

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 643 448**

51 Int. Cl.:

**G06N 3/04**

(2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **10.04.2001 PCT/NZ2001/00059**

87 Fecha y número de publicación internacional: **18.10.2001 WO01078003**

96 Fecha de presentación y número de la solicitud europea: **10.04.2001 E 01926261 (7)**

97 Fecha y número de publicación de la concesión europea: **04.10.2017 EP 1287488**

54 Título: **Sistema y método de aprendizaje adaptativo**

30 Prioridad:

**10.04.2000 NZ 50388200**

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

**22.11.2017**

73 Titular/es:

**PACIFIC EDGE LIMITED (100.0%)  
Level 6, Otago House  
481 Moray Place Dunedin, NZ**

72 Inventor/es:

**KASABOV, NIKOLA KIRILOV**

74 Agente/Representante:

**CAMPELLO ESTEBARANZ, Reyes**

**ES 2 643 448 T3**

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

**DESCRPCIÓN**

Sistema y método de aprendizaje adaptativo

Sector de la invención

5 La invención se refiere a un sistema y método de aprendizaje adaptativo y, en particular, se refiere a un módulo de red neuronal que forma parte de un sistema de aprendizaje adaptativo.

Antecedentes de la invención

10 Los problemas del mundo real, como el análisis masivo de datos biológicos y el descubrimiento de conocimiento, el reconocimiento adaptativo del habla y la adquisición de lenguaje durante toda la vida, la predicción inteligente y adaptativa y los sistemas de control, los sistemas basados en agentes inteligentes y agentes adaptativos en la web, robots móviles, sistemas de monitorización visual, procesamiento de información multimodal, sistemas de soporte de decisión adaptativa inteligente, aparatos domésticos adaptativos y edificios inteligentes, sistemas que aprenden y controlan los estados del cerebro y el cuerpo a partir de biorretroalimentación, sistemas que clasifican datos bioinformáticos y otros sistemas requieren soluciones sofisticadas para construir sistemas de base de conocimiento adaptativo en línea.

15 Estos sistemas deben ser capaces de aprender rápidamente a partir de una gran cantidad de datos, adaptarse incrementalmente en línea, tener una estructura abierta que permita la creación dinámica de nuevos módulos, memorizar información que pueda utilizarse más tarde, interactuar continuamente con el entorno con una forma de aprendizaje «de por vida», gestionar el conocimiento así como los datos, y representar adecuadamente el espacio y el tiempo en su estructura.

20 Las técnicas de inteligencia artificial (IA) y las redes neuronales establecidas presentan dificultades cuando se aplican para el aprendizaje basado en conocimiento en línea. Por ejemplo, los perceptrones de múltiples capas (PMC) y los algoritmos de aprendizaje de retropropagación presentan numerosos problemas, por ejemplo el olvido catastrófico, problemas de mínima locales, dificultades en la extracción de reglas, incapacidad para adaptarse a nuevos datos sin actualización de los datos antiguos, y los tiempos de capacitación excesivos cuando se aplican a grandes cantidades de datos.

25 El mapa de autoorganización (MAO) puede no ser eficiente cuando se aplica en aprendizaje no supervisado de nuevos datos, ya que el MAO asume una estructura fija y una red fija de nodos conectados en un espacio de salida topológico que puede no ser apropiado para proyectar un determinado conjunto de datos. Las redes neuronales de base radial requieren primero agrupamiento y después la aplicación del algoritmo de retropropagación. Los sistemas neuronales difusos no pueden actualizar las reglas aprendidas a través de capacitación continua sobre datos adicionales sin olvidos catastróficos.

30 Este tipo de redes no son eficientes para el aprendizaje en línea y adaptativo, aunque proporcionan una mejora sobre las técnicas anteriores.

35 BERENJI H R Y COL.: «Agrupación en el espacio de producto para la inferencia difusa» PROCEDIMIENTOS DE LA CONFERENCIA INTERNACIONAL SOBRE SISTEMAS DIFUSOS. SAN FRANCISCO, MAR. 28 - ABR. 1, 1993, NUEVA YORK, IEEE, EE. UU., vol. VOL. 2 CONF. 2, 28 marzo 1993 (1993-03-28), páginas 1402-1407, XP010103946 ISBN: 0-7803-0614-7 describe un método para generar un conjunto de reglas difusas con conclusiones lineales a partir de datos utilizando funciones de base radial y agrupaciones en espacio de producto. El método está basado en un módulo de red neuronal que comprende nodos de entrada para recibir datos, nodos de regla teniendo cada uno un umbral de activación mínimo, nodos de salida y un componente de adaptación para modificar los parámetros del nodo de regla basado en los datos de salida y para fusionar nodos de regla.

40 En XP010350803 «Inserción de reglas y extracción de reglas a partir de redes neuronales difusas en desarrollo: Algoritmos y aplicaciones para construir sistemas expertos en inteligencia y adaptativos» Kasabov N; Woodford B Procedimientos de conferencia de sistemas difusos, 1999. FUZZ-IEEE'99. 1999 IEEE International Seoul, Seoul, Korea, 22-25 Agosto 1999, Piscataway, NJ, EE. UU., IEEE, VOL - 3 (1999-08-02), proporciona algoritmos para la extracción de reglas y la inserción de reglas para redes neuronales difusas en desarrollo. El módulo de red neuronal comprende nodos de entrada para recibir datos, nodos de regla teniendo cada uno un umbral de activación mínimo, nodos de salida y un componente de adaptación para modificar los parámetros y para agregar nodos de regla.

MOREIRA M Y COL.: «Redes neuronales con índices con tasa de aprendizaje adaptativo y condiciones de impulso», MENCION DE INTERNET, 1 octubre de 1995 (1995-10-01), páginas I-1, XP002635433, sugiere adaptar la tasa de aprendizaje individualmente para cada nodo de regla individual.

Explicación resumida de la invención

- 5 La presente invención proporciona un módulo de red neuronal, un sistema de aprendizaje adaptativo y un método para implementar un módulo de red neuronal como el definido en las reivindicaciones independientes adjuntas. Las realizaciones preferidas según la presente invención se definen en las reivindicaciones dependientes adjuntas.

Breve descripción de las figuras

- 10 A continuación se describirán formas preferidas del sistema y método de aprendizaje adaptativo con referencia a las figuras adjuntas en donde:

la Figura 1 es una vista esquemática del hardware en el que puede implementarse una forma de la invención;

la Figura 2 es otra vista esquemática de un sistema de aprendizaje adaptativo de la invención;

la Figura 3 es una vista esquemática de un módulo de red neuronal de la Figura 2;

la Figura 4 es un ejemplo de funciones de pertenencia para utilizar con la invención;

- 15 la Figura 5 es un ejemplo de un modo de regla de la invención;

la Figura 6 ilustra el proceso de ajuste y aprendizaje en relación con el nodo de regla de la Figura 5;

la Figura 7 muestra un sistema de aprendizaje adaptativo de la invención que tiene tres nodos;

la Figura 8 muestra un método para agrupar los nodos de regla de la Figura 7;

la Figura 9 ilustra otro método para agrupar los tres nodos de regla de la Figura 7;

- 20 las Figuras 10 y 11 ilustran la agrupación de nodos de regla asignados espacialmente;

las Figuras 12 y 13 ilustran la agrupación de nodos de regla asignados linealmente;

las Figuras 14 a 17 ilustran diferentes estrategias de asignación para nuevos nodos de regla;

las Figuras 18A y 18B ilustran el sistema aprendiendo una función caótica de series temporales complejas;

la Figura 19 es una tabla de reglas seleccionadas extraídas de un sistema basado en la función de la Figura 18;

- 25 las Figuras 20 y 21 ilustran el sistema aprendiendo de ejemplos de datos de series temporales;

las Figuras 22 y 23 ilustran aprendizaje continuo no supervisado por el sistema;

la Figura 24 ilustra nodos de regla desarrollados y la trayectoria de una palabra hablada «zoo» en el espacio bidimensional de los primeros dos componentes principales en un sistema entrenado con una mezcla de palabras habladas en inglés de Nueva Zelanda y maorí;

- 30 la Figura 25 ilustra un análisis comparativo del modelo de aprendizaje del sistema con otros modelos;

la Figura 26 es una tabla que muestra la precisión en una prueba global de un método conocido comparado con el sistema de la invención;

la Figura 27 ilustra una regla a partir de un conjunto de reglas extraídas de un sistema desarrollado a partir de una secuencia de datos biológicos para la identificación de una unión de conexión entre intrones y exones en un gen; y

la Figura 28 ilustra una regla a partir de un conjunto de reglas extraídas de un sistema desarrollado a partir de los datos de expresión genética de una micromatriz tomados de dos tipos - ALL y AML de tejidos cancerosos leucémicos.

Descripción detallada de las formas preferidas

5 La Figura 1 ilustra una forma preferida del hardware en el que puede implementarse una forma de la invención. El sistema 2 preferido comprende un procesador de datos 4 conectado a una memoria principal 6, el procesador 4 y la memoria 6 funcionando bajo el control de un hardware o software operativo adecuado. El procesador 4 puede conectarse a uno o más dispositivos de entrada 8 y uno o más dispositivos de salida 10 con un controlador I/O 12. El sistema 2 también puede incluir dispositivos 14 adecuados de almacenamiento masivo, por ejemplo, discos flexibles, discos duros o unidades de CD Rom o aparatos de DVD, una pantalla 16, un puntero 17, un módem 18 y/o un controlador de red 19. Los diferentes componentes pueden estar conectados a través del bus del sistema o mediante una red cableada o inalámbrica.

