

# Redes de Retropropagación

*Antonio Rivas & Javier Pereira*

Laboratorio RNASA

Departamento de Computación

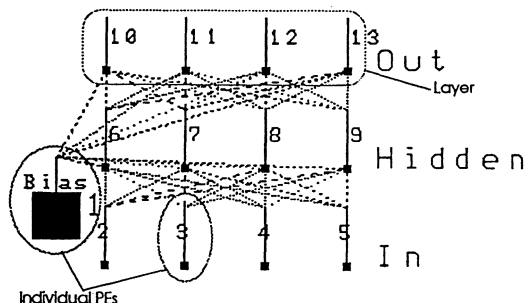
Universidade da Coruña

email: antonio@udc.es

## 1.- Introducción

Como recordatorio acerca de la estructura básica de una red de neuronas artificiales (figura 1) diremos que sus principales componentes son las neuronas, nodos o elementos de proceso (EP). Los EP's están organizados en capas y los EP's de una capa están conectados con EP's de otras capas, de su misma capa o incluso consigo mismos.

Cada EP procesa la información que le llega a través de sus conexiones de entrada y envía el resultado a otros EP's mediante sus conexiones de salida. Al igual que en los sistemas biológicos, las fortalezas de estas conexiones pueden cambiar (y de hecho lo hacen en función de las fortalezas de las conexiones de entrada y del tipo de función de transferencia usada por los EP's). Cuando creamos una red hemos de tener en cuenta la forma en la que sus EP's estarán conectados, cómo procesarán la información y cómo serán modificadas las fortalezas de las conexiones.



*Figura 1.- Estructura básica de una RNA.*

Atendiendo a la dirección del flujo de datos que circula por la red, podemos clasificar a estas en:

- Redes alimentadas hacia adelante: La dirección del flujo de información es únicamente hacia adelante.
- Redes retroalimentadas: La dirección del flujo de datos es tanto hacia adelante como lateralmente y, o, hacia atrás.
- Según el tipo de aprendizaje podemos hacer la siguiente clasificación:
  - Aprendizaje supervisado: Durante el periodo de aprendizaje se le suministra a la red la respuesta deseada además de las entradas.
  - Aprendizaje no supervisado: No se suministra la respuesta deseada.

En este capítulo, se aborda la construcción de Redes de Neuronas del tipo “BackPropagation” utilizando una herramienta comercial llamada “N.W. Explorer”. Esta herramienta presenta un menú con estructura de ventanas desplegable en cualquier punto del área inactiva de la pantalla inicial. Una ventana puede estar formada por diferentes componentes como muestra la figura 2:

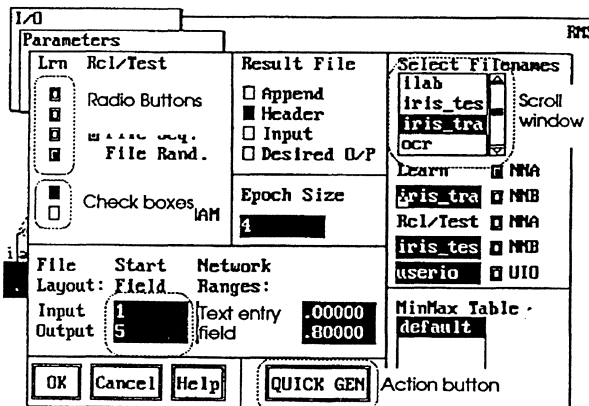


Figura 2.- Ventana de Componentes de la RNA.

Podemos identificar los siguientes grupos de elementos:

- ◆ “Radio buttons”: Se trata de un conjunto de opciones mutuamente exclusivas. Seleccionar uno implica desactivar las otras del conjunto.

- ◆ “Check boxes”: Una o más opciones pueden estar seleccionadas simultáneamente. Hacer “click” para seleccionar una opción y repetir la operación para reactivarla.

- ◆ “Action buttons”: Al hacer “click” en uno de ellos tiene lugar la acción que en el se indica (por ejemplo grabar un fichero o abandonar la ventana).

- ◆ “Text Fields”: En ellos se pueden incluir valores numéricos o texto. Para activarlos hacer “click” en el campo deseado.

- ◆ “Scroll Windows”: Permiten hacer una selección en una lista de parámetros tales como ficheros o reglas de aprendizaje.

Asimismo, NW Explorer permite el acceso a diversos objetos que componen una red de neuronas mediante las opciones presentadas en la parte izquierda de la pantalla inicial.

Estas son:



“Apuntador” : Permite regresar al menú. También se usa para mover objetos y modificar su tamaño.



“Layer”: Permite añadir, borrar, copiar y editar capas. También se pueden ocultar y presentar en pantalla las conexiones de una capa y añadir capas como elementos de prueba.



”PE”: Permite crear, editar, borrar o copiar elementos de proceso (EP’s). Asimismo se pueden ocultar o presentar en pantalla conexiones de EP’s o designarlos como componentes de pruebas.



”Graph”: Permite crear, modificar, copiar, controlar o borrar instrumentos de medida utilizados para controlar la red.



“Conexiones de capa”: Permite añadir, copiar, borrar, controlar o modificar conexiones entre capas. También podemos ocultar o presentar en pantalla conexiones, o añadirlas como elementos de prueba.



“Conexiones de EP’s”: Se permite el mismo tipo de utilidades que en los casos anteriores pero con conexiones entre elementos de proceso.



“Fan-in”: Es el conjunto de conexiones que van desde una capa a un EP. Las utilidades son similares a las anteriores. Convergencia.



“Fan-out”: Es el conjunto de conexiones que van desde un EP hacia una capa. El conjunto de utilidades es parecido a los casos anteriores. Divergencia.

Para hacer uso de estas opciones, es preciso haber construido una red o cuando menos una capa. Al hacer “click” en uno de los iconos se despliega otro menú parecido al que acabamos de ver pero referido al icono seleccionado. Este menú permanecerá desplegado mientras continuemos pulsando el botón del ratón. Para escoger una de sus opciones nos desplazaremos hasta ella con el botón del ratón pulsado y una vez allí soltaremos dicho botón.

Al hacer esto, el icono del objeto elegido aparece resaltado en la pantalla y el símbolo de la utilidad elegida se sitúa en la esquina superior izquierda de la pantalla. Además el cursor del ratón cambia de forma. Esta puede ser:

- Una flecha rodeada por un círculo que se usará para marcar puntos de origen, por ejemplo en una conexión.
- Una tarjeta que se usa para marcar puntos destino.
- Una “X” que se utiliza para marcar posiciones.

Estas utilidades permanecen activas hasta que se abandonan (haciendo “click” en el icono apuntador o seleccionando otra utilidad).

Para acceder a la ayuda basta con hacer “click” en la utilidad que se está usando, situada en la esquina izquierda de la pantalla.

## **2.- Construcción de una Red de Neuronas Backpropagation**

A continuación construiremos una red de neuronas Back-propagation que sirva para clasificar flores. Para empezar, examinaremos la red y sus partes y luego la entrenaremos, testaremos y ajustaremos algunos parámetros que influyen en el aprendizaje

### **2.1.- Construcción de la red**

Usaremos el comando Back-Prop Builder para crear la red de neuronas Back-Propagation. Esta red clasificará especies de flores en tres categorías: Setosa, Versicolor o Virginica. Los criterios que usa la red para clasificar las especies son la longitud y anchura de sépalos y pétalos.

El comando Back-Prop Builder proporciona una ventana que se puede usar para crear redes Back-Propagation simplemente seleccionando parámetros. Para crear la red Back-Propagation:

1.- Desplegar la ventana del menú principal en cualquier sitio del área inactiva de la pantalla.

2.- Seleccionar **NETWORK**.

3.- Seleccionar el comando BackProp Builder. Aparecerá una ventana con un conjunto completo de utilidades que nos permiten crear la red.

4.- Dar los siguientes valores para el número de EP’s en cada capa:

4 Input

4 Hid1

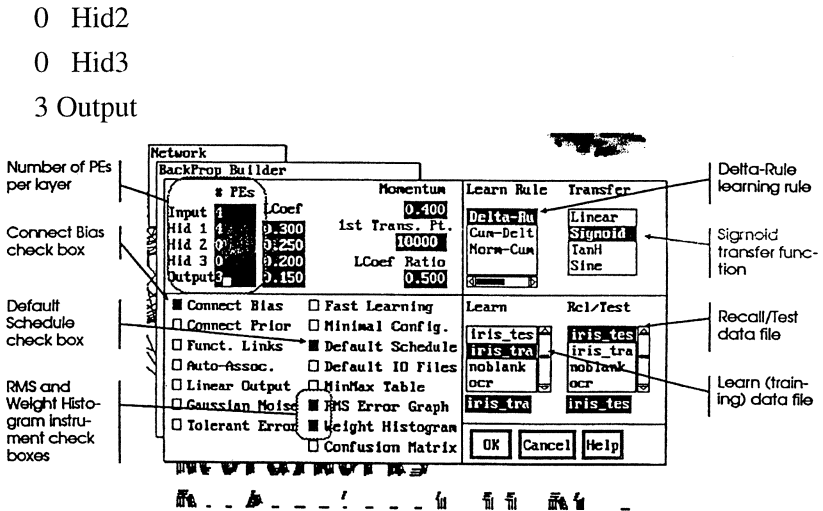


Figura 3.- Ventana de diálogo de NWExplorer construyendo una BP.

