietaria del nido, [36] esto reduce la probabilidad de que su huevo sea abandonado y aumenta su reproductividad.

### 5.2.2 Lévy Flights

En la naturaleza los animales buscan comida de una manera aleatoria, o casi aleatoria. La dirección que escogen depende implícitamente de una probabilidad que puede ser modelada

matemáticamente, por ejemplo, estudios muestran que el comportamiento de muchos animales e insectos poseen cierta característica denominada Lévy Flights [37, 38, 39].

Recientes estudios[38] muestran que la mosca de la fruta o *Drosophila melanogaster* explora el medio en una serie de vuelos rectos, interrumpidos por un repentino giro en  $90^\circ$ , dando lugar al estilo de vuelo Lévy Flights. Estudios en comportamientos humanos, como la tribu cazador-recolector denominada Ju/'hoansi, también evidencian patrones de tipo Lévy Flights.

En términos estadísticos, este es un algoritmo estocástico de optimización que busca el mínimo global. La longitud de paso puede ser calculado utilizando el algoritmo Mantegna, por medio de:

$$S = \frac{u}{|v|^{\frac{1}{\beta}}} \quad (5.3)$$

Tanto  $u$  como  $v$  son extraídos de una distribución normal:

$$u \sim N(0, \sigma_u^2), v \sim N(0, \sigma_v^2) \quad (5.4)$$

$\sigma_u^2$  y  $\sigma_v^2$  son la varianza de las distribuciones que vienen dadas por:

$$\sigma_u = \left\{ \frac{\Gamma(1 + \beta) \sin(\tau\beta/2)}{\Gamma[(1 + \beta)/2] \beta 2(\beta - 1)^2} \right\}^{\frac{1}{\beta}}, \sigma_v = 1 \quad (5.5)$$

Estos comportamientos han sido aplicados a optimización de búsqueda, dando buenos resultados [40, 39].

### 5.2.3 Algoritmo Cuckoo Search

Para simplificar el algoritmo, se idealizan las siguientes 3 reglas: [12]

- a) Cada Cuckoo pone un huevo a la vez, y lo deja en un nido escogido al azar.
- b) Los mejores nidos, con los huevos (soluciones) de mejor calidad, se trasladarán a la siguiente generación.
- c) El número de nidos disponibles es fijo, y el ave propietaria puede descubrir a un huevo ajeno con una probabilidad  $p_a \in [0, 1]$ . En este caso el ave puede lanzar el huevo fuera del nido, o abandonarlo y construir uno completamente nuevo en otro lugar.

Por simplicidad, el último punto puede ser aproximado por la fracción  $p_a$  de los  $n$  nidos que son reemplazados por nuevos nidos (con soluciones aleatorias). Además se utilizará la representación simple, en que cada huevo en un nido representa una solución, y los huevos Cuckoo

representan una nueva solución, para así utilizar estas nuevas soluciones (Cuckoo) para reemplazar las soluciones que no son buenas en los nidos. Por supuesto, el algoritmo puede ser extendido para el caso en que cada nido tiene múltiples huevos representando un set de soluciones [12].

Basado en estas tres reglas, los pasos de CS utilizando Lévy Flights, pueden ser resumidos como se muestra en el siguiente pseudocódigo:

```

Data: Función objetivo  $f(x)$ , donde  $x = (x_1, \dots, x_d)^T$ 
Result: Resultados post procesados
Inicializar la población de  $n$  nidos  $x_i (i = 1, 2, \dots, n)$  de forma aleatoria;
while ( $(t < \text{generación máxima})$  or ( $\text{criterio de término}$ )) do
    Crear una nueva generación de nidos utilizando Lévy flights;
    Evaluar la calidad/fitness  $F_i$  de los nidos creados;
    Obtener los mejores nidos entre la generación inicial y la nueva generación;
    if  $F_i > F_j$  then Reemplazar  $j$  por la nueva solución;
    Abandonar una fracción ( $p_a$ ) de los peores nidos;
    // y crear nuevos nidos en una nueva localización, utilizando Lévy flights
    Almacenar las mejores soluciones (o nidos con las mejores soluciones);
    Hacer un ranking de las soluciones encontradas en los mejores nidos;
return Resultados post procesados;

```

#### **Algoritmo 5.2:** Cuckoo Search (CS)

Cuando generamos una nueva solución  $x^{(t+1)}$ , para el Cuckoo  $i$ , se utiliza Lévy Flights 5.6.

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus Levy(\lambda) \quad (5.6)$$

Donde  $\alpha > 0$  es el tamaño de paso que debería estar relacionado con la escala del problema en interés, en la mayoría de los casos se utiliza  $\alpha = 1$ . El camino aleatorio es una *Cadena de Márkov* donde el estado/ubicación siguiente, sólo depende de la ubicación actual (primer término de la ecuación) y de la probabilidad de transición (segundo término de la ecuación). El producto  $\oplus$  significa Matriz de Multiplicación o Entrywise multiplication.

