

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

**CLASIFICACIÓN DE LA SEVERIDAD EN
ACCIDENTES DE TRÁNSITO DE LA REGIÓN
METROPOLITANA, UTILIZANDO REDES
NEURONALES ARTIFICIALES Y PSO
MODIFICADO**

EDUARDO ANDRÉS CÁRDENAS CALDERÓN

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

MARZO, 2012

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

**CLASIFICACIÓN DE LA SEVERIDAD EN
ACCIDENTES DE TRÁNSITO DE LA REGIÓN
METROPOLITANA, UTILIZANDO REDES
NEURONALES ARTIFICIALES Y PSO
MODIFICADO**

EDUARDO ANDRÉS CÁRDENAS CALDERÓN

Profesor Guía: **Nibaldo Rodríguez Agurto**

Profesor Co-referente: **Broderick Crawford Labrín**

Carrera: **Ingeniería Civil Informática**

MARZO, 2012

Dedicatoria

Dedicada a mi familia, y en particular a mis padres por su incondicional apoyo y sacrificio realizado durante este proceso.

Agradecimientos

A Dios, a mis padres, profesor guía y correferente.

Abstract

The artificial neural networks are a paradigm of learning and automatic process it was born like a concept in the year 1943 make it from this techniques that counts with a unquantifiable years of study, however their use in problems like in this investigation (classification from cars accidents) there are recently. If the cars accidents have been analyzed by different techniques and data mining algorithms, like decision trees, bayesian networks and vectorial support machines, the utilization the evolutions algorithms, like example Particle Swarm Optimization (PSO), the aren't been incorporated on this investigations, but there are positives options in the moment for good results, like on [1].

Due to the above, on this investigation are classifications models that to measure the severity of lesions from cars accidents in Metropolitan Region, using a artificial neural networks with evolutionary algorithms. All the best results are with Particle Swarm Optimizer with Dynamic Adaptation (DAPSO) variable with the function of sigmoidal activation for occult cape and softmax for scape, this one flinged a exactitude porcentual number of 82,36 %, that means it's a correctly classifications on this porcentual number for hurts and unhurt people.

Key words : Artificial Neural Network (ANN), Particle Swarm Optimization (PSO), Clasification from cars accidents, Softmax, Particle Swarm Optimizer with Dynamic Adaptation (DAPSO).

Resumen

Las redes neuronales artificiales son un paradigma de aprendizaje y procesamiento automático que nace como concepto en el año 1943 haciendo de ésta una técnica que cuenta con una cantidad no despreciable de años de estudio, sin embargo su utilización en problemas como el que se presenta en esta investigación (clasificación de accidentes de tránsito) son relativamente recientes. Si bien los accidentes de tránsito han sido analizados por distintas técnicas y algoritmos de minería de datos, tales como árboles de decisión, redes bayesianas y maquinas de soporte vectorial (SVM), la utilización de algoritmos evolutivos, como por ejemplo Optimización por enjambre de Partículas (PSO), no han sido incorporados del todo dentro de éstas investigaciones, pese a que son alternativas no despreciables al momento de obtener resultados satisfactorios, como los obtenidos en [1]

Debido a lo anterior, en ésta investigación se presentan modelos de clasificación que miden la severidad de las lesiones 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) en conjunto con las funciones de activación sigmoideal para capa oculta y softmax para la capa de salida, la cual arrojo una exactitud del 82,36 % lo cual indica que se clasifican correctamente, en dicho porcentaje a las personas lesionadas o ilesas.

Palabras Claves : Redes Neuronales Artificiales , Optimización por enjambre de Partículas, Clasificación de accidentes de tránsito, Softmax, Optimizador por Enjambre de Partículas con Adaptación Dinámica (DAPSO)

Indice General

| | |
|---|----------|
| Lista de Abreviaturas | IX |
| Indice de Figuras | X |
| Indice de Tablas | XII |
| 1. Introducción | 1 |
| 2. Marco General del Proyecto | 3 |
| 2.1. Objetivo General | 3 |
| 2.2. Objetivos Específicos | 3 |
| 2.3. Planteamiento del Problema | 3 |
| 2.4. Alcance del Estudio | 3 |
| 3. Marco Teórico | 4 |
| 3.1. Reseña Histórica | 4 |
| 3.2. Redes Neuronales Biológicas | 5 |
| 3.3. Redes Neuronales Artificiales (RNA). | 7 |
| 3.4. Topologías de Redes Neuronales. | 9 |
| 3.4.1. Según su número de capas | 9 |
| 3.4.2. Según el tipo de conexiones | 11 |
| 3.4.3. Según el grado de conexión | 11 |
| 3.5. Redes Neuronales Recurrentes | 11 |
| 3.5.1. Red de Jordan. | 13 |
| 3.5.2. Red de Elman. | 14 |
| 3.6. Aprendizaje de una RNA. | 15 |
| 3.6.1. Aprendizaje Supervisado. | 16 |

| | | |
|-----------|---|-----------|
| 3.6.2. | Aprendizaje No Supervisado. | 17 |
| 3.6.3. | Aprendizaje Hibrido | 17 |
| 3.7. | Perceptrón | 17 |
| 3.8. | Perceptrón Multicapa (MLP) | 20 |
| 3.9. | Red Neuronal de Base Radial (RBFN) | 22 |
| 3.10. | Red Wavelet (WNN) | 24 |
| 3.11. | Red Neuronal Polinomial (PNN) | 25 |
| 3.12. | Softmax. | 26 |
| 3.13. | Aplicaciones. | 27 |
| 4. | Optimización por Enjambre de Partículas (PSO) | 28 |
| 4.1. | Introducción | 28 |
| 4.2. | Fundamentos del movimiento de partículas. | 29 |
| 4.3. | Algoritmo PSO. | 29 |
| 4.4. | Parámetros PSO. | 31 |
| 4.5. | PSO con actualizaciones síncronas y asíncronas. | 31 |
| 4.6. | Exploración y Explotación con PSO. | 32 |
| 4.7. | Variaciones del Algoritmos de PSO. | 32 |
| 4.7.1. | PSO Discreto | 33 |
| 4.7.2. | LDWPSO | 33 |
| 4.7.3. | Center PSO | 34 |
| 4.7.4. | Simple PSO | 35 |
| 4.7.5. | PSO con Factor de Inercia | 36 |
| 4.7.6. | DAPSO | 36 |
| 4.7.7. | PSO Quantum | 37 |
| 4.7.8. | Métodos de Mejoramiento de PSO para Optimización Global | 38 |

| | |
|--|-----------|
| 5. Métodos de Entrenamiento | 40 |
| 5.1. Conceptos básicos | 41 |
| 5.2. Métricas de rendimiento para evaluar el clasificador | 42 |
| 6. Estado del Arte Accidentes de Tránsito | 45 |
| 7. Implementación del Modelo | 47 |
| 7.1. Obtención de los datos | 47 |
| 7.2. Estructura de los datos | 47 |
| 7.3. Procesamiento de los Datos | 50 |
| 7.3.1. Integración de los datos | 50 |
| 7.3.2. Selección y codificación de los datos | 51 |
| 7.3.3. Transformación de los datos | 53 |
| 7.3.4. Software y Hardware Utilizado | 53 |
| 7.4. Clasificación RNA con PSO | 54 |
| 7.5. IPSO (Improved Particle Swarm Optimization) | 56 |
| 7.5.1. Función de activación Tangente Hiperbólica- Tangente Hiperbólica. | 57 |
| 7.5.2. Función de activación Tangente Hiperbólica- Sigmoidal. | 59 |
| 7.5.3. Función de activación Sigmoidal- Sigmoidal. | 60 |
| 7.5.4. Función de activación Sigmoidal-Tangente Hiperbólica. | 62 |
| 7.6. DAPSO (PSO con Adaptación Dinámica) | 63 |
| 7.6.1. Función de activación Tangente Hiperbólica- Tangente Hiperbólica. | 64 |
| 7.6.2. Función de activación Tangente Hiperbólica- Sigmoidal. | 66 |
| 7.6.3. Función de activación Sigmoidal- Sigmoidal. | 67 |
| 7.6.4. Función de activación Sigmoidal-Tangente Hiperbólica. | 69 |
| 7.6.5. Comparativa entre las dos variantes PSO | 70 |
| 7.7. Función Softmax | 74 |

| | |
|---|-----------|
| 7.7.1. Softmax con IPSO | 75 |
| 7.7.2. Softmax con DAPSO | 76 |
| 7.7.3. Comparativa entre las dos variantes PSO | 78 |
| 7.8. Comparativa entre las mejores alternativas | 79 |
| 8. Conclusiones | 83 |
| Bibliografía | 84 |
| A. ANEXO | 87 |
| A.1. Plantilla SIEC2 | 87 |
| A.2. Prototipo Aplicación “Accidentes de Tránsito región Metropolitana” | 89 |
| A.3. Accidentes separados por Zona | 90 |

Lista de Abreviaturas

| | |
|---------|--|
| BP | Back Propagation |
| CONASET | Comisión Nacional de Seguridad de Tránsito |
| DAPSO | Particle Swarm Optimizer with Dynamic Adaptation |
| FN | Falsos Negativos |
| FP | Falsos Positivos |
| LDWPSO | Linearly Decreasing Inertia Weight PSO |
| LS-SVM | Least Squares Support Vector Machines |
| MLP | Multilayer Perceptron |
| OMS | Organización Mundial de la Salud |
| OPS | Organización Panamericana de la Salud |
| PNN | Polynomial Neural Network |
| PUCV | Pontificia Universidad Católica de Valparaíso |
| QPSO | Quantum Particle Swarm Optimization |
| RBFN | Radial Basis Function Neural Networks |
| ROC | Receiver Operating Characteristic |
| VN | Verdaderos Negativos |
| VP | Verdaderos Positivos |
| VPN | Valor Predictivo Negativo |
| VPP | Valor Predictivo Positivo |
| WNN | Wavelet Neural Network |

Indice de Figuras

| | |
|---|----|
| 3.1. Salto Sináptico | 6 |
| 3.2. Neurona Biológica | 7 |
| 3.3. Representación de una Neurona Básica | 8 |
| 3.4. Arquitectura básica de una red neuronal | 9 |
| 3.5. Red Neuronal Monocapa | 10 |
| 3.6. Red Neuronal Multicapa | 10 |
| 3.7. Tipos de Recurrencia | 11 |
| 3.8. Esquema de una red parcialmente recurrente | 12 |
| 3.9. Red de Jordan | 13 |
| 3.10. Red de Elman | 15 |
| 3.11. Aprendizaje Supervisado | 16 |
| 3.12. Aprendizaje No Supervisado | 17 |
| 3.13. Perceptrón Simple | 18 |
| 3.14. Separación de dos clases mediante un Perceptrón | 19 |
| 3.15. Arquitectura del Perceptrón multicapa | 20 |
| 3.16. Funciones de activación del Perceptrón multicapa | 22 |
| 3.17. Red Neuronal de base radial | 23 |
| 3.18. Arquitectura de una WNN | 25 |
| 3.19. Uso función de activación Softmax | 27 |
| 4.1. Enjambres en la Naturaleza | 28 |
| 4.2. Diagrama de Flujo PSO tradicional | 30 |
| 4.3. Distribución de las partículas: (a) en 1 iteración (b) en 10 iteraciones (c) en 40 iteración (d) en 60 iteraciones (e) en 80 iteraciones (f) en 100 iteraciones | 35 |
| 5.1. Curva ROC | 43 |
| 5.2. Tipos de Curva ROC | 44 |

| | |
|--|----|
| 7.1. Entidades base de datos CONASET | 47 |
| 7.2. Base de Datos Accidentes de Tránsito | 51 |
| 7.3. Region Metropolitana por Zonas | 55 |
| 7.4. Comparacion de Errores por función IPSO | 63 |
| 7.5. Comparación de Errores por función DAPSO | 70 |
| 7.6. Comparativa entre IPSO-DAPSO | 71 |
| 7.7. Comparativa del VPP | 72 |
| 7.8. Comparativa del VPN | 73 |
| 7.9. Comparativa entre IPSO-DAPSO Softmax (Error) | 78 |
| 7.10. Comparativa entre IPSO-DAPSO Softmax (VPP) | 78 |
| 7.11. Comparativa entre IPSO-DAPSO Softmax (VPN) | 79 |
| 7.12. Comparativa entre los 2 mejores resultados (Error) | 80 |
| 7.13. Comparativa entre los 2 mejores resultados (Especificidad) | 80 |
| 7.14. Comparativa entre los 2 mejores resultados (Sensibilidad) | 81 |
| 7.15. Comparativa entre los 2 mejores resultados (VPP) | 81 |
| 7.16. Comparativa entre los 2 mejores resultados (VPP) | 82 |

Indice de Tablas

| | |
|---|----|
| 5.1. Matriz de Confusión | 41 |
| 7.1. Atributos del Modelo | 52 |
| 7.2. Atributo Calidad y Estado de la Calzada | 52 |
| 7.3. Atributo Resultado | 52 |
| 7.4. Clases para la clasificación | 53 |
| 7.5. Parámetros de entrada Red | 56 |
| 7.6. Parámetros de entrada PSO | 56 |
| 7.7. Tangente Hiperbólica- Tangente Hiperbólica con IPSO (Entrenamiento) . . | 57 |
| 7.8. Tangente Hiperbólica- Tangente Hiperbólica con IPSO (Prueba) | 58 |
| 7.9. Tangente Hiperbólica- Sigmoidal con IPSO (Entrenamiento) | 59 |
| 7.10. Tangente Hiperbólica- Sigmoidal con IPSO (Pruebas) | 59 |
| 7.11. Sigmoidal- Sigmoidal con IPSO (Entrenamiento) | 60 |
| 7.12. Sigmoidal- Sigmoidal con IPSO (Pruebas) | 61 |
| 7.13. Sigmoidal- Tangente Hiperbólica con IPSO (Entrenamiento) | 62 |
| 7.14. Sigmoidal- Tangente Hiperbólica con IPSO (Pruebas) | 62 |
| 7.15. Tangente Hiperbólica- Tangente Hiperbólica con DAPSO (Entrenamiento) | 64 |
| 7.16. Tangente Hiperbólica- Tangente Hiperbólica con DAPSO (Prueba) | 65 |
| 7.17. Tangente Hiperbólica- Sigmoidal con DAPSO (Entrenamiento) | 66 |
| 7.18. Tangente Hiperbólica- Sigmoidal con DAPSO (Pruebas) | 66 |
| 7.19. Sigmoidal- Sigmoidal con DAPSO (Entrenamiento) | 67 |
| 7.20. Sigmoidal- Sigmoidal con DAPSO (Pruebas) | 68 |
| 7.21. Sigmoidal- Tangente Hiperbólica con DAPSO (Entrenamiento) | 69 |
| 7.22. Sigmoidal- Tangente Hiperbólica con DAPSO (Pruebas) | 69 |
| 7.23. Parámetros de entrada Red softmax | 74 |
| 7.24. Parámetros de entrada PSO con función softmax | 74 |

| | |
|--|----|
| 7.25. Sigmoidal- Softmax con DAPSO (Entrenamiento) | 75 |
| 7.26. Sigmoidal- Softmax con DAPSO (Pruebas) | 75 |
| 7.27. Sigmoidal- Softmax con DAPSO (Entrenamiento) | 76 |
| 7.28. Sigmoidal- Softmax con DAPSO (Pruebas) | 77 |

1 Introducción

Como se indica en el sitio web de la organización mundial de la salud (OMS), por medio de su oficina regional (OPS) en su informe sobre la situación mundial de la seguridad vial señala que: “En las Américas y el Caribe, cada año fallecieron alrededor de 130.000 personas, más de 1,2 millones sufrieron traumatismos, y cientos de miles resultan discapacitadas como consecuencia de colisiones, choques, volcaduras o atropellamientos en las vías de tránsito. Este problema se constituye como la novena causa de mortalidad general en la Región”. [2]. Los resultados muestran que los accidentes de tránsito siguen siendo un problema de salud pública, especialmente para países de ingresos bajos y medios. Los peatones, ciclistas y motociclistas representan casi la mitad de los muertos en las autopistas, destacando la necesidad por parte de los usuarios de adquirir mayor atención en los programas de seguridad vial [3].

En Chile la situación no es menor. Los valores por siniestros de tránsito en el año 2009 llegaron a un total de \$357.462.613.856 pesos chilenos, esto considerando los costos sociales y privados. Los accidentes de tránsito se han posicionado como una verdadera epidemia lo que ha llevado a alcanzar altas cifras. Esta situación se observa como un problema complejo de solucionar, según la Comisión Nacional de Seguridad de Tránsito (CONASET), solamente en 2009 se contabilizaron un total de 56.330 accidentes, siendo la colisión el tipo de siniestro más recurrente. La región metropolitana se transforma en la zona con más accidentes a nivel nacional y el 2009 contribuyó con un total de 21.675 siniestros, de los cuales 417 se tradujeron en pérdidas de vidas humanas y 15.512 se encontraron en la categoría de lesionados, desglosando ésta en graves (2.016 personas), menos graves (1.117 personas) y leves (12379 personas) [4].

Se han propuesto modelos predictivos que permiten anticiparse a las causas de éste tipo de incidentes, en los cuales se han obtenido resultados que hacen avanzar en éste ámbito, como por ejemplo [1]. Técnicas como la Minería de Datos (Data Mining), permite visualizar patrones y variables significativas que ayudan a identificar de mejor manera las condiciones en las que ocurren los accidentes, tomando en cuenta que éste tipo de evento contempla un sin número de variables, de las cuales se deben considerar sólo las más relevantes y así poder cuantificar la cantidad de accidentes que se producen en la actualidad.

La Minería de Datos cuenta con diversas técnicas que permiten el análisis de datos, dentro de las cuales se puede mencionar las Redes Neuronales Artificiales (RNA), que corresponden a un paradigma de aprendizaje y procesamiento automático inspirado en la forma en que funciona el sistema nervioso de los animales. Estos mecanismos son utilizados para la clasificación y predicción de datos, y mediante algoritmos evolutivos, como por ejemplo PSO, se podrá establecer patrones que permiten clasificar a las personas que se ven envueltas en un accidente de tránsito.

Actualmente, existen diversas investigaciones en accidentes de tránsito que utilizan técnicas como Redes Bayesianas, Perceptrón multicapa (MLP) y máquinas de soporte

vectorial (SVM), sin embargo, el uso de redes neuronales artificiales en clasificación de accidentes de tránsito es reducido y su enfoque depende plenamente de cómo el investigador construya el modelo de clasificación. El presente trabajo de investigación desarrolla y evalúa un modelo de clasificación utilizando redes neuronales artificiales con algoritmos de optimización por enjambre de partículas (PSO), lo que permite clasificar el grado de severidad (lesionadas o ilesas) de las personas involucradas.

El presente informe expone en primer lugar marco general escribiendo los objetivos; plan de trabajo; planteamiento del problema y el alcance del estudio. Continúa con el marco teórico el cual contiene una reseña histórica sobre las redes neuronales; redes neuronales biológicas; redes neuronales artificiales; topología de las redes; tipos de redes neuronales artificiales; aprendizaje y aplicación. Se presenta el modelo evolutivo propuesto en donde se menciona el algoritmo PSO; los parámetros; la topología de la vecindad local y global; conceptos referentes al algoritmo y su aplicación. Posteriormente se menciona el estado del arte de los accidentes de tránsito para finalizar con las conclusiones del trabajo realizado.

2 Marco General del Proyecto

2.1 Objetivo General

Desarrollar y evaluar un modelo de clasificación utilizando redes neuronales artificiales con algoritmos de optimización de enjambre de partículas (PSO) para clasificar el grado de severidad en el cual resultan las personas involucradas en accidentes de tránsito de la región metropolitana de Chile.

2.2 Objetivos Específicos

- Explicar el marco teórico de las redes neuronales artificiales y Optimización por Enjambre de Partículas (PSO).
- Diseñar la estructura del modelo clasificación y estimar sus parámetros usando PSO-Modificado.
- Evaluar y contrastar resultados de los modelos de clasificación propuestos.

2.3 Planteamiento del Problema

Los accidentes de tránsito son una constante en Chile, diariamente se registra una gran cantidad de eventos de este tipo a lo largo del país, siendo un riesgo constante para las personas que circulan por calles y carreteras. Existen diversas maneras para poder identificar las causas en las que puede producirse un accidente, sin embargo, al realizar ésta investigación se ha observado que en Chile existen escasos trabajos referentes a la clasificación de la severidad en accidentes de tránsito. Dentro de los trabajos que se han realizado recientemente se puede mencionar [1], donde se utilizaron técnicas como SVM y LS-SVM (Least Squares Support Vector Machines) .

2.4 Alcance del Estudio

En vista de los estudios realizados por la Escuela de Ingeniería en Transporte de la PUCV, el estudio se enmarca en accidentes de tránsito de la zona de Santiago. Es por ello que se crea un modelo de clasificación para determinar la severidad de los accidentes de tránsito, clasificándolos en lesionados o ilesos.

3 Marco Teórico

3.1 Reseña Histórica

La historia de las redes neuronales se remonta hacia el año 1936 en donde Alan Turing fue el primero en estudiar el cerebro como una forma de ver el mundo de la computación. Sin embargo, los primeros teóricos que concibieron los fundamentos de la computación neuronal fueron: el neurofisiólogo Warren McCulloch y el matemático Walter Pitts quienes, en 1943, describen una lógica de cálculo de redes neuronales que une estudios de neurofisiología y lógica matemática. Su modelo formal de una neurona sigue la ley “todo o nada”. Con un suficiente número de estas simples unidades, correctas conexiones sinápticas y operando de manera síncrona, demostraron que una neurona constituida de esta manera puede, en principio, calcular cualquier función computable.

En 1949 Donald Hebb, fue el primero en explicar los procesos del aprendizaje desde un punto de vista psicológico, desarrollando una regla de cómo el aprendizaje ocurría, planteaba que el aprendizaje se daba cuando ocurrían ciertos cambios en una neurona activando éstas. También propone que la conectividad del cerebro cambia continuamente conforme un organismo aprende cosas nuevas, creándose asociaciones neuronales con estos cambios. Los trabajos de Hebb formaron las bases de la Teoría de las Redes Neuronales.

En 1950 Karl Lashley resume su investigación de 30 años y destaca que el proceso de aprendizaje es un proceso distribuido y no local a una determinada área del cerebro.

Haibt y Duda en 1956 realizan una de las primeras simulaciones computacionales para probar una bien formulada teoría neuronal basándose en el postulado de aprendizaje de Hebb. Este mismo año, Uttley, demuestra que una red neuronal con sinapsis modificables puede aprender a clasificar un simple conjunto de patrones binarios en sus respectivas clases.

Quince años luego de la publicación de McCulloch y Pitts un nuevo acercamiento al problema de reconocimiento de patrones fue introducido por Rosenblatt (1958) en su trabajo sobre el perceptrón. Éste constaba de 2 niveles y los pesos se ajustaban en proporción al error entre las salidas deseadas y obtenidas. Dos años después Widrow y Hoff introdujeron el algoritmo “least mean-square” y lo usaron para formular el ADALINE (Adaptative Linear Element). Éste último, se diferencia del perceptrón en el proceso de entrenamiento.

En 1969 se produjo la “muerte abrupta” de las redes neuronales cuando Minsky y Papera probaron matemáticamente que el Perceptrón no era capaz de resolver problemas tan fáciles como el aprendizaje de una función no lineal. Esto demuestra una gran debilidad, dado que las funciones no-lineales son ampliamente utilizadas en computación y en problemas del mundo real.

En 1982 Hopfield publica un trabajo clave para el resurgimiento de las redes neuronales. En él, desarrolla la idea del uso de una función de energía para comprender la

dinámica de una red neuronal recurrente con uniones sinápticas simétricas. El principal uso de estas redes ha sido como memorias y como instrumento para resolver problemas de optimización como el problema del viajante.

En 1986 Rumelhart, Hinton y Williams, desarrollan el algoritmo de aprendizaje de retro propagación (backpropagation) para redes neuronales multicapa, dando una serie de ejemplos en los cuales se demuestra el potencial del método desarrollado.

A partir de ese año, el número de trabajos sobre redes neuronales ha aumentado exponencialmente apareciendo un gran número de aportaciones tanto a los métodos de aprendizaje como a las arquitecturas y aplicaciones de las redes neuronales. Se podría destacar entre todas estas aportaciones los trabajos de Broomhead y Lowe y el de Poggio y Girosi sobre el diseño de redes neuronales en capas usan RBF (Radial Basis Function).

En la actualidad, son numerosos los trabajos que se realizan y publican cada año, las aplicaciones nuevas que surgen (sobre todo en el área de control) y las empresas que lanzan al mercado productos nuevos, tanto hardware como software (sobre todo para simulación).

3.2 Redes Neuronales Biológicas

El aparato de comunicación neuronal de los animales y del hombre formado por el sistema nervioso y hormonal tiene la misión de recibir la información, transmitirla y elaborarla, en parte también almacenarla y enviarla de nuevo en forma elaborada. El sistema de comunicación neuronal se compone de tres partes.

- Los receptores, que se encuentran en las células sensoriales, recogen la información en forma de estímulos desde el ambiente o del interior del organismo.
- El sistema nervioso que recibe la información, la elabora, la almacena y la envía de forma elaborada a los órganos efectores y a otras zonas del sistema nervioso.
- Órganos diana o efectores reciben la información y la interpretan en forma de acciones motoras, hormonales, etc.

El elemento estructural y funcional más esencial en el sistema de comunicación neuronal es la neurona. La mayoría de estas utilizan sus productos de secreción como señales químicas para la transmisión de la información, enviando ésta a través de distintas neuronas formando redes, las cuales se elabora y permiten almacenar la información.

Una parte de la neurona esta en relación con los receptores, a través de ellos llega la información proveniente del exterior o del interior del organismo a las redes neuronales. Otra parte conduce las informaciones elaboradas en forma de órdenes hacia los efectores. Una de las prolongaciones es la encargada de la conducción de los impulsos, la cual se denomina Axón.

La mayoría de las neuronas codifican sus salidas como una serie de breves pulsos periódicos, llamados potenciales de acción, que se originan cercanos al soma de la célula y se propagan a través del axón. Luego, éste pulso llega a las sinapsis y de ahí a las dendritas de la neurona siguiente. Una sinapsis es una interconexión entre dos neuronas, un dibujo esquemático de ella se incluye en la figura 3.1, en ella el botón sináptico corresponde al término del axón de una neurona pre-sináptica, y la dendrita es la correspondiente a una neurona post-sináptica.

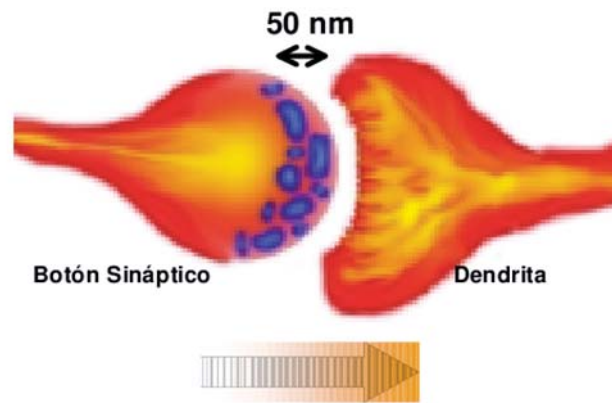


Figura 3.1: Salto Sináptico

En la figura 3.2 se aprecia que la neurona se encuentra formada por un soma o cuerpo celular que en su interior contiene el núcleo. Cuenta con el axón, el cual se encuentra envuelto en la Vaina de mielina, que es la encargada de propagar una serie de impulsos electro-químicos. Además, las ramificaciones de entrada con las que cuenta la neurona se conocen como dendritas, estas difunden la señal al interior de la neurona y los terminales axónicos permitiendo la interacción con las dendritas de la siguiente neurona para la entrega de la señal. El mecanismo es el siguiente [5]:

- La sinapsis recoge información electro-química de las neuronas vecinas por medio de las dendritas.
- La información llega al núcleo, en donde es procesada para generar la respuesta que viajará por el Axón.
- Luego, la señal propagada por el axón se ramifica en los terminales axónicos y llega a las dendritas de otras neuronas en lo que se denomina Sinapsis.

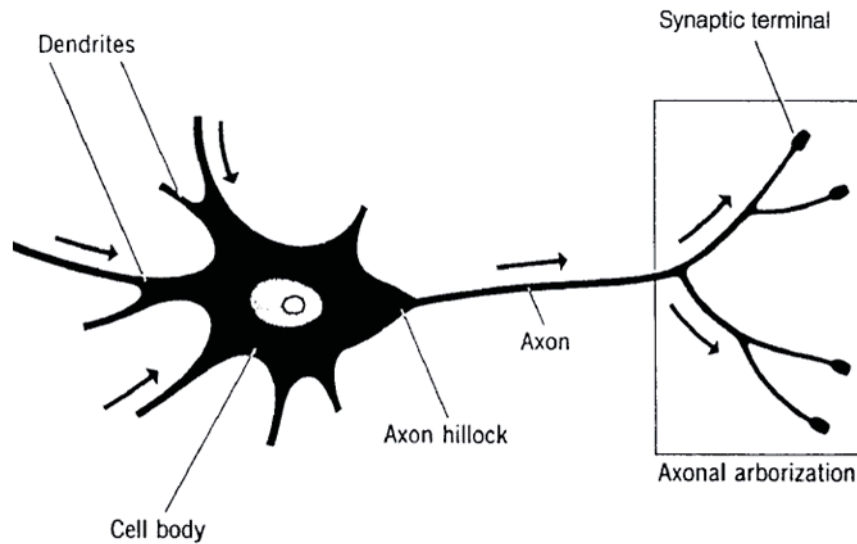


Figura 3.2: Neurona Biológica

La capacidad de una red neuronal de aprender y de adaptarse a su ambiente puede lograrse mediante dos mecanismos: la creación de nuevas conexiones sinápticas y la modificación de las conexiones sinápticas existentes [6]. Esto permite a la red modificar su estructura, además de otorgarle la cualidad de ser tolerante a fallos, dado que se pueden realizar más de un camino para que un impulso neuronal viaje a través de la red y provoque la respuesta adecuada. Además la comunicación entre neuronas se basa en el paso de mensajes pequeños, por lo que la información crítica no se transmite directamente, mas se captura y distribuye entre las interconexiones.

