

## Análisis de electroencefalograma usando redes neuronales artificiales

### Electroencephalogram analysis using artificial neural networks

Karina Delgado<sup>1\*</sup>, Sergio Ledesma<sup>1</sup> & Horacio Rostro<sup>1</sup>

<sup>1</sup> Departamento de Ingeniería Electrónica, División de Ingenierías, Campus Irapuato-Salamanca, Universidad de Guanajuato. Carretera Salamanca - Valle de Santiago km 3.5 + 1.8. Comunidad de Palo Blanco Salamanca, Gto. C.P. 36885.

Correo electrónico: kadelgadoa@gmail.com

\*Autor de correspondencia

### Resumen

A través de la electroencefalografía se detecta la comunicación entre señales eléctricas creadas por las neuronas que, al conectarse entre sí, crean conexiones sinápticas. Esta técnica ha sido muy importante en la detección de trastornos neurológicos como la epilepsia. Caracterizada por cambios temporales en el funcionamiento bioeléctrico del cerebro, la epilepsia provoca convulsiones que afectan a la conciencia, el movimiento o la sensibilidad. Las redes neuronales artificiales (RNA) proporcionan modelos con diversas alternativas para detección, clasificación y predicción de muestras mediante el análisis del electroencefalograma a partir de la estructura de los datos, los cuales determinan la topología de la red. Este artículo propone la implementación de un modelo basado en RNA para analizar, clasificar y procesar señales epilépticas a partir del entrenamiento. Particularmente, la base de datos cuenta con muestras que registraron la actividad cerebral de pacientes sanos, pacientes que controlaron las crisis y pacientes que aún registraban oscilaciones en las señales emitidas por la actividad cerebral. Después de aplicar la transformada de Fourier, estas señales se integraron en una matriz aplicando tres tipos de umbral, procediendo a seleccionar los datos de entrada de la RNA para su entrenamiento y validación. Se consideran dos métodos de aprendizaje: redes neuronales multicapa con validación clásica (back propagation) y redes neuronales con validación cruzada (LOOCV, por sus siglas en inglés), para ello se calcula el error cuadrático medio (MSE, por sus siglas en inglés) así como la cantidad de errores por umbral con la finalidad de comparar los resultados obtenidos y precisar el método que proporciona los mejores resultados. Ambas redes se entrenaron usando un método híbrido basado en el templado simulado y el gradiente conjugado. Finalmente, se presenta el análisis de las RNA como sistemas de clasificación a través de los dos métodos en funcionamiento, alcanzando resultados satisfactorios que manifiestan la aplicación como herramienta de apoyo al diagnóstico médico para la detección de este trastorno.

**Palabras clave:** Inteligencia artificial; red neuronal artificial; electroencefalograma; error cuadrático medio; validación cruzada.

### Abstract

Through electroencephalography it is possible to detect the electrical signals created by interconnected neurons that create synaptic connections. This technique has been very important in the detection of neurological disorders such as epilepsy. Characterized by temporary changes in the bioelectric function of the brain, epilepsy causes seizures that affect awareness, movement, or sensation. Artificial neural networks (ANN) provide alternative models for detection, classification and prediction of samples by analyzing the electroencephalogram from the structure of the data, which determine the topology of the network. This article proposes the implementation of a system based on ANN to analyze, classify and process signals from an epileptic training model. In particular, the database has samples with recorded brain activity in healthy patients, patients who controlled the crisis and patients that still recorded oscillations in the signals emitted by the brain activity. After applying the fast Fourier transform, these signals were integrated into a matrix using three types of threshold and selecting the input data of an ANN for training and validation. Two methods of learning are considered: multilayer neural networks with classic validation (back propagation) and neural networks using leave one out crossed validation (LOOCV), for which the mean square error (MSE) and the amount of errors threshold are calculated in order to compare the results obtained and to find the method that provides the best results. Both networks were trained using a hybrid method based on simulated annealing and conjugate gradient. Finally, the analysis of ANN as classification systems through the two methods in operation is presented, achieving satisfactory results that show the application as a tool to support the medical diagnosis for the detection of this disorder.

**Keywords:** Artificial intelligence; artificial neural network; electroencephalogram; mean squared error; cross validation.

**Recibido:** 13 de noviembre de 2016

**Aceptado:** 16 de octubre de 2018

**Publicado:** 08 de abril de 2019

**Como citar:** Delgado, K., Ledesma, S., & Rostro, H. (2019) Análisis de electroencefalograma usando redes neuronales artificiales. *Acta Universitaria* 29, e1672. doi: <http://doi.org/10.15174/au.2019.1672>

## Introducción

Las neuronas que conforman el cerebro establecen comunicación mediante señales eléctricas llamadas impulsos electroquímicos, esta actividad se detecta a través de la electroencefalografía, permitiendo explorar y registrar la actividad eléctrico-cerebral mediante electrodos que se colocan en diferentes sitios del cuero cabelludo y que, conectados a un amplificador y a una grabadora, se convierten en patrones que se pueden analizar en una computadora.

La electroencefalografía desempeña un rol importante en la detección de trastornos neurológicos tales como la esquizofrenia, la epilepsia y un sin fin de psicopatologías diferentes y complejas.

La epilepsia es una enfermedad grave caracterizada por cambios temporales en el funcionamiento bioeléctrico del cerebro. Estos cambios provocan una sincronización neuronal anormal tales como convulsiones que afectan la conciencia, el movimiento o la sensibilidad. La detección temprana en la predicción de este trastorno aumentaría considerablemente la calidad de vida de los pacientes que no pueden ser tratados con éxito mediante estrategias terapéuticas comunes.

Los avances científicos y tecnológicos permiten tener un conocimiento más amplio acerca de los trastornos neurológicos y de las diversas alternativas para su detección, clasificación y predicción mediante el análisis del electroencefalograma (EEG). En ese sentido, técnicas de inteligencia artificial (IA) así como las redes neuronales artificiales (RNA) establecen relaciones complejas entre variables, aportando modelos con importantes propiedades clasificadoras y predictivas a partir de la estructura de los datos, los cuales determinan la topología de la red.

Las RNA basan su diseño en las neuronas biológicas del cerebro humano, permiten el tratamiento no lineal de los datos de acuerdo con el modelo fisiopatológico vigente, son ampliamente utilizadas en una gran variedad de aplicaciones de aprendizaje supervisado y no supervisado para tareas relacionadas con clasificación y reconocimiento de patrones y predicen con gran precisión.

Este modelo matemático inteligente, RNA, se fundamenta en el aprendizaje, lo que simplifica el proceso de clasificación por reconocimiento de patrones, siendo los datos los que determinan el comportamiento de la RNA a través de su estructura y de los parámetros a considerar.

Investigaciones demuestran que las redes neuronales permiten la detección y clasificación de las señales EEG mediante la implementación de diversas técnicas de IA. Castaño, Zapata & Villegas (2009) proponen detectar puntas epilépticas de un EEG, emplean análisis multi-resolución y sistemas híbridos que descomponen la señal con *wavelets* para clasificar propiedades y parámetros estadísticos para su clasificación en una RNA.

Guo, Rivero, Dorado, Rabuñal & Pazos (2010) presentan un método para la detección automática de crisis epiléptica que utiliza características de la longitud de línea basado en la descomposición multi-resolución sobre la transformada *wavelet*, el cual se combina con una RNA para clasificar las señales del EEG sobre la existencia o la no existencia de una crisis.

Srinivasan, Eswaran & Sriraam (2007) proponen una red neuronal basada en un sistema automatizado de detección de EEG de epilepsia que utiliza la entropía aproximada (ApEn, por sus siglas en inglés) como la función de entrada, midiendo la predictibilidad de los valores de la amplitud de corriente de una señal fisiológica basado en sus valores de amplitud anteriores, mediante la implementación de dos redes neuronales: Elman y redes neuronales probabilísticas.