Lévy Flights esencialmente provee un camino aleatorio donde la longitud del paso se obtiene de la distribución Lévy 5.7.

$$Levy \sim u = t^{-\lambda}, \quad (1 < \lambda \leq 3) \quad (5.7)$$

La que posee una media y una varianza infinita. Algunas de las nuevas soluciones deberían ser generadas por Lévy alrededor de la mejor solución obtenida hasta aquí, esto acelerará la búsqueda local. Sin embargo, una considerable fracción de nuevas soluciones deberían ser generadas en una ubicación que sea lo suficientemente lejana de la actual mejor solución, esto hará que no se quede atrapado en un óptimo local [12].

Análisis y resultados obtenidos, nos indican que la tasa de convergencia, hasta cierto punto, no es sensitiva a los parámetros usados. Esto significa que el ajuste de los parámetros del algoritmo no es necesariamente dado por el problema. Para la mayoría de los problemas de optimización, basta con considerar  $n$  entre 15 y 25,  $p_a$  entre 0,15 y 0,30 y  $\alpha = 1$  [12, 13].

## 6 Suavizado de Datos

La serie temporal de  $n$  observaciones es posible descomponerla en una suma de ondas de diferentes frecuencias. Las componentes de baja frecuencia varían suavemente, mientras que las componentes de alta frecuencia varían rápidamente. Se asume, para las series de datos, que los componentes de alta frecuencia constituyen el ruido de fondo que es necesario eliminar.

La aplicación de las técnicas de suavizado es preciso realizarla de forma exploratoria, debido a que el cambio del análisis del dominio del tiempo al dominio de la frecuencia no es en este caso conocida y es esta su principal limitación [41].

La Media Móvil (MA) es una estrategia de suavizado utilizada para crear series de promedios utilizados para identificar o extraer la tendencia de una serie disminuyendo el ruido de la misma.

La MA es aplicada sobre 3 puntos de una serie de tiempo de longitud  $n$ , suavizando  $n - 2$  elementos de la siguiente forma:

$$\tilde{S}_k = \sum_{i=k-1}^{k+1} \frac{x_i}{3} \quad (6.1)$$

$$\tilde{S}_1 = x_1 \quad (6.2)$$

$$\tilde{S}_n = x_n \quad (6.3)$$

Donde  $\tilde{S}_k$  es el  $k$ -ésimo elemento suavizado de la serie original  $k = 2, \dots, n - 1$  y  $x_i$  es cada elemento de la serie original.  $\tilde{S}_1$  y  $\tilde{S}_n$  tienen los mismos valores que  $x_1$  y  $x_n$  respectivamente.

## 7 Métricas de Calidad

La calidad de los resultados obtenidos se puede calcular siguiendo distintos parámetros estadísticos. En la mayor parte de las investigaciones de predicción que utilizan RNA, la red se entrena buscando minimizar la Raíz del Error Cuadrático Medio o el Error Porcentual Absoluto Medio.

Para esta investigación se utilizará la Raíz del Error Cuadrático Medio para ajustar los parámetros de la red, durante la etapa de Training. Para la etapa de Testing además de las métricas mencionadas, se calculará el Coeficiente de Correlación y de Determinación, para así estudiar la dependencia entre los resultados observados y los resultados esperados.

A continuación se detallan las métricas a utilizar: [42]

### 7.1 Error Cuadrático Medio

El Error Cuadrático Medio o Mean Squared Error (*MSE*), es la media aritmética de la suma de las diferencias cuadráticas entre los valores de la serie y los valores producidos por el modelo para iguales momentos de tiempo. Tiene el inconveniente de que se expresa en unidades cuadradas de la medida de la variable en la serie. Mientras menor sea, más adecuado es el modelo.

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (7.1)$$

Dónde  $y_i$  corresponde a la serie observada e  $\hat{y}_i$  a la serie estimada.

### 7.2 Raíz del Error Cuadrático Medio

La Raíz del Error Cuadrático Medio o Root Mean Square Error (*RMSE*), es la raíz cuadrada del promedio de la suma de diferencias cuadradas entre los valores observados en la serie y los esperados según el modelo de tendencia. El resultado se expresa en la misma magnitud de medida de la variable. Mientras menor sea, más adecuado es el modelo.

$$RMSE = \sqrt{MSE} \quad (7.2)$$

### 7.3 Porcentaje de Error Absoluto Medio

El Porcentaje de Error Absoluto Medio o Mean Absolute Percentage Error (*MAPE*), es el Error Absoluto Medio expresado en porcentaje. El MAPE es la media aritmética de la suma de las diferencias absolutas entre valores de la serie y valores obtenidos con el modelo en iguales

momentos de tiempo. La diferencia entre el Error Absoluto Medio y el Error Medio radica en que el Error Medio viene dado por el absoluto de la suma de las diferencias, mientras que el Error Absoluto Medio viene dado por la suma de las diferencias absolutas.

Se expresa en la misma unidad de medida de las observaciones. Mientras menor sea, más adecuado es el modelo.