3.3 Redes Neuronales Artificiales (RNA).

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

Según [5] el modelo de una neurona presenta tres elementos básicos:

- Una serie de sinapsis, donde cada una está caracterizada por un peso el cual puede tomar valores negativos como positivos.
- Un sumador que suma las señales de entrada pesadas por la respectiva sinapsis de la neurona.

- Una función de activación para limitar la amplitud de la salida de la neurona. Cabe destacar que ésta corresponde a una función no lineal que normaliza el estado interno de la neurona a un intervalo cerrado $[0,1]$ o $[-1,1]$ según sea el caso. Ejemplos de función de activación serían: la función escalón, sigmoideal, gaussiana, polinomial, entre otras.

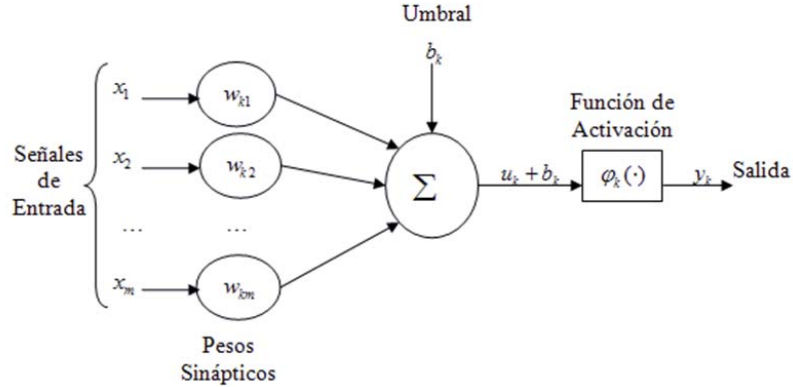


Figura 3.3: Representación de una Neurona Básica

La figura 3.3 muestra la estructura de una neurona básica, la que incluye las señales de entradas, los pesos sinápticos, un valor numérico conocido como umbral simbolizado por b_k , una función de activación φ_k lo que lleva a la salida de la neurona. El umbral es a un valor opcional que puede aumentar o disminuir el estado interno de la neurona dependiendo si este es positivo o negativo respectivamente. En términos matemáticos, una neurona k puede ser descrita por medio de las ecuaciones 3.3.1 y 3.3.2.

$$u_k = \sum_{j=1}^m w_{kj} x_j \quad (3.3.1)$$

$$y_k = \varphi(u_k + b_k) \quad (3.3.2)$$

Donde x_1, x_2, \dots, x_m corresponden a las señales de entrada a la neurona artificial, $w_{k1}, w_{k2}, \dots, w_{km}$ a los pesos sinápticos de la neurona k , u_k es la salida obtenida de la combinación lineal dada la señal de entrada. Luego, b_k corresponde al umbral, φ_k corresponde a la función de activación e y_k es la señal de salida de la neurona k .

Las neuronas se organizan en capas, las cuales corresponden a un conjunto de neuronas cuya entrada se refiere a la salida de otra capa o a una entrada desde el exterior, y su salida hace alusión a la entrada de una capa siguiente o a la salida de la red neuronal. En la figura 3.4 se muestra de forma gráfica dicha organización.

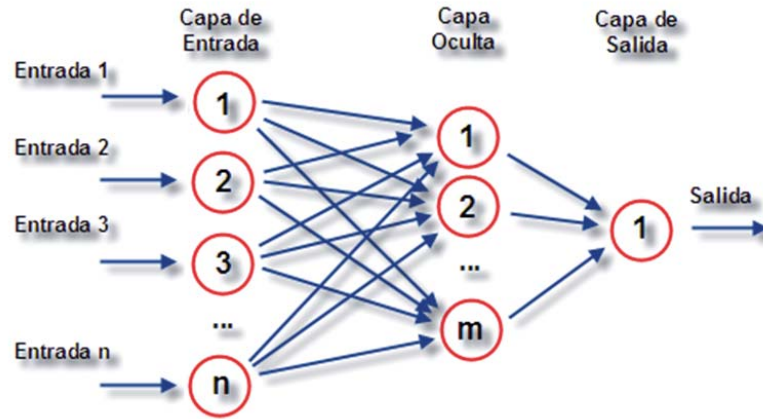


Figura 3.4: Arquitectura básica de una red neuronal

Ya definido el concepto de capa y en base a la figura 3.4 se pueden definir las distintas capas que forman parte de una red neuronal. La capa de entrada corresponde al primer nivel que recibe los valores de patrones (representados como vectores) que sirven como entrada a la red. A continuación, hay una serie de capas intermedias conocidas como capas ocultas, cuyas unidades corresponden a rasgos particulares y pueden corresponder a 1 o más niveles. Finalmente, la última corresponde a la capa de salida, la cual sirve de salida de toda la red.

Cada interconexión entre las distintas neuronas actúa como una ruta de comunicación, a través de la cual viajan los valores numéricos de una neurona a otra. Así, la Red Neuronal Artificial podría definirse como un grafo cuyos nodos poseen características idénticas y cuya información se transporta a través de los arcos.

3.4 Topologías de Redes Neuronales.

La topología o arquitectura de una red neuronal consiste en la organización y disposición de las neuronas que la conforman. Estas estructuras o modelos conexionistas se encuentran fuertemente ligados al algoritmo de aprendizaje utilizado en su elaboración y se pueden clasificar de diferentes formas dependiendo del criterio de uso.

3.4.1 Según su número de capas

- Redes neuronales monocapa:** Son redes que se forman por una única capa. Para unirse las neuronas crean conexiones laterales para su conexión con otras neuronas de su capa. Las redes más representativas son la red de Hopfield, la red brain-state-in-a-box o memoria asociativa y las maquinas estocásticas de Botzmann y Cauchy. Entre las redes neuronales monocapa, existen algunas que permiten que las neuronas tengan conexiones a sí mismas y se denominan autorecurrentes. Estas redes

se utilizan generalmente en relacionadas con lo que se conoce como autoasociación (consiste en la regeneración de la información de entrada a la red que se presenta de forma incompleta o distorsionada). Un ejemplo se muestra en la figura 3.5

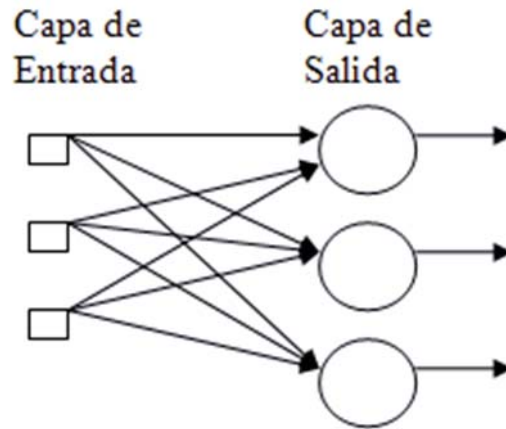


Figura 3.5: Red Neuronal Monocapa

- **Redes neuronales multicapa:** Son aquellas que disponen de un conjunto de neuronas agrupadas en varios niveles o capas, como se muestra en la figura 3.6. Es una generalización de la anterior existiendo un conjunto de capas intermedias, entre la capa de entrada y salida, conocidas como capa oculta. Estas redes se pueden a su vez clasificar dependiendo de la forma en que se conectan sus capas, véase 3.4.3.

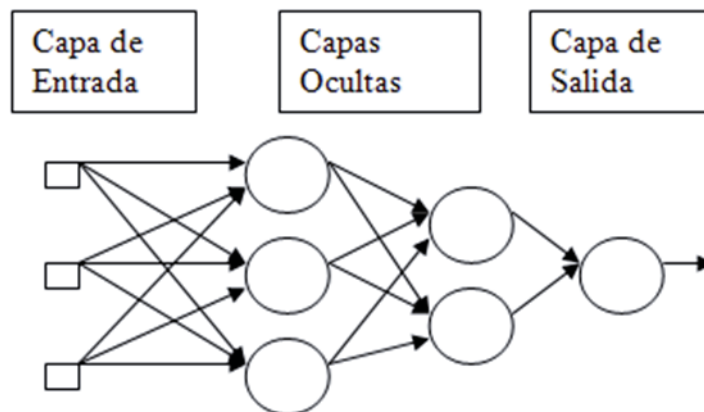


Figura 3.6: Red Neuronal Multicapa

3.4.2 Según el tipo de conexiones

- **Redes neuronales no recurrentes:** También conocidas como redes feed-forward, son aquellas donde los nodos se conectan sólo a capas posteriores, sin generar ciclos de información. La figura 3.6 correspondiente a la red neuronal multicapa también corresponde a una red neuronal no recurrente.
- **Redes neuronales recurrentes:** También conocida como redes de retro-propagación (feedback), son aquellas donde las neuronas de una capa se conectan con neuronas de distintas capas (anteriores y/o posteriores), neuronas de la misma capa, o incluso con la misma neurona. Se detallara en el apartado 3.5.

3.4.3 Según el grado de conexión

- **Redes neuronales totalmente conectadas:** En éstas redes todas las neuronas de una capa se encuentran conectadas con las neuronas de la capa siguiente y con las de la capa anterior, esto en caso de ser una red neurona recurrente.
- **Redes neuronales parcialmente conectadas:** En este caso no se da la conexión total entre neuronas de diferentes capas.

3.5 Redes Neuronales Recurrentes

También conocida como redes de retro-propagación (feedback), son aquellas donde las neuronas de una capa se conectan con neuronas de distintas capas (anteriores y/o posteriores), neuronas de la misma capa, o incluso con la misma neurona. En la figura 3.7 se muestran los tipos de conexiones de las redes recurrentes. a) Conexión con la misma neurona, b) Conexión con neuronas de la misma capa y c) Conexión con neuronas posteriores y anteriores.

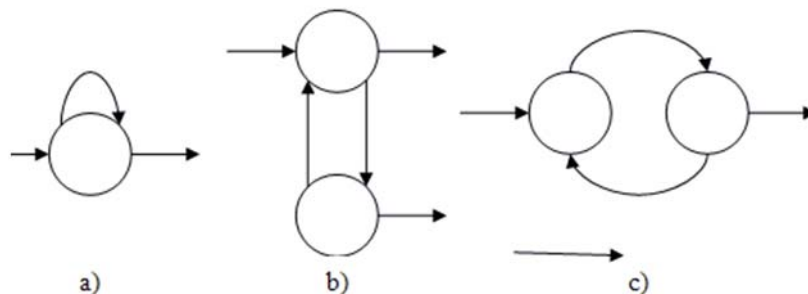


Figura 3.7: Tipos de Recurrencia

La activación ya no depende sólo de activaciones ocurridas en la capa anterior, sino que también depende del estado o activación de cualquier otra neurona que se encuentre

conectada a ella, o incluso su propia activación. Por lo tanto, es necesario incluir la variable tiempo en la activación de la neurona, la cual viene dada por la ecuación 3.5.1.

$$a_i(t + 1) = f_i\left(\sum_j w_{ij}a_j(t)\right) \quad (3.5.1)$$

Donde el índice j varía en el conjunto de todas las neuronas que se encuentran conectadas a la neurona i . La presencia de $a_i(t + 1)$ en las activaciones de las neuronas recurrentes hace que éstas posean un comportamiento dinámico o temporal.

Pese a la complejidad con la que cuentan las redes neuronales recurrentes y la dificultad, en ocasiones, para realizar su “aprendizaje”, éstas han sido utilizadas para abordar distintos tipos de problemas. Existen distintos tipos de redes neuronales recurrentes, dentro de las más reconocidas se encuentran las redes de Elman y Jordan, definidas por los investigadores de mismo nombre.

Cuando se habla de redes neuronales parcialmente recurrentes, existe un grupo de neuronas especiales ubicadas en la capa de entrada, conocidas como neuronas de contexto o neuronas de estado. Por lo anterior se pueden distinguir dos tipos de neurona en la capa de entrada de la red, las que actúan de entrada y reciben las señales desde el exterior, y las neuronas de contexto, que actúan como neuronas receptoras de las conexiones recurrentes y funcionan como una memoria de la red en la que se almacenan las activaciones de las neuronas de una cierta capa de la red en el instante o iteración anterior. En la figura 3.8 se muestran los dos tipos de neuronas.

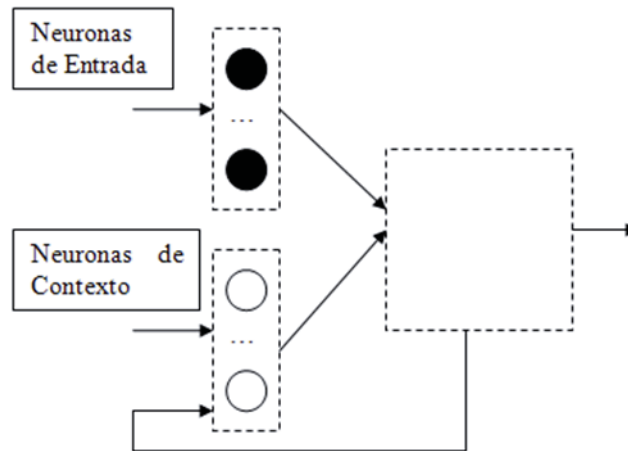


Figura 3.8: Esquema de una red parcialmente recurrente

La diferencia fundamentalmente que existe entre las redes de Jordan y Elman se basa en que en las redes de Jordan las neuronas de contexto reciben copias de las neuronas de

la capa de salida y de sí mismas, y en las redes de Elman las neuronas de contexto reciben copias de las neuronas ocultas.

3.5.1 Red de Jordan.

Propuesto por Jordan en 1986 [7, 8], se caracteriza por que las neuronas de contexto reciben una copia de las neuronas de salida de la red y de ellas mismas. Además, las conexiones recurrentes de la capa de salida a las neuronas de contexto llevan un parámetro asociado, el cual toma un valor constante positivo y menos que 1, como se muestra en la figura 3.9.

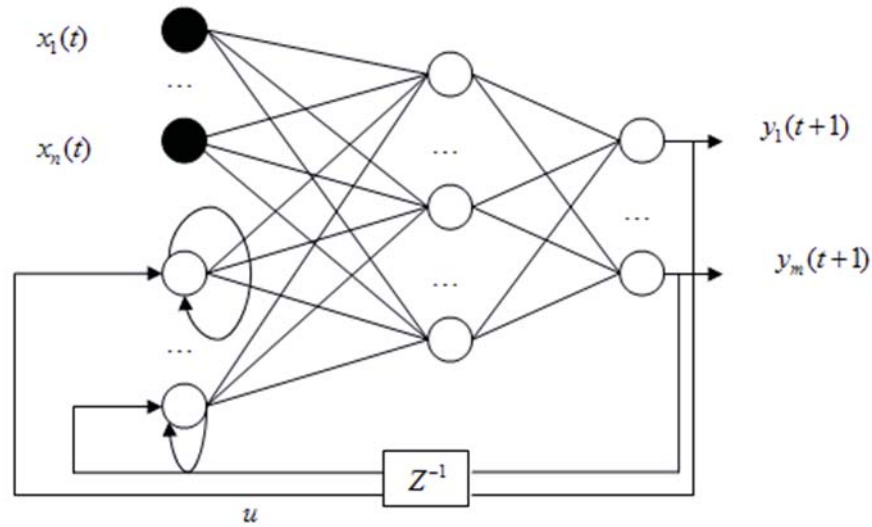


Figura 3.9: Red de Jordan

La activación de cada neurona de contexto en el instante de tiempo o iteración t , denotada por $c_i(t)$, viene dada por las ecuaciones 3.5.2 y 3.5.3.

$$c_i(t) = \mu c_i(t-1) + y_i(t-1) \quad \text{Para } i = 1, 2, \dots, m \quad (3.5.2)$$

$$c_i(t-1) = \mu c_i(t-2) + y_i(t-2) \quad (3.5.3)$$

Donde m corresponde al número de salidas de la red, e $y(t-1) = (y_1(t-1), y_2(t-1), \dots, y_m(t-1))$ corresponde al vector de salida de la red en el instante $t-1$.

El resto de las activaciones se calculan como en una red multicapa con conexiones hacia adelante, la diferencia radica en el vector de entrada en el tiempo t , dado que corresponderá a la concatenación de las activaciones de las neuronas de entrada junto con las de contexto. Éste vector $u(t)$ se define en la ecuación 3.5.4.

$$u(t) = (x_1(t), \dots, x_n(t), c_1(t), \dots, c_m(t)) \quad (3.5.4)$$

Donde $x_i(t)$ corresponde a la señal que la red recibe desde el exterior.

Las neuronas de contexto tienen una función de activación lineal, como se muestra en la ecuación 3.5.2. Esto hace que la activación de las neuronas de contexto se pueda desarrollar en el tiempo como muestran las ecuaciones 3.5.5 y 3.5.6.

$$c_i(t) = \mu^2 c_i(t-2) + \mu y_i(t-2) + y_i(t-1) \quad (3.5.5)$$

$$= \mu^{t-2} y_i(1) + \mu^{t-3} y_i(2) + \dots + \mu y_i(t-2) + y_i(t-1) \quad (3.5.6)$$

Y se obtiene la expresión que se muestra en la ecuación 3.5.7.

$$c_i(t) = \sum_{j=1}^{t-1} \mu^{j-1} y_i(t-j) \quad \text{Para } i = 1, 2, \dots, m \quad (3.5.7)$$

En esta expresión se observa que las neuronas de contexto acumulan las salidas de la red en todos los instantes anteriores de tiempo, y el parámetro determina la sensibilidad de estas neuronas de retener dicha información. Así, valores cercanos a 1 permitirán memorizar estados muy lejanos al tiempo actual, y a medida que este valor tiende a 0, dichos estados tienen una menor presencia en la activación actual de las neuronas de contexto.

3.5.2 Red de Elman.

La red de Elman [9] se caracteriza por que las neuronas de contexto, reciben copias desde neuronas ubicadas en la capa oculta de la red, tal como se muestra en la figura 3.10, y no traen asociado ningún tipo de parámetro (como en el caso de la red de Jordan). Por lo tanto, existirán tantas neuronas de contexto como neuronas ocultas tenga la red.

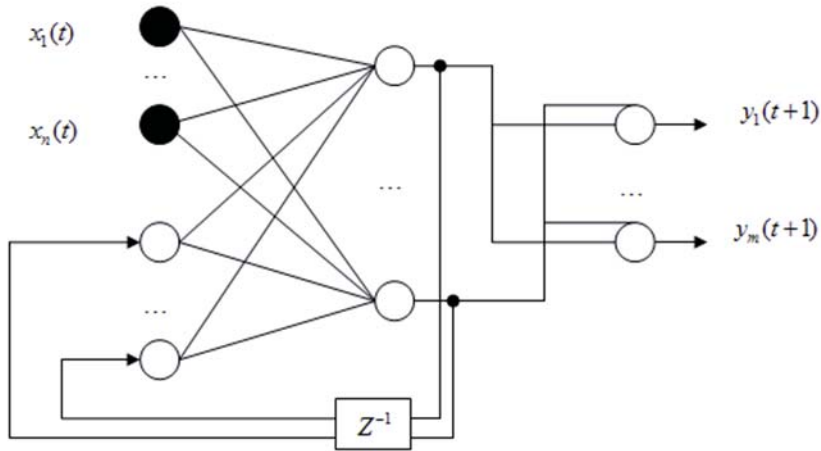


Figura 3.10: Red de Elman

La activación de la neurona de contexto está dada por la ecuación 3.5.8.

$$c_i(t) = a_i(t - 1) \quad \text{Para } i = 1, 2, \dots, r \quad (3.5.8)$$

Donde r corresponde al número de neuronas ocultas de la red de Elman, y $a_i(t - 1)$ corresponde a las activaciones de las neuronas ocultas en el instante $t - 1$.

Al igual que las redes de Jordan, el resto de las activaciones se calculan como en una red multicapa con conexiones hacia adelante, formándose como entrada a la red el vector $u(t)$, definido previamente en la ecuación 3.5.4.

3.6 Aprendizaje de una RNA.

Una de las principales características que poseen las redes neuronales artificiales es la capacidad de “aprender” del ambiente y mejorar su rendimiento a través del aprendizaje. El aprendizaje consiste en algoritmos basados en formulas matemáticas, que usando técnicas como minimización del error o la optimización de alguna ”función de energía”, modificando el valor de los pesos sinápticos en función de las entradas disponibles y con ello optimizan la respuesta de la red a las salidas que se desea.

El aprendizaje se basa en el entrenamiento de la red con patrones, que usualmente son llamados patrones de muestra o entrenamiento. El proceso usual del algoritmo es que la red ejecuta los patrones iterativamente, cambiando los pesos de las sinapsis, hasta que convergen a un conjunto de pesos óptimos que representan a los patrones, entonces se mostrará una respuesta satisfactoria para esos patrones. Sus pesos sinápticos se ajustan para dar respuestas correctas al conjunto de patrones de entrenamiento.

Dentro del aprendizaje se destacan tres tipos, aprendizaje supervisado; aprendizaje no supervisado y aprendizaje híbrido, los cuales pasarán a ser detallados a continuación.

3.6.1 Aprendizaje Supervisado.

Se caracteriza por ser un tipo de aprendizaje que se encuentra controlado por un agente externo (supervisor, maestro) el cual determinan cuales debieran ser las respuestas entregadas por la red dada una entrada determinada. Este agente externo controla la salida y en caso de no corresponder a la deseada, se procede a modificar los pesos de las conexiones, con el fin de que la salida obtenida se aproxime lo más posible a la deseada. véase figura 3.11.

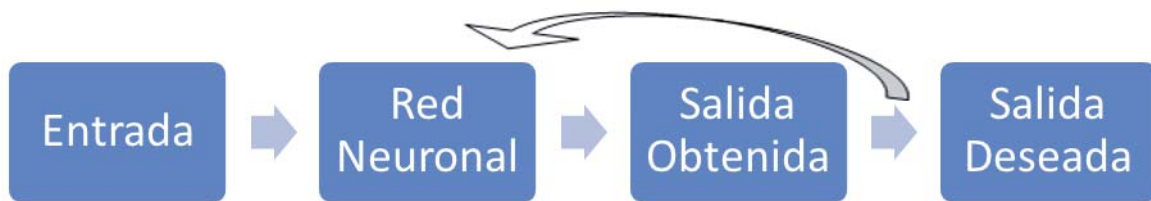


Figura 3.11: Aprendizaje Supervisado

Se suelen considerar 3 formas de llevar a cabo este tipo de aprendizaje, dando lugar a los siguientes aprendizajes supervisados.

- **Aprendizaje por corrección de error.** Consiste en ajustar los pesos de la red en función de las diferencias entre los valores deseados y los valores obtenidos de la salida de la red.
- **Aprendizaje por refuerzo.** Tipo de aprendizaje más lento que el anterior, en el cual el supervisor o maestro toma un papel de crítico dado que indica mediante una señal de refuerzo si la salida de la red obtenida se ajusta a la deseada (éxito = +1 o fracaso = -1), y en función de ello se ajustan los pesos basándose en un mecanismo de probabilidades.
- **Aprendizaje estocástico.** Asocia a la red con un sólido físico que tiene cierto estado energético, de tal forma que un estado de mínima energía correspondería a una situación en que los pesos de las conexiones consiguen que su funcionamiento sea el que más se ajusta al objetivo deseado. De esta manera, el aprendizaje consistiría en asignar valores aleatorios a los pesos y determinar la energía de la red. Si el valor energético es menor después del cambio (comportamiento se acerca al deseado) se acepta el cambio, de lo contrario se aceptaría el cambio en función de una determinada y preestablecida función de probabilidades.

3.6.2 Aprendizaje No Supervisado.

Este tipo de aprendizaje se caracteriza por que la red no recibe ninguna información por parte del entorno que le indique si la salida generada en respuesta de una determinada entrada es o no es correcta. Estas redes deben encontrar características, regularidades, correlaciones o categorías que se puedan establecer entre los datos que se presentan en su entrada. Existen varias posibilidades en cuanto a la interpretación de la salida de estas redes, que depende de su estructura y algoritmo de aprendizaje utilizado, véase figura 3.12.

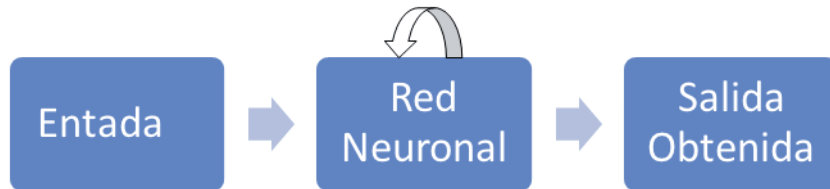


Figura 3.12: Aprendizaje No Supervisado

- **Aprendizaje Hebbiano.** Sugerida por Donald Hebb en 1949, a partir de estudios con neuronas reales. Corresponde a una regla de aprendizaje que pretende medir la familiaridad o extraer características de los datos de entrada. Consiste en que si 2 neuronas toman el mismo estado simultáneamente (ambas activas o ambas inactivas, dado que el origen de la regla es en base a la neurona biológica clásica), el peso de la conexión entre ambas se incrementa.
- **Aprendizaje Competitivo y Comparativo.** Si un patrón nuevo se determina que pertenece a una clase previamente reconocida, entonces la inclusión de este nuevo patrón a esta clase matizara la representación de la misma. En caso contrario, la estructura y los pesos de la red serán ajustados para reconocer la nueva clase.

3.6.3 Aprendizaje Híbrido

El aprendizaje híbrido corresponde a una mezcla de los tipos de aprendizaje: el aprendizaje supervisado y el aprendizaje no supervisado, descritos en las secciones 3.6.1 y 3.6.2

3.7 Perceptrón

Es un tipo de red neuronal artificial desarrollado por Frank Rosenblatt. El Perceptrón fue diseñado como un sistema capaz de realizar tareas de clasificación de clases de manera automática. Su arquitectura consta de una estructura mono capa en la cual hay un conjunto

de células de entrada y una o más células de salida, donde cada célula de entrada está conectada con todas las células de salida.

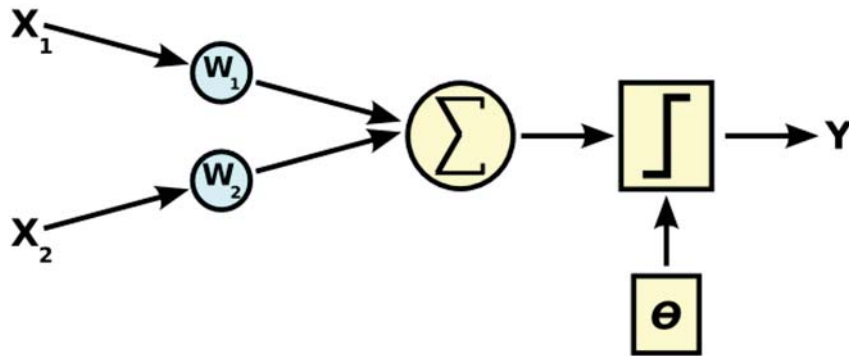


Figura 3.13: Perceptrón Simple

Como se aprecia en figura 3.13 , la red consta de entradas x_1 y x_2 , una salida "y" y un umbral θ , el cual es utilizado como un factor de comparación para producir la salida.

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, matemáticamente se representa con la ecuación 3.7.1.

$$y' = \sum_{i=1}^n w_i \cdot x_i \quad (3.7.1)$$

En el Perceptrón la función de salida es una función escalón que dependerá del umbral, se detalla en las ecuaciones 3.7.2 y 3.7.3.

$$y = F(y', \theta) \quad (3.7.2)$$

$$F(s, \theta) = \begin{cases} 1 & \text{Si } s > \theta \\ -1 & \text{en otro caso} \end{cases} \quad (3.7.3)$$

Según pase θ al lado contrario de la ecuación, es posible obtener la ecuación 3.7.4.

$$y' = \sum_{i=1}^n w_i \cdot x_i + \theta \quad (3.7.4)$$

$$F(s) = \begin{cases} 1 & \text{Si } s > 0 \\ -1 & \text{en otro caso} \end{cases} \quad (3.7.5)$$

Con las ecuaciones 3.7.4 y 3.7.5 ya es posible determinar fácilmente la forma en que se puede clasificar mediante el modelo del Perceptrón.

Como se puede apreciar en la figura 3.14, si se produce una salida -1, la entrada pertenece a la clase B. Si se produce una salida 1, la entrada pertenece a la clase A.

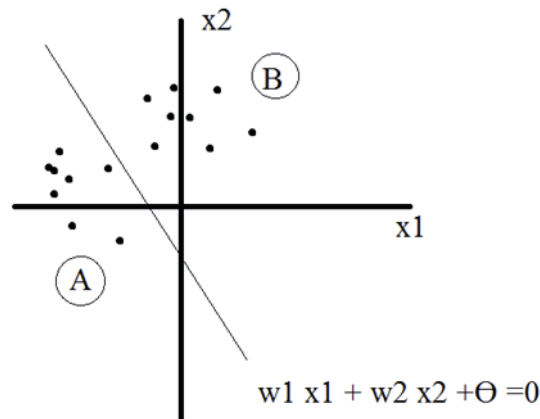


Figura 3.14: Separación de dos clases mediante un Perceptrón

Este ejemplo sólo es para dos dimensiones, viéndolo desde n dimensiones se podría expresar de la siguiente forma [5]:

“Dados conjuntos de puntos en R^n : $A = (a_1, \dots, a_{na})$ y $B = (b_1, \dots, b_{nb})$, obtener el hiperplano: $w_1 a_1 + \dots + w_n a_n + \theta = 0$, de tal forma que”:

$$\forall \vec{a} \in A : w_1 \times a_1 + w_2 \times a_2 + \dots + w_n \times a_n \quad (3.7.6)$$

$$\forall \vec{b} \in B : w_1 \times b_1 + w_2 \times b_2 + \dots + w_n \times b_n \quad (3.7.7)$$

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

$$\forall \vec{a} \in A : y(\vec{a}) = 1 \quad (3.7.8)$$

$$\forall \vec{b} \in B : y(\vec{b}) = -1 \quad (3.7.9)$$

Con lo cual ya no es una recta la que separa los puntos, sino un hiperplano.