Costa, Oliveira, Rodrigues, Leitão & Dourado (2008) plantean un sistema de clasificación automático para epilepsia basado en redes neuronales y señales EEG, emplean 14 características (extraídas del EEG) con el fin de clasificar el estado del cerebro en uno de los cuatro posibles comportamientos epilépticos: inter-ictal, pre-ictal, ictal y pos-ictal. Los experimentos se realizaron en: a) un solo paciente, b) diferentes pacientes y c) múltiples pacientes, utilizando dos conjuntos de datos.

Las RNA han sido propuestas por diferentes investigadores como sistemas de clasificación para diagnosticar epilepsia a través de métodos diferentes, con resultados satisfactorios, y manifiestan que son capaces de clasificar señales de pacientes sanos, pacientes enfermos y pacientes controlados.

Las RNA aprenden de la experiencia a partir de un proceso llamado entrenamiento, este proceso involucra ajustes de los pesos sinápticos entre neuronas debido al entrenamiento. Una vez que la RNA obtiene un valor óptimo, el ajuste de los pesos se detiene, y es entonces cuando se dice que la RNA ha aprendido.

Una RNA procesa información específica y no puede utilizarse en problemas para los cuales no ha sido diseñada.

En este trabajo se lleva a cabo la implementación de la RNA multicapa con retro propagación (*Backpropagation*). En el proceso de aprendizaje de la red, el entrenamiento supervisado calcula el error cuadrático medio (MSE, por sus siglas en inglés) a través de definir el umbral para la selección de los casos de entrenamiento y datos de entrada, continúa con el proceso de entrenamiento y validación de la RNA para el primer caso y, a su vez, se realiza el proceso de aprendizaje a través de validación cruzada (LOOCV, por sus siglas en inglés), el cual consiste en dejar un caso de entrenamiento fuera. Este caso de entrenamiento se va alternando para realizar el proceso de aprendizaje de manera que, al final, todos forman parte del entrenamiento.

El objetivo de este artículo es implementar la IA a través del estudio y aplicación de redes neuronales en el análisis de EEG para apoyo al diagnóstico médico, desarrollando un modelo basado en RNA para la clasificación del trastorno neurológico mediante un análisis de datos reales y comprobar mediante pruebas que, efectivamente, las RNA son capaces de realizar la clasificación.

Así como lo demuestran las siguientes investigaciones de Özkan, Doğan, Kantar, Akşahin & Erdamar (2016), en la implementación de la RNA realizan el análisis de señales EEG en el dominio de tiempo y frecuencia, lo que les permite obtener las principales características de este trastorno.

Çetin, Çetin & Bozkurt (2015), a través de tres redes neuronales: *Feed Forward Backpropagation*, *Cascade* y *Elman*, clasifican señales EEG para determinar la existencia o no de ataques epilépticos.

## **Materiales y Métodos**

### **Neurona Artificial y Redes Neuronales Artificiales**

La neurona artificial es la unidad básica de una red neuronal. Su procesamiento radica en recibir datos de entrada de otras neuronas y producir una salida a través de conexiones con determinado peso que se modifica mediante el aprendizaje, lo que determina el comportamiento de la neurona. Esto es, realiza la suma ponderada de los datos de entrada con los pesos correspondientes, este término es el encargado de generar la señal de activación que se filtra mediante una función de transferencia para obtener la salida de la neurona.

La simulación de una neurona artificial considera un dato de entrada, representado por la variable  $x$ , que se multiplica por el peso  $w_{ni}$  y su correspondiente valor de umbral  $b_n$ , llamado *bias* o *sesgo*; el resultado obtenido es analizado por una función de activación  $f(a)$  para inmediatamente obtener la salida esperada, como se muestra en la figura 1.

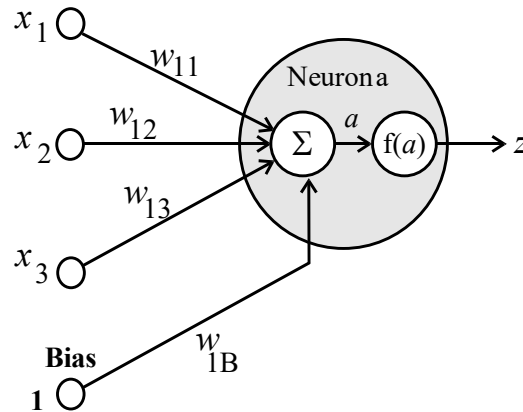


Figura 1. Esquema de una neurona artificial  
Fuente: Elaboración propia.

Matemáticamente este cálculo es expresado mediante la siguiente ecuación:

$$f(a) = \begin{cases} 1 & \text{si } \sum_{j=1}^n w_{nj} x_j + b_n > 0 \\ 0 & \text{en caso contrario} \end{cases} \quad (1)$$

Por lo tanto, una RNA es una colección de neuronas artificiales interconectadas; la base del funcionamiento de las RNA es el reconocimiento de patrones y, a través de ello, obtienen y procesan el conocimiento, partiendo de ejemplos previos ajustando los parámetros de las neuronas mediante un algoritmo de aprendizaje.

La RNA usada en problemas de clasificación denominada perceptrón multicapa (MLP, por sus siglas en inglés) o Red de Propagación hacia Delante por Ripley (1996) traslada la señal de modo unidireccional de la capa actual a la siguiente y consta de: una capa de entrada, una o más capas ocultas y una capa de salida, tal como se observa en la figura 2.

La capa de entrada coordina las muestras (datos) permitiendo su entrada a la RNA, estos datos de entrada son distribuidos a la capa oculta, aplicando ponderaciones distintas a cada muestra y, mediante una función de activación, las muestras se transforman en la salida de la neurona; la capa de salida debe tener una cantidad de neuronas igual a las variables de salida que tenga el problema que se está resolviendo.

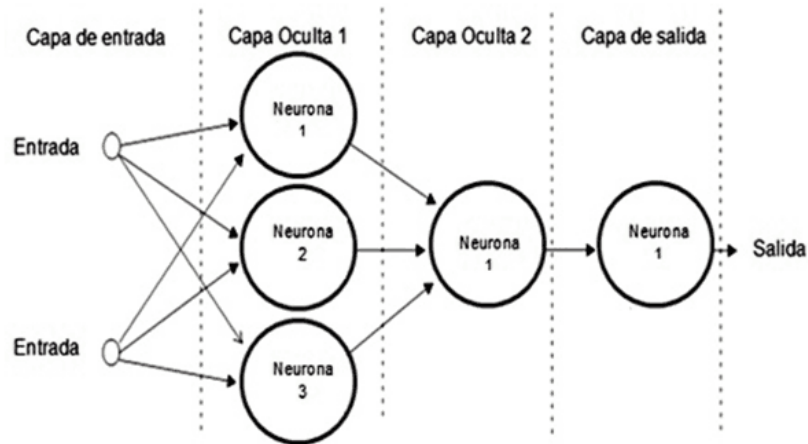


Figura 2. Esquema de una red neuronal perceptrón multicapa.  
Fuente: Elaboración propia.

Una vez que las neuronas establecen comunicación, la información debe fluir de las entradas hacia las salidas, evitando crear ciclos de realimentación; por esta razón, se denominan Redes de Propagación hacia Delante, *feed-forward* o *MLP: Multi Layer Perceptron*.

## Métodos de Aprendizaje

Las conexiones de la red proporcionan la salida deseada a través de métodos de aprendizaje clasificados como: aprendizaje supervisado y aprendizaje no supervisado.

En este caso en particular, para el aprendizaje supervisado se otorga a la red el conjunto de datos (entrada-salida), donde la entrada simboliza las propiedades destacadas que determinan el valor de la salida que la RNA debe presentar. Este tipo de aprendizaje debe ser apto de generalizar, revelando valores de salida para datos de entrada que no fueron analizados previamente.

