



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO



FACULTAD DE INGENIERÍA

**ESTUDIO COMPARATIVO DE HERRAMIENTAS PARA
REDES NEURONALES ARTIFICIALES (RNA): WEKA,
MATLAB Y NEUROSOLUTIONS**

TESINA PARA OBTENER EL TÍTULO DE:

Licenciatura en Ingeniería en Computación

PRESENTA:

Sergio Calixto Aldama

ASESORADO POR:

Dra. Rosa María Valdovinos Rosas

JUNIO - 2018



DEPTO. DE EVALUACIÓN PROFESIONAL

No. Oficio: 31/2018

Ciudad Universitaria, Toluca, Méx. a 21 de junio del 2018

C. SERGIO CALIXTO ALDAMA
PASANTE DE INGENIERÍA EN COMPUTACIÓN
PRESENTE.

En respuesta a su solicitud, a continuación transcribo el tema aprobado por esta Dirección, que propuso la **DRA. ROSA MARÍA VALDOVINOS ROSAS**, con el fin de que lo desarrolle en la modalidad de **TESINA**, le informo que se autoriza la **impresión de su trabajo** para presentar su Evaluación Profesional.

"ESTUDIO COMPARATIVO DE HERRAMIENTAS PARA REDES NEURONALES ARTIFICIALES: WEKA, MATLAB Y NEUROSOLUTIONS."

ÍNDICE GENERAL

INTRODUCCIÓN

RESUMEN

CAPÍTULO 1: MARCO TEÓRICO Y ESTUDIO DEL ARTE

CAPÍTULO 2: CARACTERÍSTICAS DE LAS HERRAMIENTAS

CAPÍTULO 3: PRUEBAS Y USABILIDAD

CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO

Ruego a usted tomar nota de que en cumplimiento a lo especificado por la Ley de Profesiones, deberá prestar Servicio Social durante un tiempo mínimo de seis meses, como requisito indispensable para sustentar su Evaluación Profesional.

Así mismo, para la elaboración de la **TESINA** y demás trámites, deberá sujetarse a la reglamentación respectiva de esta Universidad.



ATENCIÓN
PATRIA, CIENCIA Y TRABAJO

"2018, Año del 190 Aniversario de la Universidad Autónoma del Estado de México".

FACULTAD DE INGENIERÍA
U. A. E. M.

DRA. MARÍA DOLORES DURÁN GARCÍA
DIRECTORA DE LA FACULTAD DE INGENIERÍA

**/Saha.

Cerro de Coatepec S/N
Ciudad Universitaria; Toluca México
Tel. (722) 2-14-08-55 / 2-15-13-51

FI

RESUMEN

La Tesina consiste en una investigación, en la que se busca examinar las características de tres herramientas de Redes Neuronales Artificiales: Weka, Matlab y NeuroSolutions, a través de una revisión bibliográfica de artículos, libros, sitios web, investigaciones académicas y algunas relacionadas al objeto de estudio. Para entender el funcionamiento de cada una de las herramientas analizadas, se desarrolló un ejemplo con cada una y se resaltaron las bondades así como las dificultades que tienen.

Según lo presentado, se puede concluir que Weka es de fácil uso, que no requiere conocimientos previos de programación gracias a que su interfaz gráfica es sencilla, Matlab es una herramienta más completa, pero es necesario tener conocimientos de programación para manipular la aplicación, así como hacer el pago de la licencia correspondiente. Por último, se tiene NeuroSolutions, que es la herramienta más completa y no tan compleja de utilizar, no obstante, al igual que Matlab es un software de paga, por lo que se necesita comprar una licencia.

Respecto al precio, la licencia de NeuroSolutions cuesta *USD* \$295.00 para la versión de Estudiante (*Student*) y Matlab cuesta *USD* \$99.00 en la versión de Estudiante (*Student*). Es sencillo identificar porque la licencia de NeuroSolutions es más costosa pues contiene más herramientas y es más fácil de usar en comparación de Matlab.

En la última parte del trabajo, se realizarán algunas pruebas con un ejemplo en común para cada una de las herramientas y con ello poder mencionar algunas particularidades así como unas recomendaciones para el estudiante.

ÍNDICE GENERAL

	Pág.
Introducción.....	V
1. Marco teórico y estado del arte	9
1.1 Generalidades de las RNA.....	9
1.2 Topología de las RNA	11
1.3 Conectividad de la RNA	12
1.4 Paradigmas de aprendizaje en RNA	13
1.5 Proceso de reconocimiento.....	14
1.6 Aplicaciones de Redes Neuronales	14
2. Características de las herramientas	17
2.1 <i>Weka</i>	17
2.1.1 <i>Explorer</i>	18
2.1.2 <i>Experimenter</i>	19
2.1.3 <i>Simple cli</i>	19
2.1.4 <i>Knowledge Flow</i>	20
2.1.5 RNA con Weka.....	20
2.1.6 Ventajas	22
2.1.7 Desventajas	22
2.2 Matlab	22
2.2.1 Métodos de aprendizaje	23
2.2.2 Entorno de trabajo de Matlab	24
2.2.3 Ventajas	26
2.2.4 Desventajas	27
2.3 NeuroSolutions	27
2.3.1 Métodos de Aprendizaje	28
2.3.2 Entorno de trabajo de NeuroSolutions	28
2.3.3 Funcionamiento de Redes Neuronales.....	31
2.3.4 Ventajas	32
2.3.5 Desventajas	33
3. Pruebas y usabilidad	34
3.1 Realización de pruebas con las herramientas	34
3.1.1 Ejercicio con Weka	34

3.1.2 Ejercicio con Matlab	36
3.1.3 Ejercicio con NeuroSolutions	40
3.2 Prueba de usabilidad	44
Conclusiones, recomendaciones y trabajo futuro	47
Anexo. Instrumento de evaluación de usabilidad	55

ÍNDICE DE FIGURAS

Figura 1: Esquema de una neurona biológica	9
Figura 2: Red neuronal monocapa	11
Figura 3: Red neuronal multicapa	12
Figura 4: RNA con propagación <i>feedforward/feedback</i>	13
Figura 5: Resultados experimentales	16
Figura 6: Pantalla principal del <i>Explorer</i>	18
Figura 7: Pantalla principal del <i>Experimenter</i>	19
Figura 8: Pantalla principal de <i>simple cli</i>	20
Figura 9: Pantalla principal de <i>Knowledge Flow</i>	20
Figura 10: Pantalla de <i>Neuronal Toolbox</i>	24
Figura 11: Pantalla principal de NeuroSolutions	28
Figura 12: Menú de paletas	29
Figura 13: Paleta de la familia Áxon	29
Figura 14: Paleta de la familia Áxon y unos componentes	30
Figura 15: Paleta de la familia Sinapsis y unos componentes	30
Figura 16: Paleta de la familia Criterios de Error	31
Figura 17: RNA con plano de retro propagación	31
Figura 18: RNA con los tres planos	32
Figura 19: Pantalla de explorer	34
Figura 20: Pantalla de todos los atributos	35
Figura 21: Resultados del algoritmo	35
Figura 22: Pantalla de inicio	36
Figura 23: Pantalla de datos	37
Figura 24: Selección de archivo	37
Figura 25: Entrenamiento	38
Figura 26: Red creada	38

Figura 27: Matrices de confusión	39
Figura 28: Gráfica de resultados	39
Figura 29: Elección del problema	40
Figura 30: Selección de archivo	41
Figura 31: Selección de atributos y clases	41
Figura 32: Elección del nivel de complejidad.....	42
Figura 33: Red creada.....	42
Figura 34: Resultados de entrenamiento	43
Figura 35: Resultados finales	43

ÍNDICE DE TABLAS

Tabla 1: Despliegue de resultados	44
Tabla 2: Resumen de las 3 Aplicaciones	47

ÍNDICE DE GRÁFICAS

Gráfica 1: Usabilidad de Weka	45
Gráfica 2: Usabilidad de Matlab.....	45
Gráfica 3: Usabilidad de NeuroSolutions.....	46

Introducción

Una Red Neuronal Artificial (RNA) es un modelo matemático que consta de varias neuronas que trabajan en paralelo, cuyas conexiones tienen pesos que representan el aprendizaje, inspiradas en los sistemas nerviosos biológicos [1]. De este modo, las RNA son una forma de emular ciertas características propias de los humanos, como la capacidad de memorizar y de asociar hechos. Si se examinan con atención aquellos problemas que no pueden expresarse a través de un algoritmo, puesto que resultan ser muy complejos, se observará que todos ellos tienen una característica en común: la experiencia.

Las primeras propuestas de RNA surgieron en 1943 y hasta la fecha, son utilizadas para la solución de distintos problemas que se presentan en la vida real. Fueron los primeros métodos computacionales con la capacidad inherente de tolerancia a fallos, comparados con los sistemas computacionales tradicionales, los cuales pierden su funcionalidad cuando sufren un pequeño error de memoria. De este modo, en las RNA, si se produce un fallo en alguna de las neuronas, no sufre una caída repentina aunque el comportamiento del sistema se ve influenciado [1].

Al día de hoy, existen distintas herramientas para trabajar con RNA como lo son: Weka, Matlab, RapidMiner, NeuroSolutions, SPSS, PSPP, Python, entre otros. De estas herramientas; la mayoría son de paga, lo que dificulta su acceso. En esta Tesina se realiza el análisis de 3 herramientas de software; una de ellas sin licenciamiento, de uso libre (Weka) y las otras de licencia, de las más utilizadas (Matlab y NeuroSolutions), con la intención de conocer las características, ventajas y desventajas de cada una de ellas en cuanto a su uso y con ello poder identificar la factibilidad de uso de cada una de éstas por parte de los alumnos de la Facultad de Ingeniería.

Planteamiento del Problema

Las RNA son modelos complejos de estudio y de programación. Requieren conocimientos puntuales, bases matemáticas y habilidades importantes en programación de sistemas, por parte de aquellos interesados en su estudio. De este modo, si no se cuenta con la asesoría de un investigador experto en el tema que guíe y resuelva dudas, es prácticamente imposible realizar una investigación seria. Aunado a esto, el material bibliográfico existente, o bien es demasiado técnico que dificulta su comprensión o carece de profundidad, dificultando la comprensión del verdadero sentido y potencial del uso de las RNA [2].

Algo semejante sucede con las herramientas existentes para trabajar con RNA, las cuales demandan conocimientos sobre el tema para la interpretación de resultados o son tan “cuadradas” que complican su adaptación y en muchos casos la información disponible es muy limitada de tal suerte que se abandona su uso. Al respecto, en estos momentos en la Facultad de Ingeniería no se cuenta con suficientes manuales de ayuda para la programación, descripción funcional o interpretación de resultados en RNA, pese a ser una materia de la línea de acentuación de Interacción Hombre Máquina. Esto último obliga a recurrir a libros o artículos que en su mayoría están en inglés, son demasiado técnicos o no detallan procesos, situación que en muchos casos orilla al alumno a abandonar el estudio en esta área del conocimiento.

En esta tesina se realiza un estudio acerca de RNA donde se explica lo relacionado a ellas, la importancia que tienen en la actualidad para solución de problemas reales y brindar un acercamiento a la facilidad o dificultad de cada una de las herramientas objeto de estudio, lo que puede brindar elementos de uso de una u otra por los interesados en trabajar en el área de RNA.

Importancia de la Temática

Debido a la importancia que tienen las RNA en la solución de problemas reales con computación no convencional, en la actualidad (2018), se han incluido como una materia en la línea de acentuación Interacción Hombre Máquina de la Facultad de Ingeniería de la UAEMex.

Pese a haber surgido en 1943 [3], hasta el día de hoy las RNA son ampliamente utilizadas en la solución de problemas reales. No obstante, para utilizar los diferentes modelos de RNA, se requieren habilidades de programación y conocimiento de los modelos matemáticos, en caso de que se desee modificar la estructura y/o funcionamiento para adaptarlo a un problema específico.

Actualmente existen diversas herramientas de software que incluyen los principales modelos de RNA programados, por ejemplo: Weka [4], RapidMiner [5], Matlab [6], NeuroSolutions [7], entre otros. Con su utilización de alguna de estas herramientas para fines de investigación se podrían disminuir el impedimento de la programación, orientándose más a la interpretación de resultados.

Pese a lo anterior, para poder utilizar cualquiera de estas herramientas, los jóvenes investigadores que desean incursionar en esta área necesitan invertir tiempo de acuerdo al interés que tengan sobre el tema para comprender el funcionamiento de los modelos de RNA, su programación y experimentación, tiempo que en muchas ocasiones no se dispone. Esto último es lo que motiva a realizar la investigación documental aquí presentada, de la que se busca identificar los aspectos que hacen atractivas o dificultan el uso de las herramientas.

Objetivo

Realizar un estudio para describir el funcionamiento de tres herramientas de RNA: Weka, Matlab y NeuroSolutions, con el propósito de brindar las ventajas y desventajas de cada una de ellas, y de este modo, brindar un panorama general de uso a estudiantes interesados a incursionar en el área.

Para dar cumplimiento al objetivo planteado, fueron utilizadas 3 técnicas de investigación:

- **Documental.** Para el caso de esta Tesina, la investigación documental fue utilizada al realizar la lectura y revisión bibliográfica de distintos documentos y fuentes de información formal, como repositorios de Redalyc y otros más sobre RNA, para así obtener la información más importante relacionada al tema, así como también saber la importancia de RNA en la actualidad de acuerdo a información consultada [8].
- **Exploratoria.** Para el caso de esta Tesina, se realizó la exploración de información en algunos artículos seleccionados e identificar aquella información que ayudaría a comprender la importancia de las RNA, el uso que se le da en distintas ramas de la ciencia y la solución de casos reales [9].
- **Analítica.** Para el caso de esta Tesina, se aplicó una investigación analítica al momento de tener distintos usos de algunas herramientas enfocadas a RNA, saber la diferencia que hay entre estas herramientas y con ello poder dar una conclusión final de cuál es la mejor herramienta para trabajar con RNA exponiendo algunos puntos de vista que deducen porqué dicha herramienta es la mejor según la información consultada. Para esto, se realizaron ejercicios y se manipularon las aplicaciones a fin de identificar las ventajas y desventajas que cada una de ellas tiene [10].

Estructura de la tesina

La tesina se encuentra estructurada de la siguiente forma: en el Capítulo 1 se incluye el marco teórico que sustenta la revisión bibliográfica, en el Capítulo 2 se describen las herramientas caso de estudio, en el Capítulo 3 se incluyen los ejercicios que se realizaron para validar las características de las herramientas y, finalmente se incluye una sección de Conclusiones, recomendaciones y trabajo futuro.