Para el modelo del Perceptrón se utilizan iteraciones de modo que los pesos de las conexiones se modifiquen paulatinamente.

3.8 Perceptrón Multicapa (MLP)

El Perceptrón multicapa (MLP, según sus siglas en inglés Multi Layer perceptron) o red multicapa con conexiones hacia adelante, el cual se muestra en la figura 3.15, surge como consecuencia de las limitantes presentadas por el Perceptrón simple en lo relacionado al problema de separabilidad no lineal.

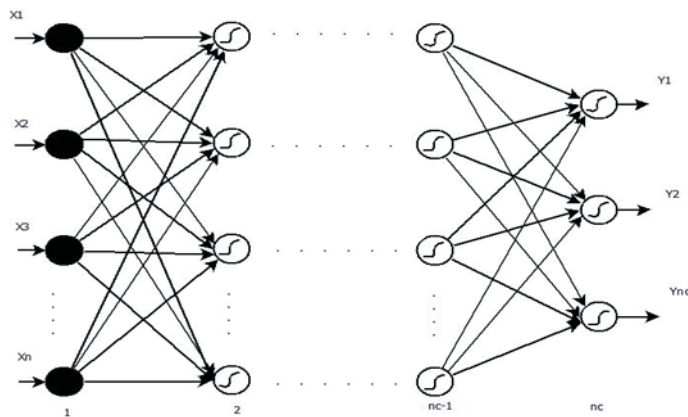


Figura 3.15: Arquitectura del Perceptrón multicapa

Esta arquitectura se caracteriza por agrupar neuronas por capas, dentro de las cuales se encuentran:

- **Capa de entrada:** en esta capa se encuentran las neuronas encargadas de recibir las señales o patrones del exterior para luego propagarlos a la siguiente capa que es la primera capa oculta.
- **Capas ocultas:** las capas ocultas son las capas que se encuentran entre la capa de entrada y la capa de salida, pudiendo haber una o más capas ocultas. Las neuronas de las capas ocultas realizan un procesamiento no lineal de los patrones recibidos.
- **Capa de salida:** encargada de proporcionar al exterior la respuesta de la red para cada uno de los patrones de entrada.

Por lo general cada neurona de la capa t está conectada con todas las neuronas de la capa $t + 1$, estas conexiones están dirigidas hacia adelante, razón por la cual se les conoce como redes feed forward (alimentadas hacia adelante), además estas conexiones tienen un determinado peso que se va ajustando en la etapa de entrenamiento. Todas las neuronas

de la red tienen un umbral que se trata como una conexión más a la neurona, su valor es constante e igual a uno.

Ahora para entender más en profundidad el Perceptrón multicapa se debe considerar:

- C capas, con $C - 2$ capas ocultas y n_c neuronas en la capa c , para $c = 1, 2, \dots, C$.
- $W^c = (w_{ij}^c)$ la matriz de pesos asociada a las conexiones de la capa c a las $c + 1$ para $c = 1, 2, \dots, C + 1$, con w_{ij}^c es el peso de la conexión de la neurona i de la capa c a la neurona j de la capa $c + 1$.
- $U^c = (u_i^c)$ es el vector de umbrales de las neuronas de la capa c para $c = 2, \dots, C$.
- a_i^c es la activación de la neurona i de la capa c . Esta se calcula de tres formas según sea el caso.

La activación de la capa de entrada para $i = 1, 2, 3, \dots, n_1$ se obtiene según la ecuación 3.8.1.

$$a_i^1 = x_i \quad (3.8.1)$$

La activación de las neuronas de la capa oculta para $i = 1, 2, 3, \dots, n_c$ y $c = 2, 3, \dots, C - 1$ se obtiene mediante la ecuación 3.8.2.

$$a_i^c = f \left(\sum_{j=1}^{n_{c-1}} w_{ji}^{c-1} \times a_j^{c-1} + u_i^c \right) \quad (3.8.2)$$

La activación de las neuronas de la capa de salida para $i = 1, 2, 3, \dots, n_C$ se obtiene por medio de la ecuación 3.8.3.

$$y_i = a_i^c = f \left(\sum_{j=1}^{n_{c-1}} w_{ji}^{c-1} \times a_j^{c-1} + u_i^c \right) \quad (3.8.3)$$

Las funciones de activación más utilizadas se pueden apreciar en la figura 3.16, estas son:

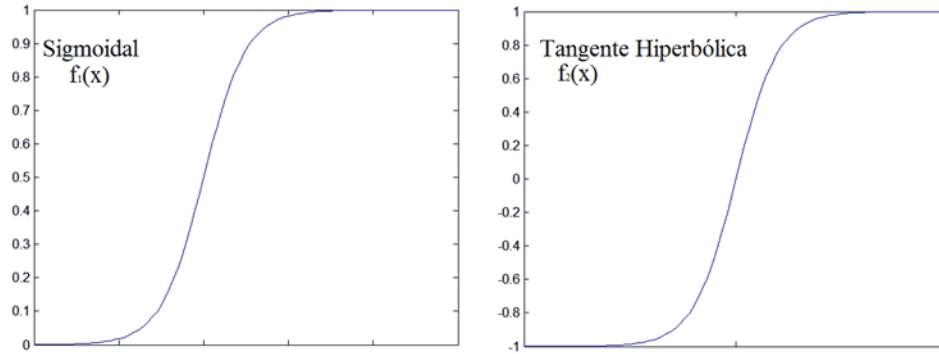


Figura 3.16: Funciones de activación del Perceptrón multicapa

- La función sigmoideal (ecuación 3.8.4), cuya imagen esta en el rango $[0,1]$

$$f_1(x) = \frac{1}{1 + e^{-x}} \quad (3.8.4)$$

- La función tangente hiperbólica (ecuación 3.8.5), cuya imagen esta en el rango $[-1,1]$

$$f_2(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \quad (3.8.5)$$

- Ambas funciones se relacionan mediante la ecuación 3.8.6

$$f_2(x) = 2 \times f_1(x) - 1 \quad (3.8.6)$$

La función de activación elegida dependerá de tipo de problema o de cómo se quieran expresar las salidas. Por lo general se suele usar una misma función para todas las neuronas de la red, pero esto puede variar según lo requiera el creador de la red.

3.9 Red Neuronal de Base Radial (RBFN)

La red neuronal de base radial (RBFN, según sus siglas en ingles Radial Basis Function Network), es una red multicapa con conexiones hacia adelante, es un aproximador universal al igual que el MLP, en el sentido que puede aproximar cualquier función continua en R^n .

Esta fue pensada para aprender en un tiempo menor al que toma el MLP, a modo de poder ser usado en aplicaciones de tiempo real. Esto se logra gracias a la arquitectura de la RBFN, la cual cuenta con una sola capa oculta, en la que se alojan neuronas con funciones de activación local, donde cada neurona de la capa oculta se activa en una región

diferente del espacio de patrones de entrada, de modo que sólo unas pocas neuronas sean procesadas para cada patrón de entrada. Las conexiones entre la capa de entrada y la oculta no tienen pesos, pero las conexiones entre la capa oculta y la capa de salida sí los tienen. En las RBFN sólo las neuronas de salida tienen umbral, como se aprecia en la figura 3.17.

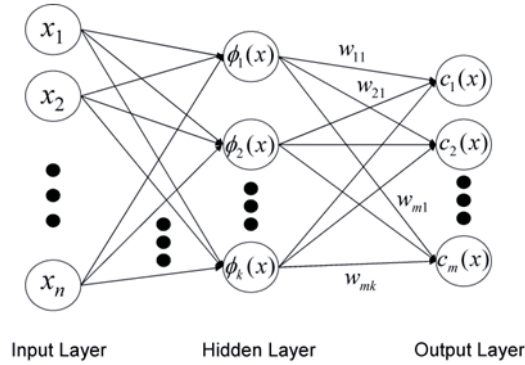


Figura 3.17: Red Neuronal de base radial

Tomando una red RBFN con p neuronas de entrada, m neuronas en la capa oculta y r de salida, donde el vector de entrada está dado por $X = (x_1, x_2, \dots, x_p)$ y el vector de salida por $Y = (y_1, y_2, \dots, y_r)$, este último se obtiene de la ecuación 3.9.1.

$$y_k = \sum_{i=1}^m w_{ik} \times \phi_i + u_k \quad (3.9.1)$$

Donde w_{ik} son los pesos entre la conexión de la neurona oculta i a la de salida, ϕ_i es la activación de la neurona oculta y u_k es el umbral de la capa de salida k .

Se puede observar claramente entonces que la salida de la red es una combinación lineal de las activaciones de las neuronas de la capa oculta.

La función ϕ_i , conocida como función de base radial, es la que determina la activación de la neurona de la capa oculta, la cual está definida por la ecuación 3.9.2, para $i = 1, 2, \dots, m$.

$$\phi_i = \phi \left(\frac{\|X - C_i\|}{d_i} \right) \quad (3.9.2)$$

Donde ϕ es la función de base radial, $C_i = (c_{i1}, c_{i2}, \dots, c_{ip})$ son los vectores de los centros de la función de base radial, d_i es un número real que representa la desviación de la función de base radial y $\|X - C_i\|$ representa la distancia euclidiana entre el vector X y el vector C_i , esta distancia está definida por la ecuación 3.9.3.

$$X - C_i = \left(\sum_{j=1}^p (x_j - c_{ij})^2 \right)^{\frac{1}{2}} \quad (3.9.3)$$

La función de base radial es de carácter local, ésta se acerca al máximo cuando el patrón de entrada está cerca del centro, y se acerca al mínimo cuando le patrón de entrada se lejos del centro.

Las funciones de base radial más utilizadas, donde c , $\theta > 0$ y r pertenecen a los reales, son:

- Función Gausiana (ecuación 3.9.4):

$$\phi(r) = e^{\left(\frac{-r^2}{2\theta^2}\right)} \quad (3.9.4)$$

- Función inversa cuadrática (ecuación 3.9.5):

$$\phi(r) = \frac{1}{c^2 + r^2} \quad (3.9.5)$$

- Función inversa multicuadrática (ecuación 3.9.6):

$$\phi(r) = \frac{1}{\sqrt{c^2 + r^2}} \quad (3.9.6)$$

3.10 Red Wavelet (WNN)

Las técnicas Wavelet pueden ofrecer un conocimiento y rendimiento agregado en el análisis de datos en situaciones donde las técnicas de Fourier han sido usadas. Las redes neuronales Wavelet (WNN , según sus siglas en ingles Wavelet Neural Network) son útiles en tareas de clasificación, predicción y estimación de funciones. Este tipo de redes tienen una estructura similar a las RBFN, con la diferencia de la función usada en la capa oculta, como se muestra en la figura 3.18.

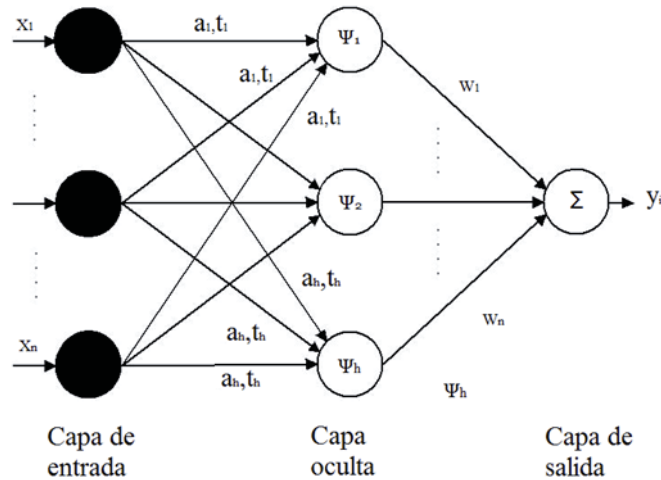


Figura 3.18: Arquitectura de una WNN

En donde se utiliza la ecuación 3.10.1, basada en la transformada discreta en el tiempo de las Wavelets:

$$\psi\left(\frac{x - t_i}{a_i}\right) \times \frac{1}{\sqrt{a_i}} \tag{3.10.1}$$

Donde ψ es una función Wavelet madre, hay varias de estas funciones [10], una de las más utilizadas es la Segunda Derivada de una Gaussiana, la cual se representa por la ecuación 3.10.2:

$$\psi(t) = (1 - t^2) \times e^{-\frac{t^2}{2}} \tag{3.10.2}$$

3.11 Red Neuronal Polinomial (PNN)

Las redes neuronales polinomiales (PNN, según sus siglas en inglés Polynomial Neural Network) son del tipo feed-forward, éstas no tienen un único enfoque, existen una gran cantidad de enfoques como por ejemplo las implementaciones de red neuronal de modelos discretos Volterra, implementadas, MLP con una capa oculta de funciones polinomiales de activación, Pi-Sigma y Sigma-Pi entre otras, esto se debe a su potencial dado que tienen habilidades de aproximación [11].

Las principales diferencias en cada enfoque radican en:

- La topología de la red polinomial y cuál es su conectividad.
- Qué funciones de activación polinomial son usadas en los nodos de la red (lineales, cuadráticas o asignaciones altamente no lineales en una o varias variables)

- Cuál es la técnica de aprendizaje para los pesos.
- Si hay algoritmos diseñados que busquen la adecuada estructura de la red polinomial.
- Qué criterios son considerado para el control de la búsqueda al evaluar el desempeño de los datos.

3.12 Softmax.

Cuando el problema tratado es un problema de clasificación, se considera un número de nodos de salida igual al número de clases que posee el modelo a evaluar. De este modo, si J es el número de clases, $\vartheta = (\vartheta_1, \dots, \vartheta_J)$ es el conjunto de coeficientes asociados a la RNA y $\vartheta_j = (\beta_j, W)$ es el conjunto de parámetros asociados al nodo de salida j , el modelo funcional se representa en 3.12.2.

$$B_j(x, w_j) = \frac{1}{1 + \exp(-(w_{j0} + \sum_{i=1}^k w_{ji} \cdot x_i))} \quad (3.12.1)$$

$$f_j(x, \theta_j) = \sum_{i=1}^M \beta_i^j B_i(x, w_i) \quad 1 \leq j \leq J - 1 \quad (3.12.2)$$

Siendo $\beta_i^j = \{\beta_1^j, \beta_2^j, \dots, \beta_M^j\}$ el valor de las conexiones entre capa oculta y capa de salida para el nodo j .

El enfoque adoptado a la hora de interpretar las salidas de los nodos de la capa de salida es un enfoque probabilístico que considera la función de activación softmax [12][13] en cada uno de dichos nodos, la cual viene dada por 3.12.3.

$$g_j(x, \theta_j) = \frac{\exp(f_i(x, \theta_j))}{\sum_{i=1}^J \exp(f_i(x, \theta_i))} \quad (3.12.3)$$

Donde $g_j(x, \theta_j)$ es la probabilidad de que el patrón x pertenezca a la clase j . La transformación softmax produce estimaciones positivas en todas las salidas, siendo la suma de todas ellas igual a 1, lo que hace que puedan ser interpretadas como la probabilidad de pertenencia a la clase correspondiente. Teniendo en cuenta esta consideración, se puede comprobar que la clase predicha por la RNA, $C(x, \theta)$, es la correspondiente al nodo de la capa de salida cuyo valor de salida es mayor. La expresión formal se describe en la ecuación.

$$C(x, \theta) = \Gamma \quad \text{donde} \quad \Gamma = \arg \max_j g_j(x, \theta_j) = \arg \max_j f_j(x, \theta_j), \quad 1 \leq j \leq J \quad (3.12.4)$$

En la figura 3.19 se puede apreciar de forma gráfica el funcionamiento descrito anteriormente.

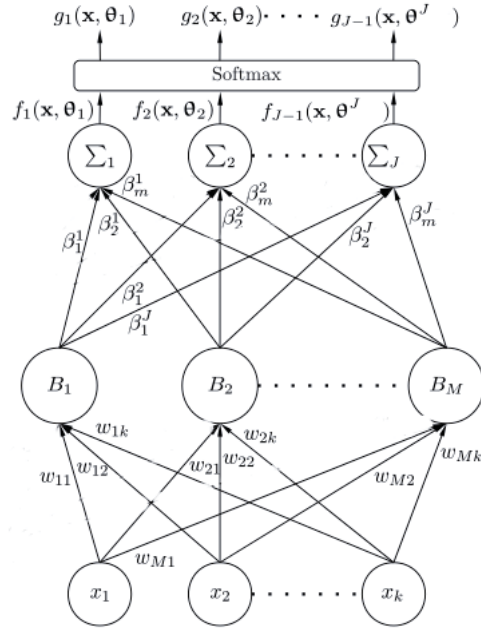


Figura 3.19: Uso función de activación Softmax

3.13 Aplicaciones.

Las redes neuronales artificiales son una tecnología computacional que puede utilizarse en un gran número y variedad de aplicaciones. A continuación se presentan algunas áreas en las cuales se aplican redes neuronales artificiales:

- Optimización de soluciones que satisfacen ciertas restricciones, con el fin de maximizar o minimizar la función objetivo. En la gestión empresarial, son decisiones de optimización encontrar los niveles de tesorería, de existencias, de producción, construcción de carteras óptimas, entre otras.
- Agrupamiento y categorización de datos que siguen un patrón común indeterminado, como es el caso de aplicaciones de compresión y minería de datos.
- Control de sistemas dinámicos
- Asociación y clasificación. Donde los patrones de entrada estáticos o señales temporales deben ser clasificados o reconocidos. Idealmente, un clasificador debería ser entrenado para que cuando se le presente una versión distorsionada ligeramente del patrón, pueda ser reconocida correctamente sin problemas.
- Pronóstico y predicción de series de tiempo y que tienen aplicaciones en una gran variedad de áreas.

4 Optimización por Enjambre de Partículas (PSO)

4.1 Introducción

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



Figura 4.1: Enjambres en la Naturaleza

Sus orígenes se remontan de los estudio realizados por Kennedy y Eberhart en el año 1995, quienes intentaron simular de manera gráfica el movimiento sincronizado e impredecible de los bancos de peces o bandadas de aves (figura 4.1), intrigados por la capacidad que tienen estos grupos para separarse, reagruparse o encontrar su alimento. Paralelamente con trabajos previos en el ámbito de la biología y la sociología, concluyen que el comportamiento, inteligencia y movimiento de estas agrupaciones está relacionado directamente con la capacidad para compartir información y aprovecharse de la experiencia acumulada por cada uno de los congéneres.

Kennedy y Eberhart introducen el término partícula, la cual representa a cualquier tipo de individuo que muestre algún tipo de comportamiento social como grupo en forma de una colección de agentes que interactúan entre sí. Cada partícula depende de dos factores: la memoria autobiográfica de la partícula o nostalgia y la influencia social de todo el enjambre. El proceso evolutivo se reduce a mover cada partícula dentro del espacio de soluciones con una velocidad que variará de acuerdo a su velocidad actual, a la memoria de la partícula y a la información global que comparte el resto del enjambre, utilizando una función de fitness para cuantificar la calidad de cada partícula en función de la posición que ésta ocupe.

La optimización por enjambre de partículas llega a una solución próxima a la global,

en lugar de converger hacia soluciones locales como puede suceder con otros métodos tradicionales. Otros aspectos, tales como el hecho de realizar la búsqueda utilizando un conjunto de puntos sobre el espacio de soluciones en lugar de un único punto, el utilizar una función objetivo para dirigir la búsqueda en lugar de derivadas, o que los nuevos puntos a explorar se determinen de acuerdo a reglas de decisión estocásticas en lugar de deterministas, reafirman la superioridad de PSO sobre técnicas de optimización local.

4.2 Fundamentos del movimiento de partículas.

Para aplicaciones relacionadas con el ámbito de la vida artificial se deben respetar cinco principios básicos de lo que se conoce como inteligencia de grupo, estos se conocen como:

- **Proximidad.** La población debiera ser capaz de realizar cálculos sencillos de espacio y tiempo, lo cual en PSO se traduce a movimientos en d-dimensiones llevados a cabo durante un intervalo de tiempo que coinciden con movimientos de la población a una determinada velocidad.
- **Calidad.** Los factores de calidad en PSO se consiguen en base a la memoria de la partícula junto con la historia o conocimiento social que comparten todos los congéneres.
- **Diversidad de respuesta.** Representadas por las tendencias marcadas por la memoria personal de cada partícula y por la historia de la mejor posición visitada por el conjunto.
- **Estabilidad y Adaptabilidad.** Resaltan aspectos contrapuestos, dado que por un lado, la población solo cambia su comportamiento como grupo cuando se actualiza la mejor posición históricamente visitada por alguno de los miembros que lo integran (principio de estabilidad) y, por otro lado, la población debe ser capaz de modificar su comportamiento y movimiento cuando hay alguna señal que así lo recomienda. Esto último, desde el punto de vista de ahorro computacional o mejora de precisión. En PSO esto se consigue cuando alguna partícula alcanza una solución global que mejora el resultado, en ese instante la población cambia de rumbo.

4.3 Algoritmo PSO.

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

- **Paso 1:** Se debe definir el tamaño n del enjambre y se deben inicializar las posiciones de las velocidades de las partículas en un espacio D-dimensional.

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

Lo anterior puede ser visualizado en la figura 4.2

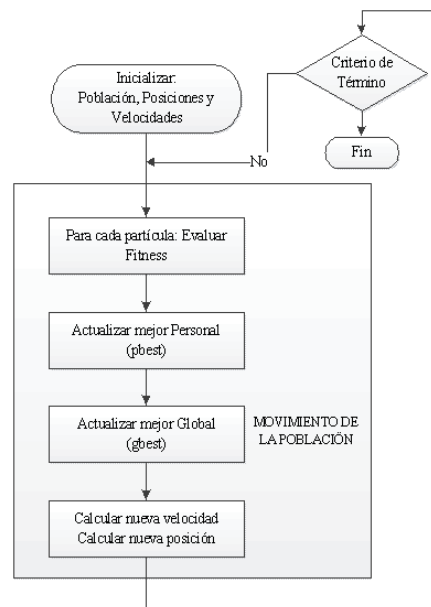


Figura 4.2: Diagrama de Flujo PSO tradicional

4.4 Parámetros PSO.

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

Para acotar la velocidad de la partícula se define un valor máximo V_{max} , el cual restringe a la velocidad en cada dimensión del intervalo $[-V_{max}, V_{max}]$. Si el valor de V_{max} tiende a ser demasiado pequeño, las partículas exploraran el espacio de soluciones muy lento y podrán quedar atrapadas dentro de soluciones locales. Por otro lado, la no restricción de la velocidad llevaría a la no convergencia del enjambre en un punto, ocurriendo lo que se conoce como explosión del PSO, el cual corresponde a un comportamiento oscilatorio y creciente de la posición de las partículas provocando la ineficiencia del PSO como algoritmo de optimización.

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

Los parámetros c_1 y c_2 , luego de una buena sintonización, pueden acelerar la convergencia del algoritmo y aliviarlo de los mínimos locales. Típicamente a estos parámetros se les asigna 2 o 1.49. Trabajos recientes revelan que podrían ser un parámetro cognitivo c_1 más grande que el social c_2 , siempre y cuando se cumpla que $c_1 + c_2 \leq 4$ [14].

El tamaño de la población debe ser seleccionado de manera rigurosa, dado que valores muy grandes pueden explorar de manera minuciosa el espacio de búsqueda pero el costo computación se eleva de manera considerable debido al aumento del número de evaluaciones de la función fitness. Generalmente se utilizan poblaciones ente 10 y 50 o 100 y 200 para problemas completos, en el primer intervalo de población el costo computacional no es alto, no así en el segundo intervalo que el costo computacional aumenta considerablemente.

4.5 PSO con actualizaciones síncronas y asíncronas.

La diferencia entre una actualización asíncrona con una síncrona, radica en el instante en el cual se realiza la actualización de la memoria de cada partícula y el conocimiento social del grupo. En el modelo síncrono, todas las partículas se mueven en paralelo, el fitness de las partículas se evalúa en cada iteración, se actualiza su memoria P_{id} y el conocimiento social P_{gd} . Por otro lado, en el PSO asíncrono al momento de desplazarse, cada partícula aprovecha la información actualizada por sus inmediatos predecesores. Es

decir, en cada iteración k , la i -ésima partícula se desplaza hacia un nuevo punto utilizando la información de los vectores P_{id} y P_{gd} , actualizados por las $i - 1$ partículas previas. Posteriormente, la partícula evalúa la calidad del nuevo punto y actualiza, si se requiere, las variables P_{id} y P_{gd} . Esta información se transmite a las restantes partículas. Al actualizar la información partícula a partícula, el modelo asíncrono acelera la optimización, aunque la naturaleza del modelo síncrono lo hace susceptible de ser ejecutado en paralelo, sobre múltiples procesadores.

4.6 Exploración y Explotación con PSO.

Uno de los problemas de los algoritmos evolutivos, es que, en ocasiones quedan atrapados en óptimos locales, lo cual en la literatura se conoce como convergencia prematura del algoritmo y corresponde a la incapacidad de salir de un mínimo local, impidiendo que se exploren otras regiones del espacio de búsqueda, haciendo que, en ocasiones, no se logre la obtener un mejor resultado.

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

Se define como exploración a la búsqueda de la solución del problema en el espacio de búsqueda de manera amplia, privilegiando la diversificación del enjambre. Por otro lado, la explotación consiste en la búsqueda de la solución del problema en un rango (sub-espacio) dentro del espacio de búsqueda donde el enjambre trate de tomar todos los valores posibles, privilegiando la convergencia del enjambre.

Estudios realizados en [14] revelan que altos valores para el coeficiente de inercia, factores de cognición y V_{max} ($0,75 < w < 1$ y $2 < c_1, c_2 < 4$), favorecen el modo de exploración del PSO, mientras que valores bajos ($0,4 < w < 0,75$ y $1 < c_1, c_2 < 2$) favorecen el modo de explotación del PSO.

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

4.7 Variaciones del Algoritmos de PSO.

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

4.7.1 PSO Discreto

PSO discreto [15, 16]: si bien la mayoría de los algoritmos de optimización de enjambre de partículas están pensados para buscar soluciones en un dominio continuo, hay variantes que están pensadas para espacios discretos. La primera variante para un dominio discreto fue PSO binario [17]. Aquí se considera la posición discreta, mientras su velocidad es continua. El componente j -ésimo del vector de velocidad de la partícula es usado para calcular la probabilidad con la cual el j -ésimo componente del vector de posición de la partícula toma el valor uno. La velocidad se actualiza según la ecuación 4.7.3, mas la posición es actualizada usando la ecuación 4.7.1:

$$x_{ij}^{t+1} = \begin{cases} 1 & \text{si } r < \text{sig}(v_{ij}^{t+1}) \\ 0 & \text{si } r \geq \text{sig}(v_{ij}^{t+1}) \end{cases} \quad (4.7.1)$$

Donde x_{ij} es la j -ésima componente del vector de posición de la partícula i , r es un número aleatorio en el intervalo $[0, 1]$ y $\text{sig}(v_{ij})$ que representa la probabilidad de que el bit j -ésimo de la partícula i sea uno, $\text{sig}(v_{ij})$ está dada por la siguiente ecuación 4.7.2:

$$\text{sig}(v_{ij}) = \frac{1}{1 + e^{-v_{ij}}} \quad (4.7.2)$$

4.7.2 LDWPSO

[18] (LDWPSO del ingles Linearly Decreasing Inertia Weight PSO) fue propuesto por Shi y Eberhart en 1998 en donde fue introducido el concepto de peso de inercia linealmente decreciente w . En ésta variante la partícula se define como un vector formado por la velocidad, posición y memoria de la misma. En la primera iteración las partículas se inicializan aleatorias y se optimiza sus soluciones a medida que aumentan el numero de iteraciones. La posición del enjambre es representado por $X = (x_{i1}, x_{i2}, \dots, x_{id})$ donde $i = 1, 2, \dots, n$ corresponde a la i -ésima partícula y $d = 1, 2, \dots, D$ corresponde a sus dimensiones, la velocidad se representa por $V_i = (v_{i1}, v_{i2}, \dots, v_{id})$.

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

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

Ambos valores se actualizan en cada iteración y se utilizan para ajustar la velocidad con la que se mueve una determinada partícula en cada una de las dimensiones. La influencia

que tiene la mejor posición personal sobre la velocidad de una partícula se conoce como el factor de cognición y la influencia de la mejor partícula del enjambre se conoce como componente social.

Se incorpora el factor de inercia w en la ecuación que determina la velocidad, de tal manera de que se balancee la búsqueda global y local. La velocidad y la posición están dadas por las ecuaciones 4.7.3 y 4.7.5

$$v_{id}(t+1) = wv_{id}(t) + c_1rand()(p_{id}(t) - x_{id}(t)) + c_2rand()(p_{gd}(t) - x_{id}(t)) \quad (4.7.3)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (4.7.4)$$

Donde r_1 y r_2 corresponden a valores aleatorios independientes en el intervalo $[0, 1]$, y c_1 y c_2 corresponden a las variables que controlan la influencia de las componentes cognitiva y social, $rand()$ corresponde a valores aleatorios en el rango de $[0, 1]$.

