

# HERRAMIENTAS APLICABLES A LA ENSEÑANZA DE REDES NEURONALES ARTIFICIALES

N.J. Medrano, B. Martín del Brío, J. Blasco

Dpto. de Ingeniería Eléctrica, Electrónica y Comunicaciones. Facultad de Ciencias, 50009 - Zaragoza. España. Telf.: (976) 761000 ext. 3358. Fax: (976) 761159

E-mail: niki@zape.unizar.es

**RESUMEN.-** Debido a sus especiales características computacionales, las redes neuronales artificiales se han convertido en los últimos años en una herramienta especialmente útil cuando se trata de abordar problemas complejos, difícilmente resolubles con las técnicas convencionales. Este hecho ha propiciado la aparición de gran número de programas de ordenador destinados a la simulación de diversos modelos de redes neuronales artificiales. En nuestro trabajo presentamos una visión general de las diversas alternativas existentes en este campo, así como diversas direcciones electrónicas donde acudir para ampliar información. Finalmente, exponemos nuestra propia experiencia desarrollada en el marco de los cursos de la Universidad de Verano de Teruel en su pasada edición de 1995.

## 1.- INTRODUCCION

Las redes neuronales artificiales (RNA) son sistemas adaptativos, diseñados en la forma de programas de ordenador o circuitos electrónicos, inspirados en la biología del sistema nervioso de los seres vivos. Una red neuronal artificial se compone de un elevado número de procesadores muy simples, llamados neuronas, altamente interconectados entre sí que, trabajando en paralelo, son capaces de procesar eficientemente la información que proviene del mundo exterior, caracterizada por su alto contenido en ruido, redundancia e imprecisión [1].

Durante los últimos años las RNA vienen resolviendo con éxito un buen número de aplicaciones prácticas de interés, como reconocimiento de patrones, estimación de series temporales o modelización de procesos no lineales [2], aprovechando algunas de sus características de procesamiento que las diferencia de técnicas más convencionales, como la no necesidad de establecer condiciones iniciales demasiado restrictivas (por ser estimadores de modelo libre), o su capacidad de aprendizaje a partir de ejemplos.

La metodología más natural para la realización de una red neuronal artificial es su implementación hardware, pues ello permite aprovechar al máximo sus propiedades de procesamiento paralelo [3]. Sin embargo, este tipo de implementación resulta costosa y en ocasiones excesiva, si la tarea que se pretende resolver no precisa de tiempos de respuesta críticos. En estos casos, el desarrollo de un programa software ejecutable sobre un computador convencional (como un PC) que simule el comportamiento de la red neuronal resulta suficiente para resolver el problema. Además, la simulación del modelo por programa constituye siempre el primer paso en el desarrollo de un sistema neuronal, incluso si el objetivo último es la implementación hardware.

## 2.- HERRAMIENTAS DE SIMULACIÓN DISPONIBLES

Dada la difusión que las redes neuronales están teniendo, existe un gran número de programas orientados a su simulación. Básicamente, podemos agruparlos en tres tipos: programas comerciales, desarrollados por empresas, programas de libre distribución o gratuitos, realizados por grupos de investigadores de Universidades o particulares, y programas de desarrollo propio, elaborados expresamente para realizar una tarea concreta.

A la hora de decidirse por un determinado programa, sea comercial o gratuito, hay que buscar un producto que se adapte a nuestras necesidades o preferencias. Debemos averiguar las plataformas y sistemas operativos para los que se encuentra disponible, los modelos de RNA y reglas de aprendizaje que contempla, el enfoque del programa (introductorio y genérico o indicado para desarrollo de aplicaciones), la forma de introducir las RNA (interface gráfico, lenguaje de descripción), si permite la generación de ejecutables o código fuente con los que reproducir más tarde la fase de recuerdo o de entrenamiento de la RNA, o si admite la inclusión de rutinas externas de usuario.