El algoritmo de aprendizaje realiza el ajuste de los pesos de la red, de manera que es capaz de clasificar sin requerir un punto de partida.

## Método Propuesto: Algoritmo de Aprendizaje.

Para que una RNA profundice en el aprendizaje de la actividad solicitada se requiere llevar a cabo la variación de los pesos a través del proceso denominado entrenamiento, donde cada neurona de la RNA es responsable de adaptar los pesos a los datos de entrada que obtiene de las neuronas que le anteceden.

Por lo tanto, una RNA aprenderá lo que los datos de entrada le enseñen, alcanzando un aprendizaje óptimo si en el entrenamiento se engloban los datos más relevantes, así como la elección del algoritmo de aprendizaje que proporcione resultados inmejorables, logrando obtener la clasificación apropiada de los datos que no pertenecen al entrenamiento.

Dado que las salidas adquiridas para los datos de entrenamiento deben ser lo más semejantes a las salidas pretendidas, es preciso establecer un valor que depende de estas salidas denominado función error.

El propósito de disminuir el error generado por la red radica en evaluar la diferencia entre el valor deseado y el proporcionado por la RNA respecto a los datos de entrada; inclinarse por esta función error condiciona la actividad de la RNA una vez realizado el entrenamiento, además de indicar el momento en el que se debe dejar de ejecutar el entrenamiento.

Los factores que influyen en la búsqueda de los pesos para disminuir la función del error son: la estructura de la red que lleva consigo la disminución del error, la función de error elegida, los datos de entrenamiento y las salidas deseadas.

La función de error implementada en este trabajo, destinada a la clasificación de señales EEG, es el MSE. La ecuación correspondiente es:

$$mse = \frac{1}{n} \sum_{i=1}^n (t_i - z_i)^2 \quad (2)$$

donde se calcula la suma de los cuadrados de las diferencias entre las salidas obtenidas y las salidas pretendidas para determinados datos de entrenamiento.

La RNA que lleva a cabo su aprendizaje haciendo variación a sus pesos implementa el algoritmo de retro propagación, también denominado *backpropagation* o propagación hacia atrás, el cual extiende la estimación y disminución de la función error en dirección a las capas de entrada, adquiriendo la ruta en que deben variar los pesos de las neuronas. O'Connor (1993) manifiesta la disminución del error como: la forma en la que el gradiente descendente evalúa la derivada de la función error MSE, disminuyendo el error total de la red haciendo el ajuste de los pesos.

Los parámetros de la RNA consiguen ser modificados cuando los datos de entrenamiento durante el aprendizaje alcanzan un error mínimo.

Otro algoritmo de aprendizaje para funciones de error continuas (MSE) es el denominado LOOCV o *Validación Cruzada por LOO*, que determina la organización de las clases al medir el error sobre todos los datos de entrenamiento menos uno.

Este método distribuye los datos de entrenamiento en  $n$  entradas, por lo que en este proceso se implican  $n - 1$  iteraciones para realizar el entrenamiento de la RNA, eliminando de manera temporal un dato de entrada. Dado que el entrenamiento se realiza con el resto de los datos de entrada, esto permite obtener el error como la media aritmética de los errores atribuidos en las diferentes iteraciones de los datos de entrada que forman parte del entrenamiento.

En cada iteración el dato de entrada que se elimina de manera temporal se alterna, por lo que al final del algoritmo todos los datos forman parte del entrenamiento.

El método LOOCV se basa en evaluar  $n$  cantidad de veces los datos de entrada, intentando con un dato de entrada distinto. Este procedimiento se describe en la figura 3, el área resaltada como prueba es la que se elimina de manera temporal en la iteración que le corresponde:

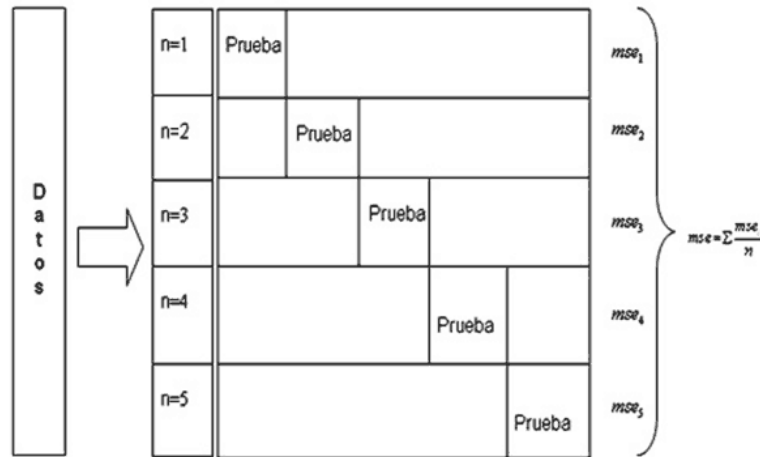


Figura 3. Estructura del modelo validación cruzada con  $n = 5$ .  
Fuente: Elaboración propia.

La validación cruzada se utiliza para estimar el error de generalización. Se aconseja realizar el entrenamiento de la RNA con solo una capa escondida y la menor cantidad posible de neuronas; ya que demasiadas neuronas sobre-entrenarán (*overfitting*) la RNA, lo que hará que aprenda los datos de entrada durante el entrenamiento.

Para prevenir el sobre-entrenamiento es necesario contar con datos de entrenamiento denominados datos de validación, los cuales se encargan de evaluar el error de la RNA una vez realizada cada iteración, además de precisar el instante en el que el error se incrementa. Como resultado del error que se produce de estos datos se ejecuta la detención del entrenamiento.

Si el MSE que se obtiene durante el entrenamiento es mucho más pequeño que el MSE que se obtiene en la validación, se habrá cometido sobre-entrenamiento o exceso de ajuste.

La base de datos a emplear para el entrenamiento de la red se ha obtenido en el sitio web del Departamento de Epilepsia del Hospital de la Universidad de Bonn en Alemania, donde Andrzejak *et al.* (2001) hicieron posible el acceso a la información, esta base de datos permitirá a la RNA realizar la clasificación de las señales correspondientes al trastorno neurológico que se estudia.

Esta base de datos se integra por cinco conjuntos denominados por Andrzejak *et al.* (2001) como A-E. Cada conjunto contiene 100 segmentos de registro monopolar de 23.6 s de duración. Los segmentos fueron seleccionados y extraídos de grabaciones EEG multicanal continuas tras la inspección visual en busca de artefactos, debido a la actividad muscular o movimientos de los ojos.

En la figura 4, Andrzejak *et al.* (2001) exhiben que los conjuntos A y B consisten en segmentos extraídos de los registros de las grabaciones EEG captados de la superficie, los cuales se registraron sobre cinco voluntarios sanos utilizando un esquema estándar para la colocación de electrodos. Los voluntarios se relajaron en estado de vigilia con los ojos abiertos (A) y los ojos cerrados (B), respectivamente.

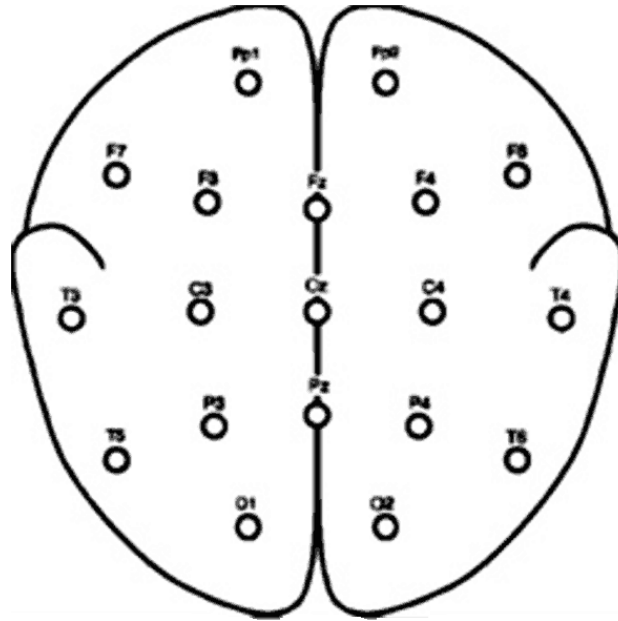


Figura 4. Esquema de localización de electrodos superficiales de acuerdo con el sistema internacional 10-20.  
Fuente: Elaboración propia.