Para prevenir que las velocidades de las partículas incrementen infinitamente, se incorpora un parámetro V_{max} que delimita el rango de velocidad $[-V_{max}, V_{max}]$ que puede tomar una partícula. V_{max} estará dada por las ecuaciones 4.7.5, 4.7.6 y 4.7.7

$$if \{(v_{ij} > V_{max}) \text{ o } (v_{ij} < -V_{max})\} \quad (4.7.5)$$

$$v_{ij} = sign(v_{ij}) \cdot V_{max} \quad (4.7.6)$$

$$sign(x) = \begin{cases} +1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases} \quad (4.7.7)$$

4.7.3 Center PSO

Center PSO [19, 18]: es una modificación del algoritmo LDWPSO, surgiendo como resultado de implantar una partícula central en el enjambre, esta partícula no tiene una velocidad explícita y se ubica en el centro del enjambre en cada iteración, esto porque el centro del enjambre generalmente presenta un buen fitness. La actualización de la velocidad de las demás partículas sigue el mecanismo de LDWPSO. La partícula del centro se actualiza por medio de la ecuación 4.7.8:

$$x_{cj}^{t+q} = \frac{1}{N-1} \sum_{i=1}^{N-1} x_{ij}^{t+1} \quad (4.7.8)$$

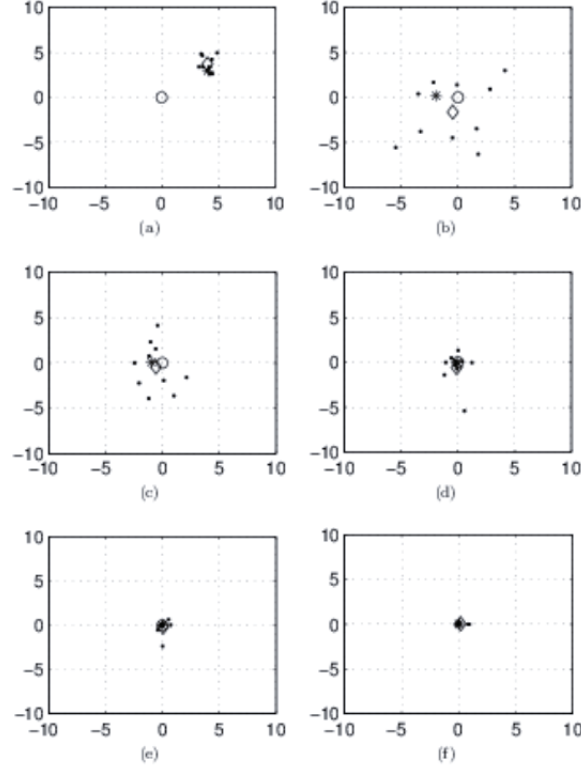


Figura 4.3: Distribución de las partículas: (a) en 1 iteración (b) en 10 iteraciones (c) en 40 iteración (d) en 60 iteraciones (e) en 80 iteraciones (f) en 100 iteraciones

A diferencia de otras partículas, la partícula del central no tiene velocidad, pero está involucrado en todas las operaciones de las demás partículas, tales como la evaluación del fitness, la revisión de la mejor posición de la partícula. Dentro de estas operaciones no se incluye el cálculo de la velocidad, véase 4.3.

4.7.4 Simple PSO

Simple PSO [19]: Se diferencia del PSO en que la velocidad de la partícula está dada por la ecuación 4.7.9:

$$v_{ij}^{t+1} = C_1 \times rand() \times (p_i^t - x_i^t) + C_2 \times rand() \times (g^t - x_i^t) \quad (4.7.9)$$

Por lo que se puede decir que la partícula no tiene memoria sobre la velocidad anterior.

4.7.5 PSO con Factor de Inercia

Shi y Eberhart [20] introdujeron el factor de inercia w en la ecuación que determina la velocidad. Esta modificación surgió como mecanismo de control de las habilidades de exploración y explotación del enjambre y además para que la velocidad no excediera los límites establecidos. Numerosos estudios concluyeron que como mecanismo de control de exploración/explotación funciona correctamente, aunque no logró eliminar completamente la necesidad del chequeo de la velocidad.

De esta forma la ecuación 4.7.10 de velocidad del algoritmo PSO queda de la siguiente manera:

$$v_{id}(t+1) = wv_{id}(t) + c_1rand()(p_{id}(t) - x_{id}(t)) + c_2rand()(p_{gd}(t) - x_{id}(t)) \quad (4.7.10)$$

Para cada iteración el factor de inercia es calculado con la siguiente ecuación:

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} \cdot iter \quad (4.7.11)$$

Donde w_{max} es el peso inicial y w_{min} el peso final, $iter_{max}$ es el número máximo de iteraciones e $iter$ es la iteración actual. w debe mantenerse entre 0.9 y 1.2. Valores altos provocan una búsqueda exhaustiva (más diversificación) y valores bajos una búsqueda más localizada (más intensificación).

Bin, Zhigang y Xingsheng [21] propusieron IPSO (Improved Particle Swarm Optimization), el cual plantea un peso de inercia dinámico que es decrementando de acuerdo aumentan a las iteraciones. Si bien es bastante parecido al método nombrado en los párrafos anteriores, este algoritmo calcula el peso de inercia de la siguiente forma:

$$w' = w \cdot \mu^{-k} \quad (4.7.12)$$

Donde:

$$w \in [0, 1]$$

$$\mu \in [1, 0001, 1, 005]$$

k es el número de iteraciones.

4.7.6 DAPSO

En DAPSO [22] (particle swarm optimizer with dynamic adaptation), a diferencia de PSO tradicional, se realiza una modificación en la ecuación de la velocidad, con el fin de

poder ser actualizada para reducir el azar en el curso de dicha actualización. Se propone una estrategia de cambio en el peso de inercia basado en el estado de la búsqueda, lo que implica que el peso de inercia cambia para cada partícula.

Una de las premisas de esta modificación, trata de que la velocidad de las partículas al momento de converger no deben tender a cero. Es por ello que se introduce el factor de inercia w , el cual varía en cada iteración. Los autores plantean que un peso de inercia que decrece linealmente puede no reflejar realmente el proceso actual de búsqueda. Por lo tanto, en DAPSO w es afectado por el estado evolutivo del algoritmo y determinado por dos factores, el factor de velocidad evolutivo y el grado de agregación del enjambre.

El factor de velocidad evolutivo se define de la siguiente forma:

$$h_i^t = \left| \frac{\min(F(pbest_i^{t-1}), F(pbest_i^t))}{\log \max(F(pbest_i^{t-1}), F(pbest_i^t))} \right| \quad (4.7.13)$$

Donde $F(pbest_i^t)$ es el fitness de $pbest_i^t$

Por otra parte, el grado de agregación se define de la siguiente forma:

$$s = \left| \frac{\min(F_{tbest}, \bar{F}_t)}{\max(F_{tbest}, \bar{F}_t)} \right| \quad (4.7.14)$$

Donde \bar{F}_t es el promedio del fitness de todas las partículas en la t -ésima iteración.

De esta forma el peso de inercia viene dado por la ecuación. 4.7.15

$$w_i^t = w_{in} - \alpha(1 - h_i^t) + \beta s \quad (4.7.15)$$

Donde w_{in} es el valor inicial de w , h y s están en el rango de $[0,1]$. La elección de α y β están típicamente en el rango de $[0,1]$.

4.7.7 PSO Quantum

Este algoritmo también llamado QPSO [23] fue planteado basándose en la lógica de la mecánica cuántica.

En términos de mecánica clásica la trayectoria de la partícula la podemos determinar según el vector de posición X_i y el vector de velocidad V_i , esto en la mecánica Newtoniana, sin embargo, en el caso de la mecánica cuántica no tiene sentido tratar de determinar la trayectoria, ya que X_i y V_i no pueden ser determinadas simultáneamente según el principio de incertidumbre. Por lo tanto si se considera el caso en que la partícula de PSO posea un comportamiento cuántico, el algoritmo PSO se debe trabajar de otra forma.

El estado de una partícula es representado por la función de onda $\psi(x, t)$ (Schrodinger equation), en vez de la posición y la velocidad. Teniendo en cuenta que la posición y la velocidad no pueden ser determinadas simultáneamente, entonces la probabilidad de que la partícula aparezca en la posición X_i dependerá de la función de densidad de probabilidad $|\psi(x, t)|^2$ y la forma en la cual depende de los campos potenciales se encuentra en [24].

Al utilizar el método de Monte Carlo, el movimiento está determinada por la ecuación 4.7.16.

$$\begin{cases} x_i(t+1) = p + \beta \times |M_{best} - x_i(t)| \times \ln\left(\frac{1}{u}\right) & k \geq 0,5 \\ x_i(t+1) = p - \beta \times |M_{best} - x_i(t)| \times \ln\left(\frac{1}{u}\right) & k < 0,5 \end{cases} \quad (4.7.16)$$

Donde β es un parámetro llamado expansión-contracción el cual toma un valor en el intervalo $[0; 1,7]$, u y k son valores generados acordes a la distribución de probabilidad uniforme entre el rango $[0, 1]$.

M_{best_i} es un vector que tiene las medidas de las mejores posiciones encontradas por las partículas, su valor esta dado por la ecuación 4.7.17

$$M_{best} = \frac{1}{N} \sum_{i=1}^N p_i = \frac{1}{N} \sum_{i=1}^N p_{i1} = \frac{1}{N} \sum_{i=1}^N p_{i2}, \dots, \frac{1}{N} \sum_{i=1}^N p_{id} \quad (4.7.17)$$

Donde N es el número de partículas del enjambre y p_i corresponde a la mejor posición personal, es decir se promedia la mejor posición personal de todas las partículas en la dimensión correspondiente.

Ahora se determina p usando la ecuación 4.7.18.

$$p = \varphi \times p_{id} + (1 - \varphi) \times p_{gd} \quad (4.7.18)$$

Donde φ un valor aleatorio en el intervalo $[0, 1]$

4.7.8 Métodos de Mejoramiento de PSO para Optimización Global

Hasta ahora se han visto variantes de PSO las cuales se basan en PSO usual. A continuación se mostrará un método que permite mejorar la eficiencia y velocidad de PSO, el cual puede ser aplicado en la mayoría de las variantes de PSO. Este método considera tres modificaciones al método convencional [19].

1. **Nueva regla de detención:** ésta nueva regla se crea con el fin de disminuir el número de iteraciones, esto se logra calculando en cada iteración la variación del

mejor valor descubierto v , deteniéndose cuando v es menor o igual a s , donde s es el mejor valor descubierto cuando un nuevo mejor valor fue recientemente descubierto. La razón de esta regla es que de no haber progreso en un número de iteraciones, podemos creer que hemos encontrado el mínimo global, por lo que debemos detener las iteraciones.

2. **Verificación similar:** ésta modificación evita que la función objetivo evalúe el punto x_i aun cuando este no cambie en la última iteración. Esto controlando los cambios en el punto x_i , por ejemplo, evaluar $f(x_i)$ si y solo si $|x_i^{t+1} - x_i^t| \geq \varepsilon$, donde ε representa un pequeño número positivo cercano a cero.
3. **Aplicación de búsqueda local:** un procedimiento de búsqueda local con el fin de evitar quedar atrapado en un mínimo local y para evitar llamadas innecesarias a la función que establece ésta probabilidad de $p_1 = \frac{1}{m}$ donde m es el número de partículas.

5 Métodos de Entrenamiento

El aprendizaje en las redes neuronales artificiales 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 con conexión hacia delante consisten en proponer una función de error que mida el rendimiento de la red en función de los pesos sinápticos. El objetivo que cumple el entrenar la red es obtener un conjunto de pesos sinápticos que minimizan (o maximizan) la función. 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 resultado en la obtención de estos pesos, es mediante la utilización de algoritmos evolutivos como por ejemplo PSO, el cual será utilizado en éste proyecto.

Uno de los métodos mas utilizados en las redes es el backpropagation (retropropagacion) es el método de entrenamiento más utilizado en redes con conexión hacia delante. 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 el error cuadrático medio, pero este es utilizado mayormente para fines predictivos, esta dada por la ecuación 5.0.19

$$E = \frac{1}{N} \times \sum_{i=1}^N ((d_i(n) - y_i(n))^2) \quad (5.0.19)$$

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.

Otra forma de entrenar es utilizando la entropía cruzada, la cual está dada por la ecuación 5.0.20, de manera de buscar E cercanos a 0. La siguiente ecuación considera que la red tiene una sola salida, en donde d es la salida deseada de la red e y es la salida de la red.

$$E = -\frac{1}{N} \times \sum_{i=1}^N (d_i(n) \times \ln(y_i(n)) + (1 - d_i(n)) \times \ln(y_i(n))) \quad (5.0.20)$$

Para calcular la entropía cruzada de una red de múltiples salidas se ocupa la ecuación 5.0.21, donde d es la salida deseada de la red e y es la salida de la red, además y_l representa los tipos de clases de las salidas de la red de 1 a J y θ es el vector $(\theta_1, \dots, \theta_J)$.

$$E = -\frac{1}{N} \times \sum_{i=1}^N \left[-\sum_{l=1}^J d_i(n) \times y_l(x(n), \theta(l)) + \log \sum_{l=1}^J \exp(y_l(x(n), \theta(l))) \right] \quad (5.0.21)$$

5.1 Conceptos básicos

Para un modelo de clasificación se utilizan otro tipo de métricas, las cuales serán descritas mas adelante. Para ello se debe tener conocimiento de los siguientes conceptos.

- **Verdaderos Positivos (VP):** número de éxitos. Para éste caso corresponde al número de personas detectadas lesionadas correctamente.
- **Verdaderos Negativos (VN):** número de rechazos correctos. Para éste caso corresponde a las personas detectadas ilesas correctamente.
- **Falsos Positivos (FP):** número de falsas alarmas. Para éste caso corresponde al número de personas detectadas lesionadas, siendo que en realidad resultaron ilesas (error tipo I o α).
- **Falsos Negativos (FN):** Para éste caso corresponde al número de personas detectadas ilesas, siendo que en realidad resultaron lesionadas (error tipo II o β).

Los VP, VN, FP y FN, para el caso de estudio, están resumidos tabla 5.1:

Cuadro 5.1: Matriz de Confusión

| | | Resultados Reales | |
|---------------------|------------|-------------------------------|-------------------------------|
| | | Lesionados | Ilesos |
| Resultados Obtenido | Lesionados | VP | FP (error tipo I o α) |
| | Ilesos | FN (error tipo II o β) | VN |

5.2 Métricas de rendimiento para evaluar el clasificador

Teniendo en cuenta lo mencionado anteriormente (VP,VN,FP,FN), se pueden construir las métricas que permiten medir el rendimiento del modelo de clasificación por medio de los ratios de error que se detallan a continuación:

- **Sensibilidad:** Probabilidad de que un individuo realmente lesionada sea detectada como tal por la prueba.. La sensibilidad es, por lo tanto, la capacidad de la prueba para detectar a los lesionados. Está definida por la ecuación 5.2.1:

$$Sensibilidad = \left(\frac{VP}{VP + FN} \right) \quad (5.2.1)$$

- **Especificidad:** Probabilidad de que un individuo ileso sea realmente clasificado como ileso por la prueba. La especificidad es la capacidad para detectar a las personas ilesas. Está definida por la ecuación 5.2.2:

$$Especificidad = \left(\frac{VN}{VN + FP} \right) \quad (5.2.2)$$

- **Exactitud:** Porcentaje de personas bien clasificadas, tanto lesionadas como ilesas. La exactitud está definida por la ecuación 5.2.3:

$$Exactitud = \left(\frac{VP + VN}{VP + VN + FP + FN} \right) \quad (5.2.3)$$

A partir de la tabla 5.1 se puede definir:

- **Valor predictivo positivo (VPP),** Es la proporción de resultados válidos entre los resultados positivos (lesionados) de la prueba, por ejemplo si se tiene un VPP igual a 80 %, significa que éste porcentaje de personas que se detecten como lesionadas, estarán realmente lesionadas. Se expresa por la ecuación 5.2.4.

$$VPP = \left(\frac{VP}{VP + FP} \right) \quad (5.2.4)$$

- **Valor predictivo negativo (VPN):** Es la proporción de resultados válidos entre los negativos (ilesos) de la prueba, por ejemplo si se tiene un VPN igual a 75 %, significa que éste porcentaje de personas que se detecten como ilesos, estarán realmente ilesos. Se expresa por la ecuación 5.2.5.

$$VPN = \left(\frac{VN}{VN + FN} \right) \quad (5.2.5)$$

Las métricas que fueron citadas (Sensibilidad, Especificidad, Exactitud), son condiciones necesarias para medir el grado de validez del modelo propuesto. A diferencia de lo anterior, el VPP y el VPN hacen mención a con qué seguridad el clasificador predice la presencia o ausencia de la lesión. Además, se debe medir el grado de reproductividad del modelo, lo que corresponde a la capacidad que tiene el modelo para obtener los mismos resultados en circunstancias similares, es decir, validándolo con distintos tamaños de muestras.

Otra métrica utilizada en modelos de clasificación es la curva ROC (Receiver Operating Characteristic), la cual corresponde a “un gráfico en el que se observan todos los pares sensibilidad/especificidad resultantes de la variación continua de los puntos de corte en todo el rango de resultados observados. En el eje y de coordenadas se sitúa la sensibilidad o fracción de verdaderos positivos. En el eje x se sitúa la fracción de falsos positivos o $1 - \text{especificidad}$ ”[25] (véase figura 5.1).

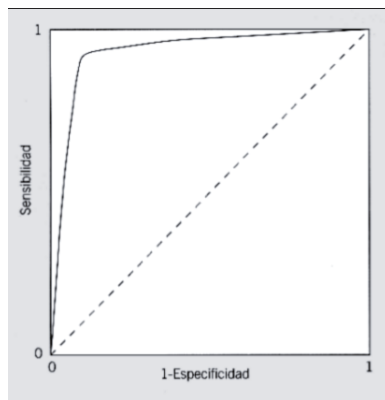


Figura 5.1: Curva ROC

Cualitativamente, cuanto más próxima es una curva ROC a la esquina superior izquierda, más alta es la exactitud global de la prueba, como se muestra en la figura . De la misma forma, si se dibujan en un mismo gráfico las curvas obtenidas con distintas pruebas diagnósticas, aquella que esté situada más hacia arriba y hacia la izquierda tiene mayor exactitud: por simple observación se obtiene una comparación cualitativa.

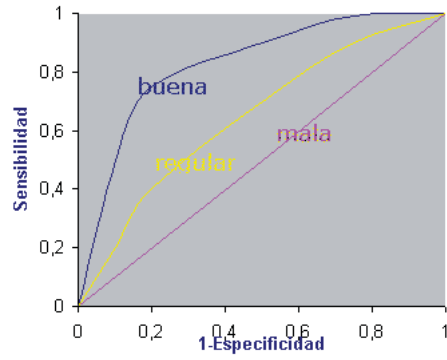


Figura 5.2: Tipos de Curva ROC

El área bajo la curva ROC es el mejor indicador global de la precisión de una prueba, ya que hace factible expresar el desempeño de una prueba mediante un número simple. Esta área es siempre mayor o igual a 0,5. El rango de valores se mueve entre 1 (discriminación perfecta) y 0,5 (no hay diferencias en la distribución de los valores de la prueba entre los 2 grupos).

6 Estado del Arte Accidentes de Tránsito

Por su parte, Mussone, Ferrari y Oneta en 1999 [26] utilizaron redes neuronales para analizar accidente ocurridos en intersecciones en Milán, Italia. Eligieron las técnicas de feed-forward (MLP) con Back Propagation (BP). El modelo contó con 10 nodos de entrada para ocho variables día o noche; flujos de tráfico en las intersecciones; número de puntos virtuales conflictivos; número de puntos reales conflictivos; tipo de intersección; tipo de accidente; estado de superficie de las carreteras y condiciones climáticas. Los nodos de salida fueron llamados “índice de accidente”, los cuales fueron calculados como un ratio entre el número de accidentes para una cierta intersección y el número de accidentes para la intersección más peligrosa. Los resultados mostraron que el mayor índice de accidentes ocurrió en intersecciones no señalizadas en la noche.

El año 2002 So Young Sohn y Sung Ho Lee [27] usaron red neuronal y árbol de decisión para mejorar la exactitud de clasificación individual. Esto se desarrollo para dos categorías de gravedad en accidentes de tránsito, lesiones corporales y daño de propiedad. Se usaron tres enfoques distintos (classifier fusión basado en el algoritmo de Dempster-Shafer, procedimiento Bayesiano y modelo logístico); fusión de datos basado en arcing y bagging. El ultimo enfoque fue clustering basado en el algoritmo k-means. Los resultados obtenidos se midieron en base a la precisión obteniéndose como resultados un 76,10 % en el método de Clustering (con árbol de decisión) seguido de 74.78 % en el método bagging (con árbol de decisión). Las redes neuronales dieron como resultado un 70.86 % de precisión.

Se realizo una clasificación de accidentes según la severidad del accidente, clasificando este en cinco clases (no lesión, posible lesión, lesiones no incapacitantes, lesiones incapacitantes y lesiones fatales) y se compararon las técnicas de árbol de decisión, red neuronal (usando un entrenamiento que combinaba Back Propagation y conjugate gradient descent), hibrid DT-ANN y SVM usando la función de base radial para el kernel. Como resultado se obtuvo que en todas las clases el árbol de decisión presento mejor precisión. Los resultados se presentan en el informe referenciado [28].

En el año 2008 Juan Zúñiga presento un modelo de redes bayesianas para identificar los factores determinantes en los accidentes de tránsito que afecten a las personas. Los registros de los accidentes contienen principalmente la fecha en la que ocurrieron éstos y sus características (Causas, Estado Atmosférico, Personas Involucradas, Vehículos Involucrados, etc.). Dentro del informe se muestran los resultados obtenidos, los cuales brindan una variedad importante de causas que provocan los accidentes de tránsito, el detalle de los resultados obtenidos se muestran en el informe referenciado [29].

Se realizo un estudio [30] que comparo dos modelos analíticos (una modificación del modelo Smeed y el modelo de Andreassen adaptado) y una red neuronal artificial para estimar el número de muertos en accidentes de tránsitos, para ellos se probaron utilizando los datos históricos de 20 años de tres grandes metrópolis de Turquía, los resultados mostraron que la red neuronal artificial predecía mejor los resultados al dar los menores errores.

El último informe de seguridad vial realizado por la organización mundial de la salud en el 2009 [31], sostuvo como conclusión que la tasa más alta de letalidad por cien mil habitantes correspondían a países de ingresos bajos y medios. Sin embargo, este estudio no utilizó herramientas de minería de datos sino que solo técnicas y herramientas estadísticas.

En Julio del 2010 Félix Castro presento un modelo de clasificación de accidentes de tránsito [1], para lo cual uso un modelo LS-SVM basado en Optimización por enjambre de partículas. Este modelo se basa en el hecho de seleccionar los parámetros que usara LS-SVM mediante un algoritmo de PSO, una vez obtenidos estos parámetros se vuelve a entrenar para obtener una buena clasificación. Para este trabajo se clasificó en lesionados e ilesos, dando como mejor resultado una exactitud del 70 %.

Redes Neuronales, Arboles de Decisión, Redes Bayesianas y Clustering son técnicas predominantes dentro de los estudios realizados referentes a los accidentes de tránsito, sin embargo, los resultados obtenidos no son los más óptimos al momento de decidir si un modelo es realmente efectivo. Es por ello que se busca mejorar las técnicas ya realizados con el fin de obtener mejores resultados y poder contribuir con un modelo que permita ser un aporte al estudio de la seguridad vial.

7 Implementación del Modelo

7.1 Obtención de los datos

Los datos que se utilizaran para realizar el estudio, fueron proporcionados por Cecilia Montt, profesora de la escuela de Ingeniería de Transporte de la Pontificia Universidad Católica de Valparaíso (PUCV), cuya escuela tiene el nexo directo con la CONASET. Dichos datos están almacenados en plantillas excel, separados por año entre 2003-2009 de las distintas regiones de Chile, cada plantilla excel contiene tres hojas las cuales contienen el detalle de los Accidentes; las Personas y los Vehículos involucrados. Estos datos son obtenidos por carabineros de Chile los cuales registran los eventos de forma manual en una plantilla llamada SIEC2 (Anexo A.1) cada accidente ocurrido, siguiendo un instructivo preestablecido. Posteriormente los datos son ingresados a una base de datos central de la CONASET.

Los datos son separables en tres entidades, cada una con sus respectivos atributos para fines de registro:

1. **Accidentes:** identificador del accidente, fecha, hora de accidente, comuna, causas, tipo de accidente, estado atmosférico, condición calzada, tipo de calzada, estado calzada, urbano/rural, y la información que indica la cantidad de ilesos, leves, graves, menos graves y muertos.
2. **Persona:** identificador del accidente, sexo, calidad, edad, y resultado.
3. **Vehículo:** identificador del accidente, servicio, y el tipo de vehículo.

7.2 Estructura de los datos

Como se indica en la figura 7.1, en un accidente (el cual cuenta con un identificador único) puede haber una o muchas personas involucradas y/o uno o muchos vehículos involucrados. Una persona y un vehículo pueden estar sólo una vez envueltos en un accidente con ese identificador único.

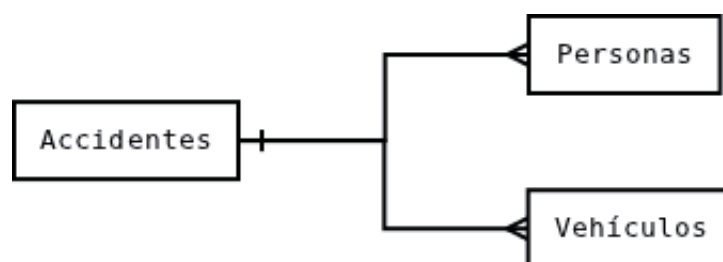


Figura 7.1: Entidades base de datos CONASET

A continuación se detallan los atributos propios de cada entidad [1]

1. **Accidentes.** Entidad correspondiente a los accidentes ocurridos en una determinada fecha y lugar geográfico. Los atributos se nombran a continuación.

- **Idaccidente:** identificador único y correlativo por cada accidente ocurrido en todo Chile.
- **Fecha:** fecha del suceso.
- **Hora:** hora del suceso.
- **Región:** región del suceso.
- **Comuna:** comuna de la región en donde ocurrió el accidente.
- **Tipo de accidente:** existen diez tipos accidentes:
 - **Atropello**
 - **Caída**
 - **Colisión** (colisiones frontal, lateral, por alcance y perpendicular).
 - **Impacto con animal**
 - **Choque con objeto:** (choques con objeto frontal, lateral y posterior).
 - **Choque con vehículo detenido:** (choque con vehículo detenido, frente/frente, frente/lado, frente/posterior, lado/frente, lado/lado, lado/posterior, posterior/frente, posterior/lado y posterior/posterior).
 - **Volcadura**
 - **Incendio**
 - **Descarrilamiento**
 - **Otro tipo**
- **Causa:** corresponde al origen del accidente, existiendo dieciocho tipos:
 - **Fallas mecánicas:** (fallas del tipo frenos, dirección, eléctrica, suspensión, neumáticos, motor y carrocería).
 - **Adelantamiento:** (adelantamientos sin el espacio o tiempo suficiente, sin efectuar la señal respectiva, por la berma, sobrepasando línea continua, en cruce, curva, cuesta, puente, conducción, bajo la influencia del alcohol, bajo la influencia de drogas o estupefacientes, contra el sentido del tránsito, en estado de ebriedad, físicas deficientes (cansancio, sueño), por izquierda eje calzada, no atento a las condiciones del tránsito, sin mantener distancia razonable ni prudente y cambiar sorpresivamente pista de circulación).
 - **No respetar derecho preferente de paso:** (derechos al peatón y vehículo).
 - **Pasajero:** (sube o desciende de vehículo movimiento, viaja en pisadera de vehículo, imprudencia y ebriedad).
 - **Peatón:** (permanece sobre la calzada, cruza calzada forma sorpresiva o descuidada, imprudencia, ebriedad, cruza calzada fuera paso peatones, y cruza camino o carretera sin precaución).

- **Señalización:** (mal instalada o mantenida forma defectuosa, no respetar luz roja de semáforo, las indicación de carabinero en servicio, la señal de ceda el paso, la señal de pare, la luz intermitente semáforo; semáforo en mal estado o deficiente y desobedecer).
- **Velocidad:** (mayor que máxima permitida, menor que mínima establecida, no razonable ni prudente, no reducir en cruce de calles, cumbre, curva, exceso en zona restringida) .
- **Carga:** (mayor que la autorizada vehículo, obstruye visual conductor, derrama en la calzada, sobresale estructura vehículo).
- **Virajes indebidos.**
- **Animales sueltos en vía pública.**
- **Vehículos en retroceso.**
- **Vehículos en panne sin señalización o deficiente.**
- **Pérdida control vehículo.**
- **Suicidio.**
- **Causas no determinadas.**
- **Otras causas.**
- **Fuga por hecho delictual.**
- **Ubicación relativa:** Existen diecisiete ubicaciones relativas:
 - **Cruce con semáforo funcionando.**
 - **Cruce sin señalización.**
 - **Enlace a desnivel.**
 - **Plaza de peaje.**
 - **Acera o berma.**
 - **Enlace a nivel.**
 - **Tramo de vía curva vertical.**
 - **Túnel.**
 - **Cruce con señal "pare".**
 - **Cruce regulado por carabinero.**
 - **Tramo de vía recta.**
 - **Acceso no habilitado.**
 - **Tramo de vía curva horizontal.**
 - **Rotonda.**
 - **Otros no considerados.**
 - **Cruce con señal "ceda el paso".**
 - **Cruce con semáforo apagado y puente.**
- **Estado Atmosférico:** (Despejado, nublado, lluvia, llovizna, neblina, nieve).
- **Tipo calzada:** (Concreto, asfalto, adoquín, mixto, ripio y tierra).
- **Estado Calzada:** (Bueno, regular, malo).