15 En una realización, la invención puede disponerse para usarse para el reconocimiento de voz y puede alimentarse con señales de voz modelo. En esta realización, el (los) dispositivo(s) de entrada 8 puede(n) comprender un micrófono y/o un dispositivo de almacenamiento adicional en el que se almacenen señales de audio o representaciones de señales de audio. El (los) dispositivo(s) de salida 10 puede(n) comprender una impresora para mostrar el proceso del lenguaje o el habla por el sistema, y/o un altavoz adecuado para generar sonido. El habla o el lenguaje también pueden mostrarse en la pantalla 16.

20 Si la invención se adapta para clasificar datos de estudio de casos de bioinformática, estos datos podrían almacenarse en un dispositivo de almacenamiento masivo 14, al que accede el procesador 4 y mostrar los resultados en una pantalla 16 y/o en otro dispositivo de salida 10.

25 Cuando el sistema 2 se adapta para utilizar un robot móvil, el (los) dispositivo(s) de entrada 8 pueden) incluir sensores u otros aparatos dispuestos para formar representaciones de un entorno. Los dispositivos de entrada también pueden incluir un almacenamiento secundario en el que se almacene una representación de un entorno. El (los) dispositivo(s) de salida 10 puede(n) incluir una unidad de monitorización o visualización para mostrar el entorno procesado por el sistema. El procesador 4 también puede estar interconectado con medios de control de motor para transportar el robot desde una ubicación en el entorno procesado a otra ubicación.

30 Se apreciará que el sistema de aprendizaje adaptativo 2 puede disponerse para operar en numerosos entornos diferentes y para resolver numerosos problemas diferentes. En cada caso, el sistema 2 desarrolla su estructura y funcionalidad en el tiempo a través de la interacción con el entorno a través de los dispositivos de entrada 8 y los dispositivos de salida 10.

35 La Figura 2 ilustra los aspectos implementados por ordenador de la invención almacenados en la memoria 6 y/o almacenamiento masivo 14 y dispuestos para funcionar con el procesador 4. El sistema preferido se dispone como un sistema conexionista en desarrollo 20. El sistema 20 se proporciona con uno o más módulos de red neuronales o MRN 22. La disposición y funcionamiento del (de los) módulo(s) de red neuronal 22 forma la base de la invención y se describirá más detalladamente abajo.

El sistema incluye un componente de representación o de memoria 26 que comprende uno o más módulos de red neuronal 22. El componente de representación 26 incluye, preferiblemente, un componente de adaptación 28, como se describirá con detalle más adelante, que permite que los nodos de regla se inserten, extraigan y/o agrupen.

40 El sistema 20 puede incluir una cantidad de componentes conocidos adicionales, por ejemplos un componente de selección de características 24 dispuesto para realizar el filtrado de la información de entrada, la extracción de características y la formación de los vectores de entrada.

45 El sistema también puede incluir un componente de decisión 30 de nivel más alto que comprende uno o más módulos que reciben retroalimentación del entorno 34, un componente de acción 32 que comprende uno o más módulos que toman valores de salida del componente de decisión y pasan información de salida al entorno 34, y una base de conocimiento 36 que se dispone para extraer información abstracta comprimida a partir del componente de representación 26 y del componente de decisión 30 en forma de reglas, asociaciones abstractas y otra información. La base de conocimiento 36 puede utilizar técnicas como algoritmos genéticos u otras técnicas informáticas desarrolladas para evaluar y optimizar los parámetros del sistema durante su funcionamiento.

50 La Figura 3 ilustra una forma preferida del módulo de red neuronal 22. La estructura preferida es una red neuronal difusa que es una estructura conexionista que implementa reglas difusas. El módulo de red neuronal 22 incluye una capa de entrada 40 que tiene uno o más nodos de entrada 42 dispuestos para recibir datos de entrada.

- 5 El módulo de red neuronal 22 también puede comprender una capa de entrada difusa 44 con uno o más nodos de entrada difusos 46. Los nodos de entrada difusos 46 transforman los datos desde los nodos de entrada 42 para el posterior uso del sistema. Cada uno de los nodos de entrada difusos 46 puede tener una función de pertenencia unida a él. Un ejemplo de una función de pertenencia es la función de pertenencia triangular mostrada en la Figura 4. La función de pertenencia también puede incluir funciones Gaussianas o cualquier otra función conocida adecuada para este propósito. El sistema se dispone preferiblemente de manera que el número y tipo de la función de pertenencia pueda modificarse como se describirá más abajo. El propósito principal de los nodos de entrada difusos 46 es transformar los valores de entrada de los nodos de entrada 42 en grados de pertenencia a los que pertenezcan los valores de la función de pertenencia.
- 10 El módulo de red neuronal 22 también comprende una capa base de regla 48 con uno o más nodos de regla 50. Cada nodo de regla 50 está definido por dos vectores de ponderación de conexión  $W1(r)$  y  $W2(r)$ . La ponderación de conexión  $W1(r)$  se ajusta preferiblemente por aprendizaje no supervisado basado en medidas de similitud dentro de un área local del espacio del problema. Por otro lado,  $W2(r)$  se ajusta preferiblemente por aprendizaje supervisado basado en un error de salida, o en aprendizaje de refuerzo basado en advertencias de salida. Las ponderaciones de conexión ( $W1(r)$  y  $W2(r)$ ) se describen con más detalle más abajo.
- 15 El módulo de red neuronal 22 también puede comprender una capa de salida difusa 52 con uno o más nodos de salida difusos 54. Cada nodo difuso 54 representa una cuantificación de las variables de salida, similares a los nodos de entrada difusos 46 de la capa de entrada difusa 54. Preferentemente, se utilizan una función de entrada de suma ponderada y una función de activación lineal saturada para que los nodos calculen los grados de pertenencia a los que pertenece el vector de salida asociado al vector de entrada para cada una de las funciones de pertenencia de salida.
- 20 El módulo de red neuronal también incluye una capa de salida 56 con uno o más nodos de salida 58. Los nodos de salida 58 representan los valores reales de las variables de salida. Preferiblemente, se utiliza una función de activación lineal para calcular los valores no difusos para las variables de salida.
- 25 La capa base de regla 48 preferida comprende uno o más nodos de regla 50 que representan prototipos de asociaciones de datos de entrada-salida que pueden representarse gráficamente como asociaciones de hiperesferas desde los espacios de la capa de entrada difusa 44 y los espacios de la capa de salida difusa 52. Cada nodo de regla 50 tiene un umbral de activación mínimo que se determina preferiblemente mediante una función de activación lineal.
- 30 Como muestra la Figura 3, el módulo de red neuronal 22 también puede incluir una capa de memoria a corto plazo 60 con uno o más nodos de memoria 62. El objetivo de la capa de memoria a corto plazo 60 es memorizar de forma estructurada relaciones temporales de los datos de entrada. La capa de memoria a corto plazo se dispone, preferiblemente, para recibir información de y enviar información a la capa base de regla 48.
- 35 Como se ha descrito anteriormente, cada nodo de regla 50 representa una asociación entre una hiperesfera del espacio de entrada difuso y una hiperesfera del espacio de salida difuso. Estas esferas se describen con referencia a la Figura 5, que ilustra el ejemplo del nodo de regla 70 mostrado como  $r_j$ . El nodo de regla  $r_j$  tiene una hiperesfera 72 inicial en el espacio de entrada difuso. El nodo de regla  $r_j$  tiene un parámetro de umbral de sensibilidad  $S_j$  que define el umbral de activación mínimo del nodo de regla  $r_j$  a un vector  $x$  de entrada nuevo a partir de un ejemplo nuevo o una entrada  $(x,y)$  de manera que se considere la asociación del ejemplo a este nodo de regla. Un nuevo vector  $x$  de entrada acciona un nodo de regla si  $x$  satisface el umbral de accionamiento mínimo  $y$ , posteriormente, se considera su asociación el nodo de regla. El radio de la hiperesfera de entrada 72 se define como  $R_j = 1 - S_j$ , siendo  $S_j$  el parámetro del umbral de sensibilidad.
- 40 El nodo de regla  $r_j$  tiene una matriz de ponderación de conexión  $W1(r_j)$  que representa las coordenadas del centro de la esfera 72 en el espacio de entrada difuso. El nodo de regla  $r_j$  también tiene una hiperesfera de espacio de salida difuso 74, siendo las coordenadas del centro de la esfera 74 ponderaciones de conexión  $W2(r_j)$ . El radio de la hiperesfera 74 de salida se define como  $E$ , que representa el umbral de error o tolerancia al error del nodo de regla 70. De esta forma es posible que algunos nodos de regla se accionen mucho más que otros nodos de regla por los datos de entrada.
- 45 Un nuevo par de vectores de datos  $(x, y)$  se transforma en vectores de datos de entrada/salida difusos  $(x_f, y_f)$  que se asignarán al nodo de regla 70 si  $x_f$  cae dentro de la hiperesfera de entrada 72 y  $y_f$  cae dentro de hiperesfera de salida 74 cuando el vector de entrada  $x$  se propaga a través del nodo de entrada. La distancia de  $x_f$  desde el centro de la hiperesfera de entrada 72 y la distancia de  $y_f$  desde el centro de la hiperesfera de salida 74 proporciona una base para calcular y asignar la magnitud o resistencia a la activación. Esta resistencia al accionamiento proporciona una base para comparar las resistencias a la activación de diferentes nodos de regla. Por tanto, otra base para la asignación es que el nodo de regla 70 reciba la mayor activación entre otros nodos de regla. Los vectores de datos  $(x_f, y_f)$  se asociarán

al nodo de regla 70 si la diferencia difusa y normalizada entre  $x_f$  y  $W1(r_j)$  es menor que el radio  $R_j$ , y el error de salida normalizado  $Err = \|y - y'\| / N_{out}$  es menor que un umbral de error  $E$ , siendo  $N_{out}$  el número de las salidas y siendo  $y'$  producida por el sistema de salida. El parámetro  $E$  establece la tolerancia al error del sistema.