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (7.3)$$

Dónde  $y_i$  corresponde a la serie observada e  $\hat{y}_i$  a la serie estimada.

## 7.4 Coeficiente de Determinación

El Coeficiente de Determinación ( $R^2$ ) corresponde al cuadrado del Coeficiente de Correlación de Pearson y adquiere valores entre 0 y 1.

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y}_i)^2} \quad (7.4)$$

Dónde  $y_i$  corresponde a la serie observada,  $\hat{y}_i$  a la serie estimada e  $\bar{y}_i$  al promedio de la serie observada.

## 7.5 Coeficiente de Correlación

El Coeficiente de Correlación de Pearson ( $R$ ) es la medida de la relación entre dos variables. Es independiente de la escala de medida de las variables. Por lo tanto se puede definir como un índice para medir el grado de relación entre dos variables cuantitativas.

El Coeficiente de Correlación varía en el intervalo  $[0, 1]$ .

- a) Sí  $R = 1$  existe una correlación total entre las dos variables (relación directa).
- b) Sí  $R \in ]0, 1[$  existe una correlación positiva.
- c) Sí  $R = 0$  puede que no existan relaciones lineales o que las dos variables sean independientes.

$$R = \sqrt{R^2} \quad (7.5)$$

## 8 Datos

Para esta investigación se utilizó tanto la cantidad de lesionados leves como de lesionados graves ocurridos en la región de Valparaíso, agrupados por semana desde el año 2003 hasta el año 2012. Si bien los datos de entrada reflejan la cantidad de lesionados en accidentes de tránsito, no se señalan las causas de estos.

De un total de 530 registros agrupados por semana se utilizó un 85 % para la fase de Training y el 15 % restante para la fase de Testing.

La Figura 8.1 (a) de la izquierda muestra los datos agrupados por semana sin suavizar, mientras que la Figura 8.1 (b) de la derecha muestra los datos agrupados por semana suavizados por medio de MA.

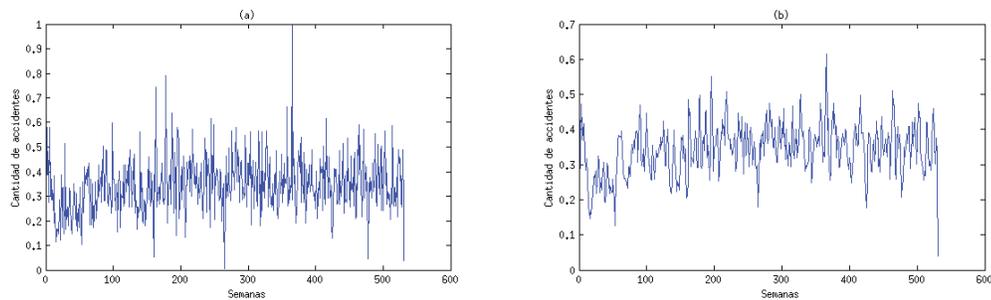


Figura 8.1: Cantidad de accidentes de tránsito (a) Serie no suavizada (b) Serie suavizada.

En el dominio del tiempo ambos gráficos muestran una diferencia mínima en cuanto a la amplitud de la serie. Para realizar un análisis más profundo se analizó la autocorrelación de ambas series de datos, la que muestra la relación que tiene la serie consigo misma, si una serie posee autocorrelación se dice que la serie tiene *memoria*, es decir que los datos siguientes dependen de los datos previos y la predicción es posible.

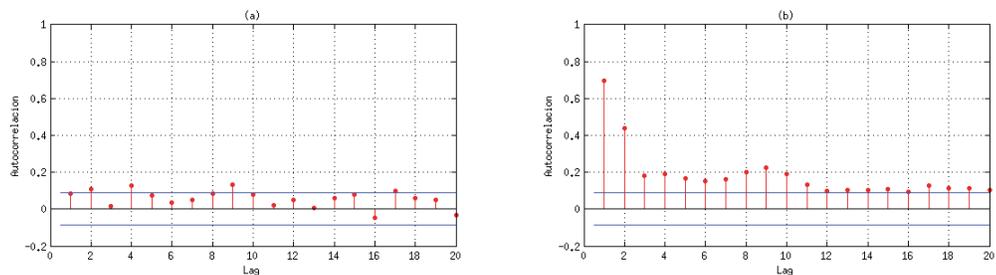


Figura 8.2: Autocorrelación de accidentes de tránsito (a) Serie no suavizada (b) Serie suavizada.

La Figura 8.2 (a) de la izquierda muestra la autocorrelación de la serie no suavizada, los puntos que se encuentran dentro de la franja no poseen una autocorrelación con una confianza de un 95 %. En esta gráfica se aprecia que la mayor cantidad de puntos no posee autocorrelación.

Por otro lado en la Figura 8.2 (b) de la derecha muestra la autocorrelación de la serie suavizada por medio de la Media Móvil (MA). En este gráfico se puede apreciar que luego de suavizar los datos y por medio de esto eliminar el ruido, todos los puntos se encuentran autocorrelacionados, lo que facilita la predicción.