Los segmentos de los conjuntos A y B fueron tomados de los electrodos mostrados, y sus nombres corresponden a la posición de los electrodos derivados de su localización anatómica. En la figura 5 se muestra un ejemplo de las señales de los conjuntos mencionados.

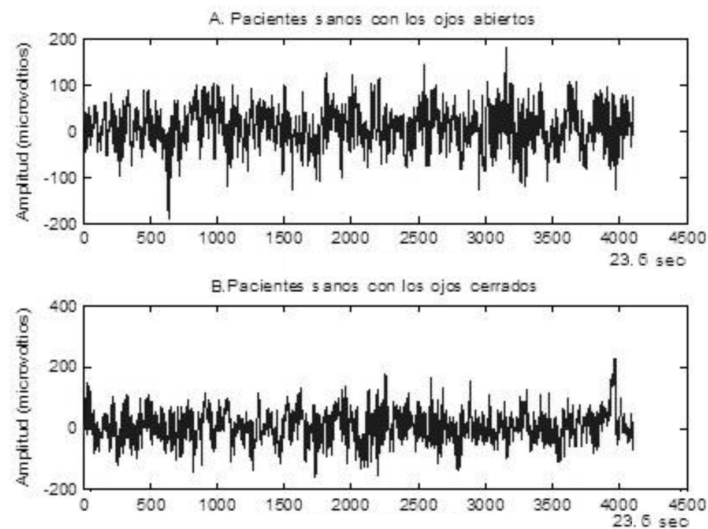


Figura 5. Ejemplo de EEG de los segmentos correspondientes a los conjuntos A y B.  
Fuente: Elaboración propia.

La amplitud de las grabaciones del EEG de la superficie son típicamente del orden de  $\mu V$ .

Los conjuntos C y D se originaron a partir del archivo EEG de diagnóstico pre-quirúrgico, para ello se seleccionaron EEG de cinco pacientes con actividad cerebral en intervalos libres de crisis: de la formación del hipocampo del hemisferio opuesto del cerebro (C) y de la zona epileptogénica (D)



La localización de los electrodos se observa en la figura 6, donde Andrzejak *et al.* (2001) manifiestan que los electrodos de profundidad se implantaron simétricamente dentro de la formación del hipocampo (cima), de donde los segmentos correspondientes a los conjuntos C y D se extrajeron de los contactos respectivos. Además, se implantaron bandas de electrodos en las regiones (lateral y basal) media e inferior de la neocorteza y los segmentos del conjunto E se registraron de los contactos de los electrodos representados.



Figura 6. Esquema de electrodos intracraneales implantados para evaluación pre-quirúrgica de pacientes epilépticos.  
Fuente: Elaboración propia.

En el conjunto E solo se contiene la actividad con crisis donde los segmentos se seleccionaron de las grabaciones de todos los sitios que exhiben actividad ictal. Los conjuntos A y B han sido registrados de forma extracraneal, mientras que los conjuntos C, D y E fueron registrados intracranealmente.

En la figura 7 se observa un ejemplo de los segmentos que corresponden a los pacientes C y D. La amplitud correspondiente a las grabaciones para EEG intracraneal se encuentra en el rango alrededor de  $100 \mu V$ .

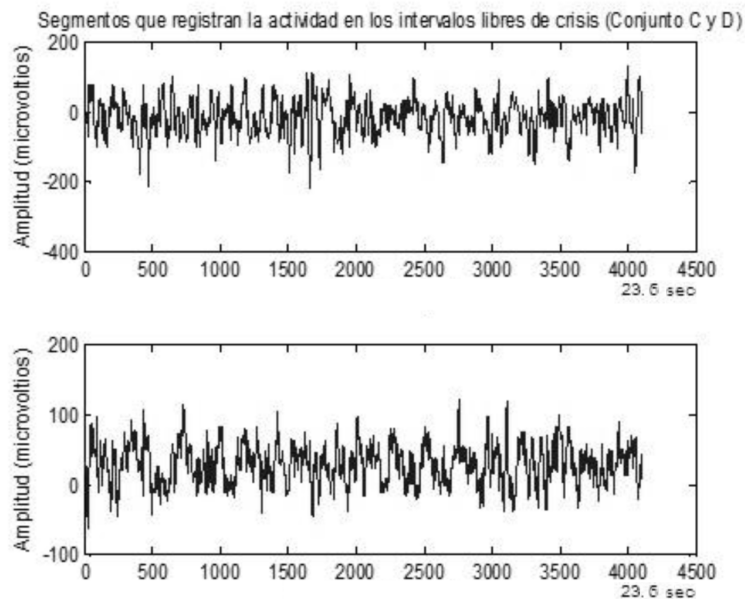


Figura 7. Segmentos que registran la actividad de los pacientes correspondientes a los conjuntos C y D.  
Fuente: Elaboración propia.

La figura 8 representa un segmento del conjunto E que registró la actividad con crisis, los voltajes pueden exceder los  $1000 \mu V$ .

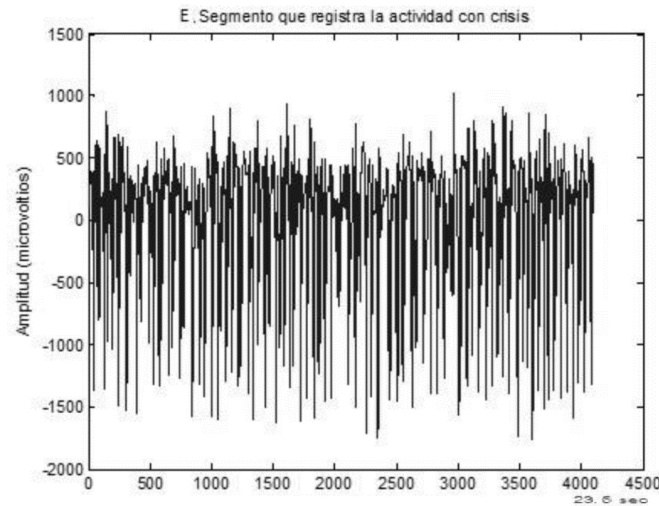


Figura 8. Segmento que registra la actividad del conjunto E.  
Fuente: Elaboración propia.

Las señales del EEG se registraron con el mismo sistema amplificador de 128 canales, utilizando un medio de referencia común (omitiendo electrodos que contienen actividad patológica [C, D y E] o fuertes artefactos del movimiento de los ojos [A y B]). Después de la conversión analógico-digital de 12 bits, los datos se escribieron de manera continua en el disco del sistema de adquisición de datos a una velocidad de muestreo de 173.61 Hz. Los ajustes del filtro pasa-banda fueron de 0.53-40Hz (12 dB/oct.).

Cada uno de estos conjuntos cuenta con 100 archivos con extensión \*.txt con 4096 datos. Para realizar el procesamiento de estas señales se requiere crear una matriz que va a permitir realizar la clasificación de cada clase antes mencionada.

En primera instancia, se forma la matriz de 4096 renglones por 100 columnas para cada uno de los conjuntos, teniendo como resultado cinco matrices diferentes de 4096 renglones por 100 columnas. Una vez que estas matrices fueron creadas, se unieron generando finalmente una matriz de 4096 renglones por 500 columnas, uniéndose la matriz B a la matriz A, la matriz C a la matriz B y así sucesivamente con la matriz D y E, la cual se denomina matriz unión.