5 En el método preferido, una diferencia (distancia) difusa y normalizada entre dos vectores  $d_{1f}$  y  $d_{2f}$  de pertenencia difusos que representen los grados de pertenencia a los que dos datos vectoriales reales  $d_1$  y  $d_2$  pertenecen a frecuencias de Mel (MF) predefinidas, se calcula como:

$$D(d_{1f}, d_{2f}) = \|d_{1f} - d_{2f}\| / \|d_{1f} + d_{2f}\| \quad (1)$$

10 donde:  $\|x - y\|$  indica la suma de todos los valores absolutos de un vector que se obtiene después de la resta de un vector (o suma en caso de  $\|x + y\|$ ) de dos vectores  $x$  e  $y$ ; “ / ” indica división. Por ejemplo, si  $d_{1f} = (0,0,1,0,0,0)$  y  $d_{2f} = (0,1,0,0,0,0)$ , entonces  $D(d_{1f}, d_{2f}) = (1 + 1)/2 = 1$ , que es el valor máximo para la diferencia difusa normalizada local.

15 A medida que se introducen nuevas entradas en el nodo de regla 70, estas entradas de datos relevantes para  $r_j$  pueden asociarse al nodo de regla 70 proporcionando una oportunidad de aprendizaje. A medida que se introduce un nuevo vector de datos de entrada/salida difusos ( $x_f, y_f$ ) en el nodo de regla 70, el centro de la hiperesfera de entrada 72 se ajusta a la nueva esfera indicada en 72A ajustando  $W1(r_j^{(1)})$  a  $W1(r_j^{(2)})$ . La hiperesfera de salida 74 también se ajusta a la nueva esfera mostrada en 74A ajustando  $W2(r_j^{(1)})$  a  $W2(r_j^{(2)})$ .

20 Los centros de las hiperesferas de los nodos se ajustan en el espacio de entrada difuso dependiendo de la distancia entre el nuevo vector de entrada y el nodo de regla a través de una tasa de aprendizaje  $l_j$ , un parámetro que se ajusta individualmente para cada nodo de regla. El ajuste de las hiperesferas en los espacios de salida difusos depende del error de salida y también en la tasa de aprendizaje  $l_j$ , a través del algoritmo Widrow-Hoff LMS, también llamado el algoritmo Delta.

Este ajuste en los espacios de entrada y de salida puede representarse matemáticamente mediante el cambio en las ponderaciones de conexión del nodo de regla  $r_j$  desde  $W1(r_j^{(1)})$  y  $W2(r_j^{(1)})$  hasta  $W1(r_j^{(2)})$  y  $W2(r_j^{(2)})$  respectivamente según las siguientes operaciones vectoriales:

$$25 \quad W1(r_j^{(2)}) = W1(r_j^{(1)}) + l_j \cdot (W1(r_j^{(1)}) - x_f) \quad (2)$$

$$W2(r_j^{(2)}) = W2(r_j^{(1)}) + l_j \cdot (A2 - y_f) \cdot A1(r_j^{(1)}) \quad (3)$$

30 donde:  $A2 = f_2(W2 \cdot A1)$  es el vector de activación de las neuronas de salida difusas cuando el vector de entrada  $x$  está presente;  $A1(r_j^{(1)}) = f_1(D(W1(r_j^{(1)}), x_f))$  es la activación del nodo de regla  $r_j^{(1)}$ ; se puede utilizar una función lineal simple para  $f_1$  y  $f_2$ , p. ej.  $A1(r_j^{(1)}) = 1 - D(W1(r_j^{(1)}), x_f)$ , donde  $D$  es la medida de distancia normalizada y difusa;  $l_j$  es la tasa de aprendizaje actual del nodo de regla  $r_j$  calculado como  $l_j = 1 / N_{ex}(r_j)$ , donde  $N_{ex}(r_j)$  es el número de ejemplos asociados en ese momento con el nodo de regla  $r_j$ . El razonamiento estadístico que está detrás de esto es que cuantos más ejemplos estén asociados en ese momento a un nodo de regla, menos se «moverá» cuando un nuevo ejemplo

35 tenga que ser ajustado por este nodo de regla, es decir, el cambio en la posición del nodo de la regla es proporcional al número de ejemplos ya asociados, lo cual es una característica estadística del método.

Cuando se asocia un ejemplo nuevo con el nodo de regla  $r_j$  no solo cambia su ubicación en el espacio de entrada, sino también su campo receptivo expresado como su radio  $R_j$  y su umbral de sensibilidad  $S_j$ :

$$40 \quad R_j^{(2)} = R_j^{(1)} + D(W1(r_j^{(2)}), W1(r_j^{(1)})), \quad R_j^{(2)} \leq R_{max} \quad (4)$$

Respectivamente (3)

$$S_j^{(2)} = S_j^{(1)} - D(W1(r_j^{(2)}), W1(r_j^{(1)})) \quad (5)$$

donde  $R_{max}$  es un conjunto de parámetros que restringen el radio máximo del campo receptivo de un nodo de regla.

5 El proceso de ajuste y aprendizaje en el espacio de entrada difuso ilustrado en la Figura 6 que ilustra esquemáticamente cómo se ajusta el centro  $r_j^{(1)}$  82 del nodo de regla  $r_j$  80, después de aprender cada punto de datos nuevo, a su nueva posición  $r_j^{(4)}$  84 basada en un aprendizaje de una pasada en los cuatro puntos de datos  $d_1$ ,  $d_2$ ,  $d_3$  y  $d_4$ .

10 El componente de adaptación del sistema preferido permite que los nodos de regla se inserten, extraigan y adapten o agrupen como se describirá más adelante. En cualquier momento o fase del proceso de desarrollo o aprendizaje, las reglas difusas o exactas pueden insertarse ajustando un nuevo nodo de regla  $r_j$  para cada regla nueva, de manera que las ponderaciones de conexión  $W1(r_j)$  y  $W2(r_j)$  del nodo de regla representen esta regla.

Por ejemplo, la regla difusa (SI  $x_1$  es Pequeña y  $x_2$  es Pequeña ENTONCES y es Pequeña) puede insertarse dentro del módulo de red neuronal 22 ajustando las conexiones de un nodo de regla nuevo a los nodos difusos  $x_1$  - Pequeño y  $x_2$  - Pequeño y al nodo de salida difuso  $y$  - Pequeño a un valor de 1 cada uno. El resto de las conexiones se ajustan a un valor de 0.

15 De forma similar, se puede introducir una regla exacta en el módulo 22, por ejemplo, SI  $x_1$  es 3,4 y  $x_2$  es 6,7 ENTONCES y es 9,5. Aquí, los grados de pertenencia a los que los valores de entrada  $x_1=3,4$  y  $x_2=6,7$  y el valor de salida  $y=9,5$  pertenecen a los valores difusos correspondientes se calculan y unen a las ponderaciones de conexión correspondientes.

20 El componente de adaptación preferido también permite la extracción de reglas en las que el sistema identifique nuevas reglas y relaciones. Cada nodo de regla  $r_j$  puede expresarse como una regla difusa, por ejemplo:

Regla  $r$ : SI  $x_1$  es Pequeño 0,85 y  $x_1$  es Mediano 0,15 y  $x_2$  es Pequeño 0,7 y  $x_2$  es Mediano 0,3 {el radio del campo respectivo de la regla  $r$  es 0,5} ENTONCES y es Pequeño 0,2 e y es Grande 0,8 {ejemplos  $Nex(r)$  asociados a esta regla dentro de los ejemplos totales aprendidos por el sistema}.

25 Los números unidos a las etiquetas difusas indican el grado al que los centros de las hiperesferas de entrada y salida pertenecen a las funciones de pertenencia respectivas.

El componente de adaptación también permite, preferiblemente, la agrupación de nodos de regla. A través de esta técnica, se combinan varios nodos de regla en uno, como se muestra en las Figuras 7, 8 y 9 en un ejemplo de 3 nodos de regla  $r_1$ ,  $r_2$  y  $r_3$ .

30 La Figura 7 ilustra un módulo de red neuronal similar al módulo de la Figura 3. El módulo puede comprender, por ejemplo, una capa de entrada 40, una capa de salida difusa 44, una capa base de regla 48, una capa de salida difusa 52 y una capa de salida 56. La capa base de regla 48 incluye, por ejemplo, los nodos de regla  $r_1$ ,  $r_2$  y  $r_3$  indicados como 90, 92 y 94 respectivamente.

35 Para la agrupación de estos tres nodos de regla  $r_1$ ,  $r_2$ , y  $r_3$  pueden utilizarse las dos estrategias siguientes para calcular las conexiones del nuevo nodo de regla agrupado  $r_{agg}$   $W1$  (las mismas formas se utilizan para calcular las conexiones  $W2$ ):

- como un centro geométrico de los tres nodos:

$$W1(r_{agg}) = (W1(r_1) + W1(r_2) + W1(r_3)) / 3 \quad (6)$$

- como un centro estadístico ponderado:

40 
$$W1(r_{agg}) = (W1(r_1) \cdot Nex(r_1) + W1(r_2) \cdot Nex(r_2) + W1(r_3) \cdot Nex(r_3)) / Nsum \quad (7)$$

$$Nex(r_{agg}) = Nsum = Nex(r_1) + Nex(r_2) + Nex(r_3); \quad (8)$$

$$Rr_{agg} = D(W1(r_{agg}), W1(r_j)) + R_j \leq R_{max}; \quad (9) \quad \text{s su} \\ \text{ado}$$

es menor que el radio  $R_{max}$  máximo predefinido.

La Figura 8 muestra un ejemplo de agrupación como un centro geométrico de los tres nodos mientras que la Figura 9 muestra la agrupación como un centro estadístico ponderado.