La estrategia de resolución, tanto con los datos suavizados como con los datos sin suavizar, consiste en la creación de cuatro modelos de predicción. Dos de estos modelos utilizarán los datos suavizados por medio de la Media Móvil (MA) como se muestra en la Figura 8.3 (a), mientras que los dos modelos restantes utilizarán los datos sin suavizar como se muestra en la Figura 8.3 (b).

Tanto para los datos suavizados como para los no suavizados se utilizará una RNA junto con CS, y por otro lado una RNA junto con PSO. Finalmente se compararán los resultados obtenidos.

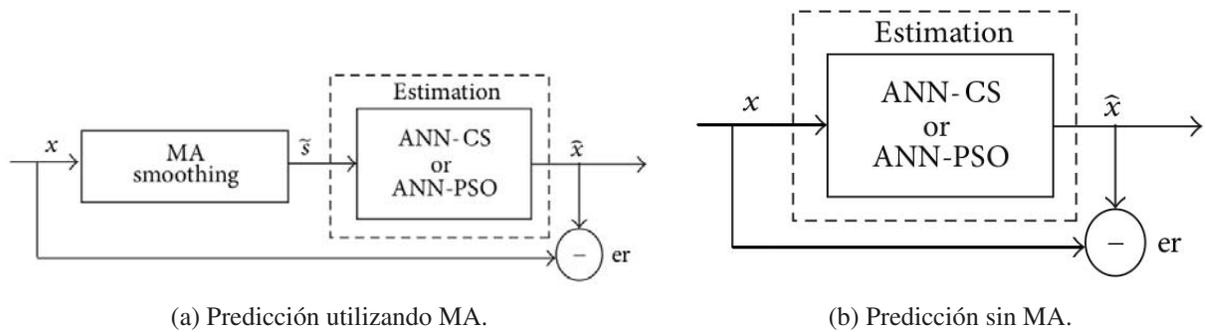


Figura 8.3: Estrategia de resolución con suavizado y sin suavizado.

## 9 Resultados

Se entrenó una red neuronal junto con el algoritmo **Cuckoo Search (CS)** para hallar el mejor resultado (menor error), posteriormente se comparó con otra red neuronal la que utilizaba el algoritmo **Optimización por Enjambre de Partículas (PSO)**, utilizado en el paper "Smoothing Strategies Combined with ARIMA and Neural Networks to Improve the Forecasting of Traffic Accidents", de la cual se obtuvieron los mejores parámetros hallados [4].

A continuación se presenta el procedimiento para calibrar la RNA con CS y posteriormente con PSO para el caso de los datos suavizados, en el caso de los datos no suavizados se presentan solamente los resultados obtenidos en la parte final de este capítulo.

### 9.1 Red neuronal con CS

Se utilizó una red neuronal **Perceptrón Multicapa** junto con el algoritmo **Cuckoo Search**, como función de activación la función Sigmoidal y como función fitness el error MAPE.

Se obtuvieron los mejores valores para la cantidad de entrada (cantidad de semanas previas a considerar), cantidad de nodos ocultos, cantidad de nidos, probabilidad de detección de un nido y cantidad de iteraciones. La estrategia a seguir se detalla a continuación:

- a) Se fijaron los parámetros cantidad de nodos ocultos igual a  $\log_2(\text{cantidad de datos}) = 9$ , cantidad de nidos igual a 25, probabilidad de detección igual a 25 % y cantidad de iteraciones igual a 1000. Posteriormente se iteraron los parámetros que se detallan en los siguientes puntos.
- b) Cantidad de entradas de 1 a 18.
- c) Cantidad de nodos ocultos de 8 a 28.
- d) Cantidad de nidos de 15 a 30.
- e) Probabilidad de detección de nido entre 15 % y 30 %.
- f) Cantidad de iteraciones entre 200 y 3000.
- g) Finalmente la mejor configuración se iteró 30 veces, con pesos iniciales seleccionados aleatoriamente para evaluar el error medio y la desviación estándar promedio.

Cada calibración de parámetros se realizó utilizando los datos suavizados por medio de la Media Móvil, en la que los datos suavizados son los que ingresan a la red neuronal, y el error es comparado utilizando la salida esperada sin suavizar, tal como se mostró anteriormente en la Figura 8.3. Los resultados de la calibración de parámetros se detalla a continuación:

1. La cantidad de entradas (semanas previas) varió de 1 a 18, obteniéndose el menor MAPE de 16.78 % para 5 semanas previas como se muestra en la Figura 9.1 (a).
2. La cantidad de nodos ocultos varió de 5 a 20, obteniéndose el menor MAPE 14.0 % para 8 nodos ocultos como se muestra en la Figura 9.1 (b).
3. La cantidad de nidos utilizados en el algoritmo Cuckoo Search varió de 15 a 30, obteniéndose el menor MAPE de 13.84 % para 29 nidos como se muestra en la Figura 9.1 (c).
4. La probabilidad de detección de un nido y posterior reemplazo varió de 0.15 a 0.30, obteniéndose el menor MAPE de 13.38 % para una probabilidad de 0.18 como se muestra en la Figura 9.1 (d).
5. La cantidad de iteraciones varió de 200 a 3000, obteniéndose el menor MAPE de 12.86 % para una cantidad de 2000 iteraciones. En la Figura 9.1 (e), se muestra el descenso del MAPE para el caso de 2000 iteraciones.
6. Finalmente la mejor configuración se ejecutó 20 veces, obteniendo un MAPE de 12.11 % en la iteración número 13, como se muestra en la Figura 9.1 (f).