En general, los programas de simulación resultan un excelente banco de pruebas al permitirnos explorar de forma bastante rápida y gráfica las distintas arquitecturas de RNA hasta encontrar la que mejor se adapta a nuestro problema.

### 2.1.- Programas comerciales

Las ventajas de los programas comerciales residen principalmente en el elevado nivel de prestaciones que ofrecen, así como la disponibilidad de productos para aplicaciones especializadas (reconocimiento de imágenes, predicción de series temporales).

El principal inconveniente es su elevado precio, aunque siempre se ofrecen descuentos para fines educativos. De entre todos ellos, podemos destacar como más representativos:

- NEURALWORKS PROFESSIONAL II PLUS (NeuralWare, USA). Es uno de los programas con más antigüedad y completos del mercado. Posee más de treinta modelos de RNA con diferentes reglas de aprendizaje (BP, ART, SOM, LVQ, RBF, ...), y diferentes visualizaciones gráficas. Permite incorporar funciones externas en C. Existe para multitud de plataformas y sistemas operativos (PC, Sun, IBM RS6000, Macintosh, SGI, DEC, HP). Dispone de una versión de demostración. Más información: [sales@nware.com](mailto:sales@nware.com)

- MATLAB NEURAL NETWORK TOOLBOX (The Mathworks, USA). Al igual que otras muchas herramientas de Matlab (lógica borrosa, procesamiento de señal e imágenes, identificación y control de sistemas, etc.) está teniendo una buena acogida y se está extendiendo en el marco de la enseñanza e investigación. Contempla gran variedad de RNA (BP, RBF, asociativas, SOM, LVQ). Más información: <http://www.mathworks.com>.

- NN / XNN (Neureka, Noruega). Proporciona un lenguaje de especificación genérico que permite describir cualquier tipo de RNA, unidireccionales o recurrentes. Los modelos más comunes ya están incorporados como ejemplos (Madalina, ART1, BP, Counterpropagation, Elman, GRNN, Hopfield, Jordan, LVQ, Perceptrón, RBF, SOFM, etc). Se trata de un producto de altas prestaciones enfocado al desarrollo de aplicaciones. Permite añadir rutinas en C. Disponible para UNIX (OSF1, Solaris, AIX, IRIX) y MSDOS. Más información en <http://www.ii.uib.no/~arnemo/neureka/neureka.html>.

- BRAINMAKER (California Scientific Software, USA). Aunque sólo contempla BP, es uno de los pocos programas comerciales preparados para manejar tarjetas aceleradoras de RNA conectadas al bus de un ordenador. Haciendo uso de estas tarjetas se consiguen velocidades de procesamiento 1000 veces mayores que con un PC. Dentro del algoritmo BP admite ciertas variantes, como recurrencias, y ciertas técnicas, como validación cruzada, análisis de

sensibilidad, optimización por algoritmos genéticos (opcional), podado de pesos, etc. Disponible para DOS, Windows y Macintosh. Más información en <http://www.calsci.com>.

- NEUROHELL2 / NEUROWINDOWS (Ward Systems Group, USA). Lleva algunos años en el mercado, lo cual ya es motivo de interés. Ofrece pocos modelos de RNA (BP, SOM, PNN y GRNN) y sólo está disponible para PC-Windows. Está formado por dos productos independientes: el primero es un entorno interactivo, mientras que el segundo se presenta en forma de librería DLL que puede ser llamada desde distintos programas de Windows (Visual Basic, Excel...).