Esta matriz se procesa para obtener la conversión de los renglones por las columnas a través de la transpuesta de la matriz para facilitar el aprendizaje de la red al momento de presentar los datos de entrada de donde resulta una matriz de 500 renglones por 4096 columnas.

Para realizar el análisis y la comparación de las señales registradas se requiere normalizar las muestras, lo que notablemente mejorará el entrenamiento además de evitar que las variaciones provoquen resultados de clasificación erróneos, así como saturar las funciones de activación de las neuronas.

Este proceso consiste en acometer los datos en un rango específico entre -1,1, obteniendo los valores máximos y mínimos de cada caso de entrenamiento (renglón) y dividirlo entre este valor máximo, obteniendo una matriz que se aproxima a una distribución normal.

Para lograr una clasificación óptima es necesario realizar una caracterización en el dominio de la frecuencia de las muestras aplicando la transformada de Fourier representada en la ecuación 3, implementada por medio del algoritmo de la transformada rápida de Fourier (FFT, por sus siglas en inglés).

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk} \quad k = 0, 1, \dots, N-1$$

donde  $W_N = e^{-j2\pi/N}$  (3)

Como resultado se obtiene información que proporciona las características más importantes, la cual se procesa a través de tres tipos de umbral, generando la matriz que facilite el proceso de identificar los patrones correspondientes al trastorno que se estudia.

Las muestras se presentan de forma aleatoria para convertir en estocástica la búsqueda en el espacio de los pesos y así minimizar la posibilidad de convergencia a un mínimo local.

Al obtener esta matriz se procede a calcular la varianza para cada columna, este cálculo permite establecer la variabilidad de la señal respecto a la media, teniendo como resultado 4096 varianzas.

Considerando el valor de las varianzas, se define un umbral que involucra aquellas que sean mayores o iguales al valor que se indica, eliminando el resto y, como consecuencia, esto genera la matriz final de entrada que va a ser procesada por la RNA.

Para llevar a cabo la clasificación de las señales correspondientes al trastorno neurológico que se estudia, se ha implementado una RNA de tipo perceptrón multicapa. Los datos correspondientes al entrenamiento se integran por los patrones de las señales que se identificaron como relevantes una vez que se define el umbral. Al aplicar el umbral 1000 sólo se consideran 139 muestras que son las que cuentan con la varianza mayor o igual a este umbral.

En la capa de entrada, los datos corresponden con el número de columnas de la matriz final, 139; para este caso, solo una capa oculta dónde se hace una variación con el número de neuronas desde 0 a 10, mientras que el número de salidas corresponde al número de clases a clasificar, 5.

## Gradiente Conjugado y Templado Simulado

La RNA ha sido entrenada utilizando el algoritmo del gradiente conjugado con retro-propagación, en conjunto con el algoritmo correspondiente al templado simulado para minimizar el MSE.

El algoritmo del gradiente conjugado se basa en que la función error es cuadrática; debido a esto, pretende adquirir nuevas direcciones de búsqueda las cuales no interfieran con las que ya se realizaron en las iteraciones anteriores. Logra gran disminución de tiempo y memoria para obtener la matriz Hessiana en grandes problemas.

El algoritmo de templado simulado es bastante sencillo de implementar y cubre diferentes programas de enfriamiento tales como ciclos exponenciales, lineales y de temperatura, además de verificar el impacto de los números aleatorios que son generados y de qué tanto afectan la velocidad y calidad del algoritmo.

Básicamente, consiste en un método de optimización que se comporta como el proceso de templado usado en metalúrgica; por ejemplo, cuando una sustancia atraviesa el proceso de templado,

primeramente, se calienta hasta alcanzar su punto de fusión para licuarlo, después se enfría lentamente de manera controlada hasta regresarlo a su estado sólido.

Finalmente, las propiedades obtenidas de esta sustancia dependen del programa de enfriamiento aplicado; esto es, si se enfría rápidamente, se desbaratará fácilmente debido a su estructura imperfecta, por lo que deberá enfriarse lentamente para que su estructura resulte fuerte y perfecta.

Para solucionar un problema de optimización implementando templado simulado, la estructura de la sustancia representa una solución codificada del problema y la temperatura es usada para determinar cómo y cuándo las nuevas soluciones son aceptadas o alteradas. El algoritmo consta de tres pasos: alterar la solución, determinar la calidad de la solución y aprobar la solución si su resultado es mucho mejor.

Las ventajas de implementar templado simulado se basan en no requerir un modelo matemático, ya que algunos problemas son muy complicados para ser utilizados o cuentan con muy poca información, por lo que en estos casos el templado simulado resulta conveniente cuando las soluciones pueden plantearse a través de implementaciones tales como: alteraciones o evaluaciones.

En otro caso, el problema consta de muchas soluciones y algunas no son óptimas, aquí la solución se encuentra acotada de soluciones que no resultan óptimas; por ejemplo, en un sistema de ecuaciones no lineales donde el MSE se usa para calcular la importancia del resultado, el cual se calcula utilizando la salida real y la salida deseada del sistema.

Este algoritmo resulta una excelente opción para disminuir el MSE y encontrar una solución adecuada; resulta ser muy útil cuando el mínimo global está definido y requiere de un tiempo de procesado mucho menor.

## Resultados y discusión

Las RNA se han implementado en el *Software Neural Lab v4.0* (Ledesma, Ibarra-Manzano, García-Hernández & Almanza-Ojeda, 2017). Para ello, los datos de entrenamiento se integran de las señales que resultaron relevantes al aplicar diferentes tipos de umbral, la implementación de la red se realizó con solo una capa oculta variando el número de neuronas de 0 a 10, mientras que el proceso que implica el entrenamiento y validación de la red se llevó a cabo en siete pasos.

El número de datos de entrada de la red corresponde al número de muestras seleccionadas para cada umbral. Para cada uno se realizó el entrenamiento seleccionando de los datos de entrada el 70%, 80% y 90% para entrenamiento y el 30%, 20% y 10% se designaron para realizar el proceso de validación.

A medida que se aplica un porcentaje diferente a cada umbral en el conjunto de datos, varían los casos de entrenamiento, permaneciendo constantes los datos de entrada; esto mismo ocurre con los datos asignados para el proceso de validación, tal como se puede observar en las tablas 1, 2 y 3.

**Tabla 1.** Matriz resultante de aplicar diferentes porcentajes en el umbral 950.

Umbral 950	Matriz de entrenamiento	Matriz de validación
70%	350 x 152	150 x 152
80%	400 x 152	100 x 152
90%	450 x 152	50 x 152

Fuente: Elaboración propia.

**Tabla 2.** Matriz resultante al aplicar porcentajes diferentes en el umbral 1000.

Umbral 1000	Matriz de entrenamiento	Matriz de validación
70%	350 x 139	150 x 139
80%	400 x 139	100 x 139
90%	450 x 139	50 x 139

Fuente: Elaboración propia.

**Tabla 3.** Resultados de aplicar los diferentes porcentajes en el umbral 1050.

Umbral 1050	Matriz de entrenamiento	Matriz de validación
70%	350 x 125	150 x 125
80%	400 x 125	100 x 125
90%	450 x 125	50 x 125

Fuente: Elaboración propia.

Generados estos conjuntos de datos, el entrenamiento consiste en variar el número de neuronas en la capa oculta de 0 a 10. Para cada umbral, los casos de entrenamiento se seleccionaron de manera aleatoria, lo que consecuentemente reinició los pesos asignados a cada una de las entradas, mientras que en el proceso correspondiente a la validación, los datos designados son utilizados para evaluar el desempeño de la RNA una vez que ha sido entrenada. Este procedimiento permite evaluar el comportamiento de la RNA con datos diferentes a los de entrenamiento.

Los resultados obtenidos de estos entrenamientos para cada umbral con sus respectivos porcentajes se observan en las tablas 4, 5 y 6:

**Tabla 4.** Resultados de seleccionar el 70% de los casos del conjunto de datos para entrenamiento y 30% para validación (umbral 950).