La mayoría de las redes Back-Propagation tienen una o dos capas ocultas y su número de EP's varía según sea el de la capa de entrada y el de la de salida, y la complejidad del problema a resolver.

##### 5.- Seleccionar las siguientes opciones:

⇒ **CONNECT BIAS**: Conecta el elemento de proceso Bias a cada capa oculta y a la de salida. El elemento de proceso Bias trabaja como una conexión a tierra que no varía en un circuito eléctrico y puede ser usado como un valor de referencia constante.

⇒ **DEFAULT SCHEDULE**: Hay varios parámetros que controlan las funciones sumatorio y transferencia, así como coeficientes de aprendizaje. El control sobre estos parámetros, así como su forma de modificarse durante la fase de entrenamiento, se lleva a cabo mediante los llamados esquemas de aprendizaje y memoria (learning and recall schedules).

Al activar esta opción tomamos el organizador por defecto para todas las capas. Normalmente esta opción no se elegirá y se tendrá un esquema diferente para cada capa.

⇒ **RMS ERROR GRAPH:** Es el error RMS para la capa de salida. A medida que avanza el proceso de aprendizaje veremos como el error se va aproximando a cero.

⇒ **WEIGHT HISTOGRAM:** Es un histograma normalizado que muestra como cambian los pesos en la capa de salida.

6.- Seleccionar la regla de aprendizaje **Delta-Ru** (delta rule).

7.- Seleccionar la función de transferencia **Sigmoid**.

8.- Especificar que la fuente de aprendizaje es el fichero **iris-tra**.

9.- Especificar que la fuente para el test es el fichero **iris-test**.

10.- La opción MinMax Table **no** debe estar seleccionada.

11.- Hacer “click” en OK para abandonar la ventana.

La red creada aparecerá en pantalla, además de dos instrumentos de medida que permiten observar la progresión de la red (Figura 4).

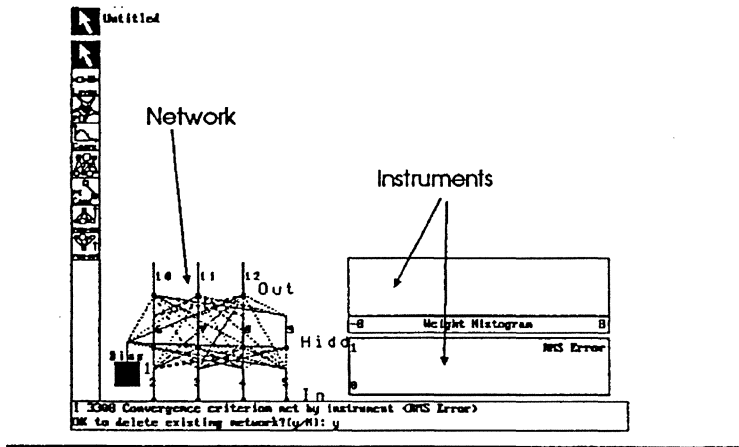


Figura 4.- RNA BP creada con sus instrumentos de control del entrenamiento.

La capa de entrada se encuentra en la parte inferior de la pantalla y la capa de salida en la parte superior. Todas las capas comprendidas entre la

de salida y la de entrada se denominan capas ocultas. Cada caja que forma parte de una capa es un elemento de proceso (EP). El color, tamaño e interior de un EP indica sus valores de salida. Las líneas continuas o de puntos que unen EP's son las conexiones. El color, así como el hecho de que una línea sea continua o de puntos es indicativo de la fuerza de la conexión.

Todos estos parámetros pueden modificarse mediante el comando Network/Display Style.

## **2.2- Movimiento de los objetos de la red.**

Vamos a mover los objetos de la red (tanto los instrumentos como la propia red) a situaciones más convenientes de la pantalla.

### **I. Movimiento de instrumentos:**



- El cursor del ratón debe tener la forma de un apuntador.
- Situar el apuntador sobre el instrumento que se quiera mover.
- Presionar el botón del ratón.
  - La forma del apuntador cambia a un pequeño símbolo gráfico y cuando empecemos a mover el instrumento aparecerá una línea alrededor de él.
- Mover esta línea al lugar deseado.
- Soltar el botón y el instrumento cambiará de situación.

### **II. Modificación del tamaño de los instrumentos:**





- Situar el cursor del ratón en la esquina del instrumento cuyo tamaño se desea cambiar.
- Presionar el botón del ratón; el cursor tomará la forma de una pequeña esquina y aparecerá una marca alrededor del instrumento.
- Mover la esquina hacia el exterior o el interior del instrumento según se quiera aumentarlo o reducirlo.
- Soltar el botón del ratón cuando hayamos alcanzado el tamaño deseado.

### III. Movimiento de la red:



Network  
drag  
pointer

- Posicionar el cursor del ratón cerca de la red pero no encima, luego presionar el botón del ratón.
- La forma del cursor cambiará a una pequeña pantalla.
- Mover la marca que delimita la red a la posición deseada mientras se mantiene pulsado el botón. Si nos habíamos situado demasiado cerca de la red sólo una capa será marcada para poder moverla. No debemos cambiar el orden vertical de las capas ya que desorganizaríamos el funcionamiento de la red.
- Soltar el botón para posicionar la red en el nuevo lugar elegido.

### **2.3.- Almacenamiento de la red**

Ahora vamos a archivar la red que acabamos de crear. Explorer tiene dos comandos para grabar ficheros: “Save” y “Save As”. Save se usa

generalmente para actualizar un fichero de red que ya había sido previamente creado. "Save As" se usa para grabar un fichero de red por primera vez o para grabar la versión actual de una red en un fichero nuevo.

Hay tres formatos en los que puede ser salvada una red:

- \* Binario.
- \* ASCII.
- \* ASCII con comentarios.

El modo binario ocupa la menor cantidad de espacio y contiene toda la información necesaria para volver a crear la red. Además, los ficheros tardan poco tiempo en grabarse o cargarse. Sin embargo, las redes en formato binario no son compatibles entre diferentes tipos de computadores.

Los ficheros de red en formato ASCII son mucho más grandes que los binarios, pero las redes almacenadas en formato ASCII son compatibles entre distintos computadores.

Los ficheros en formato ASCII con comentarios son ficheros en ASCII que llevan comentarios en cada línea, ocupan gran cantidad de espacio y se usan raramente.

Para grabar una red en el disco daremos los siguientes pasos:

- 1.- Seleccionar el comando **Save As** en el menú File. Aparecerá la ventana Save As (Figura 5).
- 2.- Seleccionar la opción **Binary**.
- 3.- Escribir el nombre **baby** en el campo reservado para el nombre de fichero. La extensión **.nnd** se asume para todos los tipos de redes.

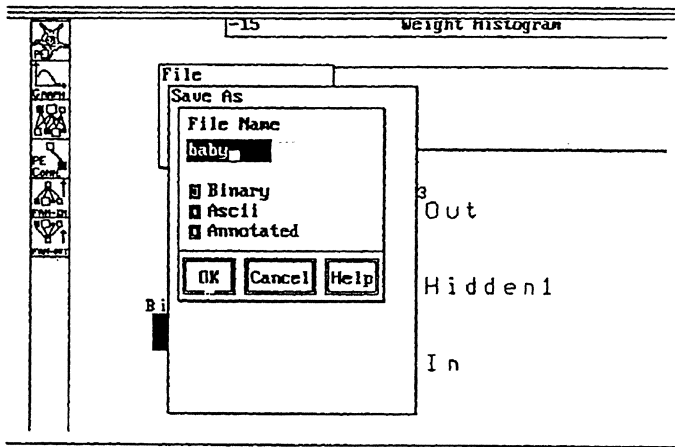


Figura 5.- Ventana para salvar una red en el disco.

4.- Hacer "click" en OK.

5.- Esperar a que el mensaje de confirmación aparezca en la pantalla.

## 2.4.- EP's y conexiones

Vamos a examinar los valores y pesos correspondientes a un elemento individual de proceso. Además, veremos que parámetros relativos a un EP se pueden cambiar.

Para ver los valores y pesos de un EP seguiremos los siguientes pasos:

1.- Seleccionar el icono **PE** (Figura 6).

2.- Seleccionar el icono **Edit** en el menú que hemos desplegado deslizando el cursor del ratón hasta él y soltando el botón.

El icono PE aparece remarcado, la casilla de la parte superior izquierda muestra que nos encontramos en la opción **Edit**, el cursor cambia de forma y aparece un mensaje diciéndonos que debemos señalar un EP.

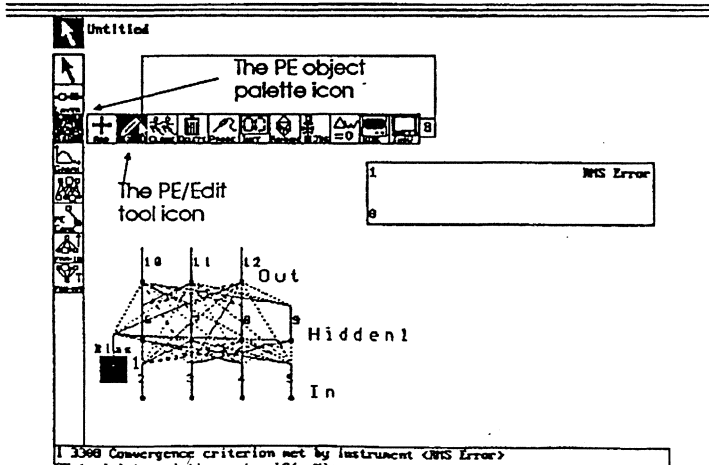


Figura 6.- Ventana de selección del icono PE.