Además de los anteriores, existen muchos más paquetes comerciales de interés, de los que una muestra se presenta en la Tabla I. Como práctica general, la mayoría de las empresas ofrecen versiones reducidas para iniciación.

| PRODUCTO/EMPRESA  | APLICACIONES  | DIRECCIÓN ELECTRÓNICA   |
|---|---|---|
| Neuroforecaster<br>Accel Infotech                                     | Series temporales.<br>Clasificación                 | accel@technet.sg  |
| Nestor Learning System<br>Nestor                                      | Reconocimiento de<br>imágenes y caracteres          |   |
| Neuralyst<br>Cheshire Engineering Corporation                         | Macros Excel  | dhighley@zarks.sgcl.lib.mo.us   |
| NeuFuz4<br>National Semiconductor Corp.                               | Fuzzy associative memories                          | <a href="http://www.commerce.net/directories/participants/ns/home.html">http://www.commerce.net/directories/participants/ns/home.html</a>           |
| Partek  | Modelización con ajuste<br>automático de parámetros | info@partek.com   |
| WAND<br>Novel Technical Solutions y el Imperial<br>College de Londres | Reconocimiento imágenes                             | <a href="http://www.neuronet.ph.kcl.ac.uk/neuronet/software/nts/neural.html">http://www.neuronet.ph.kcl.ac.uk/neuronet/software/nts/neural.html</a> |
| NeuroLab<br>Mikumi Berkeley R & D Corp.                               | Construcción con bloques.<br>Didáctico              | <a href="http://www.mikumi.com">http://www.mikumi.com</a>   |
| havBpNet++ havFmNet++<br>hav.Software                                 | Librerías de clases C++ para<br>BP y SOM            | <a href="http://www.neosoft.com/hav">http://www.neosoft.com/hav</a>   |

**Tabla I.- Software comercial de emulación de redes neuronales**

## 2.2.- Programas de libre distribución

La principal ventaja de estos programas es que están accesibles a todo el mundo de forma gratuita a través de Internet, proporcionando en la mayoría de los casos material de valor incalculable, como los códigos fuentes, manuales muy completos, ejemplos, e incluso atención por correo electrónico para las dudas que puedan surgir sobre su instalación o manejo.

A diferencia de los productos comerciales, los programas gratuitos se pueden permitir el lujo de tratar temas más minoritarios, como aspectos biológicos (GENESIS, <http://www.bbb.caltech.edu/GENESIS>, BIOSIM, <ftp://ftp.uni-kl.de/pub/bio/neurobio/>), o con fines educativos (DARTNET, <ftp://dartvax.dartmouth.edu/pub/mac/dartnet.sit.hqx>). También es más común encontrar programas dedicados a un solo algoritmo (Cascade Correlation Simulator, <ftp://ftp.cs.cmu.edu/afs/cs/project/connect/code/supported/cascor-v1.2.shar>, Neocognitron Simulator, <ftp://unix.hensa.ac.uk/pub/neocognitron.tar.Z>, LVQ\_PAK y SOM\_PAK, [ftp://cochlea.hut.fi/pub/lvq\\_pak](ftp://cochlea.hut.fi/pub/lvq_pak) y [/pub/som\\_pak](ftp://cochlea.hut.fi/pub/som_pak)). Otros están optimizados en velocidad y prometen mejoras de varios órdenes de magnitud (Adaptive Logic Network kit, <ftp://menaik.cs.ualberta.ca/pub/atree>).

Dentro del campo de los programas de libre distribución, podemos destacar:

- SNNS (Stuttgart Neural Network Simulator). Muy completo, está teniendo una gran aceptación y repercusión, llegando al punto de crearse grupos de discusión. Contempla muchos modelos: BP (con infinidad de variantes), counterpropagation, quickprop, backpercolation 1, RBF, Rprop, ART1, ART2, ARTMAP, Cascade Correlation, Recurrent

Cascade Correlation, Dynamic LVQ, Backpropagation Through Time, Hopfield, Jordan/Elman, autoasociativas, SOM, TDNN, y un largo etcétera. Permite la ampliación por el usuario, (funciones de activación, algoritmos de aprendizaje), genera código C y cuenta con la posibilidad de repartir una simulación en un cluster de estaciones de trabajo. Disponible para SunOS, Solaris, IRIX, Ultrix, OSF, AIX, HP/UX, NextStep y Linux, sus manuales son muy extensos y claros. Se encuentra disponible a través de *world wide web* en <http://vasarely.informatik.uni-stuttgart.de/snns/snns.html>.