Para que un nodo  $r_j$  dado «escoja» a los otros nodos con los que debería agruparse, se forman dos subconjuntos de nodos - el subconjunto de nodos  $r_k$  que si se activa a un grado 1 producirá un valor de salida  $y'(r_k)$  que se diferencia de  $y'(r_j)$  en menos que el umbral de error  $E$ , y el subconjunto de nodos que producen valores de salida que se diferencian de  $y'(r_k)$  en más que el umbral de error  $E$ . Las conexiones  $W2$  definen estos subconjuntos. Todos los nodos de regla del primer subconjunto que están más cerca de  $r_j$  en el espacio de entrada que el más cercano al nodo  $r_j$  del segundo subconjunto en cuanto a la distancia  $W1$  se agrupan si el radio calculado del nuevo nodo  $r_{agg}$  es menor que el límite  $R_{max}$  predefinido para un campo receptivo como se ilustra en la Figura 9.

5 En lugar de agrupar todos los nodos de regla que están más cerca de un nodo de regla  $r_j$  que el nodo más cercano de la otra clase, es posible mantener el nodo más cercano del conjunto de agrupación de la otra clase fuera del procedimiento de agrupación - como un nodo separado- una «protección», tal como se muestra en las Figuras 10, 11, 12 y 13, evitando así una clasificación errónea futura en el área limítrofe entre las dos clases.

15 La agrupación de nodos de regla asignados espacialmente se describe con referencia a las Figuras 10 y 11. En referencia a la Figura 10, se han seleccionado y ordenado dos conjuntos distintos de nodos de regla para la agrupación, mostrados generalmente como 100 y 102, respectivamente. En referencia a la Figura 11, el nodo de regla 104 se clasifica como una protección y no se agrupa. Los nodos de regla restantes que quedan en el conjunto 100 se agrupan en una nueva regla 106. De forma similar, el nodo de regla 108 no se agrupa con los nodos de regla restantes que quedan en el conjunto 102 mostrado en 110. Según la invención, el umbral de sensibilidad y el umbral de error de los nodos de regla 104 y 108 se reducen para aumentar el umbral de activación de estos nodos, dando como resultado nodos agrupados 106 y 110 que son activados con preferencia a los nodos de protección 104 y 108.

Las Figuras 12 y 13 ilustran el mismo proceso de agrupación que se describe en las Figuras 10 y 11 con la excepción de que los nodos de regla se asignan linealmente en lugar de asignarse espacialmente, como se muestran en las Figuras 10 y 11.

25 La agrupación según la invención se lleva a cabo preferiblemente después de que se presente un cierto número de ejemplos (parámetro  $N_{agg}$ ) sobre todo el conjunto del nodos de regla.

En otra forma preferida, los nodos del sistema  $r_1$  que no están agrupados pueden reducir su umbral de sensibilidad  $S_1$  y aumentar su radio  $R_1$  con un coeficiente pequeño para que estos nodos tengan más oportunidades de ganar la competición de activación para los próximos ejemplos de datos de entrada y competir con el resto de nodos.

30 Mediante la creación y agrupación consecutiva de nodos, el módulo de red neuronal 22 preferido puede ajustarse a lo largo del tiempo para cambiar en el flujo de datos y al mismo tiempo preservar sus capacidades de generalización.

Después de un tiempo determinado (cuando una cantidad determinada de ejemplos de datos se han presentado en el sistema) algunas neuronas y conexiones pueden ser podadas. Pueden aplicarse diferentes reglas de podado para un podado exitoso de nodos y conexiones innecesarias. Uno de ellos se da a continuación:

35 SI  $(\text{Edad}(r_j) > \text{MAYOR})$  Y (la activación total  $\text{TA}(r_j)$  es menor que un parámetro de podado  $\text{Pr}$  veces  $\text{Edad}(r_j)$ ) ENTONCES nodo de regla de podado  $r_j$ ,

donde  $\text{Edad}(r_j)$  se calcula como el número de ejemplos que se han presentado al sistema después de que  $r_j$  se haya creado primero; MAYOR es un límite de «edad» predefinido;  $\text{Pr}$  es un parámetro de podado en el intervalo de  $[0, 1]$ , y la activación total  $\text{TA}(r_j)$  se calcula como el número de ejemplos para los que  $r_j$  ha sido el nodo ganador correcto (o entre los nodos ganadores  $m$  en el modo de funcionamiento  $m$  de  $n$ ).

40 La regla de podado mencionada anteriormente requiere que los conceptos difusos de MAYOR, ALTO, etc. estén definidos de antemano. Como caso parcial, puede utilizarse un valor nítido, p. ej. un nodo es MAYOR si ha existido durante el desarrollo de un sistema de más de  $p$  ejemplos. La regla de podado y la forma en la que los valores para los parámetros de podado están definidos, depende de la tarea de la aplicación.

45 Los parámetros de cada nodo de regla pueden mantenerse fijos durante todo el funcionamiento del sistema, o pueden adaptarse u optimizarse según los datos de entrada. La adaptación puede conseguirse a través del análisis del comportamiento del sistema a través de una conexión de retroalimentación desde los módulos con un nivel más alto. También pueden aplicarse algoritmos genéticos y técnicas de programación evolutiva para optimizar los parámetros estructurales y funcionales del módulo de red neuronal 22.

- 5 En una forma adicional preferida de la invención, se desarrolla una población de s sistemas simultáneamente, teniendo cada sistema diferentes valores de parámetros. Una "ventana" determinada de datos entrantes se mantiene y actualiza para probar la capacidad del sistema desarrollado individualmente basado en la función de la capacidad del error cuadrático medio. Se selecciona y «multiplica» el mejor sistema a través de pequeñas derivaciones de los valores del parámetro creando así la siguiente generación población. El proceso es continuo de forma ilimitada en el tiempo.
- En cuanto a la implementación, el método y el sistema en una memoria informática, cuando se crean, los nuevos nodos de regla se asignan espacial o linealmente en la memoria del ordenador y la distribución real de nodos podría seguir una de varias estrategias diferentes como se describe a continuación.
- 10 Una de estas estrategias, como se muestra en la Figura 14, podría ser una estrategia de asignación consecutiva simple. Cada nodo de regla recién creado se asigna en la memoria del ordenador junto al nodo de regla anterior y al nodo de regla siguiente, de forma lineal, representando un orden de tiempo.
- 15 Otra estrategia posible podría ser una ubicación preagrupada como se muestra en la Figura 15. Para cada nodo difuso de salida, hay una ubicación predefinida en la memoria del ordenador donde se encuentran los nodos de regla que soportan este concepto predefinido. En el centro de esta zona se sitúan los nodos que soportan por completo este concepto. La ubicación de cada nodo de regla nuevo se define en función del error de salida difuso y de la similitud con otros nodos. En una estrategia de inserción de nodo activado más próximo, se coloca un nuevo nodo de regla más próximo al nodo altamente activado, cuya activación es todavía menor que su umbral de sensibilidad. El lado (izquierdo o derecho) donde se inserta el nuevo nodo se define por la activación más alta de los dos nodos vecinos.
- 20 Otra estrategia podría incluir la ubicación preagrupada descrita anteriormente incluyendo además conexiones de retroalimentación temporal entre las diferentes partes de los lugares de la memoria informática, como se muestra en la Figura 16. Se establecen nuevas conexiones que enlazan nodos de reglas activados consecutivamente mediante el uso de la memoria a corto plazo y los enlaces establecidos a través de la matriz de ponderación W3. Esto permitirá al módulo de red neuronal 22 repetir una secuencia de puntos de datos que empieza desde un punto determinado y no necesariamente desde el principio.
- 25 Otra estrategia podría incluir el rasgo adicional de que se establezcan nuevas conexiones entre los nodos de regla de diferentes módulos de red neuronales que se activan simultáneamente, como se muestra en la Figura 17. Este rasgo permitiría al sistema aprender una correlación entre variables conceptualmente diferentes, por ejemplo la correlación entre el sonido de la voz y el movimiento de los labios.
- 30 Un importante rasgo del sistema de aprendizaje adaptativo y el método descrito anteriormente es que el aprendizaje implica la adaptación de elementos locales. Solo un nodo de regla (o un pequeño número, si el sistema funciona en modo m de n) se actualizará para cada ejemplo de datos o, de forma alternativa, solo se creará un nodo de regla. Esto acelera el procedimiento de aprendizaje, en concreto si se utilizan las funciones de activación lineal en los módulos de red neuronal. Otra ventaja es que el aprendizaje de un nuevo ejemplo de datos no provoca que se olviden ejemplos antiguos. Además, pueden añadirse nuevas variables de entrada y de salida durante el proceso de aprendizaje, haciendo así que el sistema de aprendizaje adaptativo sea más flexible para alojar información nueva sin despreciar la información ya aprendida.
- 35 La utilización de las funciones de pertenencia, los grados de pertenencia y la distancia difusa, local y normalizada permiten que el sistema gestione los valores de atributo ausentes. En estos casos, los grados de pertenencia de todas las funciones de pertenencia será de 0,5 indicando que el valor, si existió, puede pertenecer igualmente a ellos. La preferencia, en términos de a qué funciones de pertenencia difusas puede pertenecer el valor ausente, también puede representarse a través de la asignación de grados de pertenencia apropiados.
- 40 Los algoritmos de aprendizaje supervisados y preferidos de la invención permiten que el sistema evolucione y aprenda continuamente cuando un nuevo par de datos de entrada-salida queda disponible. Esto se conoce como modo activo de aprendizaje. En otro modo, el aprendizaje pasivo, el aprendizaje se lleva a cabo cuando no se presenta ningún diseño de entrada. El aprendizaje pasivo puede llevarse a cabo después de un aprendizaje inicial. Durante el aprendizaje pasivo, se utilizan las conexiones existentes que almacenan patrones de entrada alimentados previamente como «eco» para reiterar el proceso de aprendizaje. Este tipo de aprendizaje podría aplicarse en caso de un breve tiempo de presentación de los datos, cuando solo una pequeña parte de los datos se aprende en un modo en línea de una sola pasada y, a continuación, la formación se perfecciona a través del método de aprendizaje por eco. Los patrones almacenados en las ponderaciones de conexión W1 pueden utilizarse como vectores para la mejora del sistema con los patrones W2 que indican cuáles van a ser los resultados.
- 45 Los algoritmos de aprendizaje supervisado preferidos se describen a continuación. Cada algoritmo de aprendizaje difiere en las fórmulas de ajuste de ponderación.
- 50

El primer algoritmo de aprendizaje se muestra a continuación:

Establecer valores iniciales para los parámetros del sistema: número de funciones de pertenencia; umbrales de sensibilidad iniciales (fallo  $S_j=0,9$ ); umbral de error E; parámetro de agrupación Nagg - se lleva a cabo un número de ejemplos consecutivos después de cada agrupación; parámetros de podado MAYOR y Pr; un valor para m (en modo m-de-n); límite de radio máximo Rmax; umbrales  $T_1$  y  $T_2$  para la extracción de reglas.