### 3.- Seleccionar cualquier elemento de proceso de la capa oculta.

Aparecerá en pantalla una ventana (Figura 7) que nos muestra una serie de parámetros relativos al EP seleccionado. Los valores de los campos para el EP se encuentra situados en el lado izquierdo de la ventana. Estos valores varían para los diferentes tipos de redes. Para el caso de las Back Propagation se definen de la siguiente forma:

- PE: A cada EP se le asigna una única posición de memoria, y a estas posiciones se les asigna un número que varía entre 1 y el máximo número de EP's permitidos.
- Name: designa el nombre del EP. En su defecto se toma el número.
  - Sum: Es el valor del sumatorio.
  - Tran: Es el valor de salida de la función de transferencia.
  - Output: Es el valor de salida del EP.
  - Error: Es el error propagado hacia atrás.
  - Cur Err: Es el error puro.

- Err Fac: Es un factor de escalamiento para el error.

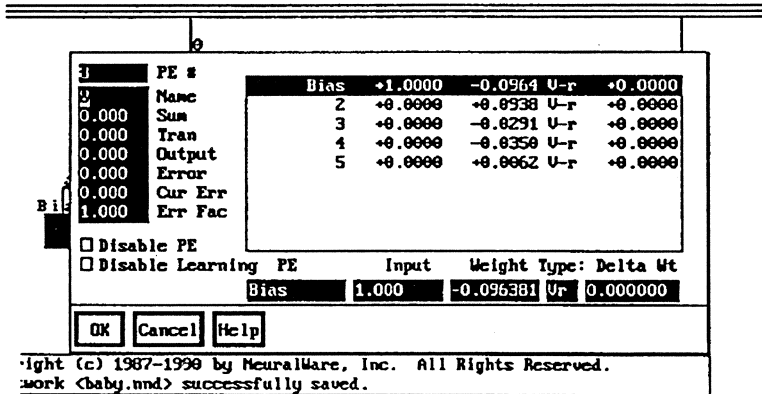


Figura 7.- Ventana de control de un EP.

Podemos desactivar un EP de manera que no se actualice su peso en la fase de aprendizaje.

En el lado derecho de la ventana podemos ver las conexiones que pertenecen al EP. Estas son las conexiones que le proporcionan una entrada.

El EP origen para la conexión, la entrada actual, el peso, el tipo de peso y el peso delta se muestran para cada una de las conexiones.

El peso puede ser de los siguientes tipos:

- \* V: Peso variable (puede cambiar durante el entrenamiento y, por lo tanto, en el aprendizaje).
- \* F: Peso fijo (no cambia durante el aprendizaje).
- \* M: Los pesos modulados son pesos fijos que se modifican mediante un factor modulador.
- \* S: Los pesos "set" son similares a los pesos "mod" salvo que los primeros se multiplican por el valor de entrada.
- \* D: Desactivado.

Normalmente, en una red Back Propagation, los pesos son de tipo V.

Una conexión puede tener una extensión que refiere como es la relación con los EP's. Puede ser:

- a Absoluta
- r Relativa
- l Local

Hacer "click" en la opción CANCEL. De este modo abandonamos la ventana sin implementar ninguno de los cambios que hayamos realizado (exceptuando los relativos al tipo del peso de las conexiones y a su extensión).

## 2.5.- Conexiones para un EP particular

A veces, es útil ver como un EP o una capa de EP's está conectada al resto de la red. Asimismo, cuando tenemos una gran cantidad de conexiones (full-connection) es común mostrar únicamente las conexiones suficientes para que la red esté representada de forma clara.

Explorer permite ver conexiones de modo selectivo. Ahora vamos a ocultar todas las conexiones para luego ir examinando estas paso a paso.

1.- Seleccionar el comando **Hide connections** en el menú Network (Figura 8).

Se ocultarán todas las conexiones.

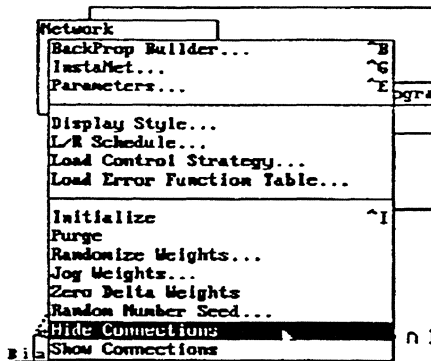


Figura 8.- Ventana de Red para ocultar las conexiones.

- 2.- Seleccionar el icono PE y desplegar el menú respectivo.
- 3.- Escoger la opción **Show Connections**.
- 4.- Hacer "click" sobre el elemento más a la izquierda de la capa de entrada. Aparecerá una ventana que permite elegir las conexiones que queremos mostrar.
- 5.- Seleccionar **Output Connections** (Figura 9) y luego **Ok**. Se muestran todas las conexiones de salida del EP elegido.

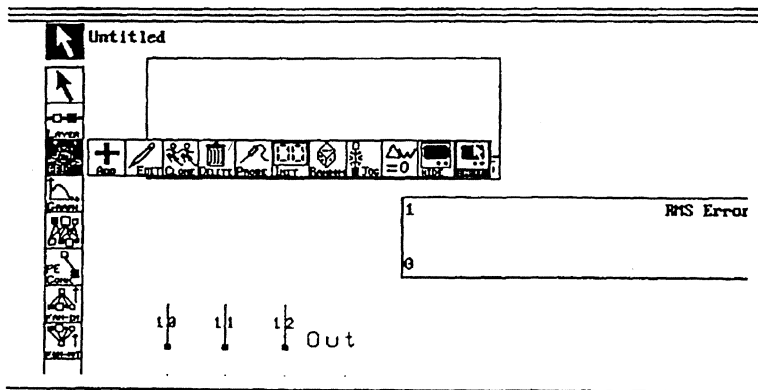


Figura 9.- Ventana con las conexiones ocultas.

Repetir la operación con el EP más a la izquierda de la capa oculta. Abandonar la utilidad Show Connections mediante el icono apuntador.

## 2.6.- Parámetros de capa

Las funciones matemáticas y las reglas de aprendizaje para los EP's se eligen en la capa base. Para ver los parámetros de capa:

- 1.- Seleccionar el icono **Layer** y luego la utilidad **Edit**.
- 2.- Escoger cualquier EP de la capa de salida. Aparece la ventana Layer Edit (Figura 10)

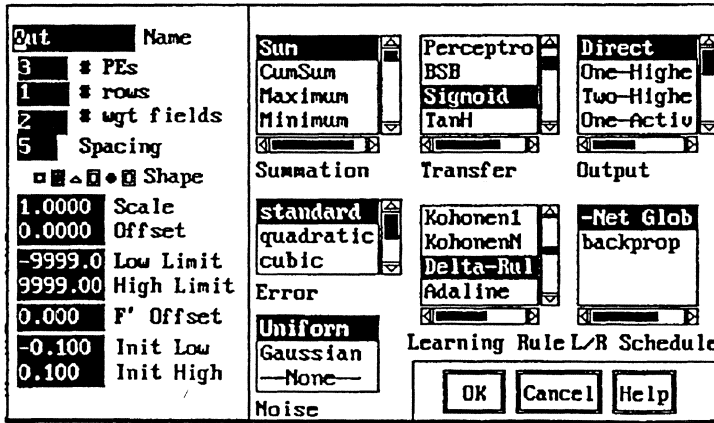


Figura 10.- Ventana de edición de capas.

El nombre de la capa se muestra en la esquina superior izquierda de la ventana. Debajo tenemos el número de EP's en la capa.

# rows indica el número de filas en las que se puede dividir la capa (esto permite mostrar gráficamente una capa con dos dimensiones).

El valor de # wgt fields asigna la memoria necesaria a cada conexión de la capa. (Este valor suele ser 1, 2 ó 3 dependiendo de la regla de aprendizaje, pesos acumulados y pesos delta).

Spacing se refiere a la separación entre EP's.

En la parte derecha hay siete ventanas que permiten escoger funciones matemáticas y reglas de aprendizaje. Estas son:

↳ Summation: Es el método para acumular las entradas.

Normalmente es la suma de las entradas multiplicadas por los pesos de las conexiones.

↳ Transfer: Es el método para transformar la entrada. Suele ser una función tangente hiperbólica o una sigmoidea.

↳ Output: La función que decide quien gana entre las capas competitivas. En el caso de Back Propagation no hay competición.



↳ Error: La función para transformar el error. También contiene el nombre de una tabla de error mediante la cual podemos aplicar un error específico basándonos en la salida actual y deseada de los EP's.

↳ Learning Rule: Se usa para actualizar pesos. Normalmente, para las Back Propagation se utiliza Regla Delta Acumulativa Normalizada.

↳ L/R Schedule: Contiene el nombre del fichero que posee muchos de los parámetros que serán usados por la red.