Capítulo 1. Marco teórico y estado del arte

En este capítulo se incluyen los conceptos básicos relacionados con las redes neuronales artificiales, mismos que son necesarios para explicar el funcionamiento general de las herramientas caso de estudio, que son descritas en el Capítulo 3.

1.1 Generalidades de las RNA

Las RNA están inspiradas en las redes neuronales biológicas del cerebro humano, y se constituyen por elementos que se comportan de forma similar a la neurona biológica en sus funciones más comunes [11].

El cerebro consta de billones de neuronas densamente interconectadas. Cada neurona es una simple unidad procesadora que recibe y combina señales desde y hacia otras neuronas mediante el axón (Figura 1). El axón (salida) de la neurona se ramifica y está conectada a las dendritas (entradas) de otras neuronas a través de uniones llamadas sinapsis. La eficacia de la sinapsis es modificable durante el proceso de aprendizaje de la red [12].

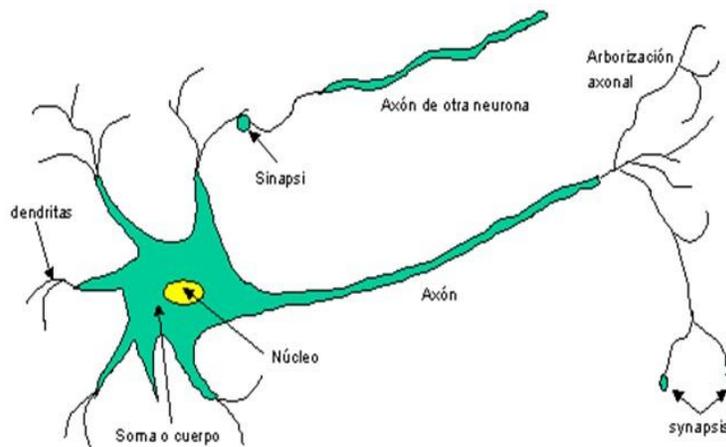


Figura 1. Esquema de una neurona biológica [12].

En contraste, una red neuronal artificial (RNA) es un sistema de procesamiento de información que tiene ciertas aptitudes en común con las redes neuronales biológicas [13]:

- El procesamiento de información ocurre en muchos elementos simples llamados neuronas.
- Las señales son transferidas entre neuronas a través de enlaces de conexión.
- Cada conexión tiene un peso asociado, el cual, típicamente, multiplica a la señal transmitida.
- Cada neurona aplica una función de activación (usualmente no lineal) a su entrada de red (suma de entradas pesadas) para determinar su salida.

Existen numerosas formas de definir a las redes neuronales, algunas de ellas son:

- Forma de emular ciertas características propias de los humanos, como la capacidad de memorizar y de asociar hechos [14].
- Redes interconectadas masivamente en paralelo de elementos simples (usualmente adaptativos) y con organización jerárquica, las cuales intentan interactuar con los objetos del mundo real del mismo modo que lo hace el sistema nervioso biológico [14].

Derivado de lo anterior, las RNA básicamente se pueden interpretar como la emulación virtual del cerebro humano, puesto que funcionan muy similar a él y de igual manera se tienen que mandar distintos pesos o como en el cerebro humano se ve como las señales que manda a distintas partes del cuerpo humano, así como también el cerebro humano tiende a pensar, las RNA tienen la parte de entrenamiento que es algo muy similar. En este sentido, algunas de las ventajas de algunas RNA son [15]:

- **Aprendizaje Adaptativo:** Capacidad de aprender a realizar tareas basadas en un entrenamiento o en una experiencia inicial.
- **Auto-organización:** Se trata de una particularidad que algunos modelos de RNA tienen para crear su propia organización o representación de la información que recibe en la etapa de aprendizaje.

- **Tolerancia a fallos:** Si se produce un fallo en un número no muy grande de neuronas y aunque el comportamiento del sistema se ve influenciado, no sufre una caída repentina.
- **Operación en tiempo real:** Los procesos neuronales pueden ser realizados en paralelo; con la capacidad de dar respuestas en tiempo real.
- **Fácil inserción dentro de la tecnología existente:** Se pueden obtener chips especializados para redes neuronales que mejoran la capacidad de respuesta en ciertas tareas, lo que facilitará la integración modular en los sistemas existentes.

1.2 Topología de las RNA

La topología o arquitectura de una RNA consiste en la organización y disposición de las neuronas en la misma, formando capas o agrupaciones de neuronas más o menos alejadas de la entrada y salida de dicha red. En este sentido, los parámetros fundamentales de la red son [16]: el número de capas, el número de neuronas por capa, el grado de conectividad y el tipo de conexiones entre neuronas.

- **Redes monocapa.** Las redes monocapa generalmente son utilizadas para realizar tareas de auto asociación. En este tipo de redes, se realizan las conexiones entre las neuronas de la única capa que constituye la red hacia la salida (Figura 2).

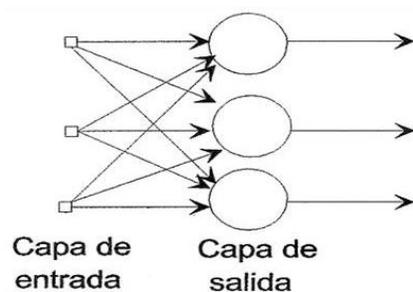


Figura 2. Red neuronal monocapa [16].

- **Redes multicapa.** Este tipo de redes son las que disponen de un conjunto de neuronas, agrupadas en distintos niveles o capas. Para estos casos, se puede distinguir la capa a la que pertenece una neurona solo con identificar el origen de las señales que recibe la entrada e identificar el destino de la señal de salida. Por lo regular, todas las neuronas de una capa reciben señales de entrada de una capa anterior y posteriormente envían señales de salida a una capa posterior (que está más cerca a la salida de la red) (Figura 3).

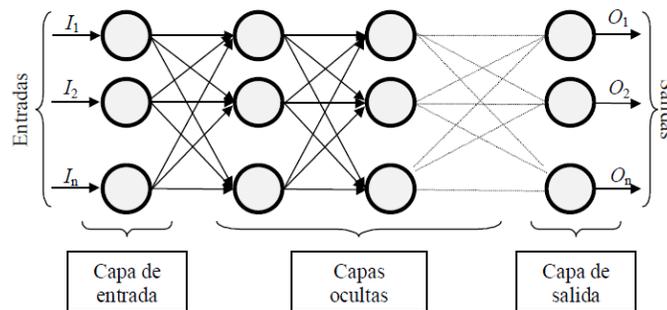


Figura 3: Red neuronal multicapa [16].

A partir de la Figura 3, se pueden distinguir tres tipos de capas [17]:

- **De entrada:** es la capa que recibe directamente la información proveniente de las fuentes externas de la red.
- **Ocultas:** son internas a la red y no tienen contacto directo con el entorno exterior. Las neuronas de las capas ocultas se interconectan con las neuronas de la capa de salida.
- **De salidas:** Recibe la información de las capas ocultas y transfieren la información de la red hacia el exterior.

1.3 Conectividad de la RNA

En un gran número de estas redes también existe la posibilidad de conectar la salida de las neuronas de capas posteriores a la entrada de capas anteriores. De acuerdo

a esto, se pueden agrupar las RNA en dos grandes grupos, dependiendo de la conectividad que éstas tengan [18]:

- RNA con conexiones hacia adelante o redes *feedforward*: son aquellas en las que, como su nombre indica, la información se mueve en un único sentido, desde la entrada hacia la salida. Estas redes están clásicamente organizadas en "capas". Cada capa agrupa a un conjunto de neuronas que reciben sinapsis de las neuronas de la capa anterior y emiten salidas hacia las neuronas de la capa siguiente. Entre las neuronas de una misma capa no hay sinapsis.
- RNA con conexiones hacia adelante y hacia atrás entre neuronas o *redes feedforward/feedback*: en este tipo de redes circula información tanto hacia adelante como hacia atrás durante el funcionamiento de la red (Figura 4) [19].

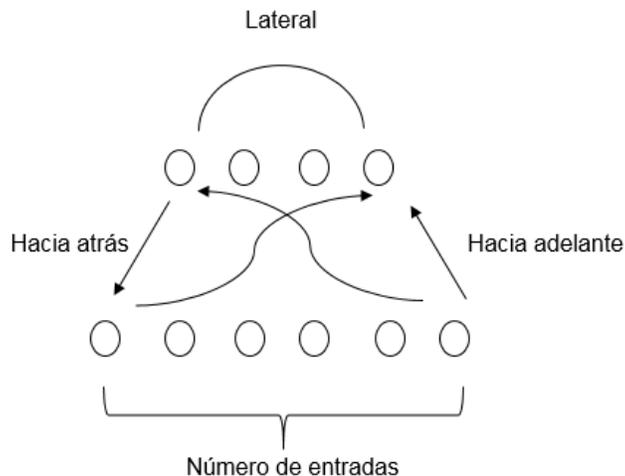


Figura 4: RNA con propagación *feedforward/feedback*

1.4 Paradigmas de aprendizaje en RNA

Una vez diseñada la arquitectura de la red (capas y número de neuronas por capa) y las funciones que la regirán, se tiene que proceder a entrenar a la red para que aprenda el comportamiento que debe tener; es decir, para que aprenda a dar la respuesta adecuada a la configuración de estímulos o patrones de entrada que se le presenten [20]. En la literatura se distinguen al menos tres diferentes estrategias

para realizar el proceso de aprendizaje de la red, mismas que son explicadas a continuación.

- **Supervisado:** El aprendizaje supervisado presenta a la red las salidas que debe proporcionar ante los patrones de entrada. Se observa la salida de la red y se determina la diferencia entre ésta y la señal deseada; para realizar esto es necesario presentar un conjunto de datos o patrones de entrenamiento para determinar los pesos (parámetros de diseño) de las interconexiones de las neuronas [20].
- **No supervisado:** En los algoritmos no supervisados no se conoce la salida que debe presentar la RN. La red en este caso se organiza ella misma agrupando, según sus características, los diferentes patrones de entrada. Estos sistemas proporcionan un método de clasificación de las diferentes entradas mediante técnicas de agrupamiento o *clustering* [20].
- **Híbrido:** En este caso existen en la red los dos tipos de aprendizaje básicos, supervisado y no supervisado en distintas capas de neuronas [20].

1.5 Proceso de reconocimiento

Una vez que la RNA ha finalizado su proceso de entrenamiento, da inicio lo que se conoce como validación y prueba. Esta etapa consiste en probar los resultados de acuerdo a los pesos obtenidos y comparar los resultados finales con los esperados.

En esta fase se le pedirá a la red que responda a estímulos diferentes a los presentados durante la fase de entrenamiento. Gracias a los ejemplos aprendidos del juego de ensayo, la red deberá ser capaz de generalizar y dar respuestas correctas ante patrones de estímulos nuevos [21].

1.6 Aplicaciones de Redes Neuronales

Las RNA pueden utilizarse en un gran número y variedad de aplicaciones. Algunas aplicaciones comerciales son las siguientes:

1. Mercadotecnia. En este ejemplo [22], se utiliza un Perceptrón Multicapa con un tipo de aprendizaje denominado *Extreme Learning Machine* (ELM), con la siguiente arquitectura:

- 4 nodos en la capa de entrada, correspondientes a cada una de las 4 variables explicativas: calidad del producto (Q), precio (P), distribución (D) y comunicación (C) normalizadas en el intervalo $[-1,1]$.
- 2 nodos en la de salida, uno para cada una de las dos categorías de clasificación: más exportadoras (E=1) y menos exportadoras (E=0).
- 10 nodos en la capa intermedia.
- La función de transferencia polinomial de tercer grado.

Con una muestra de 32 empresas, se evaluó el rendimiento de la red analizando los índices de sensibilidad y especificidad, se realizó la clasificación de las empresas como las más exportadora (E=1) o menor exportadora (E=0).

2. Video juegos. Demis Hassabis y su equipo de investigadores en Inteligencia Artificial de la empresa *Deep Mind*, de Google, han diseñado una red neuronal artificial capaz de aprender a dominar diversos videojuegos [23].

El estudio muestra que este agente artificial ha aprendido por sí mismo a jugar a 49 videojuegos clásicos de la videoconsola Atari 2600, entre los que se encuentran Pacman y *Space Invaders*, partiendo de información sobre los píxeles y el número de acciones posibles en cada juego.

El sistema es capaz de descubrir el objetivo del juego y dominar sus controles sin contar con más información que las imágenes que aparecen en pantalla y la puntuación. Gracias a sus mecanismos de adaptación y aprendizaje, la máquina actúa a un nivel comparable al de un humano profesional de los videojuegos. Derivado del éxito del sistema, se pensó en utilizar este algoritmo para mejorar el motor de búsqueda de Google para completar tareas complejas como o la planeación de viajes.

3. Medicina. Existen diversas aplicaciones en medicina que utilizan RNA. Una de ellas es la propuesta por [24], en la que se propone un sistema para detección de problemas visuales con la arquitectura multicapa de una RNA con 12 neuronas en la capa de entrada, 10 en la capa oculta y 4 en la capa de salida. Para la experimentación se contó con una muestra de 120 pacientes agrupados en las diferentes clases que representan los problemas visuales a reconocer: refracción, astigmatismo, hipermetropía y miopía.

En la elaboración del algoritmo de la RNA se utiliza el ingreso de preguntas por medio de un formulario que le realiza la anamnesis al paciente, según las respuestas obtenidas, la RNA entrenada genera un diagnóstico y el resultado es presentado mostrando el problema visual con los ojos en 3D desarrollados en OPENGL. Finalmente, al usuario se le presenta el problema visual en un Entorno Virtual que funciona de manera inteligente, de acuerdo al proceso con la RNA MLP que tuvieron sus respuestas (Figura 5). Los resultados experimentales mostraron un 99.2% de precisión.