Establecer el primer nodo de regla ro para memorizar el primer ejemplo (x,y):

$$W1(r_0)=x_r, \text{ y } W2(r_0)=y_r; \quad (10)$$

Bucle sobre presentaciones de nuevos pares de entrada-salida (x,y)

{

10 Evaluar la distancia difusa normalizada D entre  $x_1$  y las conexiones W1 de nodo de regla existentes (fórmulas (1))

Calcular la activación A1 de la capa de nodos de regla. Encontrar el nodo de regla  $r_k$  más cercano (o el nodo de regla m más cercano en caso de modo m-de-n) al vector  $x_r$  de entrada difuso para el que  $A1(r_k) \geq S_k$  (umbral de sensibilidad para el nodo  $r_k$ ),

si no hay tal nodo, crear un nodo de regla nuevo para  $(x_r, y_r)$

15 si no

Encontrar la activación de la capa de salida difusa  $A2=W2.A1(1-D(W1, x_r))$  y el error de salida normalizado

$$\text{Err} = ||y - y'| / \text{Nout.}$$

si  $\text{Err} > E$

20 crear un nuevo nodo de regla para alojar el ejemplo actual  $(x_r, y_r)$

si no

Actualizar  $W1(r_k)$  y  $W2(r_k)$  según (2) y (3) (en caso de sistema m-de-n actualizar todos los nodos de regla m con la activación A1 más alta).

Aplicar el procedimiento de agrupación de nodos de regla después de que se presente cada grupo de ejemplos Nagg

Actualizar los valores para los parámetros  $S_k$ ,  $R_k$ ,  $\text{Age}(r_k)$ ,  $\text{TA}(r_k)$  del nodo de regla  $r_k$ .

25 Nodos de regla de podado si es necesario, como están definidos por los parámetros de podado.

Extraer reglas de los nodos de regla (

}

Una versión modificada del algoritmo anterior es cuando el número de los nodos de regla ganadores se selecciona de modo que no sean 1, sino  $m > 1$  (por defecto  $m=3$ ). Este modo se llama «m-de-n».

30 El segundo algoritmo de aprendizaje se diferencia del primer algoritmo de aprendizaje en la fórmula de ajuste de la ponderación para W2 como se muestra a continuación:

$$W2(r_j^{(2)}) = W2(r_j^{(2)}) + 1_j \cdot (A2 - y_r) \cdot A1(r_j^{(2)}) \quad (11)$$

35 Esto significa que después de la primera propagación del vector de entrada y el cálculo del error Err, si las ponderaciones se van a ajustar, las ponderaciones W1 se ajustan primero usando la ecuación (2) anterior y después el vector de entrada x se propaga nuevamente a través del nodo de regla  $r_j$  ya ajustado hasta su posición  $r_j^{(2)}$  en el espacio de entrada, se calcula un nuevo error  $\text{Err} = (A2 - y_r)$  y después se ajustan las ponderaciones W2 del nodo de regla  $r_j$ . Este ajuste de ponderación es más preciso que el ajuste del primer algoritmo que puede marcar la diferencia en el aprendizaje de secuencias cortas, pero para secuencias más largas puede que no manifieste ninguna diferencia en los resultados obtenidos a través del primer algoritmo que es más rápido y más simple.

40 Además del aprendizaje supervisado, el sistema también se dispone, preferiblemente, para llevar a cabo un aprendizaje no supervisado en el que se asume que no hay valores de salida no deseados disponibles y el sistema desarrolla sus nodos de regla a partir del espacio de entrada. Una asignación de nodo se basa solo en los umbrales de sensibilidad  $S_j$  y en las tasas de aprendizaje  $l_j$ . Si una nueva unidad de datos d activa un nodo (o nodos) de regla determinado por encima del nivel de su parámetro  $S_j$ , entonces este nodo de regla (o el que tenga la activación más alta) se ajusta para alojar la nueva unidad de datos según la ecuación (2) anterior o, de forma alternativa, se crea un nuevo nodo de regla. El método de aprendizaje no supervisado de la invención se basa en las etapas descritas anteriormente como parte del método de aprendizaje supervisado cuando solo el vector de entrada x está disponible para el dato d de entrada actual.

50 Tanto el método supervisado de aprendizaje del sistema como el no supervisado se basan en los mismos principios de construcción de la capa W1 de conexiones. Podría aplicarse cualquier tipo de método en un sistema en desarrollo, de manera que si existen valores de salida conocidos, el sistema utilizará un método de aprendizaje conocido, de lo contrario aplicará el método de aprendizaje no supervisado en la misma estructura. Por ejemplo, después de haberse desarrollado de una manera no supervisada, un módulo de red neuronal de una palabra hablada de datos de entrada,

el sistema puede entonces utilizar datos etiquetados con las etiquetas de fonema apropiadas para continuar con el proceso de aprendizaje de este sistema, ahora de forma supervisada.

5 El sistema preferido también puede llevar a cabo su aprendizaje a partir de advertencias de salida o a través de aprendizaje de refuerzo, además del aprendizaje supervisado o no supervisado. Este es el caso cuando los valores de salida deseados y exactos no llegan a ser conocidos con el propósito de ajustar las ponderaciones de conexión W2. En lugar de ello, las advertencias difusas F proporcionadas en etiquetas lingüísticas difusas que se utilizan en el espacio de salida difuso pueden ser proporcionadas como retroalimentación, p. ej. «un valor de salida bajo es el deseado» mientras que el valor de salida producido por el sistema es «muy bajo». El sistema calcula entonces el error de salida difuso  $Err_f = A_2 - F$  y después ajusta las conexiones W2 mediante la fórmula (3).

10 El sistema preferido también puede producir inferencia y tener la capacidad de generalizar sobre nuevos datos de entrada. El método de inferencia es parte del método de aprendizaje cuando solo el vector x de entrada se propaga a través del sistema. El sistema calcula el ganador, o m ganadores, como se muestra a continuación: un nodo de regla r ganador para un vector x de entrada es el nodo con: (i) la activación más alta  $A_1(r)$  entre otros nodos de regla para los que, (ii):

15 
$$D(x, W_1(r)) \leq R_r, \tag{12}$$

donde:  $D(x, W_1(r))$  es la distancia normalizada difusa ente x y  $W_1(r)$ ;  $R_r$  es el radio del nodo de regla r. Si no existe ningún nodo de regla que satisfaga la condición (ii) para el vector x de entrada actual, solo se utilizará la condición (i) para seleccionar al ganador.

20 En una forma preferida de la invención con referencia a la Figura 3 anterior, una capa temporal 60 de nodos temporales 62 captura dependencias temporales entre ejemplos de datos consecutivos. Si el nodo de regla ganador del momento (t-1), al que se asocia el vector de datos de entrada del momento (t-1), es  $r_{max}^{(t-1)}$  y el nodo ganador del momento t es  $r_{max}^{(t)}$ , entonces se establece un enlace entre los dos nodos como se muestra a continuación:

25 
$$W_3(r_{max}^{(t-1)}, r_{max}^{(t)}) = W_3(r_{max}^{(t-1)}, r_{max}^{(t)}) + I_3 \cdot A_1(r_{max}^{(t-1)}) \cdot A_1(r_{max}^{(t)}) \tag{13}$$

Donde  $A_1(r^{(t)})$  indica la activación de un nodo de regla r en un momento de tiempo (t) y  $I_3$  define el grado al que el módulo de red neuronal 22 asocia los enlaces entre los nodos de regla que incluyen ejemplos de datos consecutivos. Si  $I_3 = 0$ , no se aprenden asociaciones temporales en la estructura y la capa temporal 60 se retira de forma efectiva del módulo de red neuronal 22.

30 Las asociaciones temporales aprendidas podrían utilizarse para soportar la activación de nodos de regla basados en similitudes de patrones temporales. En la presente memoria, las dependencias temporales se aprenden estableciendo enlaces estructurales. Estas dependencias pueden investigarse y potenciarse más a través del análisis sináptico, en el nivel de la memoria sináptica, en lugar de a través del análisis de activación neuronal en el nivel de comportamiento. La relación similitud espacial/correlación temporal puede equilibrarse para aplicaciones diferentes mediante dos parámetros  $S_s$  y  $T_c$ , de manera que la activación de un nodo de regla r para un nuevo ejemplo de datos  $d=(x,y)$  se define mediante las siguientes operaciones vectoriales:

35 
$$A_1(r) = | 1 - S_s \cdot D(W_1(r), x_f) + T_c \cdot W_3(r_{max}^{(t-1)}, r) |_{[0,1]} \tag{14}$$

40 donde  $|_{[0,1]}$  es una operación limitada en el intervalo [0,1], y  $r_{max}^{(t-1)}$  es la neurona ganadora en el momento de tiempo anterior. Aquí, puede darse una importancia mayor a las conexiones temporales para tolerar un distancia mayor en el tiempo para los vectores de entrada dependientes del tiempo. Si  $T_c=0$ , entonces los enlaces temporales son excluidos del funcionamiento del sistema.