Número de neuronas	Errores de entrenamiento	MSE Entrenamiento	Errores de validación	MSE Validación
0	32	0.0233321	82	0.213069
1	104	0.0979709	93	0.184304
2	38	0.0504259	82	0.228912
3	31	0.0425208	79	0.182591
4	21	0.0256486	75	0.187936
5	32	0.0265612	86	0.219699
6	9	0.00704164	72	0.189218
7	3	0.00381741	87	0.213184
8	3	0.0030678	78	0.196255
9	0	0.00137975	67	0.167914
10	0	0.000806751	72	0.168639

Fuente: Elaboración propia.

**Tabla 5.** Resultados de seleccionar el 80% de los casos del conjunto de datos para entrenamiento y 20% para validación (umbral 950).

Número de neuronas	Errores de entrenamiento	MSE Entrenamiento	Errores de validación	MSE Validación
0	34	0.0216466	56	0.223775
1	116	0.100677	52	0.166496
2	28	0.0457259	60	0.243089
3	65	0.0527178	59	0.228686
4	76	0.0563626	49	0.1469
5	24	0.0152287	59	0.23955
6	28	0.0177403	52	0.192985
7	18	0.0118333	49	0.194436
8	12	0.00878892	50	0.203591
9	1	0.00224091	36	0.143643
10	5	0.00483537	37	0.139922

Fuente: Elaboración propia.

**Tabla 6.** Resultados de seleccionar el 90% de los casos del conjunto de datos para entrenamiento y 10% para validación (umbral 950).

Número de neuronas	Errores de entrenamiento	MSE Entrenamiento	Errores de validación	MSE Validación
0	42	0.0234388	33	0.262884
1	200	0.123227	32	0.160147
2	29	0.0555787	34	0.229146
3	63	0.0346199	25	0.212016
4	37	0.0214958	19	0.165843

5	36	0.0222065	28	0.217745
6	14	0.00953704	23	0.187772
7	38	0.0212783	31	0.230822
8	7	0.00566318	15	0.128267
9	10	0.00654518	17	0.153144
10	25	0.0139412	30	0.18956

Fuente: Elaboración propia.

En las tablas 7, 8 y 9 se presentan los resultados correspondientes al umbral 1000.

**Tabla 7.** Resultados de seleccionar el 70% de los casos del conjunto de datos para entrenamiento y 30% para validación (umbral 1000).

Número de neuronas	Errores de entrenamiento	MSE Entrenamiento	Errores de validación	MSE Validación
0	36	0.0247168	79	0.236285
1	217	0.099985	122	0.18154
2	85	0.0519435	107	0.227427
3	36	0.0272225	98	0.243623
4	17	0.0157471	85	0.210837
5	32	0.0225597	91	0.214435
6	15	0.0123679	96	0.228199
7	4	0.00410279	70	0.168608
8	2	0.00318871	71	0.172365
9	1	0.00155202	75	0.168458
10	12	0.00995373	73	0.200868

Fuente: Elaboración propia.

**Tabla 8.** Resultados de seleccionar el 80% de los casos del conjunto de datos para entrenamiento y 20% para validación (umbral 1000).

Número de neuronas	Errores de entrenamiento	MSE Entrenamiento	Errores de validación	MSE Validación
0	53	0.0317008	45	0.221751
1	255	0.102659	74	0.156746
2	98	0.0508471	68	0.235594
3	45	0.0280694	67	0.238225
4	26	0.0163701	59	0.233496
5	43	0.0268917	54	0.205376
6	26	0.0173007	53	0.191975
7	18	0.0124487	57	0.21526

8	9	0.00759098	48	0.190605
9	25	0.0153933	65	0.234481
10	8	0.00615434	45	0.144777

Fuente: Elaboración propia.

**Tabla 9.** Resultados de seleccionar el 90% de los casos del conjunto de datos para entrenamiento y 10% para validación (umbral 1000).

Número de neuronas	Errores de entrenamiento	MSE Entrenamiento	Errores de validación	MSE Validación
0	64	0.0367754	31	0.256167
1	153	0.104444	29	0.168763
2	54	0.0629683	22	0.147612
3	50	0.0304009	26	0.215324
4	42	0.025035	21	0.181743
5	8	0.006157	25	0.217841
6	27	0.015792	20	0.186973
7	23	0.0143747	27	0.0245961
8	20	0.0127013	25	0.177121
9	34	0.0189145	23	0.185164
10	8	0.00538542	24	0.172633

Fuente: Elaboración propia.

A continuación, en las tablas 10, 11 y 12 se muestran los resultados para el proceso correspondiente al umbral 1050.

**Tabla 10.** Resultados de seleccionar el 70% de los casos del conjunto de datos para entrenamiento y 30% para validación (umbral 1050).

Número de neuronas	Errores de entrenamiento	MSE Entrenamiento	Errores de validación	MSE Validación
0	24	0.0168504	82	0.252242
1	108	0.102888	95	0.173135
2	32	0.0534971	83	0.208116
3	34	0.026171	89	0.243503
4	24	0.0185603	73	0.213653
5	18	0.0146868	77	0.192623
6	35	0.0244283	85	0.196509
7	10	0.00871126	77	0.205486
8	12	0.00936294	75	0.192203
9	14	0.0109507	66	0.190234
10	1	0.0014772	78	0.198993

Fuente: Elaboración propia.



**Tabla 11.** Resultados de seleccionar el 80% de los casos del conjunto de datos para entrenamiento y 20% para validación (umbral 1050).

Número de neuronas	Errores de entrenamiento	MSE Entrenamiento	Errores de validación	MSE Validación
0	42	0.0259618	55	0.242759
1	143	0.103264	62	0.177582
2	58	0.0657969	60	0.200967
3	62	0.063544	49	0.208172
4	46	0.0284876	47	0.199494
5	29	0.0241919	45	0.17594
6	34	0.0210376	43	0.178547
7	16	0.0111293	49	0.190892
8	11	0.00842928	33	0.148178
9	5	0.00429234	49	0.196357
10	1	0.00163534	50	0.163247

Fuente: Elaboración propia.

**Tabla 12.** Resultados de seleccionar el 90% de los casos del conjunto de datos para entrenamiento y 10% para validación (umbral 1050).

Número de neuronas	Errores de entrenamiento	MSE Entrenamiento	Errores de validación	MSE Validación
0	72	0.038843	21	0.169196
1	161	0.10549	32	0.191214
2	112	0.0800831	25	0.146577
3	56	0.0323492	28	0.20444
4	42	0.0232938	26	0.211798
5	28	0.0174113	26	0.218904
6	49	0.0274877	21	0.180763
7	25	0.0147733	20	0.176138
8	50	0.0276246	29	0.214142
9	22	0.0128649	25	0.220341
10	32	0.0186108	22	0.152308

Fuente: Elaboración propia.

Después de realizar el entrenamiento para cada umbral, se observa que el resultado del MSE es muy variable debido a la selección aleatoria de los patrones que conforman la matriz; además, el MSE correspondiente al entrenamiento es menor que el obtenido en el proceso de validación.

Al terminar el proceso de entrenamiento y validación, el número de errores correspondientes a cada proceso se verifica en la matriz de confusión, genera una clase llamada rechazo (*Reject*) para aquellos casos que la RNA no logra clasificar.

Por cada proceso de aprendizaje en sus pasos de entrenamiento y validación se generan estas matrices; por lo tanto, para cada umbral y sus respectivos porcentajes, estos resultados comprueban la clasificación de los casos de entrenamiento.