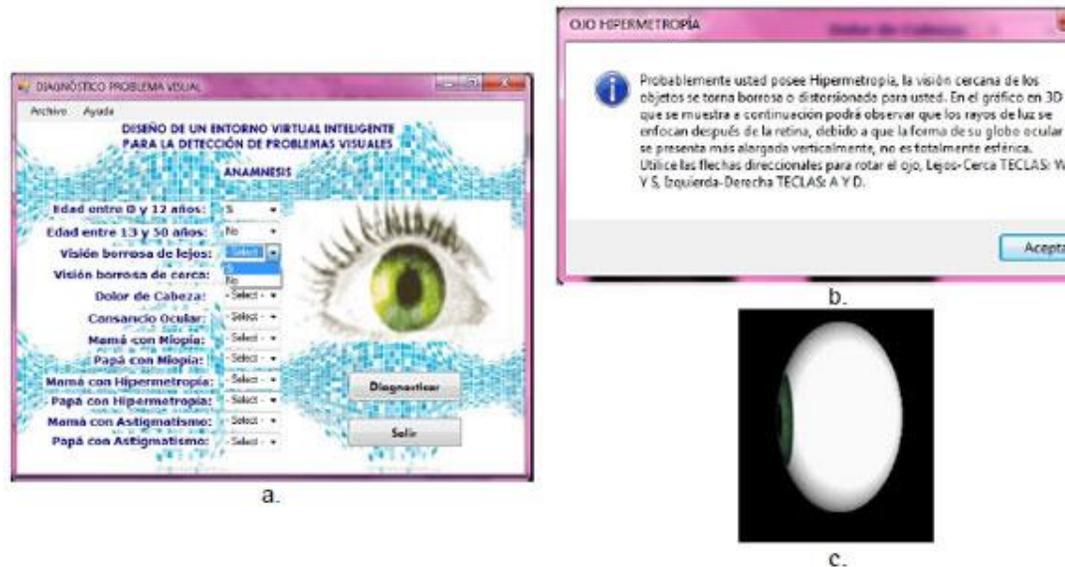


Figura 5: Resultados experimentales [20]

Capítulo 2. Características de las herramientas

En esta sección se describen algunas herramientas actualmente utilizadas para el procesamiento de RNA, así como un ejemplo de procesamiento con cada una de ellas.

2.1 Weka

WEKA en su versión 3.8 es una extensa colección de algoritmos de aprendizaje automático desarrollados por la Universidad de Waikato [25] implementados en Java, útiles para ser aplicados sobre datos mediante interfaces que ofrece o para embeberlos dentro de cualquier aplicación. Actualmente es el software con mayor reconocimiento en áreas de minería de datos y reconocimiento de patrones, contiene herramientas necesarias para realizar transformaciones sobre los datos, tareas de clasificación, regresión, *clustering*, asociación y visualización.

Además de estar diseñado como una herramienta orientada a la extensibilidad, es decir, permite añadir nuevas funcionalidades, tiene licencia de GPL (*General Public License*), lo que significa que este programa es de libre distribución y difusión. Actualmente la versión 3.8 es la más reciente, aunque ya se encuentra en desarrollo la versión 3.9 pero aún no es estable y confiable a diferencia de la versión 3.8.

Weka funciona a través de 4 métodos con los cuales se puede interactuar con la aplicación [21]:

- **Explorer:** Entorno visual que ofrece una interfaz gráfica para el uso de los paquetes.
- **Experimenter:** Entorno centrado en la automatización de tareas de manera que se facilite la realización de experimentos a gran escala.
- **Simple CLI:** Entorno consola para invocar directamente con java a los paquetes de Weka.

- **The Knowledge Flow Interface:** Entorno visual que ofrece una interfaz gráfica como alternativa al *Explorer* como un gráfico dirigido.

2.1.1 Explorer

El modo Explorador es el modo más usado y descriptivo. Éste permite realizar operaciones sobre un sólo archivo de datos. La ventana principal es la mostrada en la Figura 6:

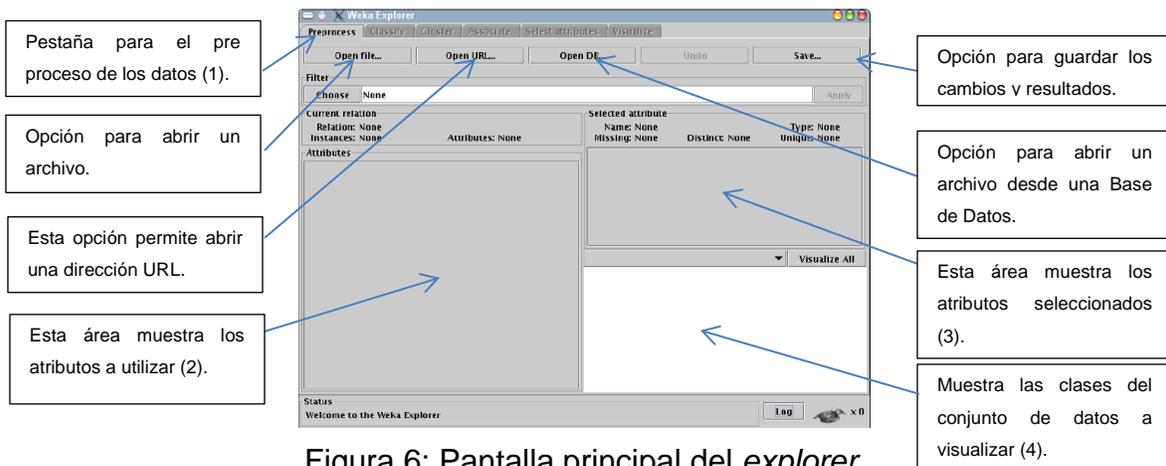


Figura 6: Pantalla principal del *explorer*.

Como se puede observar, el *explorer* permite tareas de:

- 1 **Pre-procesado de datos y aplicación de filtros:** Esta es la parte por la que se debe pasar antes de realizar cualquier otra operación, ya que se requiere de los datos para poder llevar a cabo cualquier análisis.
- 2 **Clasificación:** Esta opción permite construir un modelo para fines de clasificación, entrenamiento y evaluación del rendimiento. La clasificación se realiza bajo un enfoque supervisado, es decir, realiza la predicción de clase de nuevos patrones en función de una serie de atributos de entrada.
- 3 **Clustering:** Los algoritmos de agrupamiento buscan grupos de patrones con características similares, según un criterio de comparación entre valores de atributos de los patrones definidos en los algoritmos.
- 4 **Búsqueda de Asociaciones:** Mediante algoritmos de asociación se puede realizar la búsqueda automática de reglas que relacionan conjuntos de

atributos entre sí. Son algoritmos no supervisados, ya que no existen relaciones conocidas a priori con las que contrastar la validez de los resultados, sino que se evalúa si esas reglas son estadísticamente significativas.

5 **Selección de atributos:** Se refiere a la selección de entradas y salidas de la red, así como también la selección de topología y algoritmo de la misma red.

6 **Visualización de datos:** La herramienta de visualización de Weka permite representar gráficas 2D que relacionan pares de atributos, en las coordenadas horizontal y vertical. La idea es que se selecciona la gráfica deseada para verla en detalle en una ventana nueva.

2.1.2 Experimenter

El módulo de *Experimenter* sirve para aplicar varios algoritmos de aprendizaje automático sobre distintos conjuntos de datos y determinar de manera estadística su rendimiento en términos de precisión. La ventana principal es la que se muestra en la Figura 7.

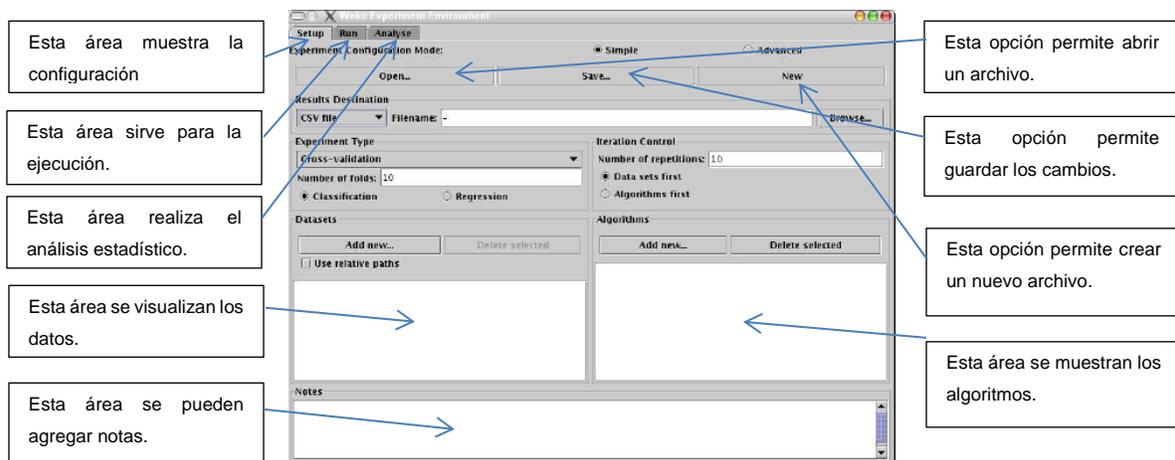


Figura 7: Pantalla principal del *Experimenter*.

2.1.3 Simple CLI

Simple CLI es una abreviación de *Simple Client*. Esta interfaz proporciona una consola para poder introducir comandos, realizar cualquier operación soportada por Weka de forma directa. Es complicada de manejar por usuarios inexpertos ya que

es necesario un conocimiento completo de la aplicación (Figura 8), por lo que se recomienda la utilización del módulo de *Experimenter*.

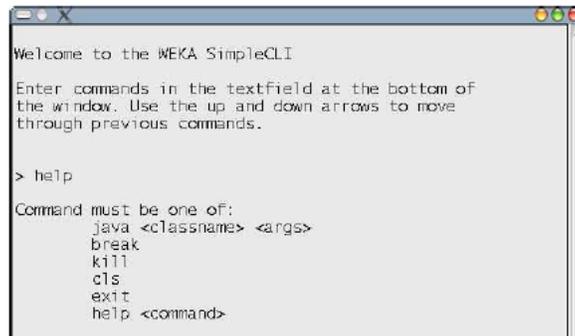


Figura 8: Pantalla principal de *Simple CLI*.

2.1.4 Knowledge Flow

Knowledge Flow permite a los usuarios seleccionar los componentes de Weka de una barra de herramientas, los colocan en un lienzo de diseño y los conectan a un gráfico dirigido que procesa y analiza los datos. Proporciona una alternativa al Explorer para aquellos que les gusta pensar en términos de cómo los datos fluyen a través del sistema. También permite el diseño y la ejecución de configuraciones para el procesamiento de datos transmitidos, lo que el Explorer no puede hacer (figura 9).

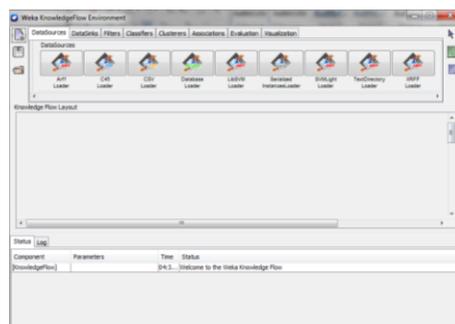


Figura 9: Pantalla principal de *Knowledge Flow*

2.1.5 RNA con Weka

Weka contiene muchos algoritmos, de los cuales los siguientes son los relacionados con RNA [2]:

- **MultilayerPerceptron.** Con el algoritmo *MultilayerPerceptron* se crea una red neuronal multicapa en donde enlaza todos los nodos de la capa de entrada con la capa oculta (de procesamiento). Luego se asigna un valor de activación a cada uno de los nodos mediante una retro propagación. A partir de estos datos, se realiza un entrenamiento de la red, lo cual conduce a la respuesta o salida de la red [26].
- **BayesNet.** Es un modelo grafo probabilístico que representa un conjunto de variables aleatorias y sus dependencias condicionales a través de un grafo cíclico dirigido que utiliza varios algoritmos de búsqueda y medidas cualitativas que determinan la red o estructura gráfica de las probabilidades de los patrones de entrada [26].
- **RBFNetwork.** (*Radial basis function network*). *RBFNetwork* implementa una red gaussiana normalizada conocida como función de base radial, la cual construye sus modelos con funciones de activación que son diferentes tanto en la capa oculta como la de salida. Esto es, las neuronas en la capa oculta se activan mediante funciones radiales de carácter no lineal con sus centros gravitacionales propios y en la capa de salida mediante funciones lineales [26].
- **SMO.** *Sequential Minimal Optimization* (SMO) es un método rápido para el entrenamiento de máquinas de soporte vectorial (SVMs). El entrenamiento de una SVM requiere la solución a un problema de programación cuadrática (QP), para lo cual SMO particiona este problema en una serie de problemas más pequeños QP. Estos problemas más pequeños son resueltos de forma analítica, con lo que se reduce significativamente el tiempo de procesamiento [26].
- **Métodos de aprendizaje.** Los métodos de aprendizaje que se encuentran disponibles en *Weka* son: *Backpropagation*, *Kohonen*, *Art 1*, *Art 2*, *Self-Organization* y *Hopfield*.

2.1.6 Ventajas

Algunas ventajas referentes al uso de RNA con Weka:

- Disponible libremente bajo la licencia pública general de GNU (*General Public License*), lo que facilita su acceso y permite tener un mejor intercambio entre redes creadas en este software.
- Es portable debido a que está totalmente implementado en Java y puede correr casi en cualquier plataforma, logrando una mejor compatibilidad de funcionamiento e interacción con el software sin importar el Sistema Operativo que se esté utilizando.
- Es fácil de utilizar por un principiante gracias a su interfaz gráfica de usuario. Gracias a esto permite una mejor interacción del usuario con el software y a su vez se hace más fácil su manejo sin necesidad de tener un nivel alto de experiencia en cuanto a programación.
- Proporciona acceso a bases de datos vía SQL, lo que quiere decir; que se pueden manejar los distintos problemas con bases de datos que sean creadas en SQL y así poder combinar las bases con el software.

2.1.7 Desventajas

- La principal desventaja de Weka es que no se cuenta con la documentación necesaria orientada al usuario. Aunque no es difícil la interacción con la herramienta, se cuenta con poca información o tutoriales que ayuden a la manipulación de la aplicación.
- Cuando se utilizan métodos de combinación de modelos, los resultados suelen ser más complicados y dada esta situación; es más difícil la comprensión de estos por parte de usuarios inexpertos.

2.2 Matlab

Matrix Laboratory es un software matemático que tiene un entorno IDE (desarrollo integrado) y un lenguaje propio que puede utilizarse en Unix, Windows y Mac OS.

El entorno de desarrollo que Matlab en su versión R2016b ofrece está totalmente integrado y orientado a llevar a cabo proyectos en donde se encuentren implicados elevados cálculos matemáticos y la visualización gráfica de los mismos. Para esto Matlab integra análisis numérico, cálculo matricial, procesamiento de señales y visualización gráfica en un entorno completo donde los problemas y sus soluciones son expresados del mismo modo en que se escribirían tradicionalmente, sin necesidad de hacer uso de la programación tradicional [27].

Entre las utilerías de Matlab, la utilería *Neural Network Toolbox* es la que cuenta con utilerías para RNA de aprendizaje supervisado y no supervisado. Incluye los mapas auto-organizados y capas competitivas. Con la caja de herramientas se puede diseñar, entrenar, visualizar y simular redes neuronales para aplicaciones como ajuste de datos, reconocimiento de patrones, agrupamiento, la predicción de series de tiempo y el modelado de sistemas dinámicos y control [27].