45 El sistema se adapta para aprender una función caótica compleja mediante un desarrollo en línea a partir de la propagación de datos de una sola pasada. El sistema también se adapta para aprender series de tiempo que cambien sus dinámicas a lo largo del tiempo y que nunca repitan los mismos patrones. Los procesos de series de tiempo con dinámicas cambiantes pueden ser de orígenes diferentes, por ejemplo biológico, medioambiental, control de procesos

industriales, financieros. El sistema también podría utilizarse para formación y pruebas fuera de línea similares a otras técnicas de redes neuronales estándares.

Un ejemplo de aprendizaje de una función caótica compleja se describe con referencia a las Figuras 18A y 18B. Aquí, el sistema se utiliza con los datos de series de tiempo caóticas Mackey-Glass generados mediante la ecuación diferencial de tiempo retardado Mackey-Glass:

$$\frac{d(x)}{d(t)} = \frac{ax(t-\tau)}{1+x^{10}(t-\tau)} - b x(t) \quad (15)$$

Esta serie se comporta como una serie temporal caótica para algunos valores de los parámetros  $x(0)$  y  $\tau$ . Aquí,  $x(0) = 1,2$ ,  $\tau = 17$ ,  $a = 0,2$ ,  $b = 0,1$  y  $x(t) = 0$  para  $t < 0$ . Los datos de entrada-salida para desarrollar el sistema de datos de series temporales Mackey-Glass tiene un vector de entrada  $[x(t), x(t-6), (t-12), x(t-18)]$  y el vector de salida es  $[x(t+6)]$ . La tarea es predecir los valores futuros  $x(t+6)$  de cuatro puntos espaciados en seis intervalos de tiempo en el pasado.

Por ejemplo, los valores para los parámetros del sistema inicialmente se establecen como aparece a continuación:

$S=0,92$ ,  $E=0,08$ ,  $1=0,005$ , el umbral de agrupación es  $R_{max}=0,15$  y el umbral de extracción de reglas  $T_1=T_2=0,1$ . La agrupación se lleva a cabo después de que se presente cada grupo de ejemplos  $N_{agg}=50$  consecutivo.

Los resultados experimentales del desarrollo en línea del sistema se muestran en las Figuras 18A y 18B. En concreto se muestran los valores deseados frente a los predichos seis pasos por delante a través del aprendizaje en línea de una sola pasada, el error absoluto, el RMSE en línea local (LRMSE) y el local en línea LNDEI (LNDEI) a lo largo del tiempo como se describe más abajo, el número de nodos de regla creados y agrupados en el tiempo y un gráfico de los vectores de datos de entrada mostrados como círculos y los nodos de regla desarrollados, las ponderaciones de conexión  $W1$  mostradas como cruces, proyectados en el espacio de entrada bidimensional de las dos primeras variables de entrada  $x(t)$  y  $x(t-6)$ . Se puede observar a partir de las Figuras 18A y 18B que el número de nodos de regla se optimiza después de cada 50 ejemplos presentados. Los nodos de regla se sitúan en los espacios del problema de entrada y salida para que representen centros de agrupación de los datos de entrada que tienen valores de salida similares sujetos a una diferencia de error  $E$ .

El error de generalización de un módulo de red neuronal en un nuevo vector (o vectores) de entrada siguiente a partir del flujo de entrada calculado mediante el proceso de desarrollo se llama error de generalización en línea local. El error de generalización en línea local en el momento  $t$  por ejemplo, cuando el vector de entrada es  $x(t)$  y el calculado por el vector de salida del módulo desarrollado es  $y(t)$ , se expresa como  $Err(t)=y(t) - y(t)$ . El error cuadrático medio en línea local y el índice de error no dimensional en línea local LNDEI(t) pueden calcularse en cada momento de tiempo  $t$  como:

$$LRMSE(t)=\sqrt{\sum_{i=1,2,\dots,t}(Err(i)^2)/t}; LNDEI(t)=LRMSE(t)/std(y(1):y(t)) \quad (16)$$

donde  $std(y(l):y(t))$  es la desviación estándar de los puntos de datos de salida desde 1 hasta  $t$ .

Para los valores escogidos de los parámetros, había 16 nodos de regla desarrollados, cada uno de los cuales representados como una regla. Tres de estas reglas se muestran en la Figura 19, a saber: Regla 1, Regla 2 y Regla 16. Estas reglas y el mecanismo de inferencia del sistema definen un sistema que es equivalente a la ecuación (16) anterior en cuanto a las variables de entrada y de salida elegidas sujetas al error calculado.

A medida que se introducen más datos de entrada después de cierto momento de tiempo, las LRMSE y LNDEI convergen a valores constantes sujetos a un pequeño error, en el ejemplo de la Figura 19 - LRMSE = 0,043, LNDEI = 0,191. En términos generales, en el caso de un espacio del problema compacto y limitado, el error puede hacerse suficientemente pequeño sujeto a una selección apropiada de los valores de parámetros para el sistema y el flujo de datos inicial. En el experimento anterior, la tolerancia al error seleccionada era comparativamente alta, pero el sistema resultante era compacto. Si el umbral de error  $E$  seleccionado fuera más pequeño (p. ej. 0,05 o 0,02), se habrían desarrollado más nodos de regla y se podría haber alcanzado una precisión de predicción mejor. Los módulos de red neuronal diferentes tienen valores de parámetros óptimos diferentes que dependen de la tarea (p. ej. predicción, clasificación de series temporales).

5 Se ha llevado a cabo otro ejemplo en el que el sistema se ha utilizado para la enseñanza y la evaluación fuera de línea. Los siguientes valores de parámetro se establecen inicialmente antes de que el sistema esté desarrollado, a saber: MF=5, S=0,92, E=0,02, m=3, l=0,005. El sistema se desarrolla en los primeros 500 ejemplos de datos de la misma serie temporal Mackey-Glass del ejemplo anterior para una sola pasada de aprendizaje. La Figura 20 muestra los valores en línea predichos frente a los deseados de la serie temporal. Después de que el sistema esté desarrollado, se prueba para una generalización global de los segundos 500 ejemplos. La Figura 21 muestra los valores deseados frente a los valores predichos por el sistema en un modo fuera de línea.

En un caso general, el error cuadrático medio (RMSE) de generalización global y el índice de error no dimensional se evalúan sobre un conjunto de p nuevos ejemplos del espacio del problema como se muestra a continuación:

10 
$$RMSE = \sqrt{\sum_{i=1,2,\dots,p} [y_i - \hat{y}_i]^2} / p; NDEI = RMSE / \text{std}(1:p), \quad (17)$$

donde std (1:p) es la desviación estándar de los datos de 1 a p en el conjunto de pruebas. Los datos evaluados en este ejemplo RMSE es 0,01 y el NDEI es 0,046. Después de haber desarrollado el sistema en una parte pequeña pero representativa del total del espacio del problema, su error de generalización global está suficientemente minimizado.

15 El sistema también se evalúa para una prueba de error en línea sobre los datos de prueba mientras se lleva a cabo una formación adicional sobre él. El error de evaluación local en línea es ligeramente menor.

20 En una aplicación experimental, el sistema preferido puede utilizarse para aprendizaje no supervisado permanente a partir de un flujo continuo de datos nuevos. Este es el caso del aprendizaje de sonidos nuevos en idiomas nuevos o nuevos acentos nunca antes escuchados. Se describe un experimento con referencia a las Figuras 22 y 23. El sistema se presenta con los rasgos acústicos de una palabra hablada inglesa «eight» con una representación fonémica de /silencio//ei//t//silencio/. En los resultados experimentales mostrados en la Figura 22, se utilizan tres retardos de tiempo de coeficiente 26 en la escala de Mel tomados de una ventana de 12 ms de la señal de voz, con una superposición del 50%, para formar vectores de entrada de 78 elementos. Los vectores de entrada se indican en función del tiempo como se muestra en la Figura 23.

25 Cada nuevo vector de entrada de la palabra hablada se asocia a un nodo de regla existente que se modifica para alojar este dato, o se crea un nuevo nodo de regla. Los nodos de regla se agrupan en intervalos regulares que reducen el número de nodos situados en los centros de agrupación de datos. Después de que se haya presentado la palabra completa, los nodos de regla agrupados representan los centros de agrupación del fonema anticipado sin que el concepto de fonema se introduzca en el sistema.

30 Las Figuras 22 y 23 muestran claramente que tres nodos de regla se desarrollaron después de la agrupación que representa los datos de entrada. Por ejemplo, las unidades 0 a 53 indicadas como 120 y las unidades 96 a 170 indicadas como 122 se asignan a un nodo de regla 1 que representa el fonema /silencio/. Las unidades 56 a 78 indicadas como 124 se asignan a un nodo de regla 2 que representa el fonema /ei/. Las unidades 85 a 91 indicadas como 126 se asignan a un nodo de regla 3 que representa el fonema /t/. Las unidades restantes representan estados transitorios. Por ejemplo, las unidades 54 a 55 representan la transición entre /silencio/ y /ei/. Las unidades 79 a 84 representan la transición entre /ei/ y /t/. Las unidades 92 a 96 representan la transición entre /t/ y /silencio/. Estas unidades se asignan a alguno de los nodos de regla más cercanos en el espacio de entrada. Si se utiliza un umbral de sensibilidad mayor, esto habría tenido como resultado nodos de regla adicionales desarrollados para representar estos sonidos transitorios cortos.