↳ Noise: Añade un valor aleatorio al valor del sumatorio antes de que sea aplicada la función de transferencia. Estas funciones se pueden aplicar capa a capa independientemente o bien para todas las capas. Hay tres funciones Noise: Uniforme, Gaussiana y nula.

Otros campos en esta ventana son:

- Shape: Se refiere a la forma de los EP's.
- Scale and Offset: Se usa para transferir datos de un rango a otro.
- Low and High Limit: Es el valor máximo y mínimo para el número de EP's por capa.
- F'offset: Un realce para la Back Propagation.
- Init Low and High: Son límites aleatorios para los pesos específicos de la capa usados durante la inicialización.

Abandonar esta ventana usa **CANCEL**.

## **2.7.- Parámetros Globales.**

Vamos a echar un vistazo a los parámetros globales de la red que hemos creado. Para hacerlo:

1.- Elegir el comando **Parameters** del menú Network. Aparece la ventana Parameters (Figura 11) que muestra que la red es Hetero-asociativa (las entradas y las salidas no tienen por que ser las mismas). El modo de pantalla es el de red. La estrategia de control ha sido escogida por el comando Back Prop Builder y es la "backprop". Esta estrategia controla el flujo de procesamiento a través de la red. El nombre del

fichero L/R Schedule es "backprop". Display Style controla los colores y tamaño de los objetos en la pantalla.

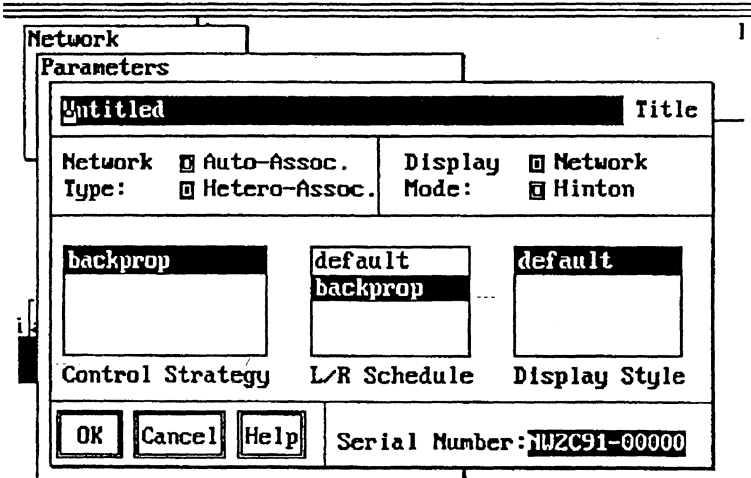


Figura 11.- Ventana de los Parámetros Globales de la Red.

2.- Abandonar la ventana mediante CANCEL.

## 2.8.- Ficheros de Entrenamiento y Test

En las redes de aprendizaje supervisado las entradas se presentan, durante la fase de entrenamiento, acompañadas de la salida deseada. La fase de test consiste en presentarle nuevas entradas junto con sus salidas deseadas a la red entrenada para ver como se comporta. Lógicamente, no vamos a testear la red con los mismos datos con los que la entrenamos ya que sólo veríamos lo bien que la red memoriza los datos de entrenamiento. Así que el conjunto de datos disponibles se divide en un conjunto de entrenamiento y un conjunto de test.

Los datos de entrada pueden estar en formato ASCII o binario. Para el caso de la red creada tenemos un fichero ASCII organizado en filas y columnas (Figura 12). Cada fila lógica contiene 4 entradas y 3 salidas separadas por blancos y con el ampersand (&) como carácter de

continuación. Los ficheros ASCII pueden tener comentarios, que irán precedidos del símbolo !. Veamos un ejemplo:

! Esto es un ejemplo de comentarios

0.2 0.2 0.1 0.2 ! Este es un comentario para un registro

0.2 0.1 0.0 ! parte de una entrada

&0.3 ! línea de continuación

En nuestro ejemplo tenemos un fichero de entrenamiento que se llama **iris-tra.nna**, que tiene el siguiente formato:

	0.224	0.624	0.067	0.043
♀	1.0	0.0	0.0	
	0.749	0.502	0.627	0.541
♀	0.0	1.0	0.0	
	0.557	0.541	0.847	1.000
♀	0.0	0.0	1.0	
	0.110	0.502	0.051	0.043
♀	1.0	0.0	0.0	
	0.722	0.459	0.663	0.584
♀	0.0	1.0	0.0	
	0.776	0.416	0.831	0.831
♀	0.0	0.0	1.0	
	0.196	0.667	0.067	0.043
♀	1.0	0.0	0.0	
	0.612	0.333	0.612	0.584
♀	0.0	1.0	0.0	

Figura 12.- Fichero ASCII con los datos de entrada en la Red.

y un fichero de test que se llama **iris-tes.nna**. (mismo formato).

Hemos identificado cada clase de flor con una salida de forma que la salida deseada tiene valor 1.0 y el resto 0.0.

## 2.9.- Parámetros de entrada/salida

Veamos, ahora, los parámetros que afectan a los datos de los ficheros de entrada anteriores.

1.- Seleccionar el comando **Parameters** en el menú I/O. Aparece la ventana de la figura 13:

<b>Lrn</b> <input checked="" type="checkbox"/> Keyboard <input checked="" type="checkbox"/> User IO <input checked="" type="checkbox"/> File Seq. <input checked="" type="checkbox"/> File Rand.		<b>Result File</b> <input type="checkbox"/> Append <input type="checkbox"/> Header <input type="checkbox"/> Input <input checked="" type="checkbox"/> Desired O/P		<b>Select Filenames</b> Image iris_tes iris_tra noblank	
<input type="checkbox"/> Binary <input type="checkbox"/> Load to RAM		<b>Epoch Size</b> 16		<b>Learn</b> <input checked="" type="checkbox"/> MNA iris_tra <input checked="" type="checkbox"/> MNB <b>Rcl/Test</b> <input checked="" type="checkbox"/> MNA iris_tes <input checked="" type="checkbox"/> MNB userio <input checked="" type="checkbox"/> UIO	
<b>File</b> <b>Start</b> <b>Network</b> <b>Layout:</b> <b>Field</b> <b>Ranges:</b>		Input    1    0.00000    1.00000 Output    5    0.20000    0.80000		<b>Min/Max Table</b> default	
<input type="button" value="OK"/> <input type="button" value="Cancel"/> <input type="button" value="Help"/>		<input type="button" value="QUICK GEN"/>			

Figura 13.- Ventana de parámetros en el menú I/O.

2.- Debajo del parámetro Lrn elegir la opción **File Rand.** (Los datos serán presentados a la red de modo aleatorio).

3.- La opción Binary para Lrn **no** debe estar seleccionada.

4.- Bajo el parámetro Rcl/Test elegir la opción **File Seq.**

5.- El fichero iris-tra aparece bajo el parámetro Learn con la extensión **.nna** asumida. Lo mismo ocurre con iris test bajo el parámetro Rcl/Test. Si usamos ficheros ASCII con extensión **.nnr** podemos decidir si los resultados se sobrescriben o se añaden, así como si contienen otras informaciones como la salida deseada.

6.- Si fuese necesario cambiar los campos File Layout Start de manera que para la entrada valga 1 y para la salida 5.

7.- Salir de la ventana con OK.

## 2.10.- Instrumentos

Explorer posee facilidades adicionales para la creación, modificación y manipulación de instrumentos (pesos, valores de elementos de proceso, performance, desarrollo de la red, etc). Estos datos, te permiten entender

las decisiones de la red y su topología, parámetros de aprendizaje, tiempos de aprendizaje, etc. Los instrumentos envían información de diagnóstico de una red a la pantalla, un fichero, o a un programa de usuario. Un pequeño número de instrumentos pueden ser creados directamente desde “BackProp Builder”, estos serán utilizadas en la red de clasificación de flores.

Es recomendable el uso de algún tipo de instrumentos en la construcción de todas las redes.

Los instrumentos están formados por dos componentes:

- Pruebas: Son los puntos de contacto del instrumento con la red. Estas pruebas pueden ser creadas para recoger datos de un PE específico, conjuntos de Pes, conexiones, la red, o alguna combinación de las anteriores.

- Instrumentos: Son las ventanas de la red que aparecen en la pantalla. Los instrumentos obtienen sus datos de las pruebas.

### **2.11.- Instrumento RMS Error**

Examinaremos el instrumento creado por “BackProp Builder” para medir el error de la capa de salida: el instrumento RMS Error (Figura 14).

Cuando seleccionas el “RMS Error Graph box” en “BackProp Builder” se crea un instrumento que mide el error medio para todos los Pes en la capa de salida.

$$Pr\ omedio = \frac{\sum (Error\_PE\_capa\_salida)^2}{n^{\circ} PE's - capa\_salida}$$

$$RMS\_error = \sqrt{Promedio}$$

El cuadrado elimina el signo del error, pero incrementa su magnitud. La raíz cuadrada retorna el error a su magnitud correcta, dejándolo en valor absoluto. El RMS Error es una medida de rendimiento común y muy valiosa durante el entrenamiento de la red.