Dentro de las aplicaciones básicas de este *toolbox*, destacan aquellas que se enmarcan dentro del campo de la industria aeroespacial y automoción (simulación, sistemas de control, auto pilotaje), banca, defensa (reconocimiento de patrones, procesamiento de señales, identificación de imágenes, extracción de características, compresión de datos), electrónica (control de procesos, análisis de errores, modelado no lineal, síntesis de voz, visión por ordenador), economía (análisis financiero, análisis predictivo), industria (control de procesos, identificación en tiempo real, sistemas de inspección), medicina, robótica (control de trayectorias, sistemas de visión), reconocimiento y síntesis del habla, telecomunicaciones (control de datos e imágenes, servicios de información automatizada, traducción del lenguaje hablado en tiempo real, diagnosis, sistemas de enrutamiento), etc.

2.2.1 Métodos de Aprendizaje

Los métodos de aprendizaje que Matlab incluye son tres:

- **Aprendizaje supervisado.** Un conjunto conocido de datos de entrada-salida se utiliza para iterativamente ajustar los pesos de la red.

- **Aprendizaje no supervisado.** únicamente se disponen de datos de entrada y una función de coste a minimizar.
- **Aprendizaje reforzado.** Combina el aprendizaje supervisado y el no supervisado. La información que se proporciona a la red es mínima, limitándose a indicar si la respuesta de la red es correcta o incorrecta.

2.2.2 Entorno de trabajo de Matlab

Matlab utiliza una interfaz única que da acceso a todas las opciones para la RNA, independientemente del tipo que esta sea, de tal manera que utilizando las distintas opciones de configuración que esta ofrece, se podrán modificar las entradas, capas, conexiones, pesos, etc. Una vez configurada la RNA, se invocan las funciones de simulación, entrenamiento, inicialización, etc., pasándole como parámetro la estructura de la RNA (Figura 10) [28]:

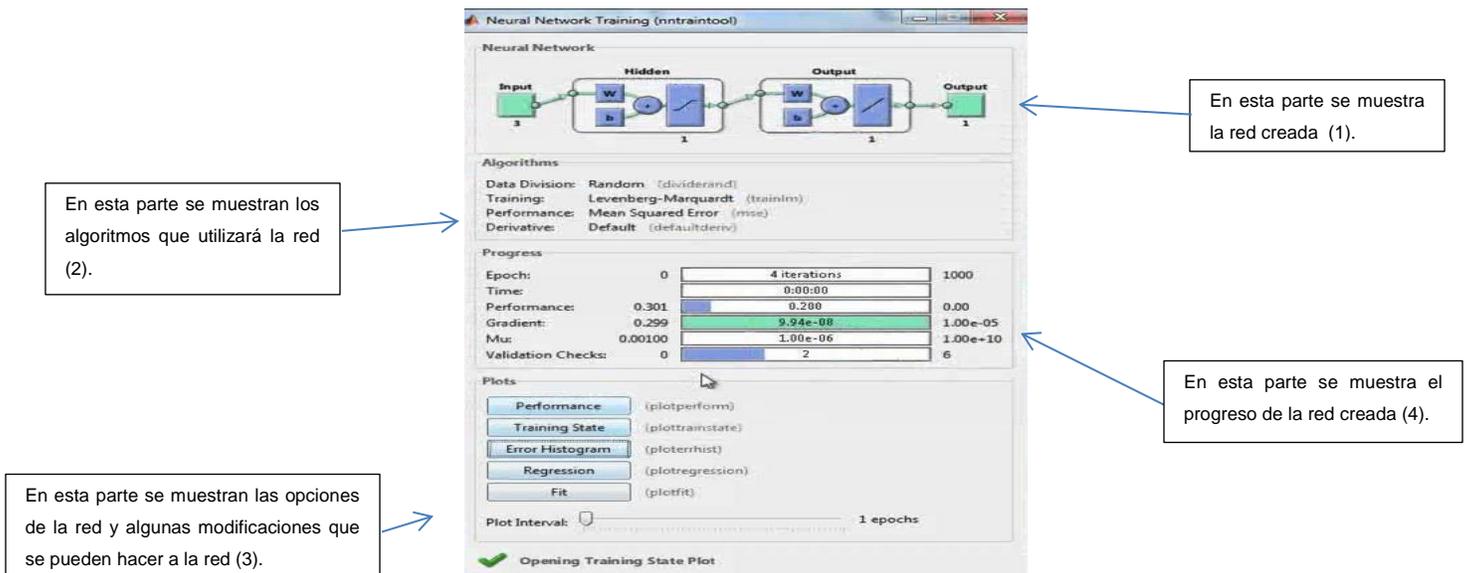


Figura 10: Pantalla de *Neuronal Toolbox*

El contenido de cada apartado, se divide en cinco secciones que se mencionan a continuación:

1. **Arquitectura:** Define las características básicas de la RNA, número de entradas, capas, conexiones, etc.

2. **Sub objetos:** Contiene referencias a las subestructuras de la RNA, que permiten configurar las propiedades de los distintos componentes que la forman (capas, entradas, salidas, etc.).
3. **Funciones:** Funciones principales de la RNA, utilizadas para ejecutar las operaciones de inicialización, entrenamiento o simulación.
4. **Parámetros:** Configuración de los parámetros asociados a las funciones seleccionadas en el bloque de funciones.
5. **Valores:** Aquí se definen las matrices con los valores de los pesos de entrada, conexiones entre capas y vías.

Una vez creada la RNA, se pueden utilizar las siguientes funciones para realizar las operaciones típicas de RNA:

1. **Inicialización (`net = init(net)`):** Mediante la función de inicialización, se obtiene una RNA con los valores de los pesos y bias actualizados según las funciones de inicialización que se le hayan asociado a la red, mediante su propiedad `net.initFcn`, o bien `net.layers{i}.initFcn` y `net.biases{i}.initFcn`.
2. **Entrenamiento (`[net, tr, Y, E, Pf, Af] = train (net, P, T, Pi, Ai, VV, TV);`):** En esta parte se realiza el entrenamiento de la red neuronal y para ello es necesario modificar los pesos que están asociados a las conexiones entre las distintas capas y neuronas de la red. Para ello, se debe indicar los patrones de entrada que tendrá la red (P , matriz de dimensiones $M \times N$ siendo M la suma de los tamaños de las capas de entrada de la red neuronal, y N el número de patrones que deseamos aplicar en el entrenamiento).
En el caso de ser un entrenamiento del tipo supervisado es necesario indicar los patrones (T , matriz de $M \times N$), con estos datos la matriz de patrones se aplica a la red neuronal, y el *toolbox* utilizando las funciones de entrenamiento que se ha indicado en las propiedades "`trainFcn`" la cual se encargará de actualizar los pesos asociados a las conexiones de la red. Los resultados del entrenamiento se visualizarán en la variable de retorno Y y los

errores para cada patrón de entrada respecto a la salida esperada en la variable de retorno E .

3. **Simulación** ($[Y, Pf, Af, E, perf] = sim(net, P, Pi, Ai, T)$): Función parecida a la anterior pero que no actualizará los pesos de la red neuronal. Una vez que se tiene entrenada la red neuronal y esta ofrezca resultados aceptables, se utilizará esta función para analizar nuevos patrones de entrada.

2.2.3 Ventajas

A continuación, se mencionan algunas de las ventajas que son consideradas como importantes de Matlab en el uso de RNA:

- Su capacidad gráfica, en 2 y en 3 dimensiones permiten visualizar de mejor manera la simulación de las RNA y gracias a ello se pueden interpretar de mejor manera los resultados.
- Lenguaje de alto nivel basado en vectores, *arrays* y matrices. Para los expertos en programación esto les permite trabajar de mejor manera la creación de las RNA.
- Tiene una colección de funciones de aplicación lo que facilita más la interacción del usuario con la aplicación.
- Al utilizar el *Neuronal Toolbox*, el usuario puede realizar las redes neuronales con mayor facilidad gracias a su interfaz gráfica y las opciones que esta proporciona.
- Los archivos de Matlab son completamente portátiles en diferentes plataformas (Linux, Apple, Windows), lo que facilita el manejo del software en los distintos Sistemas Operativos.
- Se puede encontrar una gran cantidad de archivos gratis en Internet, escritos en Matlab que pueden servir como referencia de consulta y facilitar el uso de Matlab y sus herramientas.

2.2.4 Desventajas

A continuación, se mencionan algunas de las desventajas que sobresalen acerca de Matlab en RNA:

- No es gratuito y la licencia de Matlab cuesta USD \$99.00 sin contar el costo de *Neural Network Toolbox*, pues este *toolbox* tiene un costo de USD \$250.00 para la versión de Estudiante (*Student*) tanto para la licencia de Matlab como la del *toolbox* (dato obtenido de la página oficial de Matlab consultado el día 20 de Mayo de 2018).
- Por su costo, el estudiante tiende a perder el interés de utilizar esta aplicación ya que en ocasiones es complicado tener al alcance una licencia de este tipo.
- Se debe tener conocimientos de programación puesto que, en caso de no contar con estos conocimientos, será muy difícil la manipulación de este software por lo que para algunos estudiantes de la Facultad de Ingeniería se les dificulta programar.

2.3 NeuroSolutions

NeuroSolutions es una herramienta de desarrollo gráfico de RNA que permite crear fácilmente un modelo de red neuronal para los datos, basándose en una aproximación al diseño de sistemas adaptativos orientada a objetos. De este modo, las redes son descompuestas en un conjunto fundamental de componentes que son individualmente simples pero susceptibles a ser combinados para crear sistemas capaces de resolver problemas complejos [6]. Algunas de las arquitecturas neuronales y métodos de aprendizaje que incluye son: Perceptrón Multicapa, Modular, Retroalimentación generalizada, *Jordan/Elman*, Mapas auto-organizables (SOM), Función radial básica (RBF), Red neuronal probabilística (PNN), Red neuronal de regresión general (GRNN), Red neuronal tipo *Time Delay* (TDNN), Red neuronal tipo *Time-LAg* (TLRN), Red recurrente, *Neuro-Fuzzi* (CANFIS) y SVM (*support vector machine network*).

2.3.1 Métodos de Aprendizaje

- *Backpropagation*.
- *Backpropagation* a través del tiempo (BPTT)
- *Backpropagation* recurrente
- Gradientes conjugados
- *Teacher Forcing*/predicciones iterativas
- Aprendizaje no-Supervisado
- Hebbian
- Ojas
- Sanger
- Kohonen

2.3.2 Entorno de trabajo de NeuroSolutions

La interfaz de usuario de NeuroSolutions está inspirada en el proceso de diseño de un circuito electrónico. Diversos componentes son colocados en una tablilla de conexión y después son interconectados para formar un circuito que puede ser validado, introduciendo señales y monitoreando las respuestas (Figura 11).

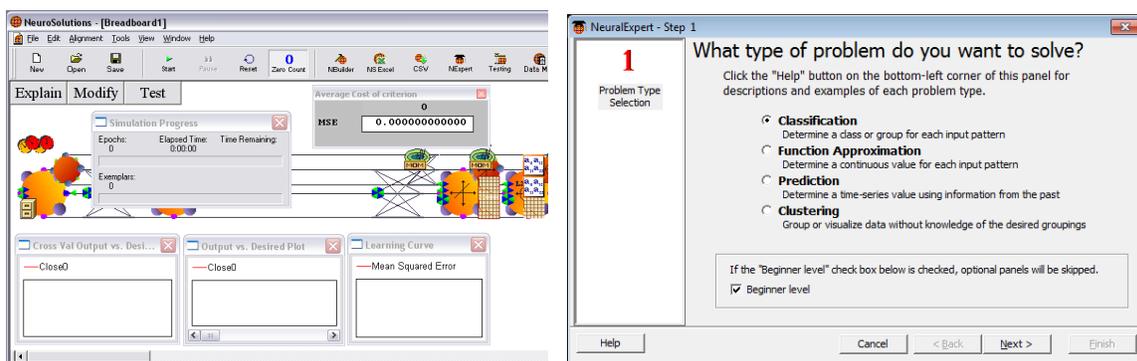


Figura 11: Pantalla principal de NeuroSolutions

La interfaz de usuario de NeuroSolutions organiza sus componentes en paletas, donde cada paleta de componentes contiene una familia de componentes que tienen una función similar. Se pueden encontrar todas las paletas en el menú de paletas como se muestra en la Figura 12.



Figura 12: Menú de paletas.

Si se abre una paleta, se marcará inmediatamente con una palomita. Una paleta abierta se ve y actúa como una barra de herramientas con un conjunto de iconos. Las paletas pueden ser desplazadas y colocadas como cualquier barra de herramientas estándar de Windows. La figura 13 muestra como ejemplo la paleta de la familia Axón.



Figura 13: Paleta de la familia Axón.

Para la construcción de una RNA, NeuroSolutions dispone de las familias de opciones Axón y Sinapsis. Por un lado, la familia Axón (Figura 14) implementa los elementos de procesamiento en la red y tiene dos funciones.

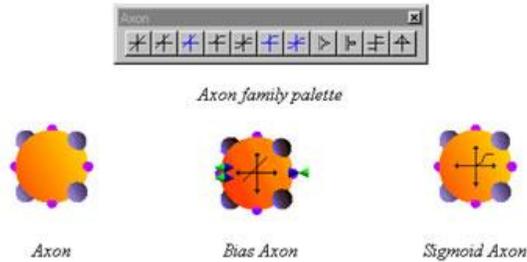


Figura 14: Paleta de la familia Axón y unos componentes

Los diferentes componentes en la familia Axón aplican una función diferente a la suma de entradas. Cada una de las diferentes imágenes en los iconos de dicha familia representa la función que será aplicada. El axón de tipo lineal (llamado simplemente Axón) solamente pasa la suma de las entradas directamente a la salida. El axón Bias suma la entrada y agrega un valor de umbral. Un sólo componente de la familia Axón puede representar cualquier número de los elementos de procesamiento. Por otro lado, la familia Sinapsis implementa las conexiones entre los axones (Figura 15).

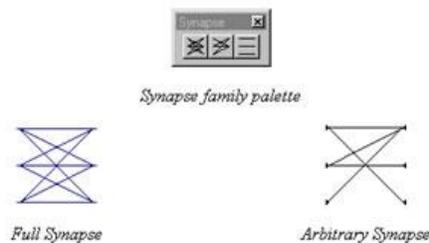


Figura 15: Paleta de la familia Sinapsis y componentes

En general, se le asigna un peso a cada conexión en una sinapsis que hace un escalamiento de los datos que pasan a través de ella. La manera en que se ajustan esos pesos será la manera en que se entrena una red o sistema adaptativo para ejecutar la tarea deseada.