40 Cuando se presentan otras pronunciaciones de la palabra «eight» u otras palabras al sistema no supervisado, el sistema perfecciona las regiones del fonema y los nodos regla del fonema o crea nuevos nodos de regla. El método de aprendizaje no supervisado descrito anteriormente permite experimentar con diferentes estrategias de aprendizaje, a saber: sensibilidad aumentada especialmente a lo largo del tiempo, sensibilidad disminuida a lo largo del tiempo y utilizando el olvido en el proceso de aprendizaje. También permite la experimentación con varios idiomas en un sistema plurilingüe.

50 En un ajuste experimental, un sistema se desarrolla tanto en palabras habladas en inglés de Nueva Zelanda como en palabras habladas en Maorí. Algunos de los nodos de regla de fonema desarrollados son compartidos por la representación acústica de los idiomas como se ilustra en la Figura 24, donde se representan los nodos de regla desarrollados así como una trayectoria de la palabra hablada «zoo» en el espacio bidimensional de los primeros dos componentes del espacio acústico de entrada. Los nodos de regla del sistema desarrollado representan una representación compacta del espacio acústico de los dos idiomas presentados al sistema. Se puede enseñar al

sistema de forma continua con más palabras de los dos o más idiomas, perfeccionando así la representación del espacio acústico con el uso del principio de sonidos (fonemas) compartidos.

5 El sistema ha sido sometido a un experimento que implica la tarea de predicción de series temporales en línea de los datos Mackey Glass. En la presente memoria se utiliza el formato de referencia estándar CMU de la serie temporal. Los datos se generan con  $\tau = 17$  utilizando un método Runge-Kutta de segunda orden con un tamaño de paso de 0,1, de cuatro entradas, a saber:  $x(t)$ ,  $x(t-6)$ ,  $x(t-12)$  y  $x(t-18)$  y una salida, concretamente  $x(t+85)$ . El dato de enseñanza es de  $t=200$  a  $t=3200$  mientras que el dato de prueba son de  $t=5000$  a  $t=5500$ . Se utilizarán los 3000 conjuntos de datos de enseñanza para desarrollar dos tipos de módulos de red neuronal.

10 Para el primer y segundo algoritmo de aprendizaje descritos anteriormente, se escogieron los siguientes valores iniciales de parámetros:  $MF=3$ ,  $S=0,7$ ,  $E=0,02$ ,  $m=3$ ,  $l=0,02$ ,  $R_{max}=0,2$ ,  $N_{agg}=100$ . El número de los centros y de los LNDEI en línea locales se calcula y compara con los resultados para el modelo RAN como se describe en Platt, J «Un recurso que asigna redes para la interpolación de funciones», Neural Computation 3.213-225 (1991) y modificaciones.

15 Los resultados se muestran en la Figura 25. Las dos modificaciones del sistema dan como resultado un error en línea más pequeño que los otros métodos y número razonable de nodos de regla. Los dos algoritmos de aprendizaje se muestran como System-su and System-dp.

20 Como el sistema utiliza preferiblemente ecuaciones lineales para calcular la activación de los nodos de regla, en lugar de las funciones de funciones Gaussianas y funciones exponenciales como en el modelo RAN, el presente procedimiento de aprendizaje del sistema es más rápido que el procedimiento de aprendizaje del modelo RAN y sus modificaciones. El sistema también produce una mejor generalización en línea, lo cual es resultado de una asignación de nodo más precisa durante el proceso de aprendizaje. Esto se suma a las características de representación de conocimiento ventajosas del sistema preferido que incluye el agrupamiento del espacio de entrada y la extracción de reglas y la inserción de reglas.

25 El sistema también se ha sometido a otros experimentos relacionados con una tarea de clasificación en un caso de estudio de datos de dígitos hablados. La tarea es el reconocimiento de las pronunciaciones independientes de hablantes de dígitos en inglés de la base de datos del corpus Otago (<http://kel.otago.ac.nz/hyspeech/corpus/>). Se utilizaron diecisiete hablantes (12 hombres y 5 mujeres) para la enseñanza y se utilizaron otros 17 hablantes (12 hombres y 5 mujeres) para la prueba fuera de línea. Cada hablante pronuncia 30 instancias de dígitos en inglés durante una sesión de grabación en una habitación silenciosa, lo que produce datos limpios, con un total de 510 unidades de habla de enseñanza y 510 unidades de habla de prueba. Se utilizan ocho coeficientes cepstrales en la escala de frecuencia de Mel (MFSCC) y logaritmo de energía como rasgos acústicos. Para evaluar el rendimiento del sistema en esta aplicación, se realiza una comparación con la Cuantificación Vectorial Lineal (LVQ). Se utiliza el discurso de enseñanza limpio para enseñar tanto a la LVQ como al presente sistema. Se introduce ruido de oficina en los datos del habla de prueba para evaluar el comportamiento de los sistemas de reconocimiento en un entorno ruidoso, con una relación señal-ruido de 10dB.

35 La precisión de la prueba fuera de línea para la clasificación del modelo LVQ y el sistema actual, así como la precisión de la prueba en línea local para el sistema, se evalúan y se muestran en la Figura 26.

40 El modelo LVQ tiene los siguientes valores de parámetro, concretamente vectores de código de libro 396, variaciones de formación 15840. El presente sistema tiene los siguientes valores de parámetro de una repetición de formación, 3 MF, 157 nodos de regla, valores iniciales para  $S=0,9$ ,  $E=0,1$ ,  $l=0,01$ . El radio máximo es  $R_{max}=0,2$  y el número de ejemplos para la agrupación  $N_{agg}=100$ .

Los resultados muestran que el presente sistema con aprendizaje fuera de línea y prueba sobre nuevos datos actúa mucho mejor que el método LVQ como se muestra en la Figura 26. Como el sistema actual permite la enseñanza continua sobre nuevos datos, pruebas adicionales así como la enseñanza del sistema sobre los datos de prueba en un modo en línea, ofrece una mejora significativa de la precisión.

45 El sistema también se ha sometido a otro experimento relacionado con la tarea de clasificación de datos de un caso de estudio de bioinformática obtenidos del repositorio de bases de datos de aprendizaje automático de la Universidad de California en Irvine. Este contiene secuencias de genes de unión de empalme de primates para la identificación de los límites del sitio de empalme dentro de estas secuencias. En los eucariotas, los genes que codifican las proteínas se dividen en regiones codificantes (exones) y regiones no codificantes (intrones) de la secuencia de ADN en los límites definidos, los llamados sitios de empalme. El conjunto de datos consiste en 3190 secuencias de ADN con una longitud de 60 nucleótidos y se clasifican ya sea como un límite exón-intrón (EI), un límite intrón-exón (IE) y un sitio de no empalme (N). El sistema utiliza 2 MF y un esquema de codificación de cuatro bits para las bases.

50

Después de formar el sistema con datos existentes, el sistema es capaz de identificar posibles sitios de empalme dentro de nuevas secuencias. Usando una ventana deslizante de 60 bases para cubrir toda la secuencia en examen, los límites se identifican como EI, IE o N. Se da una puntuación a cada límite identificado que representa la probabilidad de que el límite identificado se haya identificado correctamente. Se puede enseñar al sistema continuamente con nuevas secuencias de datos conocidos, mejorando así su rendimiento con secuencias de datos desconocidos. En cualquier momento del funcionamiento del sistema, puede extraerse conocimiento de él en forma de reglas semánticamente significativas que describen relaciones biológicas importantes. Algunas de las reglas extraídas con un umbral de extracción de regla  $T1 = T2 = 0,7$  se simplifican más, se formatean y se presentan de una manera que puede ser interpretada por el usuario, como se muestra en la Figura 27. El uso de diferentes umbrales de extracción de reglas permitiría la extracción de diferentes conjuntos de reglas que tengan diferentes niveles de abstracción, permitiendo así una mejor comprensión de las secuencias de genes.

El sistema también se ha sometido a otro experimento relacionado con la tarea de clasificación de datos de un caso de estudio de bioinformática que consiste en un conjunto de datos de 72 ejemplos de clasificación para la enfermedad del cáncer de leucemia. El conjunto de datos consiste en dos clases y un espacio de entrada más grande, valores de expresión de 7.129 genes monitorizados por chips de ADN Affymatrix (Golub y col). Los dos tipos de leucemia son la leucemia mieloide aguda (LMA) y la leucemia linfoblástica aguda (LLA).

La tarea es doble: 1) Encontrar un conjunto de genes que distingan LMA y LLA, y 2) Construir un clasificador basado en la expresión de estos genes permitiendo la entrada de nuevos datos en el sistema una vez que estén disponibles. El sistema aloja o adapta estos datos mejorando los resultados de clasificación. El sistema se desarrolla mediante la formación de una pasada en cada ejemplo consecutivo y probándolo en el siguiente.

Durante el proceso de desarrollo en línea el sistema aprende cada ejemplo y después intenta predecir la clase del siguiente. En la presente invención, el sistema se desarrolla continuamente con nuevos ejemplos alojados, a medida que están disponibles. En cualquier momento del funcionamiento del sistema pueden extraerse reglas que explican qué genes están más estrechamente relacionados con cada una de las clases. La Figura 28 muestra dos de las reglas extraídas después de que los 72 ejemplos iniciales fueran aprendidos por el sistema. Las reglas son «locales» y cada una de ellas tiene el significado de la regla de dominación en una agrupación concreta del espacio de entrada.

El sistema en un modo de aprendizaje en línea podría utilizarse como bloques de construcción para crear sistemas de reconocimiento de voz adaptativos basados en un marco conexionista en evolución. Estos sistemas podrían adaptarse a nuevos hablantes y nuevos acentos y añadir nuevas palabras a sus diccionarios en cualquier momento de su funcionamiento.