- **Condición Calzada:** (Seco, húmedo, mojado, con barro, con nieve, con aceite, escarcha, gravilla y otros).
 - **Muertos:** cantidad de muertos involucrados en el accidente.
 - **Graves:** cantidad de graves involucrados en el accidente.
 - **Menos Graves:** cantidad de menos graves involucrados en el accidente.
 - **Leves:** cantidad de leves involucrados en el accidente.
 - **Ilesos:** cantidad de ilesos involucrados en el accidente.
2. **Personas:** Entidad asociada a las personas involucradas en un accidente en particular. Un accidente por lo menos posee una persona involucrada, ya sea conductor, pasajero o peatón. Los atributos relacionados a esta entidad son:
- **Calidad:** (Peatón, conductor, pasajero).
 - **Sexo**
 - **Resultado:** (Muerto, grave, menos grave, leve, ileso).
3. **Vehículos:** Entidad asociada a los vehículos involucrados en un accidente en particular. Un accidente por lo menos presenta uno o más vehículos. Los atributos y características asociadas a esta entidad son:
- **Tipo vehículo:** (Bus/ taxi bus, minibús, trolebús, automóvil, camioneta, todo terreno, furgón, ambulancia, camión simple, camión simple con remolque, tracto-camión, tracto-camión con remolque, carro bomba, carro transporte de valores, remolque/semi remolque, motocicleta, motoneta/bicimoto, moto arenera, bicicleta, tracción animal, carro tracción humana, tractor, maquinaria agrícola, maquinaria movimiento tierras, maquinaria industrial, patín/patineta, patín motorizado, ferrocarril, dado a la fuga, otros no clasificados).
 - **Servicio:** (Carabineros, fiscal, particular, transporte escolar, taxi básico, taxi colectivo urbano, taxi colectivo rural, bomberos, salud, locomoción colectivo urbano, locomoción colectivo rural, servicio interurbano, servicio internacional, carga normal, carga peligrosa, dado a la fuga, otros sin especificar).

7.3 Procesamiento de los Datos

7.3.1 Integración de los datos

Los datos fueron extraídos desde las plantillas excel, las cuales se detallan en la sección 7.1, y puesto dentro de una base de datos Mysql, con el fin de poder ser consultados posteriormente de forma tal de facilitar la búsqueda de información relevante para éste proyecto. La base de datos se modeló por medio de las entidades accidentes y personas vistas en la figura 7.1 (la entidad vehículos fue descartada para este proyecto, ya que

se tomó como referencia el trabajo [1]), obteniéndose como resultado la base de datos mostrada en la figura 7.2.

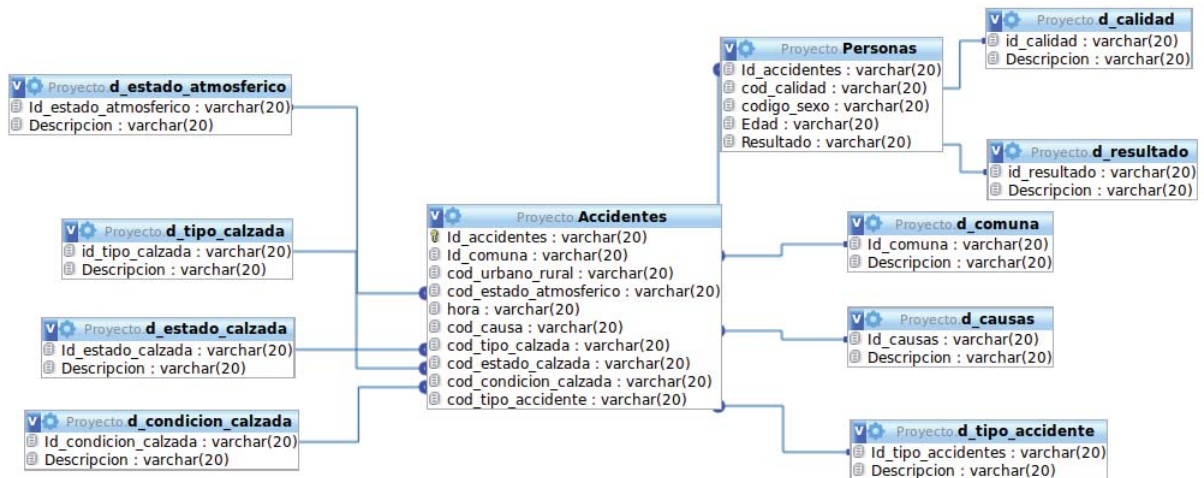


Figura 7.2: Base de Datos Accidentes de Tránsito

Para el llenado de la base de datos, se consideraron sólo los datos correspondientes a la región metropolitana de Chile, ya que el alcance del proyecto se limita a ésta zona de Chile. Además fueron descartados todos los registros que contaban con campos nulos, ingresando exclusivamente los datos que contaban con todos sus campos.

7.3.2 Selección y codificación de los datos

El modelo de clasificación se centrara en el estado que resultan las persona en un accidente de tránsito, siendo estas “Personas lesionados” o “Personas ilesos” (lo cual será visto con mayor detalle en la tabla 7.4), para ello fue necesario identificar las características que más expliquen el modelo. En función de estos se eligieron los atributos propuestos en el trabajo [29] realizado por la Escuela de Ingeniería de Transporte de la PUCV. Fue necesario obtener de la base de datos los atributos que se detallan en la tabla 7.1.

Cuadro 7.1: Atributos del Modelo

| Numero | Característica |
|--------|--------------------|
| 1 | Comuna |
| 2 | Urbano/Rural |
| 3 | Estado Atmosférico |
| 4 | Hora |
| 5 | Causa |
| 6 | Tipo Calzada |
| 7 | Estado Calzada |
| 8 | Condición Calzada |
| 9 | Tipo de Accidente |
| 10 | Calidad |
| 11 | Sexo |
| 12 | Edad |
| 13 | Resultado |

Dentro de cada atributo se asigno un código numérico a cada descripción, como ejemplo se muestra en la tabla 7.2

Cuadro 7.2: Atributo Calidad y Estado de la Calzada

| id_calidad | Descripción | id_estado_calzada | Descripción |
|------------|-------------|-------------------|-------------|
| 1 | CONDUCTOR | 1 | BUENO |
| 2 | PASAJERO | 2 | REGULAR |
| 3 | PEATON | 3 | MALO |

El atributo Resultado es utilizado para extraer la etiqueta que permitirá clasificar a las personas que estén involucradas en un accidente de tránsito. En el caso de la base de datos éste se encuentra dividido como se muestra en la tabla 7.3.

Cuadro 7.3: Atributo Resultado

| id_resultado | Descripción |
|--------------|-------------|
| 1 | MUERTO |
| 2 | GRAVE |
| 3 | MENOS GRAVE |
| 4 | LEVE |
| 5 | ILESO |

Estas características sirven para formar las clases que serán utilizadas en el modelo

Cuadro 7.4: Clases para la clasificación

| Clase | Atributos | Etiqueta |
|---------------------|--------------------------|----------|
| Personas Lesionadas | Grave, Menos Grave, Leve | 1 |
| Personas Ilesas | Ilesos | -1 |

para clasificar a las personas en **lesionados** e **ilesos**. Ambas clases son formadas como se muestra en la tabla 7.4

El atributo “Muerto” mostrado en la tabla 7.3 no es considerado para la obtención de las clases, por lo que es sacada del modelo, lo que lleva a tener un total de doce características que serán incluidas. El atributo “Resultado” es utilizado para la conformación de la clase y establecer la etiqueta.

7.3.3 Transformación de los datos

Los datos fueron normalizados permitiendo mantener la integridad de los datos, esto suele proporcionar mejores resultados que la introducción directa de datos crudos en los algoritmos. Es por ello que los datos son normalizados por columna. La normalización está dada por la ecuación 7.3.1.

$$z(i) = \left(\frac{2 \cdot x(i) - (b + a)}{b - a} \right) \quad (7.3.1)$$

Donde $x(i)$ es el elemento actual, b es el valor máximo de la columna a normalizar y a es el valor mínimo de la columna.

7.3.4 Software y Hardware Utilizado

El hardware que se utilizará para la simulación de este proyecto será:

- Procesador Intel Core i5
- Velocidad de proceso 2.67GHz
- 3 GB memoria RAM
- Disco Duro 500 GB

El software que se utilizará será:

- Sistema Operativo Ubuntu Linux 11.04 32 bits.

- Matlab para Linux 7.11.0 (R2010b) 32 bits.

7.4 Clasificación RNA con PSO

Para lograr elaborar los modelos que permitan clasificar lo propuesto en este trabajo, se debió seguir una serie de pasos que llevaron a la obtención de los modelos finales.

Primero se realizó la obtención y tratamiento de los datos (el detalle de esto se encuentra en la sección 7), esto con el fin de realizar el pre-proceso de los mismos logrando así disminuir la redundancia y la inconsistencia en ellos.

Una vez obtenidos los datos, se eligieron las técnicas que permitieron realizar los distintos modelos, para éste trabajo se utilizó la técnicas MLP (descritos en los apartados 3.8 y 3.9), en donde se probaron diversas funciones de activación (véase figura 3.16) . Posteriormente se eligió el algoritmo de entrenamiento, para la investigación se utilizó PSO y alguna de sus variantes, con el fin de obtener mejores resultado. La topología de la red esta dada por doce nodos de entradas (descritos en la tabla 7.1), los nodos de la capa oculta se establecerán mediante prueba y error y la capa de salida presenta un único nodo.

La red se entrenó mediante PSO de manera de obtener los pesos de la red, para ello se utilizaron un determinado número de datos. Del total de datos se seleccionará el 80 % de ellos para el entrenamiento de la red y 20 % restante será utilizado para la etapa de testeo.

En la etapa final se contrastaron los resultados obtenidos por medio de métricas de rendimiento (descritas en la subsección 5.2), las cuales permitieron visualizar el modelo que obtuvo mejores resultados.

Tanto el número de iteraciones como la cantidad de nodos ocultos de la red fueron obtenidos mediante prueba y error. Los parámetros del algoritmo evolutivo (PSO y sus variantes) fueron establecidos de igual forma. La simulación y el contraste de los resultados que se obtendrán con los distintos modelos. Un prototipo de la aplicación que es desarrollada para la simulación de los modelos se encuentra en el Anexo A.2

La forma para la obtención de los parámetros, tanto de la red como del algoritmo evolutivo, fueron obtenidos en base a pruebas realizadas. Para la red neuronal artificial se fijo el numero de iteraciones en 100, haciendo variar entre 1 y 30 el numero de nodos de la capa oculta. La cantidad de nodos ocultos que dio mejores resultados fue 5 nodos en la capa oculta. Posteriormente se fijo la cantidad de nodos ocultos con el mejor valor obtenido previamente y se hizo variar el numero de iteraciones entre 100 y 1000, dando como mejor resultado 300 iteraciones. Ya con la configuración establecida para la red, se procede a establecer la cantidad de partículas para el algoritmo evolutivo. Se hizo variar de 5 a 50 partículas, obteniéndose como mejor resultado 15 partículas, variable cognitiva y social se establecieron mediante lo revisado en la literatura.

En ésta etapa se presentan los modelos propuestos e implementados, los cuales varían en base a las funciones de activación y las variantes del algoritmo de optimización por enjambre de partículas. Para ello se eligieron la red neuronal y las funciones de activación sigmoïdal y tangente hiperbólica, en conjunto con las variantes de PSO, IPSO (Improved Particle Swarm Optimization) y DAPSO (Dynamic Adaptation Particle Swarm Optimization).

Para el algoritmo de optimización PSO se eligió la función fitness (de costo) MAE (error absoluto medio) con lo cual se logra una mayor generalización del modelo.

El total de datos con los que se cuentan para la región Metropolitana son aproximadamente 800.000 datos, los cuales fueron separados por zonas ver Anexo A.3(Norte, Centro, Sur, Oriente, Poniente) como muestra la figura 7.3 , con el fin de seleccionar las comunas de cada zona mas representativas en cuanto a accidente ocurridos. Realizando la separación se reducen los datos a 350.000, de los cuales se seleccionan de forma aleatoria el 10 % para las pruebas, lo que finalmente hace que se trabaje con 36.000 datos que serán la base para probar los distintos modelos propuestos. Los datos para trabajar se separaron en porcentajes de entrenamiento y pruebas, de los cuales corresponde un 80 % y un 20 % respectivamente.

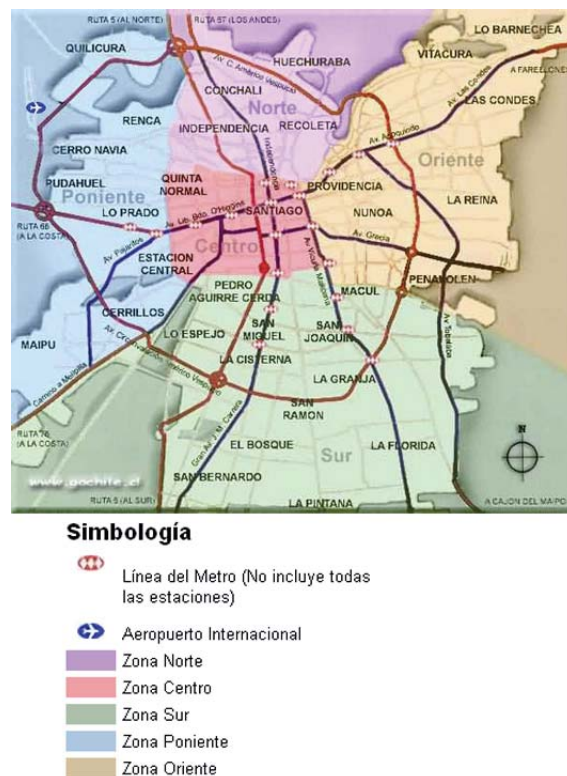


Figura 7.3: Region Metropolitana por Zonas

Para la red neuronal se utilizaron las funciones de activación sigmoïdal y tangente

hiperbólica, realizando las combinaciones correspondientes, esto con el fin de obtener los mejores resultados. Los parámetros de la red se detallan en la siguiente tabla:

Cuadro 7.5: Parámetros de entrada Red

| Parámetros de Entrada | Valores |
|-----------------------|------------|
| Iteraciones | 300 |
| Nodos de Entrada | 12 |
| Nodos Ocultos | 5 |
| Nodos Salida | 1 |
| % Entrenamiento | 80 % |
| % Pruebas | 20 % |
| Datos | 10 % Total |

Los valores de los parámetros de inicialización de PSO comunes para las 2 variantes utilizadas, se presentan en la siguiente tabla:

Cuadro 7.6: Parámetros de entrada PSO

| Parámetros de Entrada | Valores |
|------------------------|---------------|
| Tamaño de la Población | 15 partículas |
| Componente cognitiva | 2 |
| Componente social | 1.75 |
| Velocidad máxima | 1 |
| Posición máxima | 1 |
| Wmax | 0.75 |
| Wmin | 0.1 |

Tanto para IPSO como para DAPSO se utilizaron 36000 datos los cuales fueron separados en 80 % para las pruebas y 20 % para la etapa de pruebas obteniéndose los siguientes resultados.

7.5 IPSO (Improved Particle Swarm Optimization)

Esta variante posee 2 parámetros de entrada. Estos son inicializados como siguen:

- $W_{max} = 0.75$
- $\mu = 1.0005$

Para los modelos se realizaron 10 pruebas y se utilizaron las funciones de activación para capa oculta y capa de salida como sigue:

- **Capa Oculta** : Sigmoidal ; **Capa Salida:** Sigmoidal.
- **Capa Oculta** : Sigmoidal ; **Capa Salida:** Tangente Hiperbólica.
- **Capa Oculta** : Tangente Hiperbólica ; **Capa Salida:** Sigmoidal.
- **Capa Oculta** : Tangente Hiperbólica ; **Capa Salida:** Tangente Hiperbólica.

7.5.1 Función de activación Tangente Hiperbólica- Tangente Hiperbólica.

Entrenamiento

Cuadro 7.7: Tangente Hiperbólica- Tangente Hiperbólica con IPSO (Entrenamiento)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|-------------|--------|-----------|---------------|--------------|--------|--------|
| 1 | 0,2479 | 0,7521 | 0,7128 | 0,7617 | 0,9156 | 0,4222 |
| 2 | 0,2617 | 0,7383 | 0,6712 | 0,7555 | 0,8997 | 0,4129 |
| 3 | 0,2534 | 0,7466 | 0,6816 | 0,7644 | 0,8976 | 0,4419 |
| 4 | 0,2713 | 0,7287 | 0,7507 | 0,7257 | 0,9553 | 0,2718 |
| 5 | 0,2882 | 0,7118 | 0,6101 | 0,7367 | 0,8853 | 0,3620 |
| 6 | 0,2889 | 0,7111 | 0,8483 | 0,7022 | 0,9861 | 0,1566 |
| 7 | 0,2651 | 0,7349 | 0,6824 | 0,7465 | 0,9136 | 0,3745 |
| 8 | 0,3085 | 0,6915 | 0,5682 | 0,7166 | 0,8910 | 0,2894 |
| 9 | 0,2534 | 0,7466 | 0,7633 | 0,7436 | 0,9475 | 0,3413 |
| 10 | 0,2593 | 0,7407 | 0,6688 | 0,7603 | 0,8940 | 0,4315 |
| Max | 0,2479 | 0,7521 | 0,8483 | 0,7644 | 0,9861 | 0,4419 |
| Min | 0,3085 | 0,6915 | 0,5682 | 0,7022 | 0,8853 | 0,1566 |
| Prom | 0,2698 | 0,7302 | 0,6958 | 0,7413 | 0,9186 | 0,3504 |
| Desv | 0,0195 | 0,0195 | 0,0792 | 0,0209 | 0,0334 | 0,0896 |

Prueba

Cuadro 7.8: Tangente Hiperbólica- Tangente Hiperbólica con IPSO (Prueba)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|-------------|---------------|---------------|---------------|---------------|---------------|---------------|
| 1 | 0,2314 | 0,7686 | 0,6757 | 0,7924 | 0,9051 | 0,4546 |
| 2 | 0,2452 | 0,7548 | 0,6567 | 0,7770 | 0,9091 | 0,4000 |
| 3 | 0,2397 | 0,7603 | 0,6533 | 0,7882 | 0,8972 | 0,4455 |
| 4 | 0,2672 | 0,7328 | 0,6625 | 0,7415 | 0,9466 | 0,2409 |
| 5 | 0,2741 | 0,7259 | 0,5734 | 0,7633 | 0,8795 | 0,3727 |
| 6 | 0,2590 | 0,7411 | 0,8810 | 0,7325 | 0,9901 | 0,1682 |
| 7 | 0,2507 | 0,7493 | 0,6557 | 0,7682 | 0,9170 | 0,3636 |
| 8 | 0,3113 | 0,6887 | 0,4773 | 0,7357 | 0,8636 | 0,2864 |
| 9 | 0,2355 | 0,7645 | 0,7168 | 0,7733 | 0,9368 | 0,3682 |
| 10 | 0,2507 | 0,7493 | 0,6267 | 0,7813 | 0,8893 | 0,4273 |
| Max | 0,2314 | 0,7686 | 0,8810 | 0,7924 | 0,9901 | 0,4546 |
| Min | 0,3113 | 0,6887 | 0,4773 | 0,7325 | 0,8636 | 0,1682 |
| Prom | 0,2565 | 0,7435 | 0,6579 | 0,7653 | 0,9134 | 0,3527 |
| Desv | 0,0235 | 0,0235 | 0,1024 | 0,0217 | 0,0367 | 0,0934 |

Como se muestra en la tabla anterior, esta variante obtuvo un 76,86 % de exactitud en la etapa de testing, esto quiere decir que el modelo ha clasificado de manera correcta en un 76,86 % las personas que resultan en el estado lesionado o ileso. La sensibilidad obtenida fue del 79,24 % cifra correspondiente al porcentaje de personas bien clasificadas en el estado lesionado, por su parte la especificidad resultó ser un 67,57 % correspondiendo al porcentaje de personas bien clasificadas en el estado ileso. En cuanto a la seguridad del resultado, se tiene un valor predictivo positivo de 90,51 %, lo que quiere decir que el 90,51 % de las personas detectadas como lesionadas están realmente lesionadas, en tanto, el valor predictivo negativo fue de 45,46 %, este significa que éste porcentaje corresponde a las personas detectadas como ilesas que están realmente ilesas. Con las diez muestras se puede ver la variación que presenta el modelo al ser probado una cantidad determinada de veces, lo cual hace que éste no presente variaciones importantes.

7.5.2 Función de activación Tangente Hiperbólica- Sigmoidal.

Entrenamiento

Cuadro 7.9: Tangente Hiperbólica- Sigmoidal con IPSO (Entrenamiento)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|---------|--------|-----------|---------------|--------------|--------|--------|
| 1 | 0,2816 | 0,7184 | 0,6158 | 0,7604 | 0,8283 | 0,5133 |
| 2 | 0,3129 | 0,6871 | 0,5804 | 0,7180 | 0,8553 | 0,3734 |
| 3 | 0,2651 | 0,7349 | 0,6849 | 0,7496 | 0,8901 | 0,4453 |
| 4 | 0,2397 | 0,7603 | 0,7902 | 0,7534 | 0,9393 | 0,4266 |
| 5 | 0,2435 | 0,7565 | 0,7554 | 0,7568 | 0,9224 | 0,4473 |
| 6 | 0,2655 | 0,7345 | 0,7647 | 0,7289 | 0,9430 | 0,3458 |
| 7 | 0,3160 | 0,6840 | 0,5455 | 0,7627 | 0,7470 | 0,5665 |
| 8 | 0,2792 | 0,7208 | 0,6876 | 0,7284 | 0,9107 | 0,3665 |
| 9 | 0,2696 | 0,7304 | 0,7718 | 0,7233 | 0,9488 | 0,3232 |
| 10 | 0,2586 | 0,7414 | 0,6671 | 0,7690 | 0,8616 | 0,5172 |
| Max | 0,2397 | 0,7603 | 0,7902 | 0,7690 | 0,9488 | 0,5665 |
| Min | 0,3160 | 0,6840 | 0,5455 | 0,7180 | 0,7470 | 0,3232 |
| Prom | 0,2732 | 0,7268 | 0,6863 | 0,7450 | 0,8846 | 0,4325 |
| Desv | 0,0256 | 0,0256 | 0,0852 | 0,0185 | 0,0634 | 0,0812 |

Prueba

Cuadro 7.10: Tangente Hiperbólica- Sigmoidal con IPSO (Pruebas)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|---------|---------------|---------------|---------------|---------------|---------------|---------------|
| 1 | 0,2686 | 0,7314 | 0,6050 | 0,7795 | 0,8385 | 0,5106 |
| 2 | 0,2975 | 0,7025 | 0,5660 | 0,7407 | 0,8589 | 0,3798 |
| 3 | 0,2617 | 0,7383 | 0,6391 | 0,7684 | 0,8753 | 0,4557 |
| 4 | 0,2135 | 0,7865 | 0,7971 | 0,7840 | 0,9427 | 0,4641 |
| 5 | 0,2300 | 0,7700 | 0,7273 | 0,7815 | 0,9141 | 0,4726 |
| 6 | 0,2452 | 0,7548 | 0,7438 | 0,7570 | 0,9366 | 0,3798 |
| 7 | 0,2989 | 0,7011 | 0,5385 | 0,7919 | 0,7546 | 0,5907 |
| 8 | 0,2645 | 0,7355 | 0,6772 | 0,7479 | 0,9162 | 0,3629 |
| 9 | 0,2631 | 0,7369 | 0,7212 | 0,7396 | 0,9407 | 0,3165 |
| 10 | 0,2466 | 0,7534 | 0,6450 | 0,7947 | 0,8548 | 0,5443 |
| Max | 0,2135 | 0,7865 | 0,7971 | 0,7947 | 0,9427 | 0,5907 |
| Min | 0,2989 | 0,7011 | 0,5385 | 0,7396 | 0,7546 | 0,3165 |
| Prom | 0,2590 | 0,7411 | 0,6660 | 0,7685 | 0,8832 | 0,4477 |
| Desv | 0,0268 | 0,0268 | 0,0825 | 0,0209 | 0,0594 | 0,0871 |

Esta variante obtuvo un 78,65 % de exactitud en la etapa de testing, esto quiere decir que el modelo ha clasificado de manera correcta en un 78,65 % las personas que resultan en el estado lesionado o ileso. La sensibilidad obtenida fue del 78,40 % cifra correspondiente al porcentaje de personas bien clasificadas en el estado lesionado, por su parte la especificidad resultó ser un 79,71 % correspondiendo al porcentaje de personas bien clasificadas en el estado ileso. En cuanto a la seguridad del resultado, se tiene un valor predictivo positivo de 94,27 %, lo que quiere decir que el 94,27 % de las personas detectadas como lesionadas están realmente lesionadas, en tanto, el valor predictivo negativo fue de 46,41 %, este significa que éste porcentaje corresponde a las personas detectadas como ilesas que están realmente ilesas.

7.5.3 Función de activación Sigmoideal- Sigmoideal.

Entrenamiento

Cuadro 7.11: Sigmoideal- Sigmoideal con IPSO (Entrenamiento)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|-------------|--------|-----------|---------------|--------------|--------|--------|
| 1 | 0,2737 | 0,7263 | 0,5839 | 0,7775 | 0,8386 | 0,4854 |
| 2 | 0,2751 | 0,7249 | 0,5747 | 0,7856 | 0,8205 | 0,5200 |
| 3 | 0,2858 | 0,7142 | 0,7026 | 0,7153 | 0,9652 | 0,1762 |
| 4 | 0,2638 | 0,7362 | 0,6238 | 0,7678 | 0,8790 | 0,4303 |
| 5 | 0,2978 | 0,7022 | 0,6980 | 0,7024 | 0,9773 | 0,1124 |
| 6 | 0,2651 | 0,7349 | 0,5958 | 0,7880 | 0,8361 | 0,5178 |
| 7 | 0,2830 | 0,7170 | 0,6889 | 0,7199 | 0,9576 | 0,2011 |
| 8 | 0,3033 | 0,6967 | 0,5583 | 0,7168 | 0,9178 | 0,2227 |
| 9 | 0,3181 | 0,6819 | — | 0,6819 | 1,0000 | 0,0000 |
| 10 | 0,2538 | 0,7462 | 0,6781 | 0,7612 | 0,9148 | 0,3849 |
| Max | 0,2538 | 0,7462 | 0,7026 | 0,7880 | 1,0000 | 0,5200 |
| Min | 0,3181 | 0,6819 | 0,5583 | 0,6819 | 0,8205 | 0,0000 |
| Prom | 0,2820 | 0,7181 | 0,6337 | 0,7416 | 0,9107 | 0,3051 |
| Desv | 0,0199 | 0,0199 | 0,05818 | 0,0385 | 0,0646 | 0,1857 |

Prueba

Cuadro 7.12: Sigmoial- Sigmoial con IPSO (Pruebas)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|-------------|---------------|---------------|---------------|---------------|---------------|---------------|
| 1 | 0,2838 | 0,7163 | 0,6263 | 0,7500 | 0,8426 | 0,4844 |
| 2 | 0,2755 | 0,7245 | 0,6284 | 0,7658 | 0,8277 | 0,5352 |
| 3 | 0,3265 | 0,6736 | 0,6610 | 0,6747 | 0,9575 | 0,1523 |
| 4 | 0,2934 | 0,7066 | 0,6272 | 0,7307 | 0,8660 | 0,4141 |
| 5 | 0,3223 | 0,6777 | 0,7619 | 0,6725 | 0,9787 | 0,1250 |
| 6 | 0,2865 | 0,7135 | 0,6143 | 0,7539 | 0,8277 | 0,5039 |
| 7 | 0,3320 | 0,6680 | 0,6154 | 0,6732 | 0,9468 | 0,1563 |
| 8 | 0,3154 | 0,6846 | 0,6364 | 0,6922 | 0,9234 | 0,2461 |
| 9 | 0,3526 | 0,6474 | — | 0,6474 | 1,0000 | 0,0000 |
| 10 | 0,2838 | 0,7163 | 0,6736 | 0,7268 | 0,9000 | 0,3789 |
| Max | 0,2755 | 0,7245 | 0,7619 | 0,7658 | 1,0000 | 0,5352 |
| Min | 0,3526 | 0,6474 | 0,6143 | 0,6474 | 0,8277 | 0,0000 |
| Prom | 0,3072 | 0,6928 | 0,6493 | 0,7087 | 0,9070 | 0,2996 |
| Desv | 0,0260 | 0,0260 | 0,0466 | 0,0416 | 0,0638 | 0,1874 |

Esta variante clasificado de manera correcta en un 72,45 % las personas que resultan en el estado lesionado o ileso. La sensibilidad obtenida fue del 76,58 % cifra correspondiente al porcentaje de personas bien clasificadas en el estado lesionado, por su parte la especificidad resultó ser un 62,84 % correspondiendo al porcentaje de personas bien clasificadas en el estado ileso. En cuanto a la seguridad del resultado, se tiene un valor predictivo positivo de 82,77 %, lo que quiere decir que el 82,77 % de las personas detectadas como lesionadas están realmente lesionadas, en tanto, el valor predictivo negativo fue de 53,52 %, este significa que éste porcentaje corresponde a las personas detectadas como ilesas que están realmente ilesas.