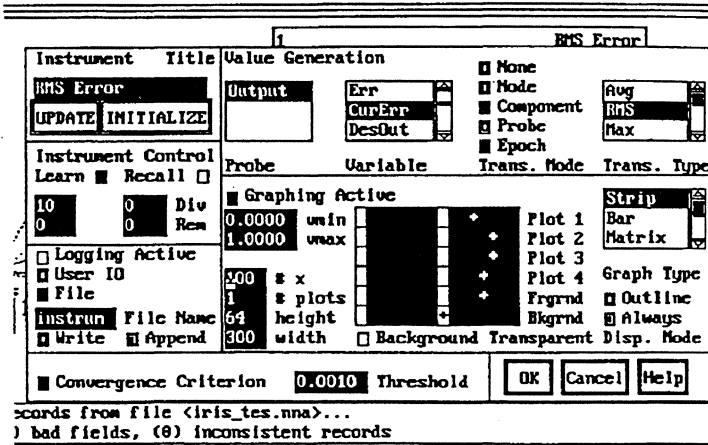


Figura 14.- Pantalla RMS Error.

Para examinar RMS Error:

- 1) Seleccione Graph Icon en la paleta de objetos.
- 2) Seleccione Edit tool.
- 3) Seleccione RMS Error instrument.

Otra forma: Hacer doble selección sobre el instrumento.

Los parámetros más interesantes son:

- ◆ *Probe*: Una prueba puede obtener sus resultados de PE's o de conexiones (o ambos).

Si "#plots"=1 ⇒ resultados limitados a capa salida.

- ◆ *Variable*: Es el tipo del dato que el instrumento está monitorizando. En este caso es el error actual.

- ◆ *Trans. Type*: Define la forma de combinar los valores. En este caso el valor es el RMS Error de todos los valores del componente "probe" (la capa de salida).

- ◆ *Epoch check box*: Epoch es el número de conjuntos de datos de entrenamiento presentados a la red (ciclos de entrenamiento)

entre actualizaciones de pesos. Este valor es especificado en el “I/O parameters command”. Se visualiza el número de ciclos realizados.

- ◆ *Instrument Control Learn Divisor*: Define el número de ciclos de aprendizaje entre dos actualizaciones del instrumento.

- ◆ *Vmin*: Mínimo valor de un punto. BackProp Builder lo define por defecto a 0.

- ◆ *Vmax*: Máximo valor de un punto, puesto a 1. Valores para “vmin” y “vmax” de 0 y 1 respectivamente son apropiados para la función de transferencia sigmoideal.

- ◆ *Converge Thereshold*: Usado únicamente por “RMS Error instrument” para detener el aprendizaje cuando se alcanza dicho parámetro. Debe ser un número positivo. Su valor por defecto en “BackProp Builder” es 0.001.

- ◆ *X*: Número de puntos gráficos en el eje x. En este caso tiene valor 100. Este valor puede cambiar de unos instrumentos a otros.

- ◆ *Plots*: Será 1 si queremos limitar al instrumento a representar datos de la capa de salida. Si le damos valor 2, también mostraremos las salidas de las capas ocultas.

Cuando ejecutamos la red de clasificación de flores, el “RMS Error instrument” nos da una indicación visual del buen o mal aprendizaje de la red.

Si “RMS Error” converge a 0  $\Rightarrow$  la red está aprendiendo.

## **2.12.- Histograma de Pesos (Weight Histogram)**

El instrumento “Weight Histogram” creado en “BackProp Builder” provee un histograma normalizado mostrando los valores de los pesos conducidos por las entradas para la capa de salida (Figura 15).

Cuando entrenamos la red, el error generado por cada ejemplo es usado para modificar los pesos de toda la red. Al cambiar los pesos, el histograma muestra la distribución de pesos en la capa de salida.

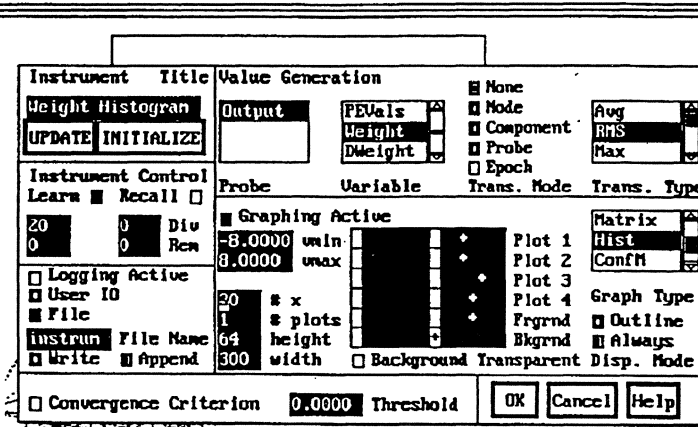


Figura 15.- Ventana "Histograma de Pesos".

Doble selección en el instrumento para pantalla Weight Histogram.

Parámetros:

- Trans.Mode: Seleccionar "none", significa que los valores son pasados sin modificar al Weight Histogram.

- Vmin=8.

- Vmax=+8.

- X=30

Definiendo las características de displayado de los pesos de la capa de salida en los 30 puntos definidos divididos en el rango -8..+8.

- Instrument Control Learn=20: La representación será actualizada cada 20 ciclos de aprendizaje.

### 2.13.- Entrenamiento de la red

Con anterioridad hemos definido:

- Construcción de la red con "BackProp Builder"

- Selección y examen de los ficheros de entrada.



- Creación y examen de aplicaciones de diagnóstico para monitorizar la fase de aprendizaje.

El paso siguiente es el entrenamiento de la red, pero antes hablaremos sobre la importancia de los coeficientes de entrenamiento.

**COEFICIENTES DE APRENDIZAJE**

Los parámetros seleccionados en "BackProp Builder" deben ser modificados en función del problema. Una opción es la de utilizar los coeficientes por defecto, para su modificación y ajuste en el aprendizaje.

Para ver los parámetros de aprendizaje seleccionados en "BackProp Builder (BPB)":

- 1) Selecciona "Network menu".
- 2) Selecciona "L/R Schedule command".

La pantalla resultante (Figura 16) está dividida en dos partes (superior e inferior). La inferior dedicada al aprendizaje.

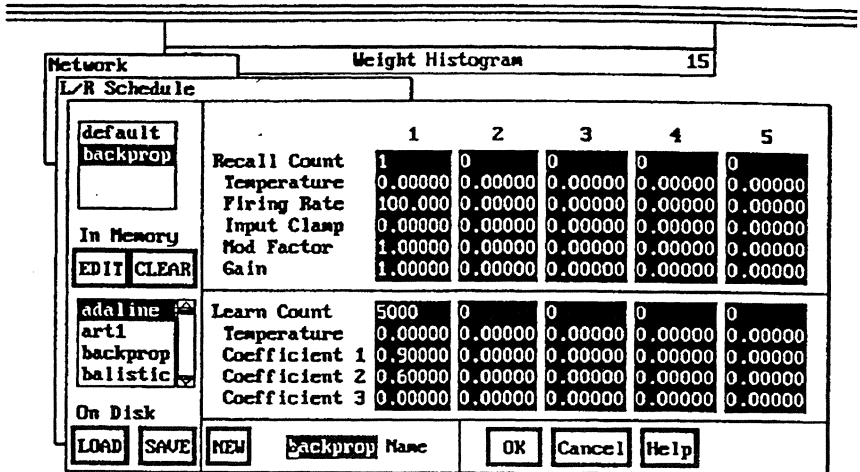


Figura 16.- Ventana esquema L/R (aprendizaje/ejecución).

Los coeficientes 1 y 2 son el coeficiente alfa y el momento respectivamente. Las diferentes columnas muestran como cambian los parámetros a lo largo del tiempo.

## ENTRENAMIENTO

Pasos para entrenar una red:

1) Selección doble del “RMS Error Instrument”.

Cambiar el criterio de convergencia a 0.05. Este no es lo bastante próximo a 0, pero reduce el número de ciclos de entrenamiento requeridos para que la red converja.

2) Seleccione “O.K”

3) Seleccione “Run menu”, seleccione “Learn”.

4) Seleccione “For”.

5) Contador “For”= 15000. Presentación de 15000 ejemplos a la red durante el entrenamiento.

6) Seleccione “O.K”. Comenzando el entrenamiento y apareciendo en pantalla la ventana de control de entrenamiento (Figura 17).

La diferencia entre la salida producida por la red y la salida deseada, es el error de la red. Este error transformado por la ecuación RMS es representado en el “RMS Error Instrument”.

Puedes interrumpir el entrenamiento en cualquier momento con la tecla Escape.

Un contador (inferior izquierda en la pantalla) indica el número de ejemplos presentados a la red. El diagrama de la red no es modificado para cada ejemplo presentado, esto produciría un proceso de entrenamiento demasiado lento. (El diagrama de la red puede ser vuelto a visualizar utilizando “Checkpoints” de “Run menu”).