- ASPIRINE/MIGRAINES. Lenguaje de descripción de RNA, que es traducido posteriormente a C. Disponible para muchas plataformas. Tiene filtros para representar resultados en diversos programas como gnuplot o Matlab. Disponible a través de ftp en <ftp://pt.cs.cmu.edu/afs/cs/project/connect/code/am6.tar.Z>.

- SESAME. Librería de clases de C++. Incluye BP, SOM, RBF, ventanas temporales, codificación en conjuntos borrosos y otros algoritmos. Permite llevar a cabo experimentos híbridos, dispone de interface gráfico (X-Window de Sun4), y un extenso manual de referencia. Disponible en <ftp://ftp.gmd.de/gmd/as/sesame> y [/gmd/as/paper](ftp://gmd.de/gmd/as/paper).

- PDP++. Desarrollado por el grupo de McClelland y Rumelhart en C++. Para novatos y expertos. Interface gráfico en X-Window (UNIX). Algoritmos: BP hacia delante y recurrente, Boltzmann, Hopfield, Mean-Field, estocásticas. Amplio manual. Disponible en <http://www.cs.cmu.edu/Web/Groups/CNBC/PDP++/PDP++.html>.

- ROCHESTER CONNECTIONIST SIMULATOR. Bastante versátil, permite tipos arbitrarios de RNA (incluye BP). Disponible con interface X11 para Sunview. [ftp://cs.rochester.edu/pub/packages/simulator/rcs\\_v4.2.tar.Z](ftp://cs.rochester.edu/pub/packages/simulator/rcs_v4.2.tar.Z).

### 3.- LA ENSEÑANZA DE REDES NEURONALES ARTIFICIALES

El éxito con el que las redes neuronales artificiales resuelven tareas tecnológicas de interés, se ve reflejado en la atención mostrada hacia el tema por un cada vez más amplio sector de la comunidad científica e industrial. Debido a ello, desde hace unos años se imparten ya en España cursos de redes neuronales (y de otras materias del alguna manera emparentadas, como los sistemas borrosos), en los que se realiza una introducción al desarrollo de sistemas neuronales, y su aplicación práctica. En algunos casos estas enseñanzas adoptan ya la forma de asignaturas de carreras tecnológicas, aunque suelen ser más habituales en cursos de especialización, como puedan ser los Cursos de Doctorado o los Cursos de Verano.

El método más empleado en ellos para presentar a los alumnos el modo como operan las redes neuronales es su simulación, ya que permite variar las propiedades y arquitectura de los modelos en estudio de forma muy sencilla, mostrando así cómo influyen en el comportamiento de la red. En los siguientes apartados comentaremos los programas de simulación empleados en el Curso de Redes Neuronales realizado el pasado año en la Universidad de Verano de Teruel.

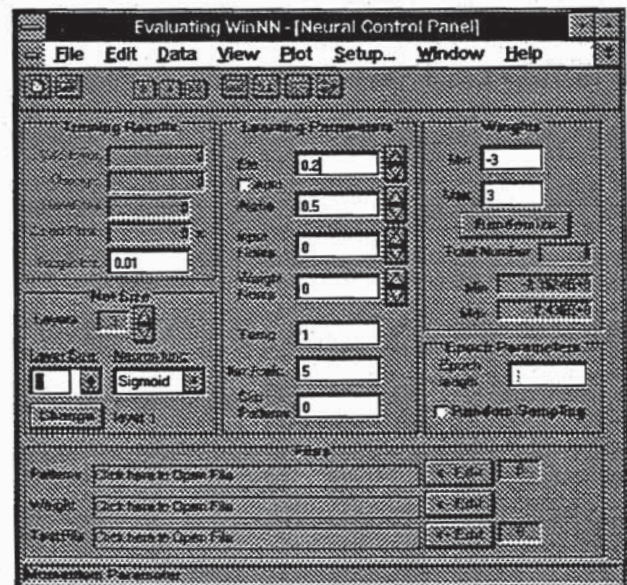


Figura 1.- Programa comercial WinNN.