7.5.4 Función de activación Sigmoidal-Tangente Hiperbólica.

Entrenamiento

Cuadro 7.13: Sigmoidal- Tangente Hiperbólica con IPSO (Entrenamiento)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|---------|--------|-----------|---------------|--------------|--------|--------|
| 1 | 0,2916 | 0,7084 | 0,7548 | 0,7048 | 0,9739 | 0,1646 |
| 2 | 0,2896 | 0,7105 | 0,6267 | 0,7255 | 0,9156 | 0,2904 |
| 3 | 0,3281 | 0,6719 | — | 0,6719 | 1,0000 | 0,0000 |
| 4 | 0,2799 | 0,7201 | 0,6087 | 0,7518 | 0,8710 | 0,4109 |
| 5 | 0,2662 | 0,7338 | 0,6991 | 0,7402 | 0,9304 | 0,3312 |
| 6 | 0,2418 | 0,7583 | 0,7175 | 0,7683 | 0,9166 | 0,4340 |
| 7 | 0,3064 | 0,6936 | 0,5562 | 0,7265 | 0,8726 | 0,3270 |
| 8 | 0,2273 | 0,7727 | 0,7244 | 0,7867 | 0,9079 | 0,4958 |
| 9 | 0,3040 | 0,6960 | 0,6434 | 0,7008 | 0,9555 | 0,1646 |
| 10 | 0,2569 | 0,7431 | 0,8000 | 0,7355 | 0,9647 | 0,2893 |
| Max | 0,2273 | 0,7727 | 0,8000 | 0,7867 | 1,0000 | 0,4958 |
| Min | 0,3281 | 0,6719 | 0,5562 | 0,6719 | 0,8710 | 0,0000 |
| Prom | 0,2792 | 0,7208 | 0,6812 | 0,7312 | 0,9308 | 0,2908 |
| Desv | 0,0312 | 0,0312 | 0,07771 | 0,0336 | 0,0426 | 0,1477 |

Prueba

Cuadro 7.14: Sigmoidal- Tangente Hiperbólica con IPSO (Pruebas)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|---------|---------------|---------------|---------------|---------------|---------------|---------------|
| 1 | 0,3072 | 0,6928 | 0,7910 | 0,6829 | 0,9698 | 0,2023 |
| 2 | 0,3113 | 0,6887 | 0,6607 | 0,6938 | 0,9181 | 0,2824 |
| 3 | 0,3609 | 0,6391 | — | 0,6391 | 1,0000 | 0,0000 |
| 4 | 0,2741 | 0,7259 | 0,6864 | 0,7379 | 0,8858 | 0,4428 |
| 5 | 0,2893 | 0,7107 | 0,7167 | 0,7096 | 0,9267 | 0,3282 |
| 6 | 0,2534 | 0,7466 | 0,7438 | 0,7474 | 0,9116 | 0,4542 |
| 7 | 0,3278 | 0,6722 | 0,5833 | 0,6942 | 0,8707 | 0,3206 |
| 8 | 0,2121 | 0,7879 | 0,8034 | 0,7829 | 0,9246 | 0,5458 |
| 9 | 0,3361 | 0,6639 | 0,6500 | 0,6652 | 0,9547 | 0,1489 |
| 10 | 0,2714 | 0,7287 | 0,8495 | 0,7109 | 0,9698 | 0,3015 |
| Max | 0,2121 | 0,7879 | 0,8495 | 0,7829 | 1,0000 | 0,5458 |
| Min | 0,3609 | 0,6391 | 0,5833 | 0,6391 | 0,8707 | 0,0000 |
| Prom | 0,2944 | 0,7057 | 0,72053 | 0,7064 | 0,9332 | 0,3027 |
| Desv | 0,0436 | 0,0436 | 0,09645 | 0,0417 | 0,0402 | 0,1593 |

Esta variante clasificado de manera correcta en un 78,79 % las personas que resultan en el estado lesionado o ileso. La sensibilidad obtenida fue del 78,29 % cifra correspondiente al porcentaje de personas bien clasificadas en el estado lesionado, por su parte la especificidad resultó ser un 80,34 % correspondiendo al porcentaje de personas bien clasificadas en el estado ileso. En cuanto a la seguridad del resultado, se tiene un valor predictivo positivo de 92,46 %, lo que quiere decir que el 92,46 % de las personas detectadas como lesionadas están realmente lesionadas, en tanto, el valor predictivo negativo fue de 54,48 %, este significa que éste porcentaje corresponde a las personas detectadas como ilesas que están realmente ilesas.

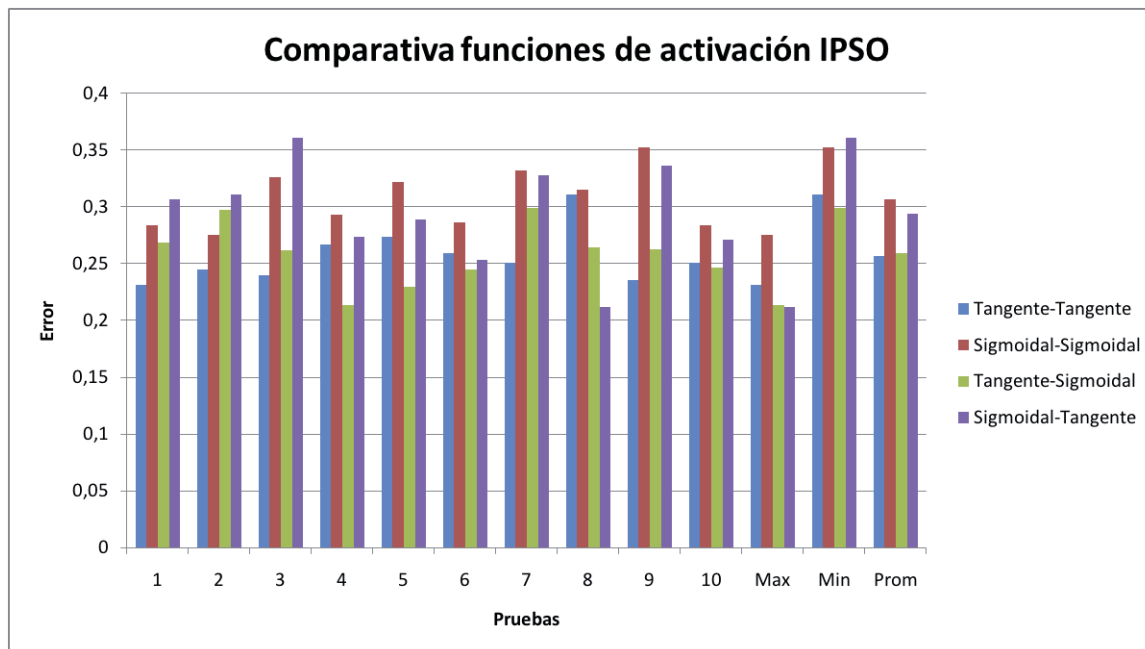


Figura 7.4: Comparacion de Errores por función IPSO

En el gráfico 7.4 se ve que el modelo que obtuvo mejores resultados en cuanto al error, que se define como 1- Exactitud, es el Hip-Hip, con lo cual se puede seguir realizando las pruebas en base a estas funciones de activación.

7.6 DAPSO (PSO con Adaptación Dinámica)

Esta variante posee 2 parámetros de entrada. Estos son inicializados como siguen:

- $W_{inicial} = 0.8$
- $\alpha = 0.4$
- $\beta = 0.8$

Para los modelos se realizaron 10 pruebas y se utilizaron las funciones de activación para capa oculta y capa de salida como sigue:

- **Capa Oculta** : Sigmoidal ; **Capa Salida:** Sigmoidal.
- **Capa Oculta** : Sigmoidal ; **Capa Salida:** Tangente Hiperbólica.
- **Capa Oculta** : Tangente Hiperbólica ; **Capa Salida:** Sigmoidal.
- **Capa Oculta** : Tangente Hiperbólica ; **Capa Salida:** Tangente Hiperbólica.

Para las dos variantes de PSO se utilizaron 36000 datos los cuales fueron separados en 80 % para las pruebas y 20 % para la etapa de pruebas obteniéndose los siguientes resultados.

7.6.1 Función de activación Tangente Hiperbólica- Tangente Hiperbólica.

Entrenamiento

Cuadro 7.15: Tangente Hiperbólica- Tangente Hiperbólica con DAPSO (Entrenamiento)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|-------------|--------|-----------|---------------|--------------|--------|--------|
| 1 | 0,1991 | 0,8009 | 0,7991 | 0,8014 | 0,9287 | 0,5522 |
| 2 | 0,1988 | 0,8012 | 0,8003 | 0,8015 | 0,9292 | 0,5522 |
| 3 | 0,1998 | 0,8002 | 0,7994 | 0,8004 | 0,9292 | 0,5491 |
| 4 | 0,2142 | 0,7858 | 0,78 | 0,7874 | 0,9256 | 0,5137 |
| 5 | 0,1991 | 0,8009 | 0,7991 | 0,8014 | 0,9287 | 0,5522 |
| 6 | 0,1991 | 0,8009 | 0,7991 | 0,8014 | 0,9287 | 0,5522 |
| 7 | 0,1991 | 0,8009 | 0,8 | 0,8012 | 0,9292 | 0,5512 |
| 8 | 0,1994 | 0,8006 | 0,8006 | 0,8005 | 0,9297 | 0,5491 |
| 9 | 0,2497 | 0,7503 | 0,8967 | 0,7317 | 0,9823 | 0,2989 |
| 10 | 0,2087 | 0,7913 | 0,7746 | 0,7965 | 0,9188 | 0,5431 |
| Max | 0,1988 | 0,8012 | 0,8967 | 0,8015 | 0,9823 | 0,5522 |
| Min | 0,2497 | 0,7503 | 0,7746 | 0,7317 | 0,9188 | 0,2989 |
| Prom | 0,2067 | 0,7933 | 0,8049 | 0,7924 | 0,933 | 0,5214 |
| Desv | 0,0160 | 0,016 | 0,0336 | 0,0218 | 0,0176 | 0,0791 |

Prueba

Cuadro 7.16: Tangente Hiperbólica- Tangente Hiperbólica con DAPSO (Prueba)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|-------------|---------------|---------------|---------------|---------------|--------------|---------------|
| 1 | 0,1956 | 0,8044 | 0,7836 | 0,8108 | 0,924 | 0,5607 |
| 2 | 0,1956 | 0,8044 | 0,7836 | 0,8108 | 0,924 | 0,5607 |
| 3 | 0,1970 | 0,803 | 0,7824 | 0,8094 | 0,924 | 0,5565 |
| 4 | 0,2121 | 0,7879 | 0,7457 | 0,8011 | 0,9097 | 0,5397 |
| 5 | 0,1956 | 0,8044 | 0,7836 | 0,8108 | 0,924 | 0,5607 |
| 6 | 0,1956 | 0,8044 | 0,7836 | 0,8108 | 0,924 | 0,5607 |
| 7 | 0,1970 | 0,803 | 0,7824 | 0,8094 | 0,924 | 0,5565 |
| 8 | 0,1970 | 0,803 | 0,7791 | 0,8105 | 0,922 | 0,5607 |
| 9 | 0,2328 | 0,7672 | 0,8723 | 0,7516 | 0,9754 | 0,3431 |
| 10 | 0,2052 | 0,7948 | 0,7557 | 0,8073 | 0,9117 | 0,5565 |
| Max | 0,1956 | 0,8044 | 0,8723 | 0,8108 | 0,9754 | 0,5607 |
| Min | 0,2328 | 0,7672 | 0,7457 | 0,7516 | 0,9097 | 0,3431 |
| Prom | 0,2023 | 0,7977 | 0,7852 | 0,8032 | 0,9263 | 0,5356 |
| Desv | 0,0120 | 0,012 | 0,0335 | 0,0184 | 0,0181 | 0,0679 |

Esta variante clasificado de manera correcta en un 80,44 % las personas que resultan en el estado lesionado o ileso. La sensibilidad obtenida fue del 81,08 % cifra correspondiente al porcentaje de personas bien clasificadas en el estado lesionado, por su parte la especificidad resultó ser un 78,36 % correspondiendo al porcentaje de personas bien clasificadas en el estado ileso. En cuanto a la seguridad del resultado, se tiene un valor predictivo positivo de 92,4 %, lo que quiere decir que el 92,4 % de las personas detectadas como lesionadas están realmente lesionadas, en tanto, el valor predictivo negativo fue de 56,07 %, este significa que éste porcentaje corresponde a las personas detectadas como ilesas que están realmente ilesas.

7.6.2 Función de activación Tangente Hiperbólica- Sigmoidal.

Entrenamiento

Cuadro 7.17: Tangente Hiperbólica- Sigmoidal con DAPSO (Entrenamiento)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|---------|--------|-----------|---------------|--------------|--------|--------|
| 1 | 0,1981 | 0,8019 | 0,7903 | 0,8053 | 0,9292 | 0,5428 |
| 2 | 0,2273 | 0,7727 | 0,8075 | 0,7658 | 0,9523 | 0,4071 |
| 3 | 0,1929 | 0,8071 | 0,7949 | 0,8107 | 0,9292 | 0,5585 |
| 4 | 0,2383 | 0,7617 | 0,7449 | 0,7655 | 0,9292 | 0,4207 |
| 5 | 0,1929 | 0,8071 | 0,7958 | 0,8105 | 0,9297 | 0,5574 |
| 6 | 0,1977 | 0,8023 | 0,7732 | 0,8115 | 0,9185 | 0,5658 |
| 7 | 0,2067 | 0,7933 | 0,7759 | 0,7983 | 0,9256 | 0,524 |
| 8 | 0,1933 | 0,8067 | 0,7938 | 0,8107 | 0,9287 | 0,5585 |
| 9 | 0,2431 | 0,7569 | 0,901 | 0,7395 | 0,9841 | 0,2944 |
| 10 | 0,1929 | 0,8071 | 0,7967 | 0,8102 | 0,9303 | 0,5564 |
| Max | 0,1929 | 0,8071 | 0,901 | 0,8115 | 0,9841 | 0,5658 |
| Min | 0,2431 | 0,7569 | 0,7449 | 0,7395 | 0,9185 | 0,2944 |
| Prom | 0,2083 | 0,7917 | 0,7974 | 0,7928 | 0,9357 | 0,4985 |
| Desv | 0,0201 | 0,0201 | 0,0404 | 0,0261 | 0,019 | 0,0926 |

Prueba

Cuadro 7.18: Tangente Hiperbólica- Sigmoidal con DAPSO (Pruebas)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|---------|---------------|---------------|---------------|---------------|---------------|---------------|
| 1 | 0,1832 | 0,8168 | 0,8095 | 0,819 | 0,9346 | 0,5738 |
| 2 | 0,2273 | 0,7727 | 0,8 | 0,7673 | 0,9509 | 0,4051 |
| 3 | 0,1832 | 0,8168 | 0,8059 | 0,8201 | 0,9325 | 0,5781 |
| 4 | 0,2273 | 0,7727 | 0,7687 | 0,7736 | 0,9366 | 0,4346 |
| 5 | 0,1818 | 0,8182 | 0,8107 | 0,8205 | 0,9346 | 0,5781 |
| 6 | 0,1942 | 0,8058 | 0,7759 | 0,8152 | 0,9202 | 0,5696 |
| 7 | 0,1956 | 0,8044 | 0,7811 | 0,8115 | 0,9243 | 0,557 |
| 8 | 0,1901 | 0,8099 | 0,7829 | 0,8185 | 0,9223 | 0,5781 |
| 9 | 0,2438 | 0,7562 | 0,8659 | 0,7422 | 0,9775 | 0,2996 |
| 10 | 0,1860 | 0,814 | 0,8 | 0,8183 | 0,9305 | 0,5738 |
| Max | 0,1818 | 0,8182 | 0,8659 | 0,8205 | 0,9775 | 0,5781 |
| Min | 0,2438 | 0,7562 | 0,7687 | 0,7422 | 0,9202 | 0,2996 |
| Prom | 0,2012 | 0,7988 | 0,8 | 0,8006 | 0,9364 | 0,5148 |
| Desv | 0,0227 | 0,0227 | 0,0275 | 0,0285 | 0,0169 | 0,0992 |

Esta variante clasificado de manera correcta en un 81,82 % las personas que resultan en el estado lesionado o ileso. La sensibilidad obtenida fue del 82,05 % cifra correspondiente al porcentaje de personas bien clasificadas en el estado lesionado, por su parte la especificidad resultó ser un 81,82 % correspondiendo al porcentaje de personas bien clasificadas en el estado ileso. En cuanto a la seguridad del resultado, se tiene un valor predictivo positivo de 93,46 %, lo que quiere decir que el 93,46 % de las personas detectadas como lesionadas están realmente lesionadas, en tanto, el valor predictivo negativo fue de 57,81 %, este significa que éste porcentaje corresponde a las personas detectadas como ilesas que están realmente ilesas.

7.6.3 Función de activación Sigmoideal- Sigmoideal.

Entrenamiento

Cuadro 7.19: Sigmoideal- Sigmoideal con DAPSO (Entrenamiento)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|-------------|--------|-----------|---------------|--------------|--------|--------|
| 1 | 0,2008 | 0,7992 | 0,7755 | 0,8066 | 0,9196 | 0,5574 |
| 2 | 0,2025 | 0,7975 | 0,7708 | 0,8059 | 0,9176 | 0,5564 |
| 3 | 0,2022 | 0,7978 | 0,7727 | 0,8057 | 0,9186 | 0,5553 |
| 4 | 0,2029 | 0,7971 | 0,7728 | 0,8047 | 0,9191 | 0,5522 |
| 5 | 0,2012 | 0,7988 | 0,7752 | 0,8062 | 0,9196 | 0,5564 |
| 6 | 0,3325 | 0,6675 | — | 0,6675 | 1 | 0 |
| 7 | 0,2029 | 0,7971 | 0,7728 | 0,8047 | 0,9191 | 0,5522 |
| 8 | 0,2823 | 0,7177 | 0,6755 | 0,7247 | 0,9304 | 0,2906 |
| 9 | 0,2022 | 0,7978 | 0,7734 | 0,8054 | 0,9191 | 0,5543 |
| 10 | 0,3325 | 0,6675 | — | 0,6675 | 1 | 0 |
| Max | 0,2008 | 0,7992 | 0,7755 | 0,8066 | 1 | 0,5574 |
| Min | 0,3325 | 0,6675 | 0,6755 | 0,6675 | 0,9176 | 0 |
| Prom | 0,2362 | 0,7638 | 0,7610 | 0,7699 | 0,9363 | 0,4175 |
| Desv | 0,0566 | 0,0566 | 0,0346 | 0,0596 | 0,0338 | 0,235 |

Prueba

Cuadro 7.20: Sigmoial- Sigmoial con DAPSO (Pruebas)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|-------------|---------------|---------------|---------------|---------------|---------------|---------------|
| 1 | 0,1942 | 0,8058 | 0,7953 | 0,809 | 0,9277 | 0,562 |
| 2 | 0,1928 | 0,8072 | 0,8072 | 0,8071 | 0,9339 | 0,5537 |
| 3 | 0,1915 | 0,8085 | 0,8084 | 0,8086 | 0,9339 | 0,5579 |
| 4 | 0,1928 | 0,8072 | 0,8072 | 0,8071 | 0,9339 | 0,5537 |
| 5 | 0,1928 | 0,8072 | 0,7965 | 0,8105 | 0,9277 | 0,5661 |
| 6 | 0,3333 | 0,6667 | — | 0,6667 | 1 | 0 |
| 7 | 0,1928 | 0,8072 | 0,8072 | 0,8071 | 0,9339 | 0,5537 |
| 8 | 0,2741 | 0,7259 | 0,7263 | 0,7258 | 0,9463 | 0,2851 |
| 9 | 0,1928 | 0,8072 | 0,8072 | 0,8071 | 0,9339 | 0,5537 |
| 10 | 0,3333 | 0,6667 | — | 0,6667 | 1 | 0 |
| Max | 0,1915 | 0,8085 | 0,8084 | 0,8105 | 1 | 0,5661 |
| Min | 0,3333 | 0,6667 | 0,7263 | 0,6667 | 0,9277 | 0 |
| Prom | 0,2291 | 0,7709 | 0,7944 | 0,7716 | 0,9471 | 0,4186 |
| Desv | 0,0605 | 0,0605 | 0,3358 | 0,061 | 0,0283 | 0,2364 |

Esta variante clasificado de manera correcta en un 80,85 % las personas que resultan en el estado lesionado o ileso. La sensibilidad obtenida fue del 80,86 % cifra correspondiente al porcentaje de personas bien clasificadas en el estado lesionado, por su parte la especificidad resultó ser un 80,84 % correspondiendo al porcentaje de personas bien clasificadas en el estado ileso. En cuanto a la seguridad del resultado, se tiene un valor predictivo positivo de 93,39 %, lo que quiere decir que el 93,39 % de las personas detectadas como lesionadas están realmente lesionadas, en tanto, el valor predictivo negativo fue de 55,79 %, este significa que éste porcentaje corresponde a las personas detectadas como ilesas que están realmente ilesas.

7.6.4 Función de activación Sigmoidal-Tangente Hiperbólica.

Entrenamiento

Cuadro 7.21: Sigmoidal- Tangente Hiperbólica con DAPSO (Entrenamiento)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|---------|--------|-----------|---------------|--------------|--------|--------|
| 1 | 0,2025 | 0,7975 | 0,8115 | 0,7929 | 0,9279 | 0,5614 |
| 2 | 0,2043 | 0,7957 | 0,7898 | 0,7979 | 0,9146 | 0,5807 |
| 3 | 0,1943 | 0,8057 | 0,8176 | 0,8017 | 0,9279 | 0,5845 |
| 4 | 0,1957 | 0,8043 | 0,8166 | 0,8002 | 0,9279 | 0,5807 |
| 5 | 0,3559 | 0,6441 | — | 0,6441 | 1 | 0 |
| 6 | 0,1943 | 0,8057 | 0,8167 | 0,8019 | 0,9274 | 0,5855 |
| 7 | 0,1974 | 0,8026 | 0,8145 | 0,7986 | 0,9274 | 0,5768 |
| 8 | 0,1946 | 0,8054 | 0,8173 | 0,8013 | 0,9279 | 0,5836 |
| 9 | 0,1933 | 0,8067 | 0,8166 | 0,8033 | 0,9269 | 0,5894 |
| 10 | 0,1939 | 0,8061 | 0,8161 | 0,8026 | 0,9269 | 0,5874 |
| Max | 0,1933 | 0,8067 | 0,8176 | 0,8033 | 1 | 0,5894 |
| Min | 0,3559 | 0,6441 | 0,7898 | 0,6441 | 0,9146 | 0 |
| Prom | 0,2126 | 0,7874 | 0,8129 | 0,7844 | 0,9335 | 0,523 |
| Desv | 0,0505 | 0,0505 | 0,00888 | 0,0494 | 0,0237 | 0,1839 |

Prueba

Cuadro 7.22: Sigmoidal- Tangente Hiperbólica con DAPSO (Pruebas)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|---------|---------------|---------------|---------------|---------------|---------------|---------------|
| 1 | 0,2052 | 0,7948 | 0,8125 | 0,7898 | 0,9371 | 0,5221 |
| 2 | 0,2094 | 0,7906 | 0,7904 | 0,7907 | 0,9266 | 0,5301 |
| 3 | 0,1997 | 0,8003 | 0,8171 | 0,7954 | 0,9371 | 0,5382 |
| 4 | 0,1997 | 0,8003 | 0,8171 | 0,7954 | 0,9371 | 0,5382 |
| 5 | 0,3430 | 0,657 | — | 0,657 | 1 | 0 |
| 6 | 0,1997 | 0,8003 | 0,8171 | 0,7954 | 0,9371 | 0,5382 |
| 7 | 0,2039 | 0,7961 | 0,8137 | 0,7912 | 0,9371 | 0,5261 |
| 8 | 0,2011 | 0,7989 | 0,8121 | 0,795 | 0,935 | 0,5382 |
| 9 | 0,2011 | 0,7989 | 0,8084 | 0,7961 | 0,9329 | 0,5422 |
| 10 | 0,2011 | 0,7989 | 0,8121 | 0,795 | 0,935 | 0,5382 |
| Max | 0,1997 | 0,8003 | 0,8171 | 0,7961 | 1 | 0,5422 |
| Min | 0,3430 | 0,657 | 0,7904 | 0,657 | 0,9266 | 0 |
| Prom | 0,2164 | 0,7836 | 0,81116 | 0,7801 | 0,9415 | 0,4811 |
| Desv | 0,0446 | 0,0446 | 0,25663 | 0,0433 | 0,0208 | 0,1692 |

Esta variante clasificado de manera correcta en un 80,03 % las personas que resultan en el estado lesionado o ileso. La sensibilidad obtenida fue del 79,54 % cifra correspondiente al porcentaje de personas bien clasificadas en el estado lesionado, por su parte la especificidad resultó ser un 81,71 % correspondiendo al porcentaje de personas bien clasificadas en el estado ileso. En cuanto a la seguridad del resultado, se tiene un valor predictivo positivo de 93,71 %, lo que quiere decir que el 93,71 % de las personas detectadas como lesionadas están realmente lesionadas, en tanto, el valor predictivo negativo fue de 53,82 %, este significa que éste porcentaje corresponde a las personas detectadas como ilesas que están realmente ilesas.

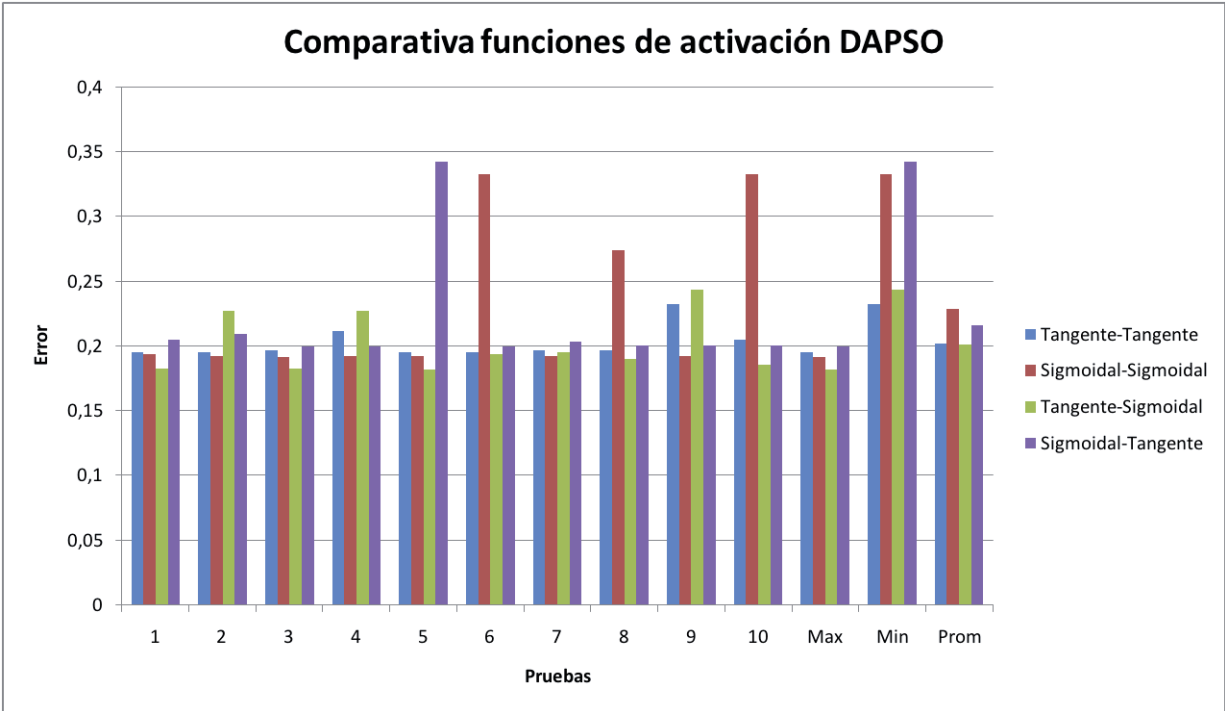


Figura 7.5: Comparación de Errores por función DAPSO

En el gráfico 7.5 se ve que el modelo que obtuvo mejores resultados en cuanto al error, que se define como 1- Exactitud, es el Hip-Sig, con lo cual se puede seguir realizando las pruebas en base a estas funciones de activación.

7.6.5 Comparativa entre las dos variantes PSO

Errores

En el apartado anterior se seleccionaron las alternativas que dieron mejores resultados en cada uno de los modelos, siendo las funciones Tangente Hiperbólica para capa oculta y Tangente Hiperbólica para la capa de salida la mejor alternativa para IPSO y Tangente

Hiperbólica para capa oculta y Sigmoidal para la capa de salida la mejor alternativa para DAPSO. Es ahora donde se contrastaran las dos mejores opciones.

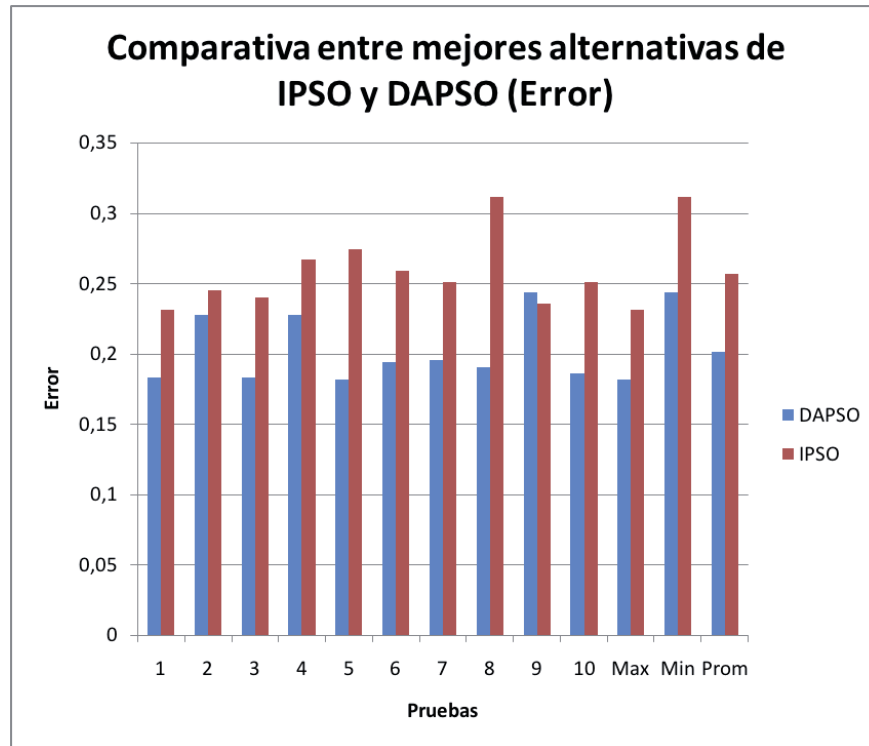


Figura 7.6: Comparativa entre IPSO-DAPSO

Como se muestra en el gráfico 7.6, los errores proporcionados por la alternativa DAPSO son menores a los obtenidos por IPSO, lo que nos permite ver que esa alternativa será la mas adecuada para poder seguir mejorando el modelo final.