- Red durante proceso de entrenamiento. Los pesos mostrados por “Weight Histogram”, están próximos a cero, esto indica que la red ha comenzado recientemente el entrenamiento. RMS muestra un error grande, por estar al comienzo del entrenamiento:

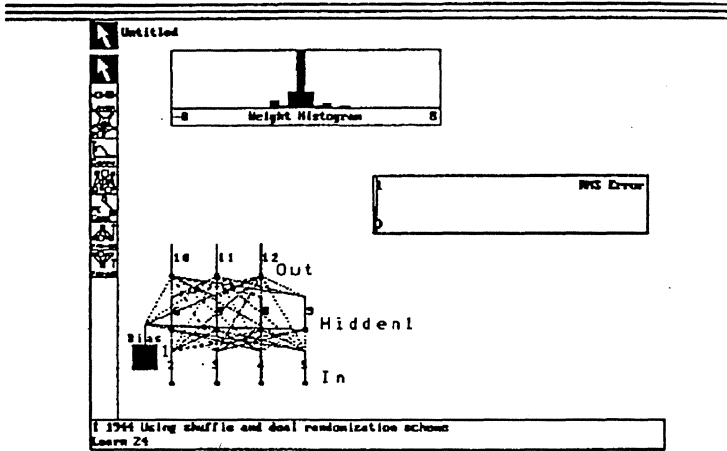


Figura 17.- Ventana de Control de entrenamiento.

- Red después de que el criterio de convergencia ha sido conseguido. “Weight Histogram” muestra los pesos modificados, con valores alejados de 0 (Figura 18). Y el “RMS Error” finaliza con error de criterio de convergencia (parar entrenamiento).

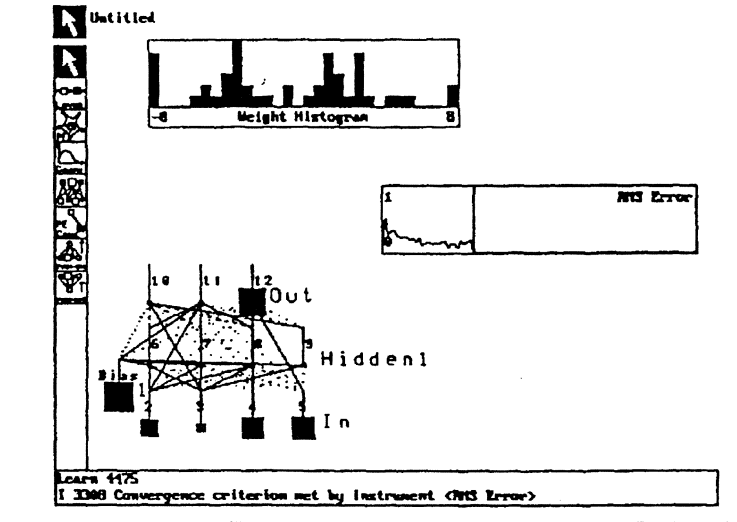


Figura 18.- Ventana de la Red cuando ya alcanzó la convergencia.

El error generado por cada ejemplo es utilizado para actualizar los pesos de toda la red. La distribución de pesos de la capa oculta y de salida aparecen en el instrumento "Weight histogram".

### **2.14.- Test de la red**

Una pregunta frecuente sobre el entrenamiento de redes es:

¿Cuánto ha de durar el entrenamiento?.

El tiempo de entrenamiento ha de estar en función del "rendimiento" deseado para la red. La fase de test es un camino para determinar la bondad del aprendizaje de la red. Durante esta fase (llamada "Recall/Test"), los casos de test son presentados a la red y esta entrega sus resultados. Si se conoce la respuesta correcta, se puede calcular el rendimiento de la red. Si los casos de test son representativos del mundo real, uno se puede hacer una idea del rendimiento de la red.

La mayor diferencia entre entrenamiento y test radica en la no modificación de los pesos durante los test.

Fase de test para la red de clasificación de flores:

1) Seleccionar "Parameters" en el menú "I/O menu".

2) En la pantalla de parámetros seleccionar:

Rcl/Test=iris tes

tipo de acceso "Recall"=File Seq"

"Result file"="Desired O/P".

3) O.K.

4) Seleccionar "Test command" en el menú "Run menu".

"Test" también escribe la salida deseada en el fichero de resultados.

5) Seleccione "Pass/All".

6) O.K.

El fichero de resultados tiene una extensión **.nnr**.

Explorer lee las entradas del fichero de test y presenta todos los ejemplos a la red entrenada. El número de ejemplos presentados aparece en la parte inferior izquierda de la pantalla. La salida deseada y la salida resultante se escriben en el fichero de resultados.

## 2.15.- Ficheros de Resultados.

1) Para leer el fichero de resultados: Use cualquier editor de texto para visualizar iris\_tes.nnr.

Esto puede hacerse utilizando "Os Call command".

Si quieres ejecutar de nuevo el test, selecciona la modalidad de "Append" en "I/O Parameters".

Hay una fila por cada ejemplo presentado en el fichero de resultados. Los tres primeros números son la salida deseada, y los tres últimos la salida de la red.

2) Ejemplo de línea:

0.0 0.0 1.0	0.1 0.15 0.86
salida deseada	resultado obtenido

Es correcta ("Si PE salida>0.5 entonces 1 sino 0")

El fichero de resultado en la clasificación de flores se puede apreciar en la figura 19.

```

=====
! Date: Fri Nov 16 12:30:32 1990 ! Result File: iris_tes.nnr
Input
File: iris_tes.nna ! Network: Untitled

1.000000 0.000000 0.000000 0.945984 0.089539 0.000000
0.000000 1.000000 0.000000 0.041111 0.673432 0.100153
0.000000 0.000000 1.000000 0.004371 0.164349 0.951315
1.000000 0.000000 0.000000 0.949121 0.080420 0.000000
0.000000 1.000000 0.000000 0.036576 0.650367 0.131851
0.000000 0.000000 1.000000 0.005389 0.198764 0.923599
1.000000 0.000000 0.000000 0.954160 0.073434 0.000000
0.000000 1.000000 0.000000 0.043646 0.683421 0.090796
0.000000 0.000000 1.000000 0.002333 0.091371 0.988030
1.000000 0.000000 0.000000 0.954732 0.074037 0.000000
0.000000 1.000000 0.000000 0.131574 0.811269 0.006911
0.000000 0.000000 1.000000 0.004432 0.168732 0.949382
1.000000 0.000000 0.000000 0.952458 0.079452 0.000000
0.000000 1.000000 0.000000 0.040453 0.647227 0.109592
0.000000 0.000000 1.000000 0.001887 0.073201 0.992665
1.000000 0.000000 0.000000 0.954354 0.072880 0.000000
0.000000 1.000000 0.000000 0.037268 0.642381 0.125731
0.000000 0.000000 1.000000 0.004419 0.167226 0.949758
=====

```

Figura 19.- Resultado de la clasificación del ejemplo de las flores.

## **2.16.- Almacenamiento de redes.**

Si se desea almacenar una red, por ejemplo, antes de hacer modificaciones en el modelo, si se guardan éstas y las modificaciones no son buenas, se puede retornar al modelo almacenado.

Para guardar una red seleccionar “Save command” en el menú “File menu”.

## **2.17.- Inicializar una red**

Una vez hemos salvado nuestra red, podemos hacer todo tipo de modificaciones sobre ella.

Para inicializar la red, seleccionar “Initialize command” en el menú “Network menu”.

Este comando inicializa los pesos en todas las capas a valores aleatorios entre un rango inferior y superior especificado para cada capa.

Además:

- El contador es puesto a cero.
- Los instrumentos son inicializados.
- Los pesos son puestos a cero, etc.

## **2.18.- Ajuste de coeficientes de aprendizaje.**

El empleo de diferentes coeficientes de aprendizaje para cada capa en una red multicapa puede decrementar el tiempo de aprendizaje.

En este ejemplo, usamos un coeficiente de aprendizaje relativamente amplio y un término momento.

Si la proporción de los coeficientes es demasiado grande, puede que la red nunca converja. Proporciones más pequeñas serán más estables.

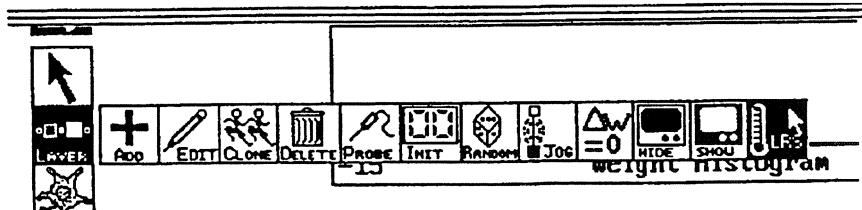
Usaremos el menú de utilidades “Tool Palette” para crear y asignar una organización a las capas de proceso y salida.

Para especificar un esquema para la capa de salida:

- 1) Seleccionar “Layer icon” en el menú de objetos.
- 2) Seleccionar “LRS tool” en el “Layer object”.

3) Selección sobre un PE de la capa de salida.

La ventana selección “Layer/LRS” puede verse en la figura 20.



Menú “LRS”

backprop untitled irisout irishid1		1	2	3	4
In Memory	Recall Count	1	0	0	0
EDIT CLEAR	Temperature	0.00000	0.00000	0.00000	0.000
adaline	Firing Rate	100.0000	0.00000	0.00000	0.000
art1	Input Clamp	0.00000	0.00000	0.00000	0.000
backprop	Mod Factor	1.00000	0.00000	0.00000	0.000
ballistic	Gain	1.00000	0.00000	0.00000	0.000
On Disk	Learn Count	500	1000	0	0
LOAD SAVE NEW	Temperature	0.00000	0.00000	0.00000	0.000
irishid1 Name	Coefficient 1	0.80000	0.30000	0.00000	0.000
OK Cancel Help	Coefficient 2	0.20000	0.10000	0.00000	0.000
	Coefficient 3	0.00000	0.00000	0.00000	0.000

Figura 20.- Ventana selección “Layer/LRS”.