En la Figura 9.2 se se muestra el valor observado versus el valor estimado.

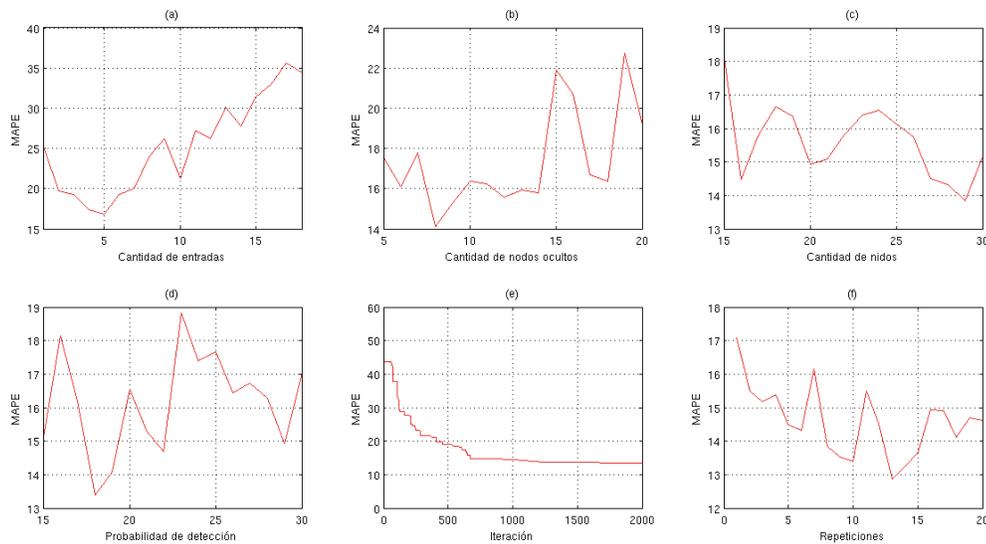
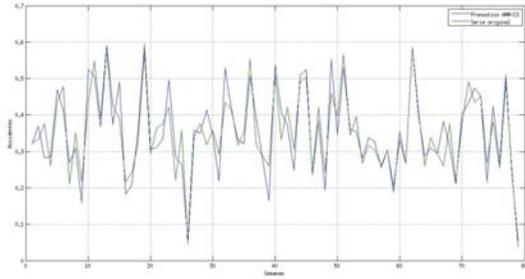


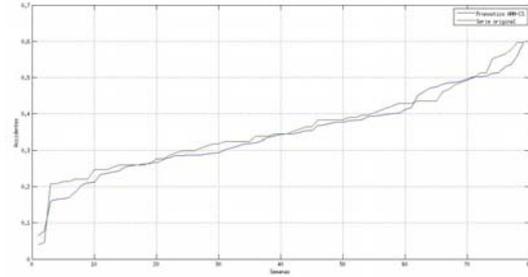
Figura 9.1: Obtención de parámetros de CS.

## 9.2 Red neuronal con PSO

Para el caso del algoritmo de **Optimización por Enjambre de Partícula (PSO)** se consideraron los resultados obtenidos en el trabajo realizado por Lida Barba *et al.* [4], en el cual se



(a) Valor observado vs Valor estimado - CS



(b) Valor observado vs Valor estimado ordenado - CS

Figura 9.2: Valor observado vs Valor estimado utilizando CS

compararon diferentes técnicas de suavizado de datos utilizando los mismos datos de accidentes de tránsito que los utilizados en este proyecto, pero en vez de utilizar Cuckoo Search se utilizó el algoritmo Optimización por Enjambre de Partículas.

Se entrenó la red neuronal utilizando los parámetros establecidos en el trabajo mencionado, los cuales corresponden a: [4]

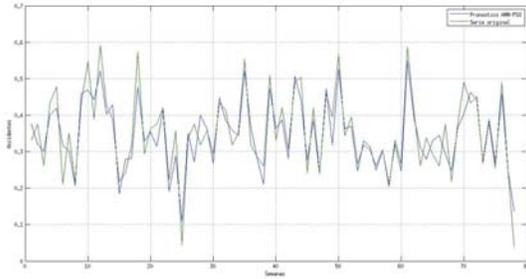
- a) Función de activación sigmoideal.
- b) Función fitness RMSE.
- c) 9 parámetros de entrada.
- d) 10 nodos ocultos.
- e) 30 partículas.
- f) 2500 iteraciones.
- g) Los parámetros C1 (1.05), C2 (2.95), Wmax (0.9) y Wmin (0.2).