El valor predictivo positivo como negativo permiten ver la seguridad en los resultados, es por ello que se contrasta variable para ver que modelo presenta una mayor seguridad en los resultados.

Valor Predictivo Positivo

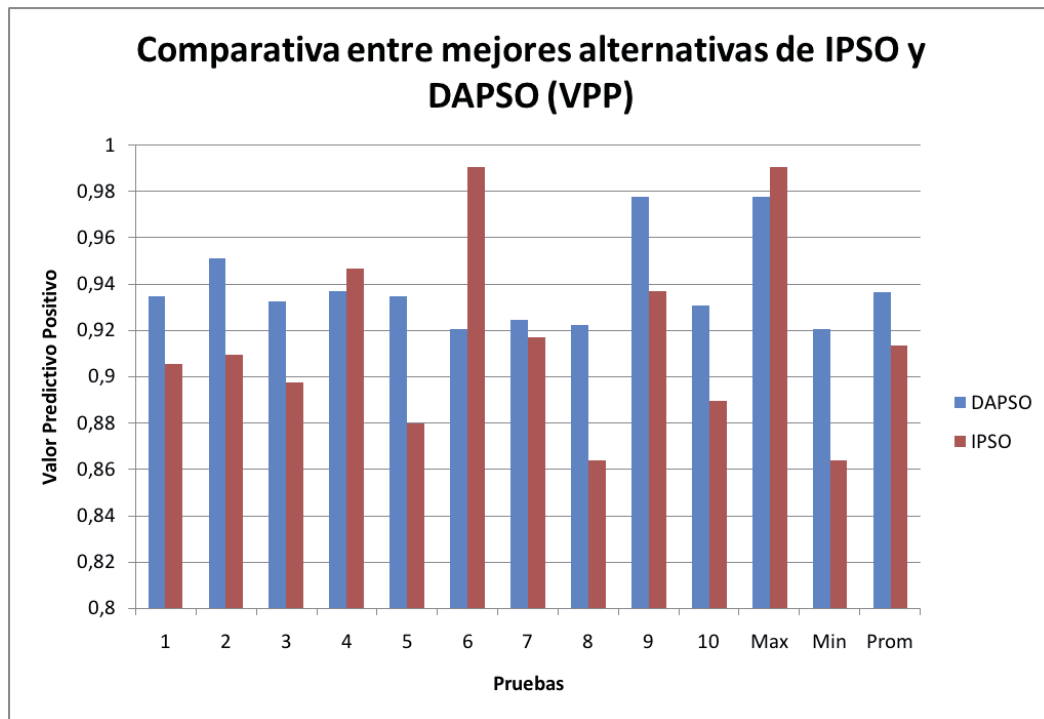


Figura 7.7: Comparativa del VPP

Valor Predictivo Negativo

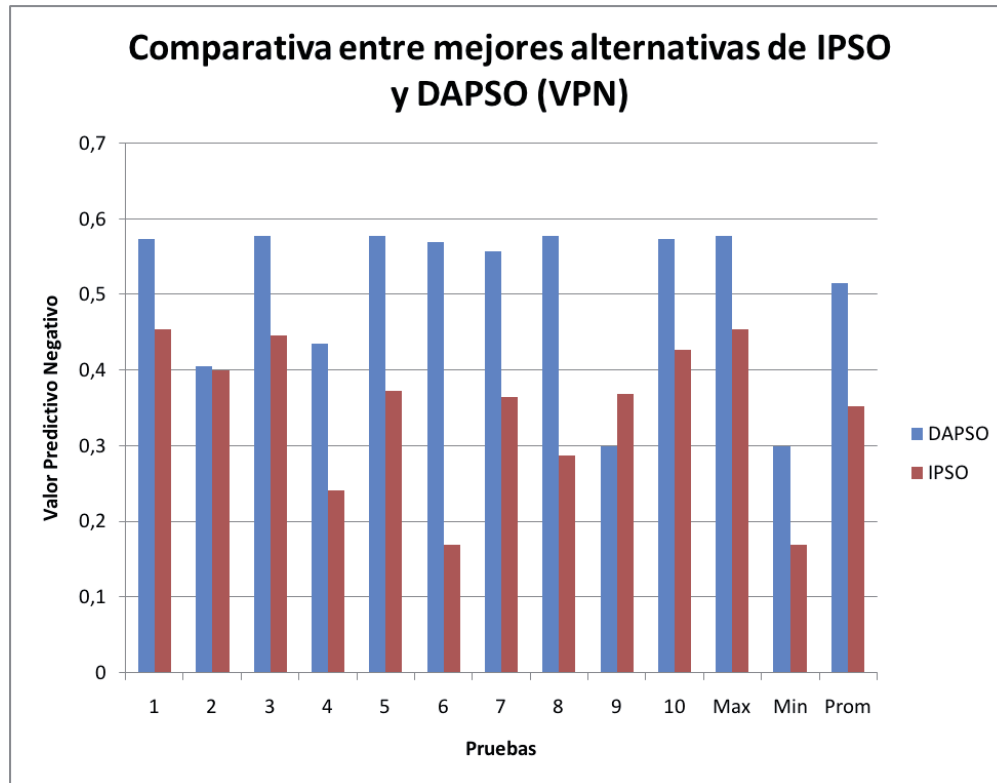


Figura 7.8: Comparativa del VPN

Como se muestra en el gráfico 7.7, el mayor valor del VPP se presenta para la alternativa IPSO, sin embargo presenta una gran variabilidad dependiente de la muestra que se presente. A diferencia de esto, el modelo que utiliza DAPSO presenta una mayor regularidad en las distintas muestras utilizadas, lo que permite tener una mayor seguridad en los resultados obtenidos. En el gráfico 7.8 se aprecia mas claramente la diferencia que existe entre las dos alternativas, ya que el modelo con IPSO presenta un muy bajo valor predictivo negativo en todas las muestras, a diferencia del que se realiza con DAPSO que presenta una regularidad en la mayoría de las muestras.

En conclusión se puede mencionar que entre todas las alternativas probadas, la mejor opción está dada por la tangente hiperbólica como función de activación para la capa oculta y una función Sigmoidal para la capa de salida, teniendo como algoritmo de optimización DAPSO. Este modelo fue el que obtuvo los mejores resultados en cuanto a exactitud y valores predictivos, por lo cual servirá como inicio para comparar con otros modelos que a futuro se realizarán.

7.7 Función Softmax

Los métodos anteriormente descritos carecen de un grado de probabilidad en la salida de la red. La función softmax asegura que la salida, que en este caso representa a las clases de la clasificación, sea positivos y que la suma de estas salidas sea igual a 1. La salida de cada clase representa la probabilidad que ese parámetro pertenezca a la clase. A continuación se realizaron pruebas con las dos variantes de PSO descrita en la sección anterior, con el fin de mejorar los resultados obtenidos. En ambos casos, la función de activación en la capa oculta será una función sigmoideal y para la capa de salida se empleara la función softmax.

Al aplicar ésta función a la red neuronal, la topología de la red cambia, esto debido a que la salida de la red esta condicionada a la cantidad de clases que posea la clasificación. Para este caso se continuará con la misma cantidad de nodos de entrada y nodos ocultos utilizados en las pruebas anteriores, modificándose la cantidad de nodos de salida de la red. Los parámetros son descritos a continuación.

Cuadro 7.23: Parámetros de entrada Red softmax

| Parámetros de Entrada | Valores |
|-----------------------|--------------------------------|
| Iteraciones | 300 |
| Nodos de Entrada | 12 |
| Nodos Ocultos | 5 |
| Nodos Salida | 2 (clases: lesionados, ilesos) |
| % Entrenamiento | 80 % |
| % Pruebas | 20 % |
| Datos | 10 % Total |

Para los parámetros de las variantes de PSO se utilizaron.

Cuadro 7.24: Parámetros de entrada PSO con función softmax

| Parámetros de Entrada | Valores |
|------------------------|---------------|
| Tamaño de la Población | 15 partículas |
| Componente cognitiva | 2 |
| Componente social | 1.75 |
| Velocidad máxima | 1 |
| Posición máxima | 1 |
| (IPSO) W_{max} | 0.75 |
| (IPSO) W_{min} | 0.1 |
| (IPSO) μ | 1.0005 |
| (DAPSO) $W_{inicial}$ | 0.8 |
| (DAPSO) α | 0.4 |
| (DAPSO) β | 0.8 |

7.7.1 Softmax con IPSO

Entrenamiento

Cuadro 7.25: Sigmoidal- Softmax con DAPSO (Entrenamiento)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|---------|--------|-----------|---------------|--------------|--------|--------|
| 1 | 0,3225 | 0,6774 | 1 | 0,677 | 1 | 0,0042 |
| 2 | 0,2998 | 0,7001 | 0,7333 | 0,6983 | 0,9796 | 0,1167 |
| 3 | 0,3112 | 0,6887 | 0,5169 | 0,7919 | 0,7319 | 0,5987 |
| 4 | 0,2912 | 0,7087 | 0,6834 | 0,7112 | 0,9582 | 0,1879 |
| 5 | 0,3239 | 0,6760 | — | 0,6760 | 1 | 0 |
| 6 | 0,3043 | 0,6956 | 0,5646 | 0,7190 | 0,9023 | 0,2643 |
| 7 | 0,3239 | 0,6760 | — | 0,6760 | 1 | 0 |
| 8 | 0,2472 | 0,7527 | 0,6807 | 0,7721 | 0,8998 | 0,4458 |
| 9 | 0,3239 | 0,6760 | — | 0,6760 | 1 | 0 |
| 10 | 0,3239 | 0,6760 | — | 0,6760 | 1 | 0 |
| Max | 0,3239 | 0,7527 | 1 | 0,7919 | 1 | 0,5987 |
| Min | 0,2472 | 0,6760 | 0,5169 | 0,6760 | 0,7319 | 0 |
| Prom | 0,3072 | 0,6927 | 0,6965 | 0,7074 | 0,9472 | 0,1617 |
| Desv | 0,0242 | 0,0242 | 0,1694 | 0,0427 | 0,0854 | 0,2145 |

Prueba

Cuadro 7.26: Sigmoidal- Softmax con DAPSO (Pruebas)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|---------|---------------|---------------|---------------|---------------|---------------|---------------|
| 1 | 0,3457 | 0,6542 | — | 0,6542 | 1 | 0 |
| 2 | 0,3347 | 0,6652 | 0,6052 | 0,6686 | 0,9684 | 0,0916 |
| 3 | 0,3126 | 0,6873 | 0,5441 | 0,7731 | 0,7389 | 0,5896 |
| 4 | 0,3168 | 0,6832 | 0,6666 | 0,6847 | 0,9557 | 0,1673 |
| 5 | 0,3457 | 0,6542 | — | 0,6542 | 1 | 0 |
| 6 | 0,3209 | 0,6790 | 0,5692 | 0,7030 | 0,8821 | 0,2948 |
| 7 | 0,3457 | 0,6542 | — | 0,6542 | 1 | 0 |
| 8 | 0,2630 | 0,7369 | 0,6875 | 0,7508 | 0,8947 | 0,4382 |
| 9 | 0,3457 | 0,6542 | — | 0,6542 | 1 | 0 |
| 10 | 0,3457 | 0,6542 | — | 0,6542 | 1 | 0 |
| Max | 0,3457 | 0,7369 | 0,6875 | 0,7731 | 1 | 0,5896 |
| Min | 0,2630 | 0,6542 | 0,5441 | 0,6542 | 0,7389 | 0 |
| Prom | 0,3276 | 0,6723 | 0,6145 | 0,6851 | 0,9440 | 0,1581 |
| Desv | 0,0263 | 0,0263 | 0,3264 | 0,0439 | 0,0848 | 0,2141 |

Esta alternativa clasificado de manera correcta, en promedio, en un 67,23 % las personas que resultan en el estado lesionado o ileso. La sensibilidad obtenida fue del 70,3 % cifra correspondiente al porcentaje de personas bien clasificadas en el estado lesionado, por su parte la especificidad resultó ser un 56,92 % correspondiendo al porcentaje de personas bien clasificadas en el estado ileso. En cuanto a la seguridad del resultado, se tiene un valor predictivo positivo de 88,21 %, lo que quiere decir que dicho porcentaje personas detectadas como lesionadas están realmente lesionadas, en tanto, el valor predictivo negativo fue de 29,48 %, este significa que éste porcentaje corresponde a las personas detectadas como ilesas que están realmente ilesas.

7.7.2 Softmax con DAPSO

Entrenamiento

Cuadro 7.27: Sigmoidal- Softmax con DAPSO (Entrenamiento)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|-------------|--------|-----------|---------------|--------------|--------|--------|
| 1 | 0,1884 | 0,8115 | 0,8058 | 0,8132 | 0,9321 | 0,5684 |
| 2 | 0,1891 | 0,8108 | 0,8053 | 0,8125 | 0,9321 | 0,5663 |
| 3 | 0,1946 | 0,8053 | 0,7810 | 0,8131 | 0,9202 | 0,5736 |
| 4 | 0,1894 | 0,8105 | 0,8041 | 0,8124 | 0,9315 | 0,5663 |
| 5 | 0,1894 | 0,8105 | 0,8041 | 0,8124 | 0,9315 | 0,5663 |
| 6 | 0,1908 | 0,8091 | 0,8011 | 0,8115 | 0,9305 | 0,5643 |
| 7 | 0,1867 | 0,8132 | 0,7944 | 0,8194 | 0,9243 | 0,5892 |
| 8 | 0,2042 | 0,7957 | 0,768 | 0,8040 | 0,9182 | 0,5487 |
| 9 | 0,2042 | 0,7957 | 0,7720 | 0,8029 | 0,9202 | 0,5446 |
| 10 | 0,2039 | 0,7960 | 0,7849 | 0,7992 | 0,9279 | 0,5300 |
| Max | 0,2042 | 0,8132 | 0,8058 | 0,8194 | 0,9321 | 0,5892 |
| Min | 0,1867 | 0,7957 | 0,7689 | 0,7992 | 0,9182 | 0,5300 |
| Prom | 0,1941 | 0,8058 | 0,7922 | 0,8101 | 0,9269 | 0,5618 |
| Desv | 0,0072 | 0,0072 | 0,0143 | 0,0060 | 0,0056 | 0,0165 |

Prueba

Cuadro 7.28: Sigmoidal- Softmax con DAPSO (Pruebas)

| Pruebas | Error | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|-------------|---------------|---------------|---------------|---------------|---------------|---------------|
| 1 | 0,1790 | 0,8209 | 0,7556 | 0,8418 | 0,9150 | 0,6045 |
| 2 | 0,1763 | 0,8236 | 0,7643 | 0,8423 | 0,9189 | 0,6045 |
| 3 | 0,1900 | 0,8099 | 0,7228 | 0,8394 | 0,8992 | 0,6045 |
| 4 | 0,1832 | 0,8168 | 0,7457 | 0,8397 | 0,9110 | 0,6 |
| 5 | 0,1790 | 0,8209 | 0,7616 | 0,8393 | 0,9189 | 0,5954 |
| 6 | 0,1804 | 0,8195 | 0,7602 | 0,8378 | 0,9189 | 0,5909 |
| 7 | 0,1859 | 0,8140 | 0,7272 | 0,8441 | 0,8992 | 0,6181 |
| 8 | 0,1969 | 0,8030 | 0,7175 | 0,8306 | 0,9011 | 0,5772 |
| 9 | 0,1873 | 0,8126 | 0,7386 | 0,8363 | 0,9090 | 0,5909 |
| 10 | 0,1887 | 0,8112 | 0,7455 | 0,8312 | 0,9150 | 0,5727 |
| Max | 0,1900 | 0,8236 | 0,7643 | 0,8441 | 0,9189 | 0,6181 |
| Min | 0,1763 | 0,8030 | 0,7175 | 0,8306 | 0,8992 | 0,5727 |
| Prom | 0,1847 | 0,8152 | 0,7439 | 0,8382 | 0,9106 | 0,5959 |
| Desv | 0,0063 | 0,0063 | 0,0169 | 0,0044 | 0,0081 | 0,0136 |

Esta variante clasificado de manera correcta en un 82,36 % promedio a las personas que resultan en el estado lesionado o ileso. La sensibilidad obtenida fue del 84,23 % cifra correspondiente al porcentaje de personas bien clasificadas en el estado lesionado, por su parte la especificidad resultó ser un 76,43 % correspondiendo al porcentaje de personas bien clasificadas en el estado ileso. En cuanto a la seguridad del resultado, se tiene un valor predictivo positivo de 91,89 %, lo que quiere decir que el 91,89 % de las personas detectadas como lesionadas están realmente lesionadas, en tanto, el valor predictivo negativo fue de 60,45 %, este significa que éste porcentaje corresponde a las personas detectadas como ilesas que están realmente ilesas.

7.7.3 Comparativa entre las dos variantes PSO

Errores

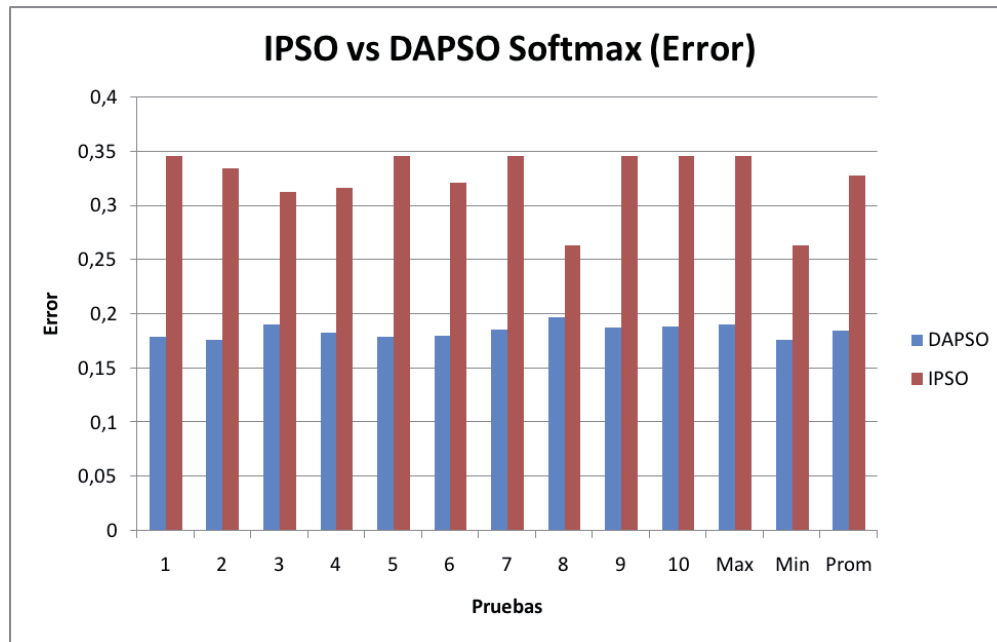


Figura 7.9: Comparativa entre IPSO-DAPSO Softmax (Error)

Valor Predictivo Positivo

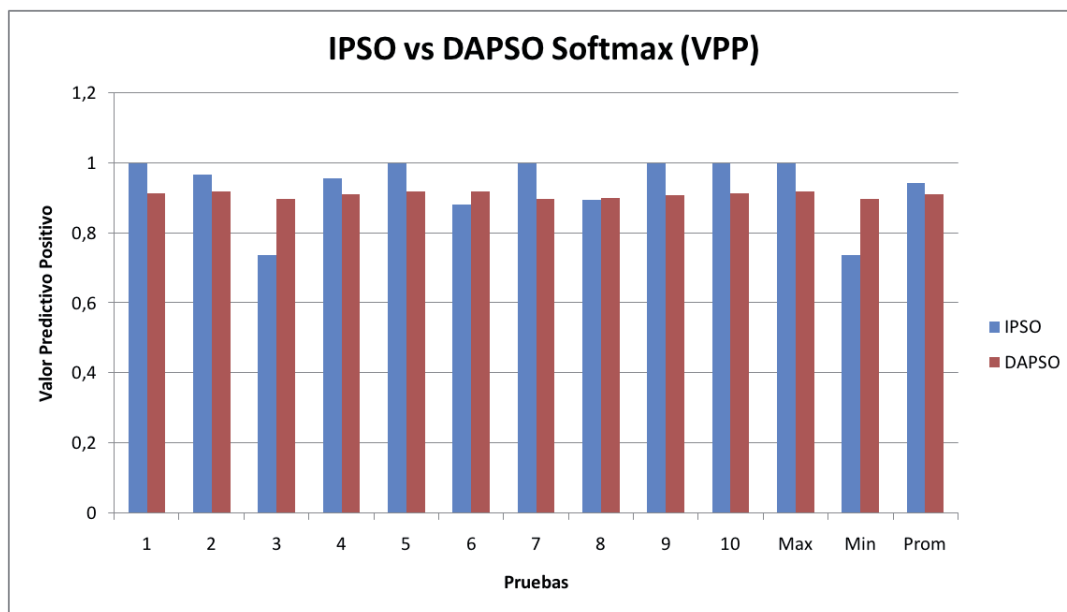


Figura 7.10: Comparativa entre IPSO-DAPSO Softmax (VPP)

Valor Predictivo Negativo

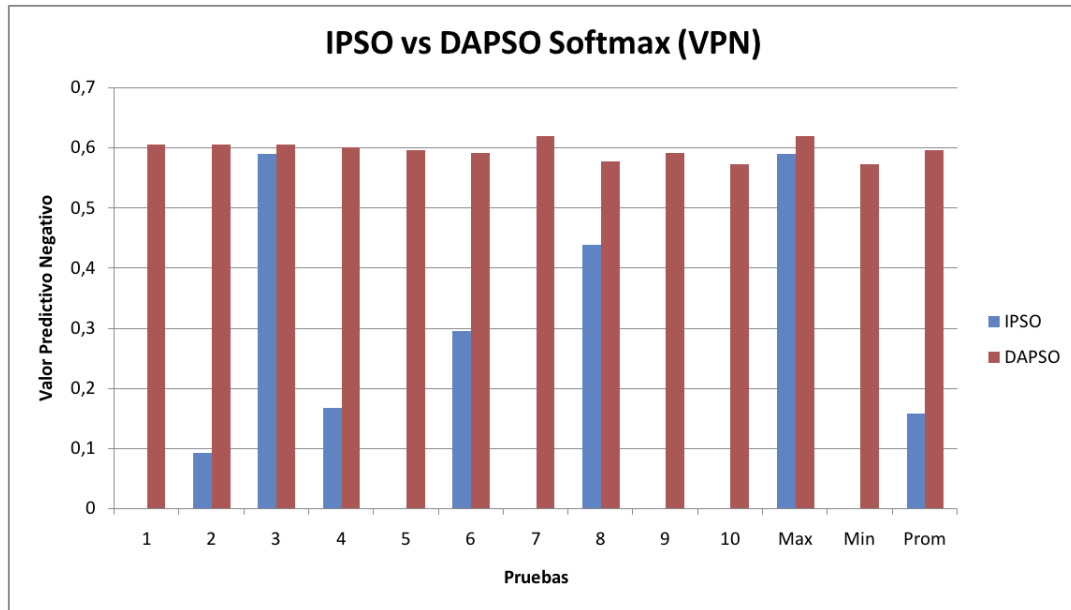


Figura 7.11: Comparativa entre IPSO-DAPSO Softmax (VPN)

Como se muestra en el gráfico 7.10, el mayor valor del VPP se presenta para la alternativa IPSO, sin embargo si se analiza el gráfico 7.11 se puede ver que el porcentaje de los valores predictivos negativos bajan considerablemente en comparación a los obtenidos con DAPSO. Es por ello que el gráfico 7.9 permite ver que el menor error es proporcionado por el modelo optimizado mediante DAPSO

En conclusión se puede mencionar que entre todas las alternativas probadas, la mejor opción está dada por DAPSO con funciones de activación Sigmoidal y Softmax en capa oculta y capa de salida respectivamente. Este modelo fue el que obtuvo los mejores resultados en cuanto a exactitud y valores predictivos, por lo que queda como base para otro tipo de comparaciones.

7.8 Comparativa entre las mejores alternativas

Se probaron diversas alternativas que se agruparon en dos grandes grupos. En el primer grupo se evaluó las distintas combinaciones entre funciones de activación para capa oculta y de salida con dos funciones específicas (Sigmoidal y Tangente Hiperbólica). A todas las combinaciones anteriores se evaluaron dos métodos de optimización (IPSO y DAPSO). De lo anterior se pudo seleccionar la mejor alternativa con el fin de poder compararla con el segundo grupo, que consistió en probar las mismas técnicas de optimización, con la diferencia de modificar la función de transferencia en la capa de salida. Para ello se eligió la función softmax, la cual tiene propiedades probabilísticas, que permite

obtener clasificar por medio de una estimación del porcentaje de pertenecer a una clase determinada.

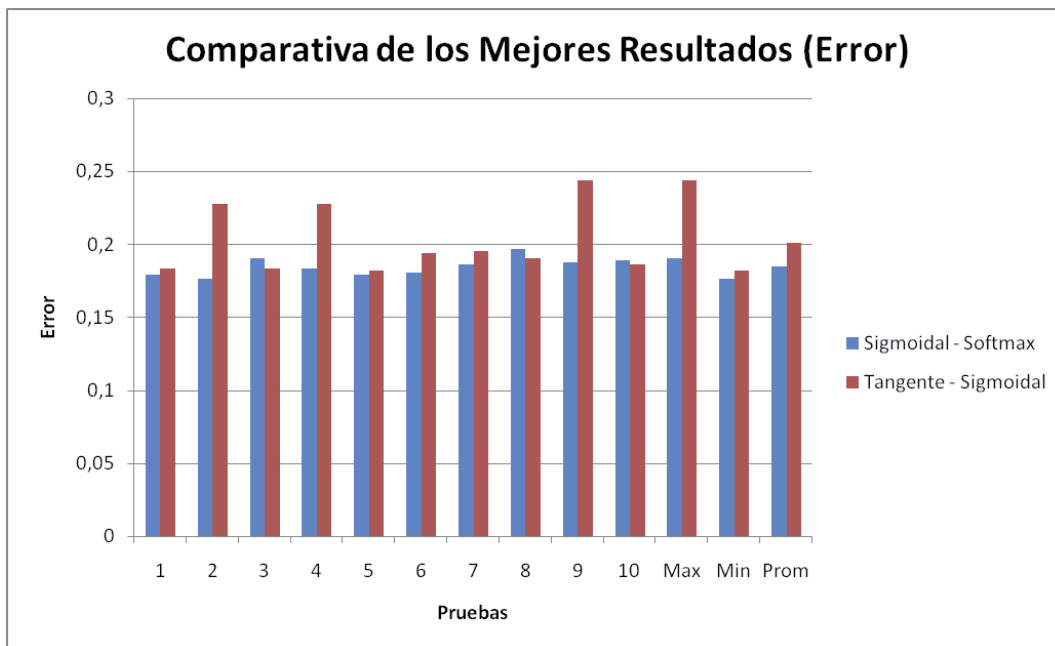


Figura 7.12: Comparativa entre los 2 mejores resultados (Error)

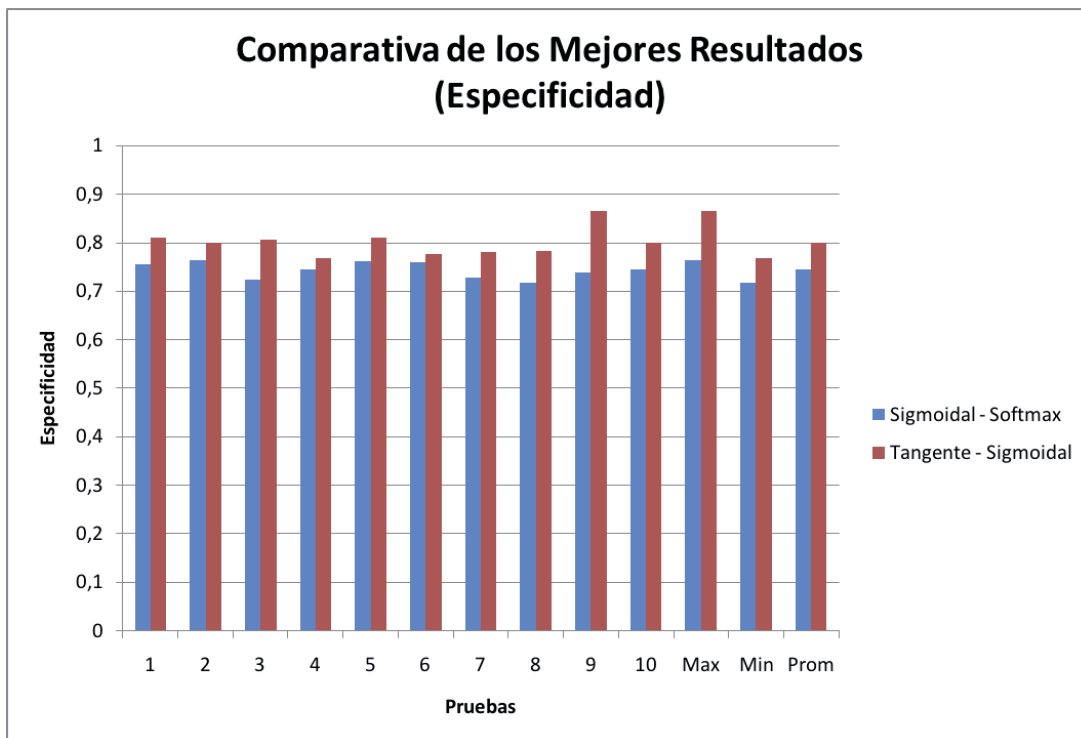


Figura 7.13: Comparativa entre los 2 mejores resultados (Especificidad)