### 3.1.- Software de tipo comercial: WINNN

Para el estudio de cómo opera uno de los modelos de red neuronal supervisada más extendidos, el BP, empleamos un programa comercial llamado WinNN, actualmente *shareware* (de bajo costo). El ejemplo seleccionado (datos de la crisis bancaria española de 1977-1985)[4] es introducido a la red para su entrenamiento, comprobando mediante la modificación de los parámetros principales, topología de la red, etc., cómo afectan estas variaciones a la eficacia de la operación de la red. Este tipo de programas (Figura 1), al pretender ser genéricos, proporcionan el acceso a un gran número de parámetros de definición del modelo neuronal, lo que puede acabar convirtiéndolo en algo complejo de manejar. Para evitar este problema, se suministra al alumno un detallado manual de operación del programa, así como de aquellas opciones de más interés. Cuando hacemos uso de programas comerciales debemos tener presente que, al tratarse de productos "cerrados", acabados, en ocasiones se muestran ineficaces para resolver la tarea concreta de quien lo utiliza en una aplicación real.

### 3.2.- Software de libre disponibilidad: KOHOMAP

Dada la elevada dificultad conceptual que presentan los mapas de Kohonen, el modelo neuronal no supervisado más popular, hacemos uso de dos programas distintos para facilitar su comprensión. El primero de ellos, KOHOMAP (Figura 2), software de libre disponibilidad procedente de [5], es un programa muy simple, orientado a mostrar la propiedad básica de los mapas autoorganizados de distribuir sus procesadores a lo largo de todo el espacio objeto del problema. Este programa también permite ilustrar los diferentes resultados que se obtienen al emplear distintas topologías de red.

Al contrario que en los programas comerciales, los de dominio público ofrecen una limitada versatilidad y, en general, una salida gráfica menos esmerada. Sin embargo, con esta clase de software se suele proporcionar en la mayoría de las ocasiones el código fuente, por lo que resultan ser una base inicial idónea para experimentar libremente con el modelo.

### 3.3.- Software de desarrollo propio: programa TERUEL

Finalmente, para estudiar con mayor profundidad el modo en el que operan los mapas autoorganizados, el alumno hace uso de un programa desarrollado en nuestro Departamento, denominado TERUEL. Debido a que un programa de este tipo está diseñado para atender a una aplicación concreta (en este caso, la enseñanza del modelo neuronal), esta clase de programas se ajusta perfectamente a necesidades específicas. Este programa (Figura 3) consta de las opciones que permiten modificar los parámetros más importantes del modelo (ritmo de aprendizaje, radio de vecindad, etc.), así como mostrar en pantalla de forma gráfica las principales características del mapa desarrollado, desde las disposiciones de los patrones en la malla

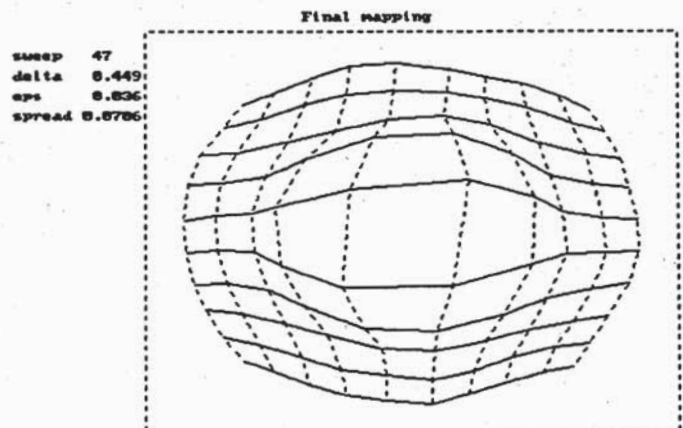


Figura 2.- Programa de libre distribución KOHOMAP.