4) Introducir Name=irisout

Este nombre se asigna a nuestro esquema.

5) Cambiar el contador de aprendizaje en la columna 1 a 500.

Esto indica que los coeficientes en la columna 1 son válidos hasta que el contador de aprendizaje sea 500.

6) Cambiar el coeficiente 1 (tasa de aprendizaje) en la columna 1 a 0.2.

7) Cambiar el coeficiente 2 (momento) en la columna 1 a 0.2.

8) Cambiar el contador de aprendizaje en la columna 2 a 1000.

Indica que, cuando el contador de aprendizaje es mayor que el valor especificado en la columna 2, serán usados los parámetros especificados.

9) Cambiar el coeficiente 1 en la columna 2 a 0.1.

10) Cambiar el coeficiente 2 en la columna 2 a 0.1.

11) Seleccione NEW.

El nuevo esquema aparecerá en la ventana “In Memory” con el nombre irisout.

12) Guardar el esquema seleccionando “SAVE”.

13) O.K.

Para crear un esquema para la capa oculta:

0) Seleccionar “Layer/LRS”.

1) Seleccionar un PE en la capa oculta.

2) Name=irishid1.

3) Cambiar el contador de aprendizaje en la columna 1 a 500.

4) Cambiar el coeficiente 1 (tasa de aprendizaje) en la columna 1 a 0.8.

5) Cambiar el coeficiente 2 (momento) en la columna 2 a 0.2.

6) Cambiar el contador de aprendizaje en la columna 2 a 0.2. (1000).

7) Cambiar el coeficiente 1 en la columna 2 a 0.3.

8) Cambiar coeficiente 2 en la columna 2 a 0.1.

9) Seleccionar NEW.

10) Seleccionar “SAVE” para guardar el esquema en disco.



## **2.19.- Establecimiento de un nuevo esquema de aprendizaje**

Un valor distinto de cero en el parámetro “Temperature” del esquema global, añade ruido a la capa de entrada.

Al definir un esquema L/R para la capa de salida y oculta, no era necesario un esquema global. Si se desea añadir una nueva capa a la red, se están asumiendo los coeficientes que serán usados. Al no especificar de forma explícita el nombre de un esquema global, el esquema por defecto definido por “BackProp Builder” será usado como esquema global.

1) Seleccionar “Parameters” en el menú “Network menu”.

2) Seleccionar esquema de aprendizaje global irisout. Este será el esquema por defecto para todas aquellas capas sin un esquema L/R definido de forma explícita.

3) O.K.

Los esquemas son almacenados en disco con la opción SAVE en el menú L/R. Para utilizar un esquema L/R previamente guardado en disco es necesario traerlo a memoria seleccionando LOAD.

**PRECAUCIÓN:** Un esquema guardado con una red cuando esta es almacenada puede tener el mismo nombre que un esquema salvado a disco con coeficientes diferentes. Al cargar la red, se cargará el esquema guardado con la red.

Los esquemas cargados en memoria pueden ser borrados seleccionando CLEAR. Clear no borra el esquema del disco.

Asegurarse siempre de tener los esquemas que deseas en cada capa antes de iniciar una fase de entrenamiento.

## **2.20.- Uso de puntos de chequeo para el almacenamiento automático de redes**

Una vez que estás familiarizado con el ajuste de parámetros de aprendizaje asociados a diferentes intervalos de tiempo (basado en el contador de aprendizaje) y has aprendido a guardar una red de forma manual. Puedes almacenar la red en los puntos en los que los coeficientes de aprendizaje son cambiados, de forma que si los nuevos coeficientes no

son buenos, no tienes que comenzar el entrenamiento desde el principio. Puedes cargar la red en el estado deseado. Esto lo puedes llevar a cabo en la ventana "Run/Check Points" (Figura 21).

Explorer presenta un método automático, llamado puntos de chequeo, basado en el contador de aprendizaje, para el almacenamiento automático de redes.

Por ejemplo, durante el entrenamiento, si los coeficientes se cambian según el contador de aprendizaje con valores 500, 1000 y 1500, debes salvar la red cada 500 ejemplos presentados.

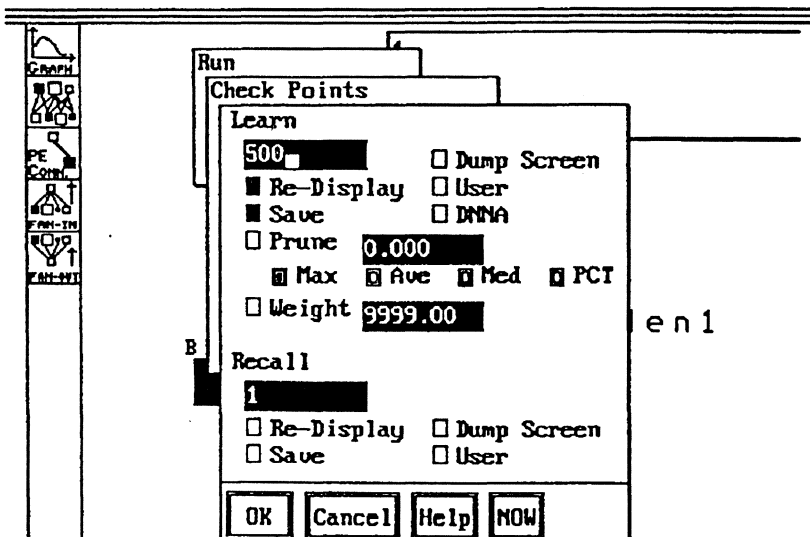


Figura 21.- Ventana "Run/Check Points".

Para guardar la red automáticamente:

- 1) Seleccionar "Check Points" del menú "Run".
- 2) Learn=500
- 3) Selecciona "Save"
- 4) Seleccionar "Re-Display"

La red presentará en pantalla un mensaje de aviso al mismo tiempo de ser almacenada.

- 5) O.K.
- 6) Seleccionar "Save As" en el "File menu".
- 7) Text field=baby0.
- 8) O.K.

El dígito del final en el nombre del "Save as" indica a Nworks que cada vez que salve la red en un punto de chequeo, el nombre usado será único. Durante los almacenamientos consecutivos, el número del final será incrementado en uno. En este ejemplo, después de 500 ejemplos presentados, la red será guardada con el nombre baby1. cuando el contador de aprendizaje sea 1000, la red será guardada con el nombre baby2, y así sucesivamente.

El almacenamiento automático puede consumir rápidamente gran cantidad de memoria en disco. Para borrar una red en disco la única forma es desde el sistema operativo.

Algunas opciones a intentar:

- Otros valores diferentes para los coeficientes.
- Estimular los pesos si la red cae en un mínimo local (RMS Error no pasa de un mínimo). Puedes inicializar los pesos para un PE o conexión usando el menú de utilidades. Examina la posible saturación sigmoidal de los PEs. Los PEs con saturación sigmoidal tienen sumatorios que son siempre demasiado bajos o demasiado altos, y la salida es 0 ó 1.

### **2.21.- Optimización de la red**

Posibles sofisticaciones de redes backpropagation:

- La Norm-Cum-Delta
- Regla de aprendizaje
- Tangente Hiperbólica (TanH). Alternativa a la sigmoidal.
- Función de transferencia
- Tablas MinMax
- Rangos. Capacidad de procesado
- Matrices de confusión. Instrumento que te permite ver cómo las salidas de la red se aproximan a sus valores derivados.

Para construir una red optimizada:

- 1) Seleccionar “BPB” en el menú “Network”.
- 2) Número de PE’s:
  - Input=4
  - Hid 1=4
  - Output=3
- 3) Seleccionar la regla de aprendizaje “Norm-Cum”.
- 4) Seleccionar la función de transferencia “TanH”.
- 5) Seleccionar en la ventana de entrenamiento el fichero iris\_tra.
- 6) Seleccionar en la ventana Rcl/Test el fichero de test iris\_tes.
- 7) Seleccionar:
  - Connect Bias
  - Connect Prior
  - MinMax Table
  - RMS Error Graph
  - Weight Histogram
  - Confusión Matrix (únicamente presente en la versión profesional).
- 8) O.K. Apareciéndonos una pantalla con la nueva red creada (Figura 22).

Opciones seleccionadas:

- *Regla de aprendizaje “Norm-Cum-Delta”*: Se define una variable “epoch” con un valor por defecto de 16. Los pesos son divididos por la raíz cuadrada de la variable epoch antes de ser aplicada la regla.

- *Función de transferencia tangencial hiperbólica (TanH)*: Es similar a la función sigmoideal. Sus rangos de salida están comprendidos entre -1 y +1, en oposición a los rangos de la sigmoideal comprendidos entre 0 y +1. La salida de la función de transferencia es utilizada como multiplicador en la ecuación de actualización de pesos, un rango entre 0 y 1 significa un multiplicador pequeño cuando la sumación es baja, y un multiplicador alto cuando la sumación es alta. Es aconsejable cuando

predominan las salidas 1. La tangencial hiperbólica da pesos iguales a valores bajos y altos.