Dado que el número de neuronas en la capa oculta debe estar en función de los casos de entrenamiento, esto es, el número de casos de entrenamiento debe ser al menos mayor o igual a 2.2 veces el número de pesos en la capa oculta, para ello la cantidad de casos de entrenamiento necesarios para realizar el proceso de aprendizaje se calcula de la siguiente manera:

$$\text{Casos de entrenamiento} \geq (\text{Número de entradas} + 1)(\text{Neuronas en la capa oculta})(2.2)$$

Para llevar a cabo este proceso se realiza la multiplicación: (el número de entradas+1) multiplicado por (el número de neuronas en la capa oculta) por (2.2).

Estos resultados quedan respaldados al calcular el número máximo de neuronas en la capa oculta con que se obtiene la correcta clasificación de las clases a través de la siguiente ecuación:

$$\text{Neuronas en la capa oculta} \leq \frac{\text{Casos de entrenamiento}}{(\text{Número de entradas} + 1)(2.2)}$$

Para cada umbral se calcula el número máximo de neuronas en la capa oculta en función de los casos de entrenamiento que se presentan por cada porcentaje, de donde se obtienen los resultados mostrados en las tablas 13, 14 y 15:

Tabla 13. Número de neuronas para cada porcentaje aplicado en el umbral 950.

$$70\% \text{ Neuronas en la capa oculta} \leq \frac{350}{(152+1)(2.2)} = 1.039$$

$$80\% \text{ Neuronas en la capa oculta} \leq \frac{400}{(152+1)(2.2)} = 1.188$$

$$90\% \text{ Neuronas en la capa oculta} \leq \frac{450}{(152+1)(2.2)} = 1.336$$

Fuente: Elaboración propia.

**Tabla 14.** Número de neuronas para cada porcentaje aplicado en el umbral 1000.

$$70\% \text{ Neuronas en la capa oculta} \leq \frac{350}{(139+1)(2.2)} = 1.136$$

$$80\% \text{ Neuronas en la capa oculta} \leq \frac{400}{(139+1)(2.2)} = 1.29$$

$$90\% \text{ Neuronas en la capa oculta} \leq \frac{450}{(139+1)(2.2)} = 1.461$$

Fuente: Elaboración propia.

**Tabla 15.** Número de neuronas para cada porcentaje aplicado en el umbral 1050.

$$70\% \text{ Neuronas en la capa oculta} \leq \frac{350}{(125+1)(2.2)} = 1.262$$

$$80\% \text{ Neuronas en la capa oculta} \leq \frac{400}{(125+1)(2.2)} = 1.443$$

$$90\% \text{ Neuronas en la capa oculta} \leq \frac{450}{(125+1)(2.2)} = 1.623$$

Fuente: Elaboración propia.

Al realizar el cálculo para cada umbral, se demuestra que el número máximo de neuronas en la capa oculta es menor o igual a 1, lo que indica que solo con las neuronas 0 y 1 se obtendrán los resultados esperados; a partir de la neurona 2 se requiere mayor cantidad de casos de entrenamiento para llevar a cabo el proceso de aprendizaje de la red. Dados los porcentajes aplicados en cada uno de los umbrales, este requerimiento excede los casos de entrenamiento existentes.

Los errores obtenidos en el entrenamiento y en la validación en las neuronas 0 y 1 para cada uno de los umbrales aplicados son elevados y el MSE calculado para entrenamiento es menor que el MSE calculado para la validación, lo que provoca sobre-entrenamiento, haciendo que estas implementaciones de RNA y estos procesos de aprendizaje sean limitados para considerarlos como referencia.

No obstante, se realiza la comparación de los resultados obtenidos de este método con otro proceso de aprendizaje denominado LOOCV, que consiste en eliminar de manera temporal un caso de entrenamiento, por lo que el entrenamiento se realiza con el resto de los casos; después, en cada iteración, el caso que se elimina es alternado y, así, finalmente todos los casos forman parte del entrenamiento.

Para realizar este entrenamiento se han definido tres tipos de umbral de la misma manera que en el proceso anterior, teniendo como resultado tres conjuntos de datos con sus respectivos casos de entrenamiento y validación que se presentan en la tabla siguiente:

**Tabla 16.** Conjunto de datos de entrenamiento resultante al aplicar el umbral en LOOCV.

	Matriz de entrenamiento	Matriz de validación
Umbral 950	499 x 152	1 x 152
Umbral 1000	499 x 139	1 x 139
Umbral 1050	499 x 125	1 x 125

Fuente: Elaboración propia.

Generado este conjunto de datos, se realizó el entrenamiento para cada umbral variando el número de neuronas en la capa oculta de 0 a 10, el conjunto de datos correspondiente al proceso de validación presenta el caso de entrenamiento que se elimina de manera temporal, procediendo de esta manera a realizar el entrenamiento con el resto de los casos que conforman el conjunto de datos.

En la tabla 17 se observan los resultados correspondientes al número de errores obtenidos mediante el método de LOOCV; en el umbral 1050 se obtiene la menor cantidad de errores correspondiente a los entrenamientos realizados.

**Tabla 17.** Número de errores obtenidos por cada umbral al realizar el aprendizaje por LOOCV.

Neuronas en la capa escondida	Número de Errores Umbral 950	Número de Errores Umbral 1000	Número de Errores Umbral 1050
0	38	31	25
1	0	0	0
2	5	12	4
3	15	11	20
4	19	20	20
5	19	16	23
6	17	19	17
7	24	20	16
8	23	13	13
9	17	14	8
10	9	17	13

Fuente: Elaboración propia.

Estos errores de entrenamiento por LOOCV se comparan con los errores correspondientes a los umbrales aplicados, los cuales se muestran a continuación (tablas 18, 19 y 20):

**Tabla 18.** Tabla comparativa de los porcentajes aplicados en el umbral 950 y LOOCV.

Número de neuronas	Errores de entrenamiento Umbral 950 – 70%	Errores de entrenamiento Umbral 950 – 80%	Errores de entrenamiento Umbral 1050 – 90 %	Número de Errores LOOCV
0	32	34	42	38
1	104	116	200	0
2	38	28	29	5
3	31	65	63	15
4	21	76	37	19
5	32	24	36	19
6	9	28	14	17
7	3	18	38	24
8	3	12	7	23
9	0	1	10	17
10	0	5	25	9
Total de Errores por Entrenamiento	273	407	501	186

Fuente: Elaboración propia.

**Tabla 19.** Tabla comparativa de los porcentajes aplicados en el umbral 1000 y LOOCV.

Número de neuronas	Errores de entrenamiento Umbral 1000 – 70%	Errores de entrenamiento Umbral 1000 – 80%	Errores de entrenamiento Umbral 1000 – 90%	Número de Errores LOOCV
0	36	53	64	31
1	217	255	153	0
2	85	98	54	12
3	36	45	50	11
4	17	26	42	20
5	32	43	8	16
6	15	26	27	19
7	4	18	23	20
8	2	9	20	13
9	1	25	34	14
10	12	8	8	17
Total de Errores por Entrenamiento	457	606	483	173

Fuente: Elaboración propia.

**Tabla 20.** Tabla comparativa de los porcentajes aplicados en el umbral 1050 y LOOCV.

Número de neuronas	Errores de entrenamiento Umbral 1050 – 70%	Errores de entrenamiento Umbral 1050 – 80%	Errores de entrenamiento Umbral 1050 – 90%	Número de Errores LOOCV
0	24	42	72	25
1	108	143	161	0
2	32	58	112	4
3	34	62	56	20
4	24	46	42	20
5	18	29	28	23
6	35	34	49	17
7	10	16	25	16
8	12	11	50	13
9	14	5	22	8
10	1	1	32	13
Total de Errores por Entrenamiento	312	447	649	159

Fuente: Elaboración propia.

Al observar cada una de las tablas comparativas por umbral, se deduce que LOOCV es el mejor método para realizar el entrenamiento de la RNA con este conjunto de datos y llevar a cabo el proceso de identificación de las clases, ya que el número de errores obtenidos por este método es mínimo en comparación con cada uno de los porcentajes aplicados para el umbral que se indica; por lo tanto, es posible implementar este método cuando los casos de entrenamiento son escasos y es posible obtener la completa clasificación de las clases con un mínimo de errores, como resulta en este caso al aplicar cada umbral.