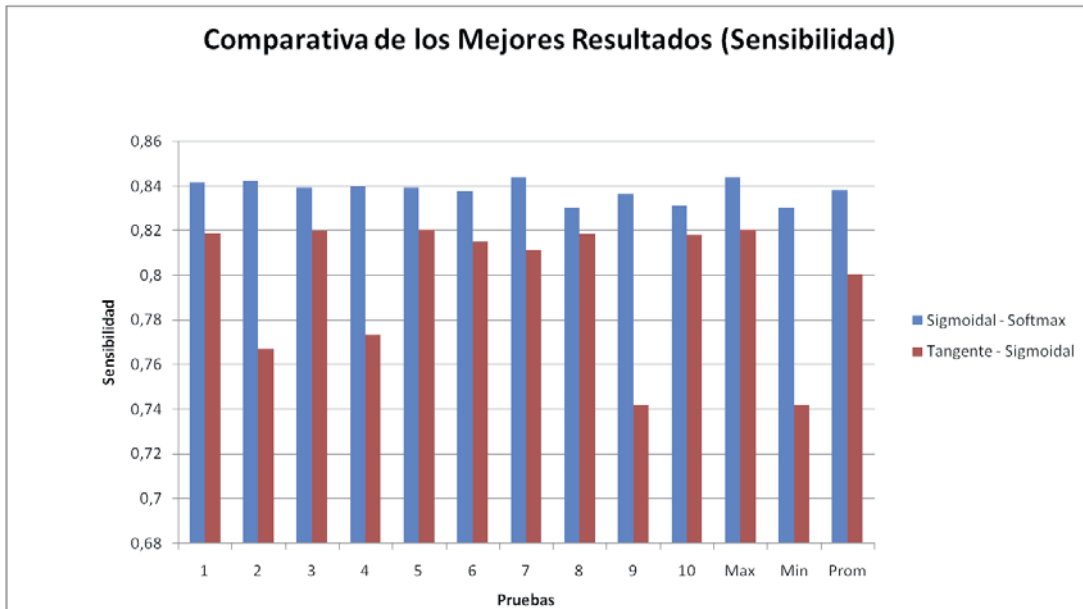


Figura 7.14: Comparativa entre los 2 mejores resultados (Sensibilidad)

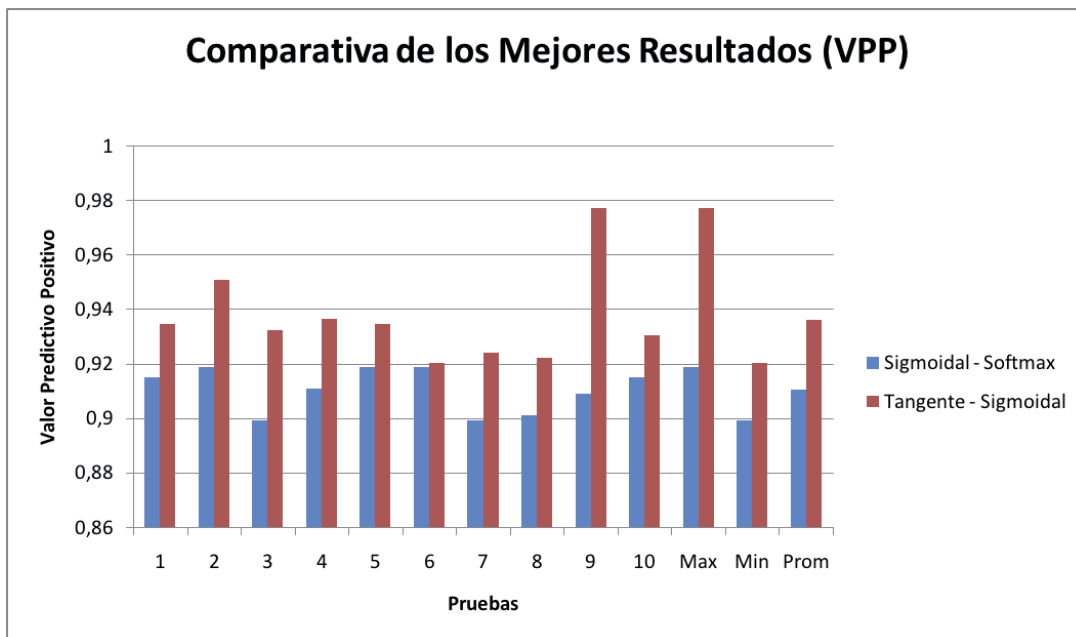


Figura 7.15: Comparativa entre los 2 mejores resultados (VPP)

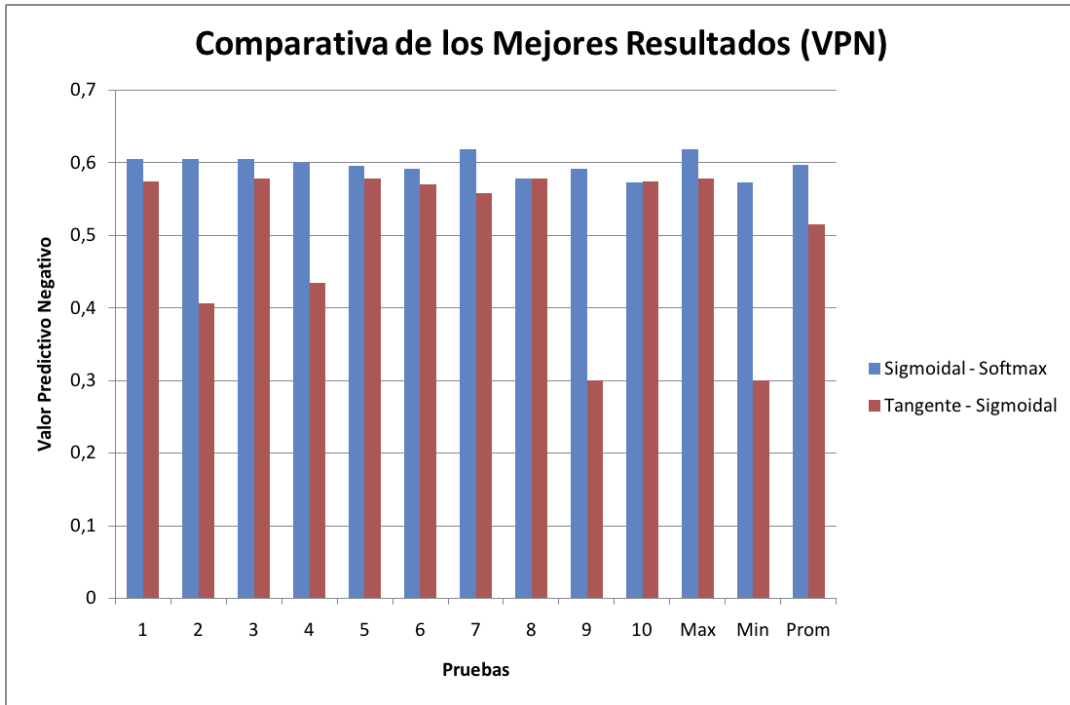


Figura 7.16: Comparativa entre los 2 mejores resultados (VPP)

A continuación se muestra en los gráficos 7.12, 7.13, 7.14, 7.15 y 7.16, ninguno de los dos modelos se destaca en forma notoria por sobre el otro. Por ejemplo si se analiza el error que entregan ambos modelos, los mejores resultados los brinda DAPSO con función de activación softmax, sin embargo si se observa la sensibilidad, el mejor resultado lo entrega el modelo DAPSO con funciones Tangente Hiperbólica para capa oculta y Sigmoidal para capa de salida. Es por ello que se debe evaluar los modelo de forma general, para visualizar cual de ellos permite clasificar a las personas que resultan lesionadas o ilesas de mejor forma. En este punto el modelo DAPSO con softmax obtiene una ventaja ya que clasifica de mejor forma a los individuos que resultan ilesos en un incidente automovilístico, y la diferencia es mínima para los individuos que resultan lesionados en comparación con el otro modelo.

8 Conclusiones

A través del presente informe se ha logrado comprender las distintas tipos de redes neuronales artificiales que existen, sus diferencias y como éstas pueden ser utilizadas en los distintos tipos de problemas. Al revisar la literatura se observaron las diferentes funciones de transferencias que poseen las redes, las cuales le dan distintas propiedades que las hacen una herramienta potente al momento de evaluar su utilización en algún problema determinado.

Se estudió un tipo de algoritmo evolutivo (PSO), lo que permitió captar la eficiencia que posee éste y además su simplicidad, lo que lo hace una buena elección al momento de decidir sobre un método para el entrenamiento de una red neuronal. Se pudo estudiar las distintas alternativas de un algoritmo evolutivo, en este caso PSO, lo que permitió tener una visión más completa de las técnicas existentes en esta área y así poder tomar una mejor decisión al momento de tener que tomar partido por una en particular.

El tratamiento realizado sobre los datos, permitió establecer las variables que fueron utilizadas para la simulación del problemas, esto permitió dar un mayor entendimiento a los resultados que se obtuvieron en las pruebas realizadas con los distintos modelos.

El estudio de las métricas dio un mayor entendimiento en cuanto a lo que el modelo arrojó, ya que permitió medir de forma cuantitativa los resultados obtenidos, y no se presta para interpretaciones del autor. La incorporación de la función softmax permitió darle un enfoque mas probabilístico a la salida de la red, haciendo que la red cuente con tantas salidas como clases tiene el modelo. Esta función cambió la topología de la red, lo que hizo que se agruparan en un nuevo grupo de modelos, para así ser comparados de mejor forma.

Los modelos realizados en el trabajo, permitieron dar un contraste diferente, ya que al contar con resultados concretos se pudo observar el comportamientos de las distintas variables que van influyendo dentro de los problemas que se han abordado. El contrastar modelos permitió visualizar diferencias entre las distintas alternativas desarrolladas en el primer grupo evaluado. En el segundo grupo las diferencias no fueron tan marcadas, pero si hubo diferencias que permitieron escoger la mejor alternativa. Una vez teniendo los mejores modelos de cada grupo, se contrastaron con el fin de elegir el mejor de ellos, fue aquí donde se produce una dificultad, ya que no fue notoria la mejora de uno u otro, lo que hizo que se evaluaran otras variables, distintas al error. Ya con la visión mas amplia se pudo escoger el modelo que dio los mejores resultados, escogiéndose la alternativa DAPSO con función de activación Softmax.

Se pudo cumplir con los objetivos específicos establecidos en un comienzo. Tomando en cuenta el trabajo hecho, se propusieron modelos que permitieron diseñar la severidad de los accidentes en la región metropolitana (separada en zonas) pudiendo concluir con el trabajo propuesto.

Bibliografía

- [1] Felix Castro. Ls-svm basado en optimización por enjambres de partículas para clasificación de accidentes de tránsito. Technical report, Pontificia Universidad Católica de Valparaíso, 2010.
- [2] Organización Panamericana de la Salud. Informe sobre el estado de la seguridad vial en la región de las américas. 2009.
- [3] OMS. Prevención de la violencia y los traumatismos y discapacidad, Agosto 2011.
- [4] CONASET. Estadísticas generales, Agosto 2011.
- [5] P.I. Viñuela and I.M.G. León. *Redes de neuronas artificiales: un enfoque práctico*. Pearson Educación, 2004.
- [6] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Macmillan, New York, 1994.
- [7] Michael I. Jordan. *Attractor dynamics and parallelism in a connectionist sequential machine*, pages 531–546. IEEE Press, Piscataway, NJ, USA, 1986.
- [8] Michael I. Jordan. *Serial order: A parallel distributed processing approach*, volume 121, pages 471–495. Elsevier, 1997.
- [9] Jeffrey L. Elman. Finding structure in time. *Cognitive Science*, 14(2):179–211, 1990.
- [10] Manuel Mena. Predicción de captura de anchoveta utilizando redes wavelets con optimización por enjambre de partículas. Technical report, 2008.
- [11] Nikolay Nikolaev and Hitoshi Iba. *Adaptive Learning of Polynomial Networks: Genetic Programming, Backpropagation and Bayesian Methods (Genetic and Evolutionary Computation)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.
- [12] Pedro Antonio Gutierrez Peña. *Nuevos Modelos de Redes Neuronales Evolutivas y Regresión Logística Generalizada utilizando Funciones de Base. Aplicaciones*. PhD thesis, Departamento de Ciencias de la Computación e Inteligencia Artificial, Junio 2009.
- [13] Dunne and Cambell. On the pairing of the softmax activation and cross-entropy penalty function and the derivation of the softmax activation function. *Conf. on Neural Networks, Melb.*, pages 181–185, 1997.
- [14] J.F. Kennedy, J. Kennedy, R.C. Eberhart, and Y. Shi. *Swarm intelligence*. The Morgan Kaufmann Series in Artificial Intelligence. Morgan Kaufmann Publishers, 2001.
- [15] Marco Dorigo. Particle swarm optimization. 2008, Agosto 2011.

- [16] R. Umarani and V. Selvi. Particle swarm optimization-evolution, overview and applications. *International Journal of Engineering Science and Technology*, 2:2802–2806, 2010.
- [17] *A discrete binary version of the particle swarm algorithm*, volume 5, 1997.
- [18] Yu Liu, Zheng Qin, Zhewen Shi, and Jiang Lu. Center particle swarm optimization. *Neurocomput.*, 70:672–679, January 2007.
- [19] Ioannis G. Tsoulos and Athanassios Stavrakoudis. Enhancing pso methods for global optimization. *Applied Mathematics and Computation*, pages 2988–3001, 2010.
- [20] Yuhui Shi and Russell Eberhart. A modified particle swarm optimizer. *Proc. of the IEEE Int.*, pages 69–73, 1998.
- [21] Bin Jiao, Zhigang Lian, and Xingsheng Gu. A dynamic inertia weight particle swarm optimization algorithm. *Chaos, Solitons & Fractals*, 37(3):698–705, August 2008.
- [22] Xueming Yang, Jinsha Yuan, Jiangye Yuan, and Huina Mao. A modified particle swarm optimizer with dynamic adaptation. *Applied Mathematics and Computation*, 189(2):1205–1213, 2007.
- [23] S. N. Omkar, Rahul Khandelwal, T. V. S. Ananth, G. Narayana Naik, and S. Gopalakrishnan. Quantum behaved particle swarm optimization (qpso) for multi-objective design optimization of composite structures. *Expert Syst. Appl.*, 36:11312–11322, October 2009.
- [24] Jing Liu, Wenbo Xu, and Jun Sun. Quantum-behaved particle swarm optimization with mutation operator. In *Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence*, ICTAI '05, pages 237–240, Washington, DC, USA, 2005. IEEE Computer Society.
- [25] Burgueñoa, García-Bastosb, and González-Buitrago. Las curvas roc en la evaluación de las pruebas diagnósticas. 104:661–670, 1995.
- [26] L Mussone, A Ferrari, and M Oneta. An analysis of urban collisions using an artificial intelligence model. *Accident analysis and prevention*, 31(6):705–718, 1999.
- [27] So Young Sohn and Sung Ho Lee. Data fusion, ensemble and clustering to improve the classification accuracy for the severity of road traffic accidents in korea. *Safety Science*, 41(1):1–14, 2003.
- [28] Miao M. Chong, Ajith Abraham, and Marcin Paprzycki. Traffic accident analysis using decision trees and neural networks. *CoRR*, 2004.
- [29] Juan Suñiga. Aplicación de redes bayesianas para identificar factores determinantes en accidentes de tránsito que afecten a las personas. Technical report, Universidad de Santiágo de Chile, 2008.

- [30] Ali Payidar Akgungor and Erdem Dogan. An application of modified smeed, adapted andreassen and artificial neural network accident models to three metropolitan cities of turkey. *Scientific Research and Essay*, 4:906–913, July 2009.
- [31] World Health Organization. *Informe Sobre la Situación Mundial de la Seguridad Vial: Es Hora de Pasar a la Acción*. World Health Organization, 2009.

A ANEXO

A.1 Plantilla SIEC2

REGISTRO DE ACCIDENTES EN EL TRANSITO Y FERROVIARIOS

SIEC 2

Membrete Comisaría o Destacamento Cód. Unidad Nº. Formulario

IDENTIFICACION

CLASE ACCIDENTE → TRANSITO 1 FERROVIARIO 2 Fecha → Hora →

SUBSECTOR (Si sucedió FUERA DEL SECTOR JURISDICCIONAL, anote "99" y Nombre Comuna en línea siguiente)

COMUNA

TIPO DE ACCIDENTE (Marque SOLO una alternativa)

| | | | | | | | | | |
|---------------|----|----------------------|----|-----------------------------------|----|---------------------|-----------|------------------|----|
| ATROPELLO | 10 | IMPACTO C/ ANIMAL | 40 | CHOQUE CON OTRO VEHICULO DETENIDO | | | VOLCADURA | 70 | |
| CAIDA | 20 | | | FRONTE/FRENTE | 61 | LADO/POSTERIOR | 66 | INCENDIO | 80 |
| COLISION | | | | FRONTE/LADO | 62 | POSTERIOR/FRENTE | 67 | DESCARRILAMIENTO | 90 |
| Frontal | 31 | CHOQUE CON OBJETO | | FRONTE/POSTERIOR | 63 | POSTERIOR/LADO | 68 | | |
| Lateral | 32 | Frontal | 51 | LADO/FRENTE | 64 | POSTERIOR/POSTERIOR | 69 | | |
| Por Alcance | 33 | Lateral | 52 | LADO/LADO | 65 | (Causante/Pasivo) | | OTRO TIPO | 99 |
| Perpendicular | 34 | Posterior | 53 | | | | | | |

UBICACION RELATIVA (Marque ALTERNATIVA y complete datos que correspondan) URBANA 1 RURAL 2 VIA FERREA 3

URBANA O RURAL

| | | | | | |
|-------------------------------|----|--------------------------------|----|-----------------------|----|
| TRAMO DE VIA RECTA | 01 | CRUCE CON SEMAFORO FUNCIONANDO | 11 | ENLACE A NIVEL | 21 |
| TRAMO DE VIA CURVA HORIZONTAL | 02 | CRUCE CON SEMAFORO APAGADO | 12 | ENLACE A DESNIVEL | 22 |
| TRAMO DE VIA CURVA VERTICAL | 03 | CRUCE REGULADO POR CARABINERO | 13 | ACCESO NO HABILITADO | 23 |
| ACERA O BERMA | 04 | CRUCE CON SEÑAL "PARE" | 14 | ROTONDA | 24 |
| PUENTE | 05 | CRUCE CON SEÑAL "CEDA EL PASO" | 15 | PLAZA DE PEAJE | 25 |
| TUNEL | 06 | CRUCE SIN SEÑALIZACION | 16 | OTROS NO CONSIDERADOS | 99 |

CALLE RUTA

VIA 1 VIA 2 VIA 3

FRENTE AL NUMERO

UBICACION (desde Km. 0) y Kilómetro (con UN decimal)

VIA FERREA

| | | | | |
|------------------|----|--|---------------------|----|
| RECINTO ESTACION | 01 | Km./Poste <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/> | CRUCE HABILITADO | 03 |
| TRAMO DE VIA | 02 | (con un decimal) | CRUCE NO HABILITADO | 04 |

CAUSA BASAL PROBABLE

(INDIQUE CON "" EL CAUSANTE)

VEHICULOS PARTICIPANTES

| IDENT. | PATENTE | TIPO (+) | SERV. (+) | CONS. (+) | VIA (⊙) | DIREC. (+) | MANIOBRA (+) | MARCA | CODIGO | AÑO |
|--------|---------|----------|-----------|-----------|---------|------------|--------------|-------|--------|-----|
| A | | | | | | | | | | |
| B | | | | | | | | | | |
| C | | | | | | | | | | |
| D | | | | | | | | | | |
| E | | | | | | | | | | |
| F | | | | | | | | | | |
| G | | | | | | | | | | |
| H | | | | | | | | | | |
| I | | | | | | | | | | |
| J | | | | | | | | | | |
| K | | | | | | | | | | |

(Si hay más VEHICULOS, detálíelos en hoja adjunta con el mismo formato de este CUADRO y marque "X")

Notas: (+) Ver CODIGO respectivo en Instructivo SIEC 2
(⊙) Anotar Nro. De Vía de Ubicación Relativa por la que circula cada vehículo

| | | | | | | | | | | | |
|--|---|------------------------|-----------------------|----------------------------|-------------------------|--|---------------------|------------------------------|---|-----------------------|------------------------|
| ESTADO ATMOSFERICO | | | | | | LUMINOSIDAD | | | | SIEI | |
| DESPEJADO | 1 | LLUVIA | 3 | NEBLINA | 5 | DIURNA | 1 | AMANECEER | 3 | LUZ ARTIFICIAL | |
| NUBLADO | 2 | LLOVIZNA | 4 | NIEVE | 6 | NOCTURNA | 2 | ÁTARDECER | 4 | NO EXISTE | 1 |
| UNIDIRECCIONAL 1 | | | | | | BIDIRECCIONAL 2 | | BIDIRECCIONAL CON BANDEJON 3 | | | SI EXISTE 2 |
| CANTIDAD DE PISTAS | | CANTIDAD DE PISTAS IDA | | CANTIDAD DE PISTAS REGRESO | | APAGADA | | | | | |
| (PISTAS DE IDA CORRESPONDEN AL VEHICULO CAUSANTE) | | | | | | ENCENDIDA SUFICIENTE | | | | | ENCENDIDA INSUFICIENTE |
| TIPO DE CALZADA | | | ESTADO CALZADA | | | CONDICION (*) TIPO = CONCRETO/ ASFALTO/ ADOQUIN MIXTO | | | | | |
| CONCRETO (*) | 1 | MIXTO (*) | 4 | BUENO | 1 | SECO | 1 | CON BARRO | 4 | ESCARCHA | |
| ASFALTO (*) | 2 | RIPIO | 5 | REGULAR | 2 | HUMEDO | 2 | CON NIEVE | 5 | GRAVILLA | |
| ADOQUIN (*) | 3 | TIERRA | 6 | MALO | 3 | MOJADO | 3 | CON ACEITE | 6 | OTROS | |
| (*) DEMARCCACION (SI TIPO DE CALZADA ES CONCRETO, ASFALTO, ADOQUIN O MIXTO) | | | Línea Continua | | Pare en Calzada | | Otras Demarcaciones | | | | |
| | | | Línea Discontinua | | Ceda el Paso en Calzada | | BORRADAS | | | | |
| | | | Línea Mixta | | Paso Peatonal | | SIN DEMARCCACION | | | | |
| (Marque TODAS las existentes) | | | | | | | | | | | |

PERSONAS PARTICIPANTES

| CANTIDAD | SEXO | RESULTADO | SOLO PARA CONDUCTORES | | | | |
|--|---------|--|-----------------------|---|-----------------------------|----------------------------|----------------------------|
| | | | CLASE DE LICENCIA | | CONDICION FISICA | NACIONALIDAD | |
| 1 - CONDUCTOR | 1. MAS. | 1. MUERTO 4. LEVE | A1 | B | PP PERMISO PROVISORIO | 1. NORMAL | 1. CHILENA |
| 2 - PASAJERO | 2. FEM. | 2. GRAVE 5. ILESO | A2 | C | BC BOLETA DE CITACION | 2. BAJO INFLUENCIA ALCOHOL | 2. ARGENTINA |
| 3 - PEATON | | 3. M. GRAVE | A3 | D | SL SIN LICENCIA | 3. EBRIEDAD | 3. BRASILEÑA |
| (Agregue ASTERISCO (*) AL CAUSANTE) | | CINT. SEGUR/CASCO (CONDUCT./PASAJ.) | A4 | E | FV FECHA DE CONTROL VENCIDA | 4. BAJO INFLUENCIA DROGAS | 4. BOLIVIANA |
| | | 0 - IGNOR 1-SI 2-NO | AS | F | PE PERMISO EXTRANJERO | 5. FATIGA | 5. PERUANA |
| | | | | | OT OTRAS | | 6. OTRAS SUDAME RICANAS |
| | | | | | | | 9. OTRAS |

| ## | CALID | SEXO | EDAD | RESULT. | CINT. | R.U.N. | VEHIC. | CL.LIC. | COMUNA | CODIGO | C.FIS. | NAC |
|----|-------|------|------|---------|-------|--------|--------|---------|--------|--------|--------|-----|
| 01 | | | | | | | | | | | | |
| 02 | | | | | | | | | | | | |
| 03 | | | | | | | | | | | | |
| 04 | | | | | | | | | | | | |
| 05 | | | | | | | | | | | | |
| 06 | | | | | | | | | | | | |
| 07 | | | | | | | | | | | | |
| 08 | | | | | | | | | | | | |
| 09 | | | | | | | | | | | | |
| 10 | | | | | | | | | | | | |
| 11 | | | | | | | | | | | | |
| 12 | | | | | | | | | | | | |
| 13 | | | | | | | | | | | | |
| 14 | | | | | | | | | | | | |
| 15 | | | | | | | | | | | | |
| 16 | | | | | | | | | | | | |
| 17 | | | | | | | | | | | | |
| 18 | | | | | | | | | | | | |
| 19 | | | | | | | | | | | | |
| 20 | | | | | | | | | | | | |

(Si hay más PARTICIPANTES, delálieles en hoja adjunta con el mismo formato de este CUADRO y marque "X")

| | | | | | | |
|---------------|---------------|------------|----------|-------------|----------------|-----------------|
| CLASIFICACION | DETENIDO(S) 1 | DENUNCIA 2 | S.I.A.T. | CONCURRIO 1 | NO CONCURRIO 2 | SI HUBO MUERTOS |
| | | | | | | MENS: DEL / / |

ACLARATORIA _____

v* B*

SUBCOMISARIO DE LOS SERVICIOS
(Nombre, Grado y Sello)

ENCARGADO S.I.E.C.
(Nombre y Grado)

A.2 Prototipo Aplicación “Accidentes de Tránsito región Metropolitana”

Region Metropolitana

Provincia: Ciudad:

N° Datos:

% de Datos:

Datos de Entrada

Red

N° Nodos de Entrada:

N° Nodos Ocultos:

% de Entrenamiento:

% de Prueba:

Variantes PSO

DAFPO

N° de Partículas:

N° Iteraciones:

Velocidad Max:

Posición Max:

Función de Activación

Tangente Hiperbólica – Tangente Hiperbólica

Opción

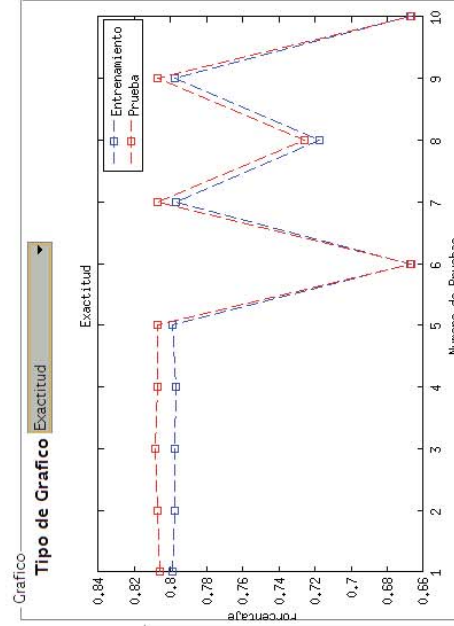
Entrenamiento

Prueba

Tiempo (seg)

Resultados

| | Exactitud | Especificidad | Sensibilidad | VPP | VPN |
|------|-----------|---------------|--------------|--------|--------|
| 1 | 0.7992 | 0.7755 | 0.8066 | 0.9196 | 0.5574 |
| 2 | 0.7975 | 0.7708 | 0.8059 | 0.9176 | 0.5564 |
| 3 | 0.7978 | 0.7727 | 0.8057 | 0.9186 | 0.5553 |
| 4 | 0.7971 | 0.7728 | 0.8047 | 0.9191 | 0.5522 |
| 5 | 0.7988 | 0.7752 | 0.8062 | 0.9196 | 0.5564 |
| 6 | 0.6675 | NaN | 0.6675 | 1 | 0 |
| 7 | 0.7971 | 0.7728 | 0.8047 | 0.9191 | 0.5522 |
| 8 | 0.7177 | 0.6755 | 0.7247 | 0.9304 | 0.2906 |
| 9 | 0.7978 | 0.7734 | 0.8054 | 0.9191 | 0.5543 |
| 10 | 0.6675 | NaN | 0.6675 | 1 | 0 |
| Max | 0.7992 | 0.7755 | 0.8066 | 1 | 0.5574 |
| Min | 0.6675 | 0.6755 | 0.6675 | 0.9176 | 0 |
| Mean | 0.7638 | NaN | 0.7639 | 0.9363 | 0.4175 |
| Desv | 0.0566 | NaN | 0.0596 | 0.0338 | 0.2350 |



Calcular

Salir

A.3 Accidentes separados por Zona

| ID | Ciudad | Accidentes | Zona |
|-------|------------------|------------|---------------|
| 13101 | SANTIAGO | 77651 | Zona centro |
| 13106 | ESTACIÓN CENTRAL | 32589 | Zona centro |
| 13127 | RECOLETA | 20780 | Zona norte |
| 13108 | INDEPENDENCIA | 17487 | Zona norte |
| 13114 | LAS CONDES | 45827 | Zona oriente |
| 13120 | ÑUÑO A | 32997 | Zona oriente |
| 13119 | MAIPÚ | 50593 | Zona poniente |
| 13117 | LO PRADO | 12662 | Zona poniente |
| 13110 | LA FLORIDA | 41745 | Zona sur |
| 13201 | PUENTE ALTO | 31022 | Zona sur |

| Zonas | N° Accidentes |
|---------------|---------------|
| Zona Norte | 54407 |
| Zona Centro | 128244 |
| Zona Sur | 68080 |
| Zona Oriente | 66607 |
| Zona Poniente | 63611 |
| Total | 380949 |

Accidentes por Zona

Region Metropolitana