La configuración propuesta se ejecutó 30 veces para obtener el menor error. La Figura 9.3 muestra el valor observado versus el valor estimado de la iteración número 5, la cual obtuvo el menor MAPE de 14.95 %. Los valores de los parámetros restantes se detallan en la Tabla 9.1.

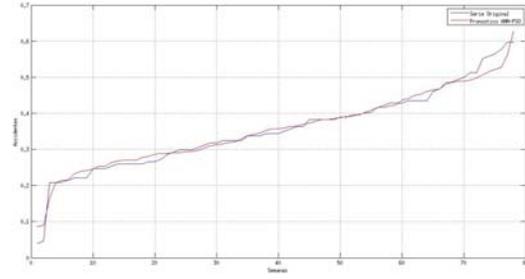
### 9.3 Discusión de Resultados

Los resultados utilizando una **RNA** tanto con **CS** como con **PSO**, junto con las configuraciones utilizadas se muestran en la Tabla 9.1.

En el caso de los datos no suavizados, se puede apreciar que tanto CS como PSO obtuvieron resultados similares y bastante deficientes. Se aprecia un error bastante alto en la predicción, con un MAPE de 38,52 % para CS y una correlación de 3,00 % para PSO.



(a) Valor observado vs Valor estimado - PSO



(b) Valor observado vs Valor estimado ordenado - PSO

Figura 9.3: Valor observado vs Valor estimado utilizando PSO

En el caso de los datos suavizados, PSO obtuvo un mejor resultado que CS en las métricas RMSE, R, R2 y en un tiempo significativamente menor, sin embargo esta diferencia fue bastante baja (en términos de milésimas). Sin embargo donde se aprecia una diferencia considerable fue en la métrica MAPE, donde CS obtuvo dos puntos porcentuales menos que PSO, superándolo notoriamente en esta métrica.

La enorme diferencia en los resultados entre la predicción de los datos suavizados y los datos no suavizados se asocia a que en el primer caso el ruido fue disminuido por medio de MA, teniendo presente que ambas metaheurísticas son sensibles a este.

Tabla 9.1: Comparación de resultados

	No suavizado		Suavizado	
	CS	PSO	CS	PSO
<b>Función de Activación</b>	Sigmoidal	Sigmoidal	Sigmoidal	Sigmoidal
<b>Entradas</b>	3	3	5	13
<b>Nodos ocultos</b>	14	19	8	10
<b>Partículas</b>	29	19	29	30
<b>Iteraciones</b>	<b>1800</b>	2800	<b>2000</b>	2500
<b>Tiempo (seg)</b>	76,43	<b>32,59</b>	57,81	<b>39,14</b>
<b>Nidos</b>	29	-	29	-
<b>Probabilidad de detección</b>	0,26	-	0,18	-
<b>Partículas</b>	-	19	-	30
<b>c1</b>	-	1,05	-	1,05
<b>c2</b>	-	2,95	-	2,95
<b>MAPE</b>	<b>38,52 %</b>	38,98 %	<b>12,11 %</b>	14,95 %
<b>RMSE</b>	<b>0,1295</b>	0,1311	0,0453 %	<b>0,0479</b>
<b>R</b>	2,84 %	<b>3,00 %</b>	91,37 %	<b>91,63 %</b>
<b>R2</b>	0,08 %	<b>0,09 %</b>	83,48 %	<b>83,96 %</b>

## 10 Conclusión

Esta investigación pretende ser un aporte al estudio de la seguridad vial en Chile, analizando las relaciones que existen entre las variables que determinan la cantidad de accidentes de tránsito ocurridos en la región de Valparaíso.

En la realización de esta investigación se indagaron distintas técnicas de inteligencia artificial que se han aplicado a variadas problemáticas, entre ellas el estudio de accidentes de tránsito en distintas ciudades del Mundo.

La problemática se abordó por medio de técnicas de Inteligencia Computacional (IC), prediciendo la cantidad de siniestros, mas no de fallecidos, como consecuencia de los accidentes de tránsito. Para esto se utilizó como insumo datos históricos de accidentes de tránsito ocurridos en Valparaíso entre los años 2003 y 2012.

Este estudio se centró en la utilización de redes neuronales artificiales (RNA), junto con un reciente algoritmo de optimización llamado Cuckoo Search (CS), el cual es una metaheurística basada en el comportamiento reproductivo de las aves Cuckoo. Los resultados obtenidos se compararon con una investigación similar realizado el año 2014, en el que se implementó una red neuronal artificial junto con el algoritmo Optimización por Enjambre de Partículas (PSO), entre otras técnicas de Inteligencia Computacional [4].

En el presente trabajo se compararon cuatro modelos predictivos utilizando redes neuronales artificiales (RNA). Para determinar la arquitectura de cada RNA, es decir la cantidad de nodos de entrada y de nodos ocultos, se iteró desde 1 a 18 y 5 a 20 respectivamente, escogiendo la estructura que predecía con el menor error, mientras que de acuerdo al problema la cantidad de nodos de salida se fijó en 1.