El miembro más utilizado de esta familia es la Sinapsis Completa y conecta todos y cada uno de los elementos de procesamiento en el axón del otro componente. Además, la Sinapsis Completa contiene nm conexiones (y pesos) para n elementos de procesamiento de entrada y m elementos de procesamiento de salida. La

Sinapsis Arbitraria permite seleccionar cuál de las nm conexiones posibles unirá un axón a otro.

2.3.3 Funcionamiento de Redes Neuronales

En esta sección se explica el proceso de entrenamiento del Perceptrón multicapa utilizando gradiente descendente sobre las tres herramientas estudiadas.

El aprendizaje adaptativo que utiliza el Perceptrón multicapa se basa en utilizar el error entre la salida del sistema y la salida del sistema deseada para entrenar la RNA. El algoritmo de aprendizaje adapta los pesos del sistema basado en el error hasta que el sistema produzca la salida deseada. La familia de Criterios de Error, que se muestra en la figura 16. NeuroSolutions calcula diferentes medidas de error que pueden ser utilizadas en el entrenamiento de una red.



Figura 16: Paleta de la familia Criterios de Error

Una vez que se ha obtenido la salida y se desea hacer la propagación del error hacia atrás para actualizar los pesos de la red Puede ser utilizado el error para poder realizar modificaciones a los pesos del sistema y con ello permitir el aprendizaje (Figura 17). El objetivo del sistema es igualar la salida del sistema a la salida deseada, para esto se necesita minimizar el error.

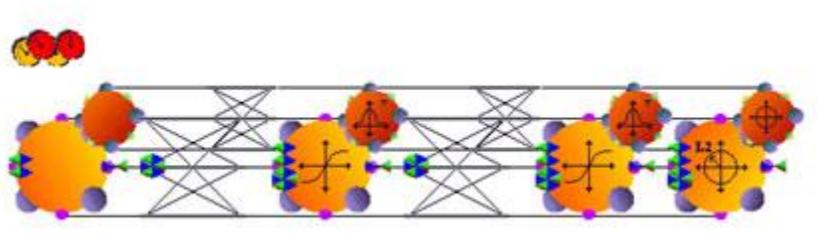


Figura 17: RNA con plano de retro propagación

NeuroSolutions visualiza el plano de retro propagación anterior gracias a que se utilizan versiones de tamaño menor de axones y de sinapsis en forma de pila. Este plano regresa los errores desde el componente de criterios hasta llegar al principio de la red. NeuroSolutions posteriormente agrega un tercer plano donde utiliza el error para poder cambiar los pesos de la red en el plano de retro propagación y es aquí donde realmente se realiza el aprendizaje.

Este tercer plano es conocido como plano de gradiente descendente que se puede encontrar en la parte de arriba del plano de retro propagación. En la Figura 18 se puede observar la simulación de una red neuronal con los tres planos:

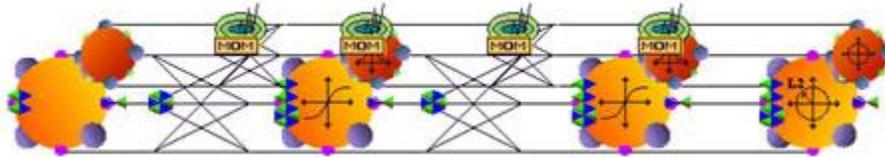


Figura 18: RNA con los tres planos: propagación hacia adelante, retro propagación y gradiente descendente.

2.3.4 Ventajas

A continuación, se mencionan algunas de las ventajas que sobresalen acerca de NeuroSolutions en RNA:

- Crear y entrenar una RNA y ver de forma gráfica el rendimiento de la red, sirve para poder visualizar la red de manera que sea más entendible para el usuario.
- Se pueden incorporar fácilmente modelos de redes neuronales en sus propias aplicaciones, lo que facilita la unión o combinación de redes creadas en el mismo software.
- Los usuarios pueden apreciar que al utilizar esta aplicación se tiene la facilidad de trabajar con RNA de una manera muy rápida ya que esta es una de las características de NeuroSolutions, su velocidad.
- Tiene una interfaz gráfica que es muy fácil para la interacción hombre – máquina y que gracias a sus componentes gráficos se puede ir

seleccionando lo se desee para la creación de la red neuronal pues cada icono muestra una descripción y al estar en el área de trabajo, se puede hacer la creación de manera sencilla.

2.3.5 Desventajas

Algunas de las principales desventajas de NeuroSolutions son la escasa documentación orientada al usuario, además de ser una aplicación costosa que maneja 3 tipos de licencias (1 año c/u):

- NeuroSolutions Pro: *USD \$1,495.00*
- NeuroSolutions: *USD \$495.00*
- NeuroSolutions *Student*: *USD \$295.00*

Capítulo 3. Pruebas y usabilidad

En este capítulo se implementa 1 mismo ejemplo para cada una de las herramientas con la finalidad de poder observar la forma que muestran los resultados y con ello comparar las herramientas. Posteriormente se muestran graficas que miden la usabilidad de cada una de las herramientas reflejando un resultado final de cuál es la que mejor le parece al usuario.

3.1 Realización de pruebas con las herramientas

Para la realización de los ejercicios se utilizó el conjunto de datos (CD) "breast-cancer", el cual cuenta con 150 patrones, 30 atributos (entradas) y 2 clases (salidas).

3.1.1 Ejercicio con weka

A continuación, se explica un ejemplo del funcionamiento de Weka en su versión 3.8 utilizando el algoritmo Perceptrón Multicapa. Utilizando el CD descrito anteriormente, empieza por abrir Weka y a continuación la herramienta Explorer. Se presiona en "Open file" y se selecciona el archivo "breast-cancer". Una vez abierto se podrá ver una serie de datos y gráficos en la ventana del Explorer, como se puede ver en la Figura 19:

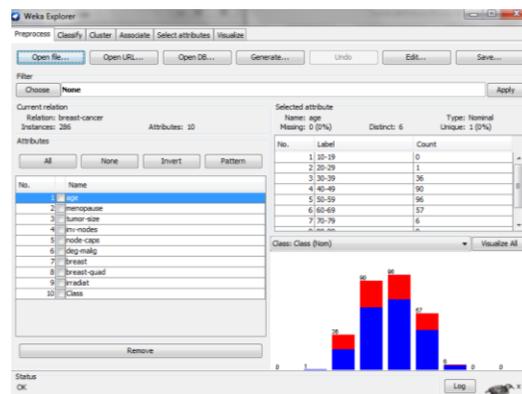


Figura 19: Pantalla de explorer

Si se desea conocer la distribución de todos los datos de manera gráfica, se debe elegir la opción “*Visualize All*” y mostrará todas las gráficas (Figura 20).

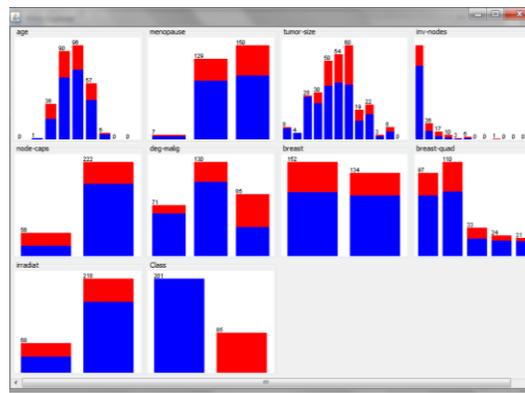


Figura 20: Pantalla de todos los atributos.

Se selecciona la pestaña “*Classify*” en la pantalla principal y se da clic en “*Choose*” para elegir el clasificador. En este caso se selecciona *MultilayerPerceptron* dentro del apartado “*functions*”. Por último, se selecciona “*Training set*” dentro del apartado “*Test options*” para realizar el proceso de entrenamiento y se da clic en “*Start*” como se muestra en la Figura 21:

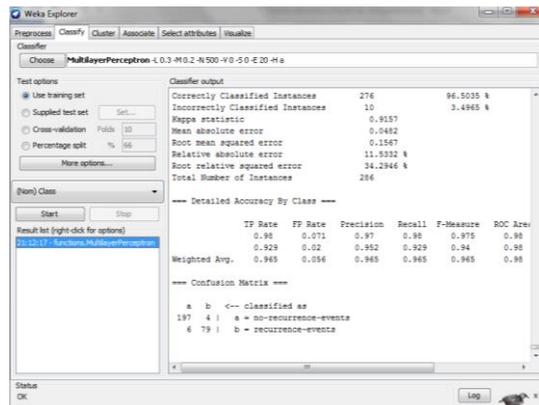


Figura 21: Resultados del algoritmo.

Como se puede observar, en la parte de resultados se muestra la hora en la que se realizó el análisis especificando el algoritmo que se utilizó. En la parte de salidas, se muestra las instancias que fueron correctamente clasificadas que en este caso

fue el 96.5035% (276 de 286) y 3.4965% (10 de 286) en la parte de instancias incorrectas, también se menciona la parte de posibles errores, así como también nos muestra una matriz de confusión donde nos indica los eventos recurrentes al igual que los eventos no recurrentes.

A modo de conclusión se puede decir lo siguiente: El ejemplo anterior se ejecutó con un Perceptrón Multicapa, pero igual se puede implementar con los otros modelos de RNA que Weka posee ya que la operación es muy similar para las demás redes y los resultados resultan sencillos de interpretar gracias a la información que se proporciona en la pantalla de resultados.

Como pudo verse, Weka es una herramienta muy fácil de utilizar ya que tiene una gran capacidad de analizar, clasificar e interpretar el contenido de bases de datos que de igual forma pueden estar conectadas desde SQL.

3.1.2 Ejercicio con Matlab

A continuación, se muestra un ejemplo del funcionamiento de Matlab en su versión R2016b (*trial*) utilizando el algoritmo Perceptrón Multicapa y el CD "*Breast Cancer*". Al iniciar el asistente, lo primero que se debe especificar es el tipo de problema que tiene que resolver la red neuronal (figura 22):

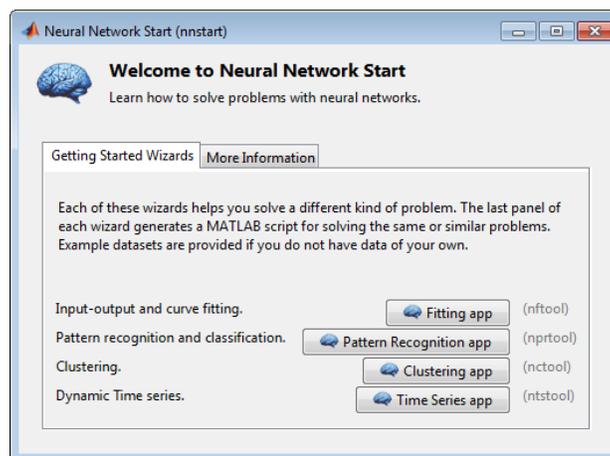


Figura 22: Pantalla de inicio

Posteriormente se tiene que seleccionar el archivo con el conjunto de datos que se quiere trabajar (figura 23):

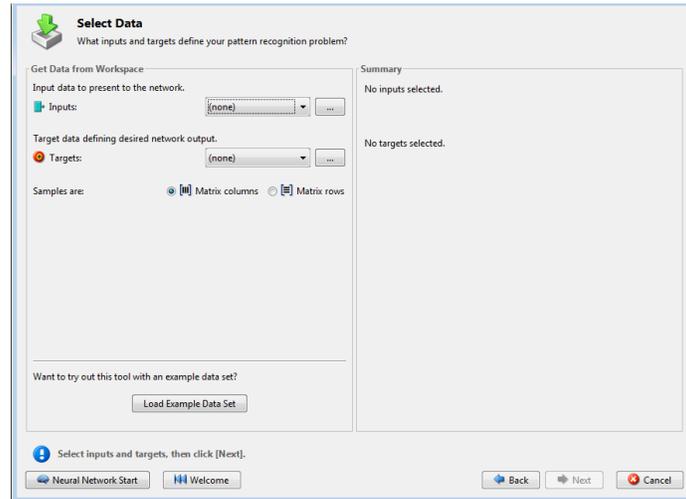


Figura 23: Pantalla de datos

Para este ejemplo se cargara el archivo “Breast Cancer” y se procede a dar clic en la opción *Import* (figura 24):

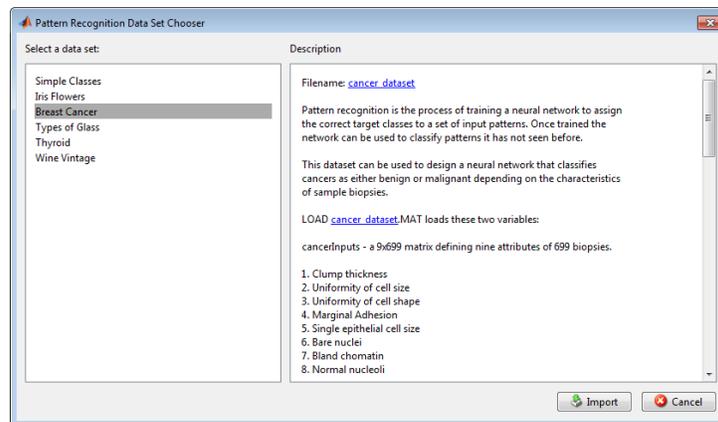


Figura 24: Selección de archivo.