de procesadores elementales que componen el mapa, a lo largo del entrenamiento y una vez concluido, hasta los rangos de los valores que adoptan las componentes de los vectores de pesos de los procesadores (mostrados en forma de escalas de color). La salida gráfica de esta aplicación ha sido realizada haciendo uso del entorno de desarrollo para Windows Visual C++ de Microsoft, mientras que los códigos de aprendizaje, recuerdo y escalados se han programado en C estándar, empleando Borland C, de forma que se facilita su exportación a otros entornos y plataformas.

Durante el proceso de elaboración del programa se ha tenido presente de manera especial evitar todos aquellos defectos que se han apreciado en el resto de programas de simulación que hemos estado utilizando: excesiva complejidad a la hora de modificar parámetros de la red, numerosas ventanas para mostrar valores de parámetros, errores y otras informaciones que, superponiéndose unas sobre otras dificultan la comprensión de la operación del programa o, por el contrario, una rigidez excesiva en el programa que impidiese su aplicación a problemas reales, para un análisis inicial.

#### 4. CONCLUSIONES

El auge que las redes neuronales artificiales están teniendo ha originado la aparición de numerosos programas dedicados a su simulación, así como la organización de cursos orientados a la enseñanza de estas técnicas de procesamiento de información. En nuestro trabajo hemos presentado un breve panorama de las diferentes posibilidades disponibles en el terreno de los programas para su simulación, así como nuestra experiencia en la aplicación de éstos a la enseñanza de las redes neuronales artificiales, realizada en el marco de los Cursos de Verano de Teruel, en la edición de 1995.

#### 5.- REFERENCIAS

- [1] Lippmann R.P. "An introduction to computing with neural nets". *IEEE ASSP Magazine*, Abril, 4-22, 1987.
- [2] Croall, I. F., Mason, J. P (eds.). "Industrial Applications of Neural Networks". Ed. Springer-Verlag, 1992.
- [3] Kung, S. Y. "Digital Neural Networks". Ed. Prentice Hall, 1993.
- [4] Martín del Brío, B. "Procesamiento Neuronal con Mapas Autoorganizados: Arquitecturas Digitales". Tesis Doctoral. Universidad de Zaragoza, 1994.
- [5] Müller, B., Reinhardt, J. "Neural Networks. An Introduction". Ed. Springer-Verlag, 1990.

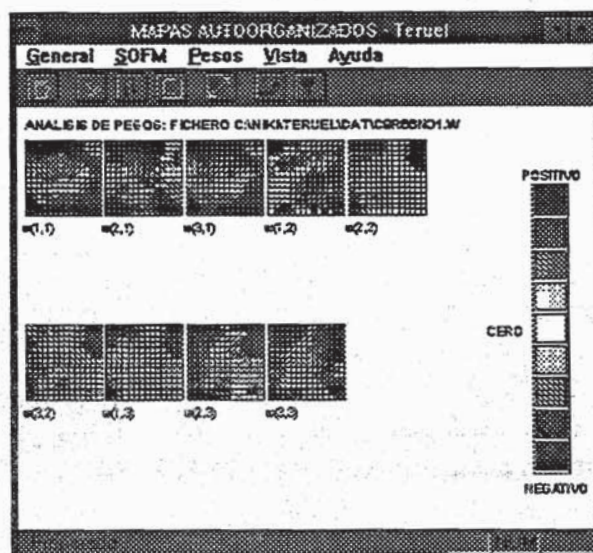


Figura 3.- Programa de desarrollo propio TERUEL.