Para el caso de la serie no suavizada utilizando CS y PSO, se obtuvo en ambos casos el menor error utilizando 3 nodos de entrada, sin embargo la cantidad de nodos ocultos fue de 14 para CS y 19 para PSO.

Para el caso de la serie suavizada, la arquitectura de la RNA junto con CS fue calculada de acuerdo a las iteraciones mencionadas anteriormente, sin embargo la arquitectura de la RNA junto con PSO fue la utilizada en el paper *Smoothing Strategies Combined with ARIMA and Neural Networks to Improve the Forecasting of Traffic Accidents* [4]. En este caso la mayor diferencia se obtuvo en la cantidad de nodos de entrada, 5 utilizando CS y 18 utilizando PSO, mientras que la cantidad de nodos ocultos fue de 8 para CS y 10 para PSO.

Por otra parte, si se compara la serie suavizada utilizando la Media Móvil (MA) con la serie sin suavizar, en la serie suavizada se evidencia una mejora considerable en los resultados de las métricas medidas, ya sea utilizando CS como PSO, esto debido a que el suavizado reduce el ruido de la serie, y ambas metaheurísticas son sensibles a este.

Si se considera la serie suavizada por medio de la MA y se analizan los resultados obtenidos por medio de CS y PSO, se puede apreciar que la red neuronal propuesta utilizando CS logró

superar a PSO en la predicción de accidentes de tránsito, obteniendo un MAPE de 12,11% y 14,95% respectivamente.

También se destaca que CS presenta una menor cantidad de parámetros a calibrar que PSO y logra alcanzar una convergencia en una menor cantidad de iteraciones.

En el caso de CS los parámetros a calibrar corresponden a la cantidad de nidos, la probabilidad de detección de un nido, la constante *alpha* que corresponde al tamaño de paso y los parámetros *u* y *v* pertenecientes al algoritmo Mantegna que complementa al algoritmo CS. Todos estos parámetros de CS de acuerdo a la literatura se encuentran acotados a ciertos intervalos, con los cuales se encuentra una solución a la mayor parte de los problemas.

Mientras que en el algoritmo PSO los parámetros a calibrar corresponden a la cantidad de partículas, el parámetro cognitivo, el parámetro social, los parámetros aleatorios  $r_1$  y  $r_2$ , la velocidad mínima y la velocidad máxima.

En el caso de este trabajo a partir de  $n$  semanas previas, se predijo la semana siguiente  $n + 1$ , sin embargo si se quisiera predecir  $h$  semanas siguientes con  $h > 1$ , con este modelo no se podría. Como trabajo futuro se propone ampliar el horizonte de predicción a más de una semana, es decir a  $n + h$ .

Finalmente se espera que esta investigación sea un aporte en el estudio tanto de las metaheurísticas tratadas como al estudio del pre procesamiento de los datos por medio de técnicas de suavizado. Se espera que también sea un apoyo en las decisiones que se adopten para controlar y disminuir la cantidad de accidentes de tránsito, que se sabe es una preocupación latente para las autoridades a nivel mundial y que cada año cobra miles de víctimas.

## Bibliografía

- [1] Organización Mundial de la Salud. Informe sobre la situación mundial de la seguridad vial 2013. 2013.
- [2] Comisión Nacional de Seguridad de Tránsito. Evolución de siniestros de tránsito Chile (1972 a 2012). 2012.
- [3] Comisión Nacional de Seguridad de Tránsito. Siniestros de tránsito y víctimas por regiones (Año 2012). 2012.
- [4] Nibaldo Rodríguez y Cecilia Montt Lida Barba. Smoothing Strategies Combined with ARIMA and Neural Networks to Improve the Forecasting of Traffic Accidents. *Hindawi*, 2014.
- [5] Félix Hernán Castro Fuentes. LS-SVM basada en optimización por enjambre de partículas para clasificación de accidentes de tránsito. *Pontificia Universidad Católica de Valparaíso*, 2010.
- [6] Eduardo Andrés Cárdenas Calderón. Clasificación de la severidad en accidentes de tránsito de la Región Metropolitana, utilizando redes neuronales artificiales y PSO modificado. *Pontificia Universidad Católica de Valparaíso*, 2012.
- [7] José David Fierro Moya. Clasificación de accidentes de tránsito en la zona de Valparaíso utilizando redes neuronales con algoritmos PSO. *Pontificia Universidad Católica de Valparaíso*, 2012.
- [8] Abdelwahab H. T. y Abdel-Aty M. A. Development of Artificial Neural Network Models to Predict Driver injury Severity in Traffic Accidents at Signalized Intersections. 1997.
- [9] Dia H and Rose G. Development and Evaluation of Neural Network Freeway Incident Detection Models Using Field Data. *Transportation Research C*, 1997.
- [10] Chen H. C Yang W.T and Brown D. Detecting Safer Driving Patterns By A Neural. 1999.
- [11] Ferrari A Mussone, L and Oneta M. An analysis of urban collisions using an artificial intelligence model. Milano : Accident Analysis and Prevention. 1999.
- [12] Xin-She Yang and Suash Deb. Cuckoo Search via Levy Flights. *IEEE Publications*, 2009.
- [13] Xin-She Yang and Suash Deb. Engineering Optimisation by Cuckoo Search. 2010.
- [14] Nashat Mansour Ahmad AL Kawam. Metaheuristic Optimization Algorithms for Training Artificial Neural Networks. *International Journal of Computer and Information Technology*, 2012.