Posterior a esto, se procede al entrenamiento de la red como se muestra en la siguiente figura dando clic en la opción *train* (figura 25):

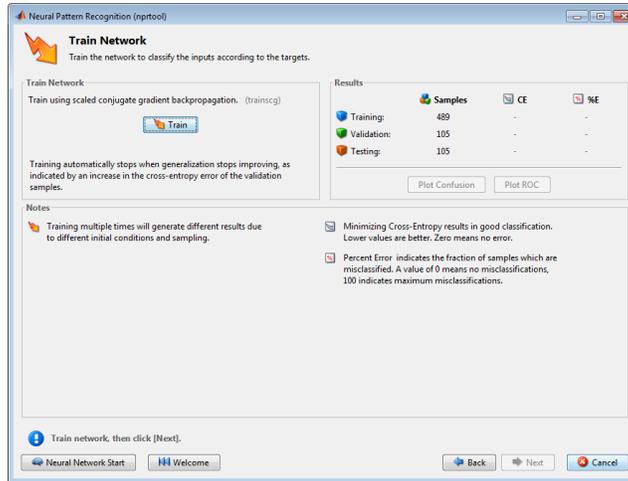


Figura 25: Entrenamiento

Después de los pasos anteriores, la herramienta muestra la red creada con información de la misma, así como también proporciona algunas opciones para visualizar resultados (figura 26):

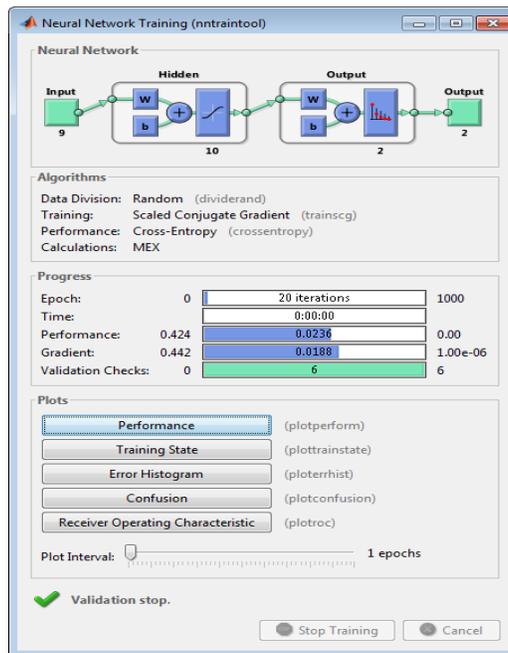


Figura 26: Red creada

Al dar clic en la opción *Confusion* de la pantalla anterior, se reflejan los resultados de la red como matrices de confusión (figura 27):



Figura 27: Matrices de confusión

Al igual nos permite visualizar los resultados de distintas maneras y en la siguiente figura se muestra una gráfica de los mismos (figura 28):

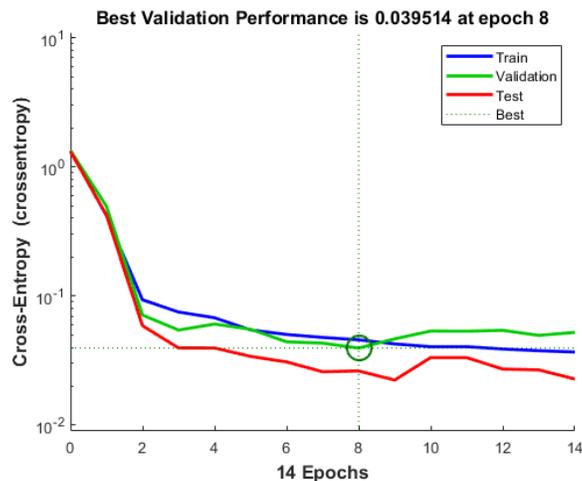


Figura 28: Gráfica de resultados

Una vez revisada la información, se puede afirmar que Matlab es una excelente opción para el análisis y simulación de las RNA para personas que tengan experiencia en programación y conocimientos teóricos de los modelos en caso de no querer trabajar y/o adquirir el *Neuronal Toolbox*. Brinda algunas bondades como la visualización de los resultados en 2D y 3D.

3.1.3 Ejercicio con NeuroSolutions

A continuación, se muestra un ejemplo del funcionamiento de NeuroSolutions en su versión 7.1.1.1 (*trial*) utilizando el algoritmo Perceptrón Multicapa y el CD Cancer descrito anteriormente.

En NeuroSolutions se puede crear rápidamente una RNA, si es primer contacto con el software y sabe muy poco acerca de las redes neuronales, se puede seleccionar el asistente para principiantes *NeuralExpert (Beginner)* al inicio del programa así como también se puede trabajar mediante Excel.

Al iniciar el asistente, lo primero que se debe especificar es el tipo de problema que tiene que resolver la red neuronal: Clasificación, Aproximación o *Clustering* (figura 29).

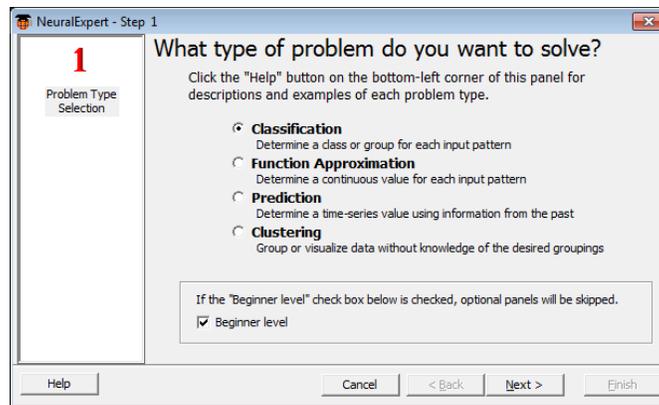


Figura 29: Elección del problema.

Posteriormente se debe especificar el archivo (extensión .csv) con los datos de entrenamiento (figura 30).

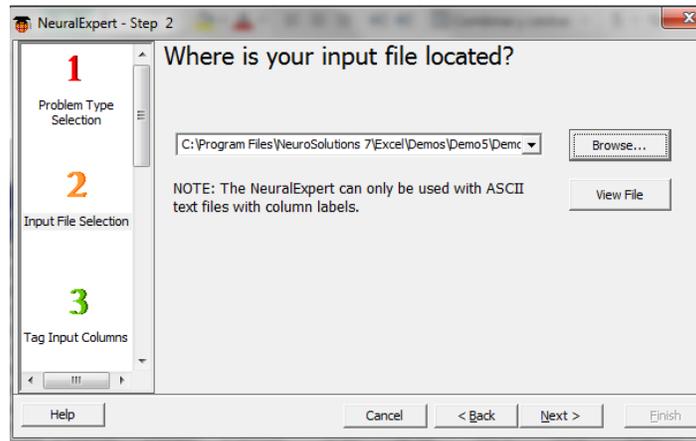
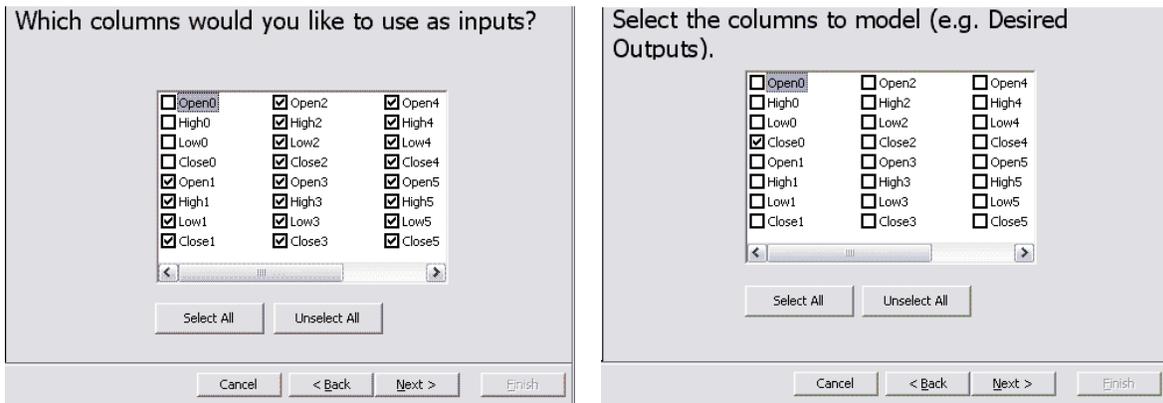


Figura 30: Selección de archivo.

Se deben seleccionar los campos que corresponden a los atributos del sector de características, distinguiendo aquellos que corresponden a las categorías del problema (Figura 31):



a) Selección de atributos

b) Selección de clases

Figura 31: Selección de Atributos y Clases.

Una vez identificados los campos de atributos y clases, el asistente propone crear la red con el grado de complejidad más adecuado al problema (Figura 32).

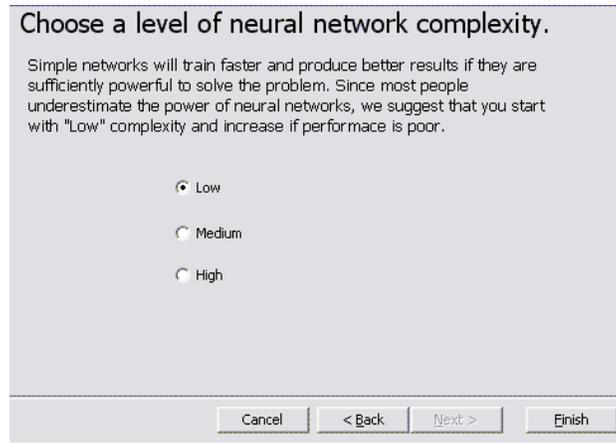


Figura 32: Elección del nivel de complejidad

Con esto, el asistente ha finalizado su trabajo creando un modelo de red neuronal e inicia la etapa de entrenamiento (Figura 33).

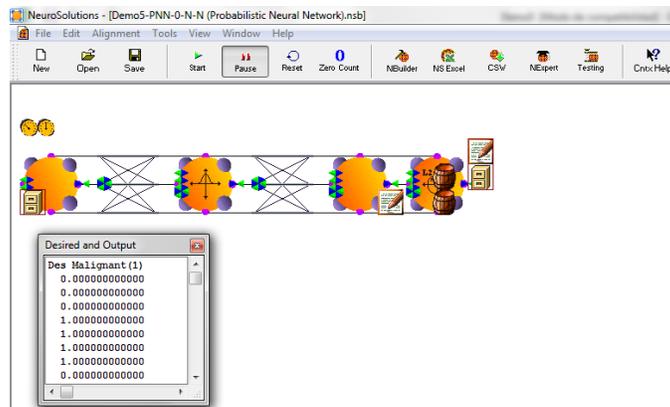


Figura 33: Red creada

Finalizado el entrenamiento, se muestran los resultados de la fase de entrenamiento así como también los resultados de la validación cruzada en la siguiente imagen (figura 34):

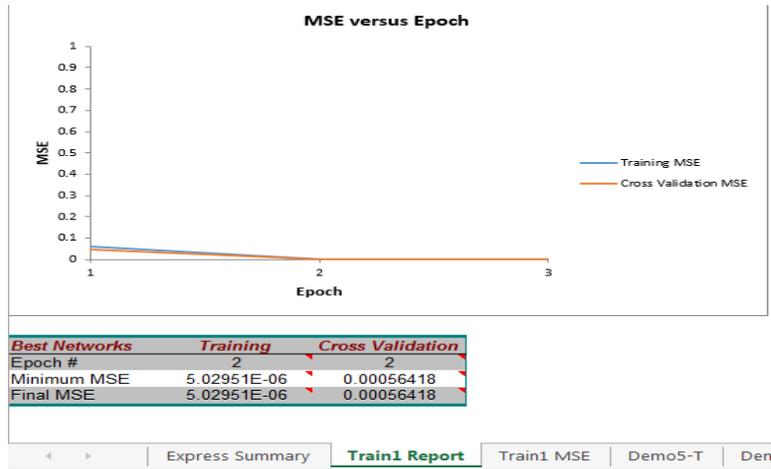


Figura 34: Resultados de entrenamiento

Por último se genera una hoja con los resultados obtenidos mostrando: las métricas de rendimiento al igual que las matrices de confusión (figura 35).

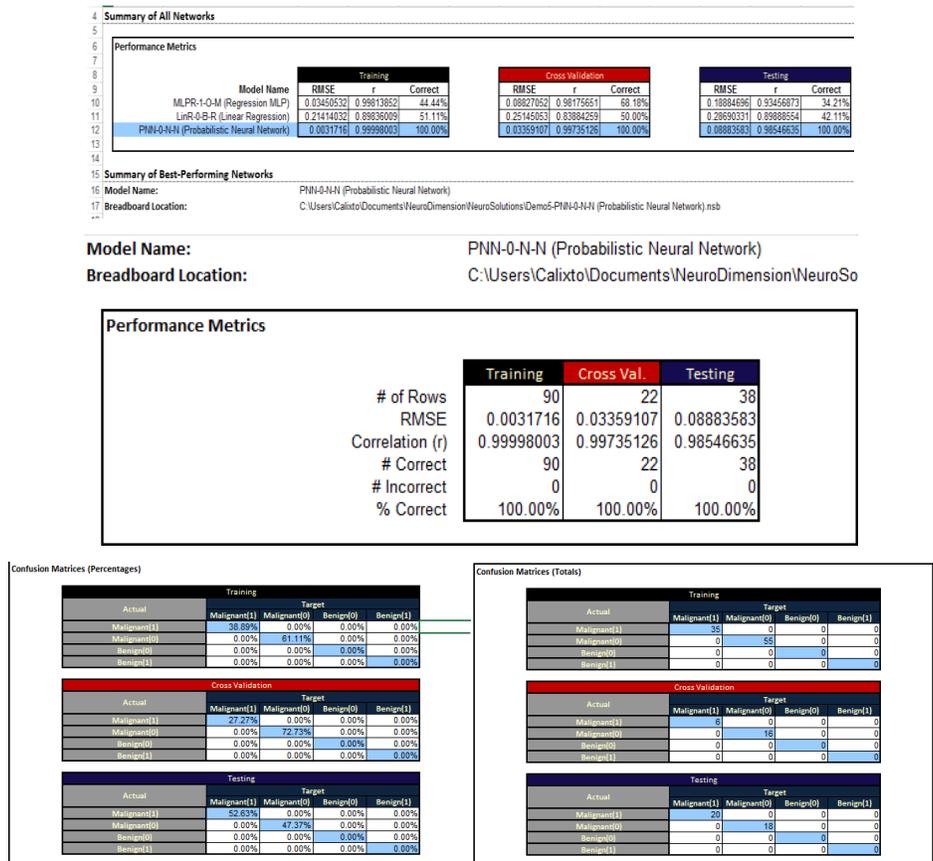


Figura 35: Resultados finales

Lo mostrado anteriormente deja de manifiesto que NeuroSolutions es una aplicación que facilita la realización de experimentos con RNA, utilizando una interfaz intuitiva y fácil de utilizar pues implementa muchos iconos que hacen que el usuario pueda interactuar de manera más simple.

Es importante mencionar que NeuroSolutions maneja conexiones con Matlab que permiten la interacción entre estas 2 aplicaciones y gracias a ello se puede importar o exportar redes creadas en cualquiera de las 2 aplicaciones.