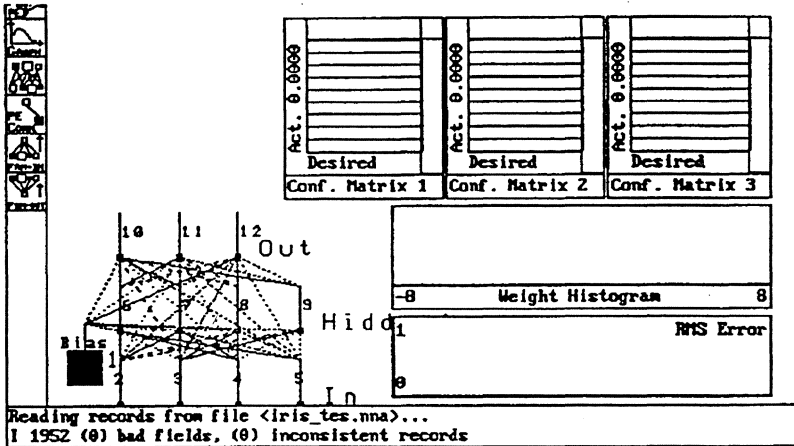


Figura 22.- Pantalla para construir una red optimizada.

### TABLAS MÍNIMO-MÁXIMO (MINMAX) Y RANGOS DE RED

Permite escalar proporcionalmente datos procedentes del mundo real a los rangos especificados para la red. Si se le presentasen estos datos sin preprocesar a la red, por ejemplo valores entre 0 y 10000, las sumaciones serán inmensas y se producirá una saturación de la red. Nworks soluciona esto con un preprocesado de los datos de entrada a red. El preproceso consiste en computar el menor y mayor de los datos del fichero de entrada. Estos valores son guardados en una tabla llamada "MaxMin Table". El usuario decide que rangos de entrada y salida serán escalados para ser presentados a la red. Nworks computa la escala para cada campo de datos. Los valores reales son escalados a los rangos de la red. Después de que la red produce un resultado escalado, el resultado es re-escalado a unidades del mundo real.

Cuando seleccionas la tabla "MinMax" y la regla "Norm-Cum-Delta", Nworks crea automáticamente la tabla "MinMax" con el fichero de datos

y establece rangos de entrada a red entre -1 y +1 y un rango de salida entre -0.8 y +0.8.

Las facilidades de preproceso son accesibles desde:

- "BPB"
- I/O//Parameters: Selección de rangos de red.
- I/O//MinMax table: Para crear y editar tablas MinMax.

Las tablas MinMax son almacenadas con la red. También pueden ser grabadas de forma independiente (ficheros con extensión .nmf).

### CAMPOS ERRÓNEOS Y REGISTROS INCONSISTENTES

MinMax señala posibles discrepancias encontradas en el fichero de datos. Por ejemplo, si alguno de los datos no es un número (campo mal). Si un campo es omitido en un registro (existe registro inconsistente).

### RANGOS DE RED

BPB establece unos rangos apropiados para la regla de aprendizaje "Norm-cum-delta". Puedes verlos en "I/O//Parameters" (Figura 23).

Parameters		Desired		Select Filenames	
Ln	Rcl/Test	Result File		image	
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Keyboard	<input type="checkbox"/> Append	irisTes		
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> User IO	<input type="checkbox"/> Header	irisTra		
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> File Seq.	<input type="checkbox"/> Input	ocr		
<input checked="" type="checkbox"/>	<input type="checkbox"/> File Rand.	<input checked="" type="checkbox"/> Desired O/P	Learn...		
<input checked="" type="checkbox"/>	<input type="checkbox"/> Binary	Epoch Size		<input checked="" type="checkbox"/> NNA	
<input checked="" type="checkbox"/>	<input type="checkbox"/> Load to RAM	16		irisTra	<input checked="" type="checkbox"/> NNB
File Layout:		Network Ranges:		Rcl/Test	<input checked="" type="checkbox"/> NNA
Input	1	-1.00000	1.00000	irisTes	<input checked="" type="checkbox"/> NNB
Output	5	-0.80000	0.80000	userio	<input checked="" type="checkbox"/> UIO
MinMax Table		default			
		irisTra			
OK		Cancel		Help	
		QUICK GEN			

Figura 23.- Pantalla de parámetros de Entrada/Salida.

Rangos de entrada a red	-1.0 a +1.0
Rangos de salida de red	-0.8 a +0.8

Recuerda que esto es para una tangencial hiperbólica. Debes utilizar 0.2 y 0.8 como rangos de salida para una red sigmoideal. La selección de los rangos de red están enlazados al número de entradas, tipo de función de transferencia en uso, y los valores iniciales de los pesos. La idea básica es la de seleccionar un rango que produce sumaciones que inicialmente no saturan la función de transferencia.

### MATRIZ DE CONFUSIÓN

Hemos visto dos formas de comprobar la bondad de la red:

- Instrumento RMS Error.
- Fichero de test.

La matriz de confusión representa el rendimiento de la red durante el aprendizaje y el test (Figura 24). Muestra una correlación entre los resultados actuales y los deseados.

Una representación sobre la diagonal inferior-izquierda a superior-derecha indica que la red está dando una salida igual a la deseada.

Por ejemplo, si la salida deseada es 0.8 y la producida por la red es 0.8, se pondrá en ON el punto de intersección entre 0.8 (eje x) y 0.8 (eje y), en caso contrario el punto estará en OFF. Muchos puntos en ON señalarán una buena red. Los puntos OFF indican que la red está haciendo predicciones bajas que deberían ser altas.

Act. 0.0000	Act. 0.0000	Act. 0.0000
Desired	Desired	Desired
Conf. Matrix 1	Conf. Matrix 2	Conf. Matrix 3

Figura 24.- Matriz de Confusión.

El valor mostrado en el eje vertical muestra el coeficiente de correlación lineal de la salida actual y deseada, utilizando para ello la variable epoch. Esto se calcula de la siguiente forma:

$$\frac{\sum (d(i) - \bar{d})(o(i) - \bar{o})}{\sqrt{\left( \sum (d(i) - \bar{d})^2 \sum (o(i) - \bar{o})^2 \right)}}$$

donde  $\bar{d} = \frac{1}{E \sum d(i)}$  y  $\bar{o} = \frac{1}{E \sum o(i)}$

E= variable epoch. Esto es una forma de calcular la bondad de la correlación entre la salida deseada (d) y la obtenida(o):

1 = perfectamente correlacionada.

0 = completamente incorrelacionada.

-1 = perfectamente anti-correlacionada.

Una matriz de confusión sólo puede representar la salida de un PE cada vez.

### EJECUCIÓN DE LA RED MEJORADA

Restaurar el valor de convergencia:

1) Doble selección sobre el instrumento RMS

2) Criterio de convergencia =0.05

3) O.K.

4) I/O//Parameters

5) Epoch=75

6) O.K.

Para ejecutar la red:

1) Seleccionar "Learn" del menú "Run".

2) Seleccionar "For".

3) Text=10000



En nuestra red, la salida deseada debe estar en el rango  $-0.8$  a  $+0.8$ . Observa la bondad de la red en la matriz de confusión. Nota también que el instrumento RMS es más consistente en su descenso de convergencia.

OTRAS POSIBILIDADES DE OPTIMIZACIÓN

- Incrementar y decrementar el valor de momento.
- Cambiar los coeficientes de aprendizaje.

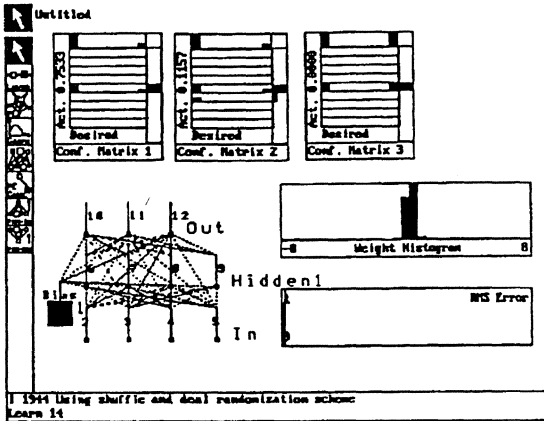


Figura 25.- Red al comienzo del entrenamiento.

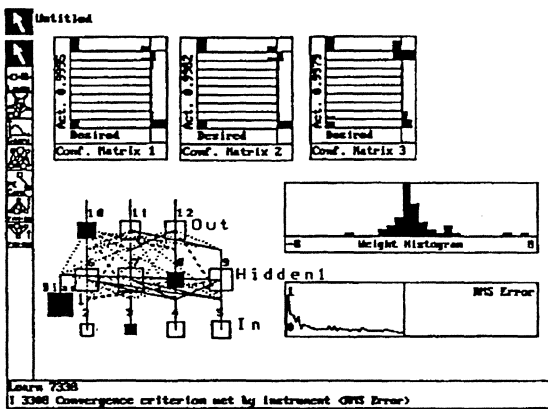


Figura 26.- Red después del entrenamiento.

**BIBLIOGRAFÍA**

- 1.- NueralWare, INC. "System Guide PC Versions". 1991.
- 2.- NeuralWare, INC. "Reference Guide". 1991.
- 3- NueralWare, INC. "UsingNworks". 1991.
- 4.- NeuralWare, INC. "Neural Computing". 1991.