- [15] Abdullah Khan Nazri Mohd. Nawi and Mohammad Zubair Rehman. A New Back-Propagation Neural Network Optimized with Cuckoo Search Algorithm. 2013.
- [16] J. Kennedy and R. C. Eberhart. Particle swarm optimization. *Proc. of IEEE International Conference on Neural Networks, Piscataway*, 1995.
- [17] Eberhart R. C. Kennedy, J. and Y. Shi. Swarm intelligence. *Academic Press*, 2001.
- [18] José Miguel Rubio y Silvana Lanata Cecilia Montt. Análisis de accidentes de tránsito con inteligencia computacional. *Pontificia Universidad Católica de Valparaíso*.
- [19] G. H. Stones M. J. Bedard M., Guyat and Hireds J. P. The Independent Contribution of Driver, Crash, and Vehicle Characteristics to Driver Fatalities. *Accident Analysis and Prevention*, 2002.
- [20] Ossiander E. M. and Cummings P. Freeway speed limits and Traffic Fatalities in Washington State. *Accident Analysis and Prevention*, 2002.
- [21] Abdelwahab M and Abdel Aty. Analysis and Prediction of Traffic Fatalities Resulting From Angle Collisions Including the Effect of Vehicles: Configuration and Compatibility. *Accident Analysis*, 2003.
- [22] Sohn So Young and Lee Sung Ho Lee. Data fusion, ensemble and clustering to improve the classification accuracy for the severity of road traffic accidents in Korea. *Safety Science*, 2003.
- [23] AjitAbraham Chong Miao and Paprzycki Marcin. Accident Data Mining Using Machine Learning Paradigms. *Oklahoma State University, USA*, 2003.
- [24] Montt C. Conductas y actitudes de los usuarios del sistema de transporte en la seguridad vial. *Pontificia Universidad Católica de Valparaíso*, 2006.
- [25] Zúñiga Alejandro y Chacon Max Montt, Cecilia. Estructura Bayesianas en Accidentes de Tránsito. *Pontificia Universidad Católica de Valparaíso*, 2009.
- [26] Zúñiga Alejandro y Chacon Max Montt, Cecilia. Análisis de Accidentes de Tránsito con Métodos de Agrupamiento. *Pontificia Universidad Católica de Valparaíso*, 2009.
- [27] F. Rosenblatt. The perceptron: A perceiving and recognizing automation. *Technical Report, Cornell Aeronautical Laboratory*, 1957.
- [28] F. Rosenblatt. Principles of Neurodynamics. *Washington: Spartan Books*, 1962.
- [29] B. Widrow. Adaptive sample data systems a statistical theory of adaption. *Welscon*, 1959.
- [30] M. Minsky y S. Paper. Perceptrons. 1969.
- [31] J. Hopfield. Computación neuronal de decisiones en problemas de optimización. 1985.

- [32] M. Pérez Delgado y Q. Martín Martín. *Aplicación de las redes neuronales artificiales a las estadística*. La Muralla, 2003.
- [33] Xabier Basogain Olabe. *Redes Neuronales Artificiales y sus aplicaciones*. *Escuela Superior de Ingeniería de Bilbao, EHU*.
- [34] P. Isasi Vinuela y I. Galvan León. *Redes Neuronales Artificiales: Un enfoque Práctico*. *Pearson Educacion*, 2004.
- [35] D. E. Goldberg. *Genetic Algorithms in Search, Optimisation and Machine Learning*. *Addison Wesley*, 1989.
- [36] Sorenson M. D. Payne, R. B. and K. Klitz. *The Cuckoos*. *Oxford University Press*, 2005.
- [37] Liebovitch L. S. Brown, C. and R. Glendon. *Levy flights in Dobe Ju/'hoansi foraging patterns*. *Human Ecol.*, 2007.
- [38] A. M. Reynolds and M. A. Frye. *Free-Flight odor tracking in Drosophila is consistent with an optimal intermittent scale-free search*. *PLoS One*, 2007.
- [39] I. Pavlyukevich. *Levy flights, non-local search and simulated annealing*. *J. Computational Physics*, 2007.
- [40] M. F. Shlesinger. *Search research*. *Nature*, 2006.
- [41] Jaume Arnau. *Métodos y técnicas avanzadas de análisis de datos en ciencias del comportamiento*. *Universidad de Barcelona*, 1996.
- [42] Armando Aguirre Jaime. *Introducción al tratamiento de series temporales*. *Editorial Diaz de Santos*, 1994.