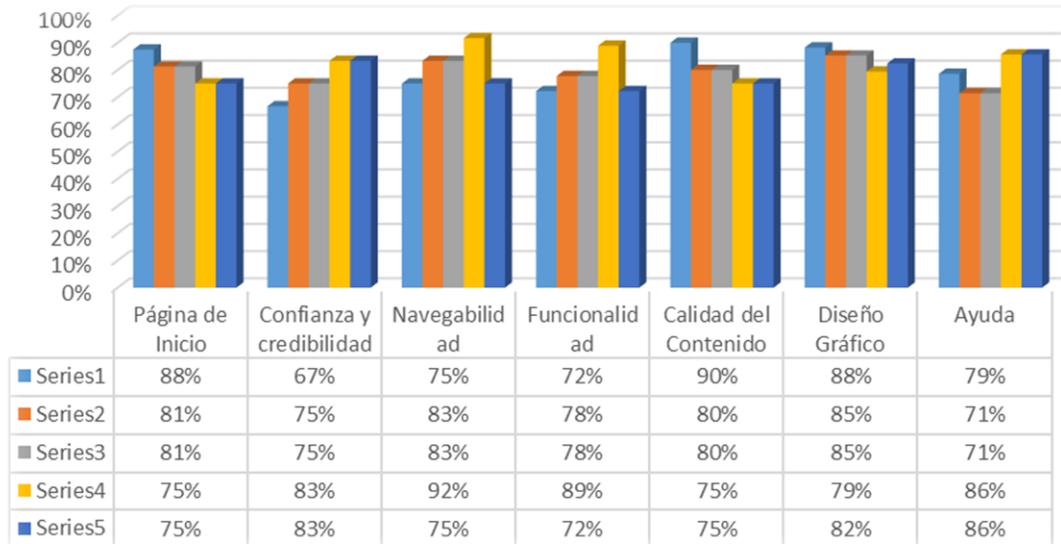
Aunque NeuroSolutions es una aplicación muy costosa, vale la pena para aquellas personas que están dedicadas en todo lo relacionado con RNA pues es la aplicación más completa que existe y no está limitada en cuanto a los modelos que se manejen.

Tabla 1: Despliegue de resultados

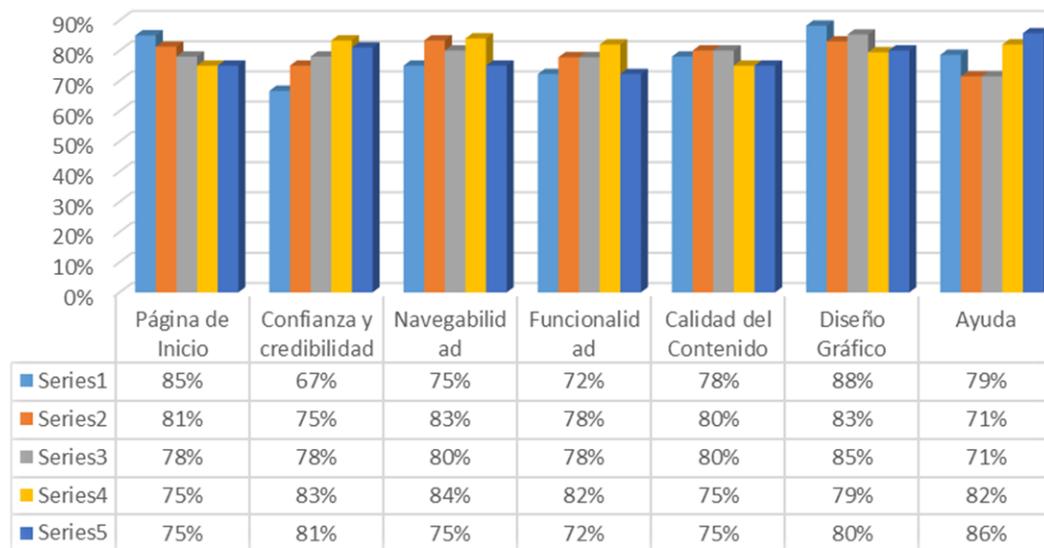
	Weka	Matlab	NeuroSolutions
Matrices de confusión	Si	Si	Si
Resultados	Sencillos	Completos	Completos

3.2 Prueba de usabilidad

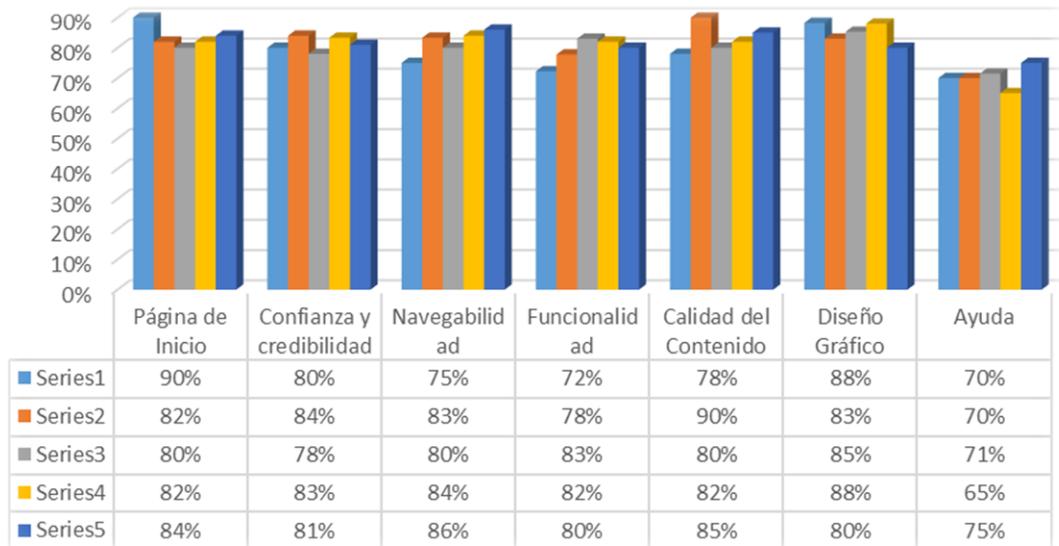
Para esta parte se utilizó la evaluación heurística, la cual sirve como un método de la evaluación de usabilidad llevado a cabo a través de una serie de preguntas que se realizaron a 5 usuarios (Series) con la finalidad de medir la calidad de la interfaz y algunos aspectos que se mencionan en las imágenes siguientes y así conocer más del que piensa el usuario acerca de cada herramienta y con ello; medir que tan fácil o difícil resulta la interacción Hombre – Máquina [29]. A continuación se muestran los resultados obtenidos de usabilidad de cada herramienta.



Grafica 1: Usabilidad de Weka.



Grafica 2: Usabilidad de Matlab.



Grafica 3: Usabilidad de NeuroSolutions

Como puede verse, los usuarios encuestados aquellos aspectos que tuvieron menor calificación son los relacionados a Confianza y credibilidad, que es lo relacionado contar con un contacto para asistencia técnica, principalmente. El resto de las heurísticas, puede observarse que tienen un rendimiento semejante.

Finalmente, al analizar el resultado global de los resultados de la prueba, es posible observar que, en términos generales las tres herramientas ofrecen un grado de usabilidad superior al 75%, predominando NeuroSolutions con un 88%.

Conclusiones, recomendaciones y trabajo futuro

En esta sección se tratan algunas conclusiones finales, así como también recomendaciones que se dan para el estudiante respecto al tema de RNA y se propone un análisis futuro de otras herramientas existentes para realizar experimentos de RNA.

Conclusiones

Esta Tesina fue realizada con el objetivo de hacer el análisis comparativo de tres herramientas: Weka, Matlab y NeuroSolutions que son de las más utilizadas para trabajar con RNA, aunque existen otras herramientas para uso de RNA como lo son: *TradingSolutions*, *Noesis*, *Neuronal Stock*, *Trader 68*, *RapidMiner*, *SSPS* y otras más que como trabajo futuro resultaría conveniente realizar su análisis. Aunque todas son de paga, sería de gran utilidad su análisis y con ello contar con más opciones para tratar RNA.

Se eligieron estas herramientas debido a las opciones implícitas para la construcción y procesamiento de información con RNA, cada una de ellas funciona de manera diferente y tiene diferentes entornos, opciones y salidas diferentes. La siguiente Tabla muestra un resumen de los principales aspectos analizados en la Tesina.

Tabla 2: Resumen de las 3 Aplicaciones

	Weka	Matlab	NeuroSolutions
Licencia	Libre	Pago (USD\$99.00)	Pago (USD \$295.00)
Complejidad de Uso	Fácil de interactuar con el usuario	Complejo (se necesitan conocimientos de programación)	Fácil de interactuar con el usuario
Interfaz	Gráfica	Gráfica	Gráfica
Documentación y Apoyo	Mucha	Mucha	Poca
Visualización de Resultados	Sencillos	Complejos	Completos

Como puede verse, para el caso de Weka, se destaca que es un software libre, pero sobre todo que es muy sencillo de utilizar porque maneja una interfaz gráfica que para el usuario es muy útil y práctico el manejo de esta herramienta sin importar sus niveles de experiencia en software. Otro aspecto importante es que, al ser implementado en Java, se puede ejecutar en casi cualquier plataforma que es una ventaja, así como también el poder permitir el acceso a bases de datos a través de SQL a través de la interfaz del explorador y que también se puede descargar código de distintas fuentes y con el poder hacer adaptaciones para lograr una mejor red. Por último, la visualización de los resultados es sencilla ya que gracias a la interfaz que maneja se pueden interpretar de manera fácil sin necesidad de ser experto en la materia o que requiera conocimientos de programación.

Para el caso de Matlab, se destaca que es una herramienta muy utilizada para realizar simulaciones de Redes Neuronales y también se puede visualizar los resultados de manera gráfica ya sea en 2 o 3 dimensiones, pero su interpretación resulta compleja ya que requiere un cierto nivel de programación para poder entenderlos de manera correcta. Aunque esta herramienta no es gratuita, es muy utilizada debido a que permite la codificación de manera rápida en un lenguaje de alto nivel siempre y cuando se cuente con un nivel poco avanzado de conocimientos en la parte de programación ya que es necesaria y su interfaz es un poco compleja, pero en Internet se pueden encontrar gran variedad de archivos escritos en Matlab.

Para el caso de NeuroSolutions, se destaca que pese a ser una herramienta de licencia un poco costosa, es muy utilizada debido a ser una de las más completas en cuanto a sus algoritmos y técnicas de RNA, a pesar de que esta herramienta no cuenta con mucha documentación de apoyo al usuario, la interacción con el sistema no es tan compleja y la visualización de sus resultados se pueden interpretar de manera sencilla y sobre todo que son completos respecto a las otras aplicaciones.

En resumen, Weka es sencillo de utilizar y es libre a diferencia de Matlab, ya que Matlab tiene un costo por su licencia de *USD* \$99.00 sin contar el costo de *Neural Network Toolbox*, pues este *toolbox* tiene un costo de *USD* \$250.00. Respecto a NeuroSolutions se habla de una aplicación más completa, pero a su vez es más

costosa la licencia ya que maneja 3 tipos de licencia; NeuroSolutions Pro: USD \$1,495.00, NeuroSolutions: USD \$495.00 y NeuroSolutions Student: USD \$295.00, por lo que podría ser una buena inversión que la Facultad de Ingeniería podría realizar y con estas licencias, el alumno tenga a la mano una herramienta más para estudiar sobre el tema.

Es importante la difusión de las diferentes herramientas de RNA para motivar a los estudiantes a incursionar en el conocimiento de estos algoritmos, pues hoy en día es muy utilizado y no se tiene duda de que en un futuro seguirá siendo un tema de interés e implementación en distintas tecnologías que se han desarrolladas.

Como se mencionó anteriormente, el estudio de las RNA es bastante amplio, lo aquí mostrado no es más que un panorama general del contexto de uso de las herramientas estudiadas. No obstante, un estudio complementario al aquí mostrado es la realización de un cuaderno de ejercicios, mismos que pueden ser utilizados como parte de las prácticas de laboratorio en materias como Redes Neuronales Artificiales y Minería de datos.

Recordar que antes de incursionarse en el tema de RNA, se recomienda que el estudiante aprenda un poco sobre el tema, principalmente su funcionamiento para poder entenderlas de manera correcta y al estudiar RNA se tenga mayor facilidad de aprendizaje e interpretación ya que este tema puede ser implementado en distintas áreas del conocimiento, como se vio anteriormente.

Como resultado de esta tesina y como experiencia personal, se recomienda NeuroSolutions ya que fue más sencillo trabajar con la herramienta puesto que al manejar una interfaz muy gráfica y sobretodo amigable, fue más fácil manipular la herramienta así como también fácil de interpretar los resultados.

Recomendaciones

Aquí se muestran algunos aspectos a tomar en cuenta por parte del estudiante y que considere respecto a RNA:

- No perder el interés en el tema, ya que actualmente puede ser implementado en distintas áreas del conocimiento, muchas de las cuales son de interés para el estudiante como lo es: los video juegos y la robótica, donde gracias a las RNA se puede lograr la creación de video juegos o el poder hacer que un dispositivo sea inteligente y tenga la facilidad de aprender por sí mismo.
- Si el estudiante tiene interés en RNA, acercarse a expertos que puedan apoyarles con asesorías, dentro de los cuales; la Facultad de Ingeniería cuenta con varios profesores que tienen conocimiento sobre el tema, así como también se cuenta con profesores que son expertos en algunas de las aplicaciones mencionadas.
- Consultar en internet documentación que facilite al estudiante el aprendizaje sobre RNA, su funcionamiento más a detalle y todo lo que sea de interés por parte del estudiante.
- Investigar acerca de las distintas aplicaciones que son utilizadas para RNA y según la que sea de mayor interés por parte del estudiante, utilizar la versión de prueba de cada aplicación y así tomar en cuenta la que se facilite más de acuerdo a los conocimientos de cada estudiante y si el interés es ser experto en una aplicación en específica, tomar en cuenta la parte del costo de la licencia.
- Consultar en internet ejercicios de apoyo que permitan al estudiante realizar prácticas e ir conociendo más sobre RNA y su funcionamiento.

Como opinión personal, tengo que reconocer el trabajo de la Facultad así como de los docentes que imparten este tema o alguno relacionado puesto que es un tema no muy fácil pero si se tiene interés se pueden lograr muchos desarrollos y que me llena de satisfacción el saber que existe una línea de acentuación relacionada con RNA, espero que los estudiantes aprovechen las instalaciones y exploten el conocimiento de los docentes.

Yo espero que con este trabajo pueda aportar algo a los compañeros estudiantes así como también a los docentes y que pueda servir como guía para conocer un poco acerca de las RNA y/o sea de utilidad para el compañero estudiante para guiarse un poco en cuanto a la manipulación de las herramientas mencionadas.

Trabajo futuro

Actualmente (2018) las RNA han ido tomando mayor importancia por investigadores, desarrolladores y gente interesada en el tema ya que son de gran utilidad en distintas áreas de la ciencia y la característica que tienen de predecir, asociar hechos, reconocimiento de patrones ha permitido implementar las RNA para cosas muy interesantes (por ejemplo: video juegos, medicina, etc.) y que se ha demostrado que funcionan de manera muy eficiente y que hoy en día se tienen muchas herramientas para trabajar RNA.

Para esta tesina, se eligieron estas 3 herramientas (Weka, Matlab, NeuroSolutions) tomando en cuenta que Weka era de licencia libre y en este caso; una herramienta totalmente desconocida, en cuanto a Matlab hubo interés ya que es una herramienta utilizada no solo para RNA sino para algunas otras cosas y me intereso ver como trabajaba con RNA. Por último se tiene a NeuroSolutions que al leer sobre esta herramienta me interesó la parte de saber que reflejaba sus RNA como un circuito eléctrico y que tiene buena reputación en cuanto a trabajar con RNA.