## Conclusiones

Para diagnosticar el trastorno neurológico denominado epilepsia, se requiere observar la actividad cerebral (EEG) del paciente, además de analizar el informe clínico, lo que conlleva mucho tiempo, tratándose sobre todo de grabaciones EEG continuas de larga duración. Es por ello que una RNA capaz de clasificar las señales correspondientes se proyecta como una gran herramienta de apoyo al diagnóstico médico.

Este trabajo presenta la implementación de una RNA del tipo perceptrón multicapa para la clasificación de señales EEG de pacientes sanos, pacientes que controlaron las crisis y pacientes enfermos de epilepsia, mediante la selección de diferentes umbrales y de casos de entrenamiento; así mismo, se realiza el entrenamiento de la red aplicando el método LOOCV para finalmente comparar los resultados y determinar el método que proporcione un resultado óptimo.

Los parámetros de las señales del EEG consideran diferentes aspectos, los cuales se implementan mediante la RNA para realizar el proceso de aprendizaje y llevar a cabo la identificación de las señales a cada una de las respectivas clases. Dado que aprenden de la experiencia, mediante el entrenamiento involucran ajustes de los pesos sinápticos; una vez que obtiene un valor óptimo, el ajuste de los pesos se detiene, entonces se dice que la RNA ha aprendido.

Para el primer método se determina por umbral cierto porcentaje que delimita los casos de entrenamiento y validación, respectivamente. Los resultados obtenidos demuestran que solo con una cantidad menor o igual a una neurona en la capa oculta, la RNA es capaz de realizar la correcta clasificación de las clases, para ello se examina el MSE obtenido en el entrenamiento, el cual es mucho más pequeño que el adquirido en el proceso de validación, lo que provoca sobre-entrenamiento, concluyendo que esta implementación es íntegramente limitada.

Pese a que con el método anterior se adquieren resultados poco favorables, al efectuar el entrenamiento de la RNA con el método de la LOOCV se logran resultados que señalan la menor cantidad de errores y, una vez que se procede con la comparación de los resultados en ambos métodos, se constata que el entrenamiento de la RNA mediante la validación cruzada proporciona excelentes resultados, variando el número de neuronas en la capa oculta, de 0 a 10, con la misma cantidad de casos de entrenamiento y obteniendo la menor cantidad de errores, lo que define a este método de procesamiento magnánimo de calidad.

Partiendo de las investigaciones que realizaron Arab, Suratgar, Martínez-Hernández & Rezaei Ashtiani (2010) y Gajic, Djurovic, Di Gennaro & Gustafsson (2014), donde plantean las RNA y la transformada *wavelet*, se demuestra que esta técnica de IA es capaz de establecer resultados satisfactorios, procesando la información mediante el aprendizaje que se determina respectivamente. Para ello, Arab *et al.* (2010) implementan la transformada *wavelet* discreta para eliminar un artefacto causado por el parpadeo y movimiento ocular y de electrodos; así mismo, elimina ruido de las señales debido a la corriente alterna. La transformada rápida de Fourier para la clasificación considera características de ondas agudas, puntas y punta-onda lenta, establece redes de cuantización vectorial (LVQ, por sus siglas en inglés) localizado en tiempo y frecuencia.

Gajic *et al.* (2014) reportan resultados que se procesan mediante la implementación de RNA en conjunto con la transformada *wavelet* y el análisis multi-resolución, estableciendo características estadísticas, donde se alcanza una alta precisión de clasificación muy satisfactoria del 99%. Estos algoritmos pueden ser utilizados para clasificar las señales de EEG y detectar las convulsiones en un entorno clínico.

Equiparando el avance de las investigaciones mencionadas con los resultados obtenidos en este trabajo, se manifiesta que el entrenamiento de la RNA aplicando la transformada de Fourier, teniendo en cuenta dos métodos de aprendizaje: redes neuronales multicapa con validación clásica (*backpropagation*) y LOOCV, donde se evalúa el MSE y la cantidad de errores obtenidos, se precisa que el método que proporciona resultados sobresalientes, productivos y convenientemente confiables es el proceso de entrenamiento LOOCV.

## Agradecimientos

Al Departamento de Epilepsia del Hospital de la Universidad de Bonn en Alemania por la publicación de bases de datos en la Internet.

## Referencias

- Andrzejak, R. G., Lehnertz, K., Mormann, F., Rieke, C., David, P., & Elger, C. E. (2001). Indications of nonlinear deterministic and finite dimensional structures in time series of brain electrical activity: Dependence on recording and brain state. *Physical Review*, 64(6). doi: <https://doi.org/10.1103/PhysRevE.64.061907>
- Arab, M. R., Suratgar, A. A., Martínez-Hernández, V. M., & Rezaei Ashtiani, A. (2010). Electroencefalogram Signals Processing for the Diagnosis of Petit mal and Grand mal Epilepsies Using an Artificial Neural Network. *Journal of Applied Research and Technology*, 8(1), 120-129.
- Castaño, N. E., Zapata, J. F., & Villegas, J. (2009). Detección de puntas epilépticas en señales electroencefalográficas para pacientes con epilepsia del lóbulo temporal utilizando wavelets. *Ingeniería y Ciencia*, 5(9), 145-165.
- Çetin, G. D., Çetin, Ö., & Bozkurt, M. R. (2015). The detection of normal and epileptic EEG signals using ANN methods with matlab-based GUI. *International Journal of Computer Applications*, 114(12), 45-50.
- Costa, R. P., Oliveira, P., Rodrigues, G., Leitão, B., & Dourado, A. (2008). Epileptic seizure classification using neural networks with 14 features. In Knowledge-Based Intelligent Information and Engineering Systems. *Knowledge-based and Intelligent Information and Engineering Systems*, 978-3-540-85565-1, 281-288. doi: [https://doi.org/10.1007/978-3-540-85565-1\\_35](https://doi.org/10.1007/978-3-540-85565-1_35)
- O'Connor, J. L. (1993). *Tecnologías computacionales para sistemas de ecuaciones, optimización lineal y entera*. España: Editorial Reverte.
- Gajic, D., Djurovic, Z., Di Gennaro, S., & Gustafsson, F. (2014). Classification of EEG signals for detection of epileptic seizures based on wavelets and statistical pattern recognition *Biomedical Engineering: Applications, Basis and Communications*, 26(2), 1450021. doi: <http://dx.doi.org/10.4015/S1016237214500215>
- Guo, L., Rivero, D., Dorado, J., Rabuñal, J. R. & Pazos, A. (2010). An automatic epileptic seizure detection in EEGs based on line length feature and artificial neural networks. *Journal of Neuroscience Methods*, 191(1), 101-109. doi: <https://doi.org/10.1016/j.jneumeth.2010.05.020>
- Ledesma, S., Ibarra-Manzano, M., García-Hernández, M., & Almanza-Ojeda, D., (2017). Neural lab a simulator for artificial neural networks. *Computing Conference*, 716-721. doi: <https://doi.org/10.1109/SAI.2017.8252175>
- Özkan, C., Doğan, S., Kantar, T., Akşahin, M., & Erdamar, A. (2016). Detection of epilepsy disease from EEG signals with artificial neural networks. *24th Signal Processing and Communication Application Conference (SIU)*, 693-696. doi: <https://doi.org/10.1109/SIU.2016.7495834>
- Ripley, B. D. (1996). *Pattern recognition and neural networks*. United Kingdom: Cambridge University Press.
- Srinivasan, V., Eswaran, C., & Sriraam, N. (2007). Approximate entropy-based epileptic EEG detection using artificial neural networks. *IEEE Transactions on Information Technology in Biomedicine*, 11(3). doi: <https://doi.org/10.1109/TITB.2006.884369>