Para esto, se sugiere que más adelante pudiera existir un análisis de otras herramientas mencionadas al inicio o alguna otra para trabajar con RNA y así como estudiantes de la Facultad de Ingeniería dejar una aportación a nuestros compañeros y puedan tener más material de apoyo e incluso material que les pueda servir para la elección de alguna herramienta ya analizada según sea su interés o en otro caso haya material que pueda servir para hacer comparaciones entre las distintas herramientas para trabajar RNA y así debatir por cual es la mejor o en su defecto por aumentar nuestros conocimientos.

Por último, otras herramientas existentes que también pudieran ser viables de ser estudiadas son: PSPP, librerías de Python, *TradingSolutions*, *Noesis* , *Neuronal Stock* y *Trader 68*.

Referencias de Consulta

1. C. Blanco, L. Casas, L. Pérez, O. Caballero. Redes neuronales artificiales en la producción de tecnología educativa para la enseñanza de la diagonalización. *Revista Academia y Virtualidad*, 8, (1), 12-20, 2015.
2. A.Flores, L.Prieto, F.Ilavero, Redes neuronales como sistemas complejos adaptativos, Estudiantes del Máster Fundamentos Celulares y Moleculares de los Seres Vivos, Universidad de Málaga, Otoño 2011.
3. X. Basogain, *Redes Neuronales Artificiales y sus Aplicaciones*, Escuela Superior de Ingeniería de Bilbao, Bilbao, España, 2014.
4. Sitio oficial de Weka. Obtenido de: <http://www.cs.waikato.ac.nz/ml/Weka/>. Consultado el día 20 de Mayo de 2018.
5. Sitio oficial de RapidMiner. Obtenido de: <https://rapidminer.com/>. Consultado el día 4 de Marzo de 2015.
6. Sitio oficial de Matlab. Obtenido de: <http://www.mathworks.com/products/Matlab/>. Consultado el día 20 de Mayo de 2018.
7. Sitio oficial de NeuroSolutions. Obtenido de: <http://www.NeuroSolutions.com/>. Consultado el día 20 de Mayo de 2018.
8. W. Casasola. *La investigación documental para elaborar un ensayo académico*. *Revista de Lenguas Modernas*, Vol. (20), pp. 475-497, 2014.
9. Tipos de investigación: Exploratoria, Descriptiva, Explicativa, Correlacional. Obtenido de: <http://metodologadelainvestigacinsiis.blogspot.mx/2011/10/tipos-de-investigacion-exploratoria.html>. Consultado el día 20 de Mayo de 2018.
10. E. Maya. *Métodos y técnicas de investigación*. Una propuesta ágil para la presentación de trabajos científicos en las áreas de arquitectura, urbanismo y disciplinas afines, Edición electrónica, pp. 1-88, Universidad Nacional Autónoma de México, 2014.
11. Dzib Goodin, Alma, *La evolución del aprendizaje: más allá de las redes neuronales*. *Revista Chilena de Neuropsicología*, 2013.
12. F. Skapura, *Redes Neuronales - Algoritmos, aplicaciones y técnicas de programación*, Addison-Wesley, pp. 1-223.
13. C. Torres. Tesis de Maestría: *Modelo Computacional para la Estimación de Oxígeno Disuelto en Estanques de Producción Acuícola Empleando Redes Neuronales Artificiales*, Universidad Autónoma del Estado de México, Atizapán de Zaragoza, México, 2017.
14. J.Carrilo, G.Espinel. *Heterogeneous networks of neurons that recognize signatures neural*, Universidad Nacional de Colombia, 2017.
15. NiñoTC, Guevara SV, González FA, Jaque RA, Infante C. Uso de redes neuronales en predicción de morfología mandibular a través de variables

- craneomaxilares en una vista poteroanteior, Universidad de Odontología. 2016, 33(74).
16. J. Bojorquez, Tesis de Maestría: *Uso de redes neuronales para estimar la respuesta sísmica de sistemas estructurales*, Universidad Nacional Autónoma de México, México, 2011.
 17. M. Gómez, *CURSO: REDES NEURONALES ARTIFICIALES INAOE*, 24 de Octubre de 2014.
 18. L. Manjarrez, Tesis de Maestría: *Relaciones neuronales para determinar la atenuación del valor de la aceleración máxima en superficie de sitios en roca para zonas de subducción*, Universidad Nacional Autónoma de México, México, 2014.
 19. M.A. Rodríguez, Tesis de Ingeniería: *Desarrollo de un sistema clasificador neuronal para reconocimiento de orugas*, Universidad Nacional Autónoma de México, México, 2010.
 20. F. Mateo, Tesis doctoral: *Desarrollo de un sistema clasificador neuronal para reconocimiento de orugas*, Universidad Nacional Autónoma de México, México, 2010.
 21. L.M. Flores. Tesis de Maestría: *Redes neuronales y preprocesado de variables para modelos y sensores en bioingeniería*, Universidad Politécnica de Valencia, Valencia, 2012.
 22. R. Araque, P. Castro, M.J. Montero, B. Gutiérrez. *Intensidad exportadora e interacción entre fortalezas del marketing mix: un análisis basado en redes neuronales artificiales*. Revista de métodos cuantitativos para la economía y la empresa, Vol. (18), pp. 88-111, 2014.
 23. La Ciencia es Noticia. Obtenido de: <http://www.agenciasinc.es/Noticias/Un-ordenador-aprende-a-jugar-a-videojuegos-por-si-mismo-gracias-a-una-red-neuronal-artificial>. Consultado el día 04 de Octubre de 2017.
 24. Aplicación de Redes Neuronales Artificiales en Entornos Virtuales Inteligentes. Obtenido de: http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-07642014000500015. Consultado el día 06 de Octubre de 2017.
 25. Sitio de apoyo para Weka. Obtenido de: <http://www.locualo.net/programacion/introduccion-mineria-datos/Weka/00000018.aspx>. Consultado el día 6 de Maro de 2015.
 26. E. Frank, M. Hall, I. Witten, *The Weka Workbench*, Appendix Online, 2016.
 27. MathWorks. Obtenido de: <https://es.mathworks.com/products/neural-network/features.html>. Consultado el día 13 de Noviembre de 2017.
 28. Sitio oficial del Toolbox de Matlab. Obtenido de: <http://www.mathworks.com/products/neural-network/>. Consultado el día 20 de Mayo de 2018.
 29. M. González, Libro: *Evaluación Heurística*, Universidad de Lleida.

ANEXO

INSTRUMENTO DE EVALUACIÓN DE USABILIDAD

Página de Inicio

Preguntas	
1	Todos los elementos de la página de inicio están claramente enfocados en las tareas solicitadas por el usuario
2	La página de inicio contiene buenos ejemplos sobre el contenido en el resto del sitio web.
3	Las opciones de navegación están ordenadas de forma lógica, es decir orientadas a las tareas más importantes
4	Con solo un vistazo a la página de inicio el usuario que ingresa por primera vez puede entender por dónde comenzar.
5	La página de inicio tiene una dirección URL fácil de recordar.
6	La página de inicio está diseñada profesionalmente y va a crear una primera impresión positiva.
7	El diseño de la página de inicio va a animar a los usuarios a explorar más del sitio.
8	La página de inicio no se confunde con las ventanas posteriores.

Confianza y credibilidad

Preguntas	
1	El contenido está actualizado, autorizado y fidedigno
2	Es claro que hay una organización real detrás del sitio (ej. Existe una dirección u oficina física)
3	El sitio evita la palabrería de mercadeo
4	Es fácil de contactar a alguien para asistencia y una respuesta recibida rápidamente
5	El contenido es fresco: es actualizado con frecuencia y el sitio no tiene contenido reciente
6	Existen personas reales detrás de la organización y son honestos y autorizados

Navegabilidad

Preguntas	
1	Existe un manera obvia y conveniente para moverse entre las páginas relacionadas y secciones y es fácil de retornar a la página de inicio
2	El sistema de navegación es amplio y sencillo (muchos ítems en un menú) en vez de un menú profundo (con muchos niveles)
3	El menú o mapa de sitio aparece en todas las páginas de sistema
4	Existe un cambio visible cuando el ratón apunta a algo "clickeable" (excluyendo los cambios del curso)
5	En todas las página existe el botón de "Regresar"

6	Dar click en el botón de "Regresar" siempre lleva al usuario de vuelta a la página de donde vino
---	--

Funcionalidad

Preguntas	
1	El sitio está libre de información irrelevante, innecesaria y distractora
2	Se ha evitado el uso excesivo de scripts, applets, video, archivos de audio, gráficos e imágenes
3	El sitio evita que los usuarios se registren de manera innecesaria
4	El sitio requiere muy poco desplazamiento y uso de clicks
5	La ruta de cualquier tarea es de longitud razonable (de 2 a 4 clicks)
6	El usuario no necesita recordar información de un lugar a otro, por ejemplo para realizar una consulta
7	Los botones de acción, como "Enviar" siempre son invocados por el usuario y no de forma automática invocados por el sistema cuando el último campo del formulario ha sido llenado
8	Las características que no se deseen (ej. Video, audio) pueden ser detenidos u omitidos en todo momento
9	El sitio es robusto y todas las características clave funcionan bien (ej. No hay errores javascript, o links rotos)

Calidad del contenido

Preguntas	
1	El sitio tiene contenido atractivo y útil.
2	Las páginas son rápidas de examinar, con títulos grandes, subtítulos y párrafos cortos.
3	El sitio evita los títulos con lenguaje difícil de entender.
4	La información está organizada jerárquicamente de lo general a lo específico y esta organización es clara y lógica
5	El sitio está libre de errores tipográficos y de errores ortográficos.
6	Cada página está claramente etiquetada con un útil y descriptivo título alusivo al contenido
7	Los efectos aplicados en las animaciones son atractivos.
8	El texto es conciso y sin instrucciones innecesarias .
9	Los links de los botones coinciden con el título de las secciones destino, así los usuarios sabrán cuando han llegado a la sección a la que querían ir.
10	Los textos de los botones son lo suficientemente cortos para que el usuario pueda saber de que se trata.

Diseño Gráfico

Preguntas	
1	El sitio es agradable a la vista.
2	El sitio tiene un consistente y claro diseño visual que es capaz de enganchar a los usuarios.
3	El tipo y tamaño de letra es estándar en todas las páginas
4	El sitio puede ser usado sin desplazamiento horizontal.
5	Todo a lo que se le puede dar click funciona correctamente.
6	Los íconos y gráficos son estándar e intuitivos
7	El tipo de fuente utilizado es agradable para el usuario.
8	Cada página del sitio comparte un diseño consistente.
9	Existe una correcta combinación de colores adecuada a sus necesidades.
10	Los elementos estándar (como títulos de páginas y menú) son fáciles de localizar.
11	Los íconos son visualmente y conceptualmente distintos pero mantienen una armonía (pertenecen a la misma familia).
12	El uso de negritas se utiliza sólo para enfatizar tópicos importantes
13	Existe una correcta combinación de colores y evita fondos complicados o no contrastados
14	El logo de la organización está ubicado en el mismo lugar en todas las páginas y si se da click sobre él, se retorna a la página de la institución
15	El tamaño de las imágenes utilizadas es el adecuadas.
16	Las características que atraen la atención (animaciones, negritas o elementos con diferente tamaño) son usados con moderación
17	Llega en algún momento ser aburrido estar en la página

Ayuda

Preguntas	
1	Los avisos al usuario son breves y no ambiguos.
2	El usuario no necesita consultar manuales de usuario u otra información externa para usar el sitio.
3	El sitio hace obvio cuándo un error ocurrió (ej. cuando un formulario está incompleto).
4	El icono de ayuda es facil de localizar.
5	Las instrucciones que se le dan al usuario son fáciles de entender.
6	Es posible deshacer una acción o cuenta con opción de "cancelar"
7	La ayuda proporcionada es suficiente para corregir errores y/o navegar en el sistema



Toluca, México a 21 de Junio de 2018

Espacio Académico, Subdirector Académico, Coordinador de Programa
Presente

Carta de autorización para publicación en el RI

Declaración de autoría original, libre de embargo, susceptible de ser depositado en el Repositorio Institucional y solicitud de evaluación de grado.

Quien firma al calce, declaro bajo protesta de decir verdad que soy el(la) autor(a) de la obra (tipo de obra) **Tesina** titulada (título de la obra) **Estudio comparativo de herramientas para redes neuronales artificiales: Weka, Matlab y Neurosolutions**

y estoy de acuerdo con la totalidad de su contenido, manifiesto mi conformidad y mi autorización para que se publique en Repositorio Institucional de la Universidad Autónoma del Estado de México, con fines académicos y culturales en formato de acceso abierto, en los términos del Reglamento de Acceso Abierto, la Normatividad vigente emitida por la Institución y las legislaciones aplicables en la materia.

Así mismo, declaro que responderé de la autoría y originalidad de la obra de mérito y del ejercicio pacífico de los derechos que autorizo en este acto, manifiesto que no existe otra persona física o moral a la que pertenezca; por lo cual libero de toda responsabilidad a la Universidad Autónoma del Estado de México de cualquier demanda o reclamación que llegara a formular alguna persona física o moral que se considere con derecho sobre la obra, asumiendo todas las consecuencias legales y económicas a que hubiera lugar.

De igual forma permito que la Oficina de Conocimiento Abierto perteneciente a esta Máxima Casa de Estudios, realice lo propio para el almacenamiento, preservación y difusión de la obra, con fines académicos y culturales en formato de acceso abierto y sin fines de lucro.

Firmo de conformidad y bajo protesta de decir verdad

Nombre y firma Sergio Calixto Aldama

No de cuenta: 0820515

Conozco y acepto los términos de privacidad de la Universidad Autónoma del Estado de México
http://web.uaemex.mx/avisos/Aviso_Privacidad.pdf



Universidad Autónoma del Estado de México
UAEM

Toluca, México a 21 de Junio de 2018

Hoja de datos del autor

Nombre: Sergio Calixto Aldama

Número de cuenta: 0820515

Grado académico obtenido: Licenciatura de Ingeniería en Computación

Programa educativo de procedencia: Facultad de Ingeniería

Institución donde labora: SSEM

Domicilio: Prolongación Francisco I Madero #114

Teléfono/ Fax: 7222693238

Correo electrónico: sca.calixto@gmail.com

Sergio Calixto Aldama

Nombre y firma

Esta información es recabada con fines administrativos para el proceso de titulación del Espacio Académico que suscribe.

Conozco y acepto los términos de privacidad de la Universidad Autónoma del Estado de México
http://web.uaemex.mx/avisos/Aviso_Privacidad.pdf

12