Las aplicaciones posibles de la invención incluyen el reconocimiento de voz adaptativo en un entorno ruidoso, sistemas evolutivos de lenguaje oral adaptativo, control de proceso adaptativo, control de robots adaptativo, sistemas adaptativos basados en conocimiento para aprender información genética, agentes adaptativos en Internet, sistemas adaptativos para la toma de decisiones en línea sobre datos económicos y financieros, sistemas automáticos de conducción adaptativos que aprenden a navegar en un nuevo entorno (automóviles, helicópteros, etc.) y la clasificación de datos bioinformáticos.

Como sigue describe la invención incluidas sus formas preferidas. Las alteraciones y modificaciones, que deducirán los expertos en la técnica, se entienden incluidas dentro del alcance de la misma, tal y como se define en las reivindicaciones adjuntas.

40

## REIVINDICACIONES

1. Módulo de red neuronal implementado en un ordenador que comprende:  
 una capa de entrada (40) que comprende uno o más nodos de entrada (42) dispuestos para recibir datos de entrada;
- 5 una capa base de regla (48) que comprende uno o más nodos de regla (50), teniendo cada nodo de regla un umbral de activación mínimo, estando configurado cada nodo de regla para ser activado donde los datos de entrada satisfagan el umbral mínimo de activación del nodo de regla, en donde cada nodo de regla está definido por dos vectores de ponderaciones de conexión  $W1(r)$  y  $W2(r)$ ;
- 10 una capa de salida (56) que comprende uno o más nodos de salida (58); y
- un componente de adaptación dispuesto para agrupar dos o más nodos de regla seleccionados (90, 92, 94) en la capa base de regla basado en los datos de entrada, y para aumentar el umbral de activación mínimo de uno o más nodos de regla no seleccionados para la agrupación,
- 15 en donde el componente de adaptación se adapta para ajustar  $W1(r)$  mediante aprendizaje no supervisado basado en una medida de similitud dentro de un área local de espacio de entrada y mediante una tasa de aprendizaje que se ajusta individualmente para cada nodo de regla individual, donde cuando un dato de entrada se asocia a un nodo de regla  $r_j$ , un campo receptivo y un umbral de sensibilidad  $S_j$  del nodo de regla se ajustan según las fórmulas:

$$R_j^{(2)} = R_j^{(1)} + D(W1(r_j^{(2)}), W1(r_j^{(1)})),$$

$$R_j^{(2)} \leq R_{\max}, y$$

$$S_j^{(2)} = S_j^{(1)} - D(W1(r_j^{(2)}), W1(r_j^{(1)})), \text{ donde}$$

- 25  $R_j$  es el radio del campo receptivo,  $D$  es la distancia normalizada difusa, y  $R_{\max}$  es un conjunto de parámetros que restringen el radio máximo del campo receptivo de un nodo de regla, y donde el componente de adaptación se adapta para ajustar  $W2(r)$  basado en la tasa de aprendizaje ajustada individualmente y mediante aprendizaje supervisado basado en un error de salida o en aprendizaje de refuerzo basado en una advertencia de salida.

2. Módulo de red neuronal según la reivindicación 1, en donde los parámetros del umbral de activación de cada nodo de regla activado por datos de entrada se ajustan en función de los datos de entrada o de los datos de salida deseados.
3. Módulo de red neuronal según la reivindicación 1 o la reivindicación 2, en donde cada nodo de regla se asigna a una magnitud de activación cuando se activan mediante un dato de entrada.
- 35 4. Módulo de red neuronal según la reivindicación 3, en donde el componente de adaptación se configura para agrupar dos o más nodos de regla en función de la magnitud de activación cuando se activan mediante datos de entrada, se configura para insertar nuevos nodos de regla dentro de la capa base de regla o se configura para extraer reglas de la capa base de regla.
- 40 5. Módulo de red neuronal según cualquiera de las reivindicaciones anteriores, comprendiendo además una memoria en la que se almacenan los datos de entrada, en donde los parámetros del umbral de activación de cada nodo de regla se ajustan en función de los datos de entrada almacenados o de los datos de entrada nuevos.
- 45 6. Módulo de red neuronal según cualquiera de las reivindicaciones anteriores, comprendiendo además una capa de entrada difusa (44) que comprende uno o más nodos de regla difusos (46) dispuestos para transformar valores de nodo de entrada para usar por la capa base de regla (48) o una capa de salida difusa (52) que comprende uno o más nodos de salida (54) dispuestos para transformar los datos salientes de la capa base de regla.
7. Módulo de red neuronal según la reivindicación 6, comprendiendo además una capa temporal (60) que comprende nodos temporales (62) adaptados para capturar dependencias entre ejemplos de datos consecutivos.
8. Sistema de aprendizaje adaptativo que comprende uno o más módulos de red neuronal según se reivindica en cualquiera de las reivindicaciones anteriores.
- 50 9. Método de implementación de un módulo de red neuronal que comprende los pasos de:  
 mantener en la memoria informática una capa de entrada (40) que comprende uno o más nodos de entrada (42) dispuestos para recibir datos de entrada;

mantener en la memoria informática una capa base de regla (48) que comprende uno o más nodos de regla (50), en donde cada nodo de regla está definido por dos vectores de ponderaciones de conexión  $W1(r)$  y  $W2(r)$ ; asignar un umbral de activación mínimo a cada nodo de regla en la capa base de regla, siendo activado cada nodo de regla si los datos de entrada satisfacen el umbral de activación mínimo del nodo de regla;

5 mantener en la memoria informática una capa de salida (56) que comprende uno o más nodos de salida (58); ajustar  $W1(r)$  mediante aprendizaje no supervisado basado en una medida de similitud en un área local de espacio de entrada y mediante una tasa de aprendizaje que se ajusta individualmente para cada nodo de regla individual, en donde cuando un dato de entrada se asocia a un nodo de regla  $r_j$ , un campo receptivo y un umbral de sensibilidad  $S_j$  del nodo de regla se ajustan según las fórmulas:

10

$$R_j^{(2)} = R_j^{(1)} + D(W1(r_j^{(2)}), W1(r_j^{(1)})),$$

$$R_j^{(2)} \leq R_{max,j}$$

15

$$S_j^{(2)} = S_j^{(1)} - D(W1(r_j^{(2)}), W1(r_j^{(1)})), \text{nde}$$

$R_j$  es el radio del campo receptivo,  $D$  es la distancia normalizada difusa, y

20  $R_{max}$  es un conjunto de parámetros que restringen el radio máximo del campo receptivo de un nodo de regla; ajustar  $W2(r)$  basado en la tasa de aprendizaje ajustada individualmente y mediante aprendizaje supervisado basado en un error de salida o en aprendizaje de refuerzo basado en una advertencia de salida; agrupar dos o más nodos de regla seleccionados en la capa base de regla basada en los datos de entrada; y aumentar el umbral de activación mínimo de uno o más nodos de regla no seleccionados para la agrupación.

25 10. Método de implementación de un módulo de red neuronal según la reivindicación 9, comprendiendo además el paso de ajustar los parámetros del umbral de activación de cada nodo de regla activado por datos de entrada basados en los datos de entrada o en los datos de salida deseados.

11. Método de implementación de un módulo de red neuronal según la reivindicación 9 o la reivindicación 10, comprendiendo además el paso de asignar a cada nodo de regla una magnitud de activación cuando son activados por datos de entrada.

30 12. Método de implementación de un módulo de red neuronal según la reivindicación 11, comprendiendo además el paso de agrupar dos o más nodos de regla en función de la magnitud de la activación cuando son activados por los datos de entrada, insertando nuevos nodos de regla en la capa base de regla, o extrayendo reglas de la capa base de regla.

35 13. Método de implementación de un módulo de red neuronal según cualquiera de las reivindicaciones 9 a 12, comprendiendo además los pasos de mantener datos de entrada en la memoria informática; y ajustar los parámetros del umbral de activación de cada nodo de regla en función de los datos de entrada almacenados o los datos de entrada nuevos.

40 14. Método de implementación de un módulo de red neuronal según cualquiera de las reivindicaciones 9 a 13, comprendiendo además el paso de mantener en la memoria informática una capa de entrada difusa (44) que comprende uno o más nodos de entrada difusos (46) para transformar valores de nodos de entrada para ser utilizados por la capa base de regla (48) o mantener en la memoria informática una capa de salida difusa (52) que comprende uno o más nodos de salida difusos (54) para transformar datos salientes de la capa base de regla.

45 15. Método según la reivindicación 14, en donde un cambio en las ponderaciones de conexión del nodo de regla  $r_j^{(1)}$  de  $W1(r_j^{(1)})$  y  $W2(r_j^{(1)})$  a  $W1(r_j^{(2)})$  y  $W2(r_j^{(2)})$  se lleva a cabo respectivamente según las siguientes operaciones vectoriales:

$$W1(r_j^{(2)}) = W1(r_j^{(1)}) + I_j \cdot (W1(r_j^{(1)}) - x_f) \quad \text{y}$$

$$W2(r_j^{(2)}) = W2(r_j^{(1)}) + I_j \cdot (A2 - y_f) \cdot A1(r_j^{(1)})$$

50 donde  $A2 = f_2(W2 - A1)$  es un vector de activación de los nodos de salida difusos cuando un vector de activación  $x$  se presenta y

$$A1(r_j^{(1)}) = f_1(D(W1(r_j^{(1)}), x_f))$$

es una activación del nodo de regla  $r_j^{(1)}$  donde  $D$  es una medida de distancia normalizada difusa.

16. Método según cualquiera de las reivindicaciones 9 a 15, en donde la tasa de aprendizaje del nodo de regla  $r_j$  se calcula como  $l_j = 1 / \text{Nex}(r_j)$  donde  $\text{Nex}(r_j)$  es el número de ejemplos actualmente asociados con el nodo de regla  $r_j$ .

5 17. Método según la reivindicación 14, comprendiendo además el mantenimiento en una memoria informática de una capa temporal (60) que comprende nodos temporales (62) adaptados para capturar dependencias entre ejemplos de datos consecutivos.

2

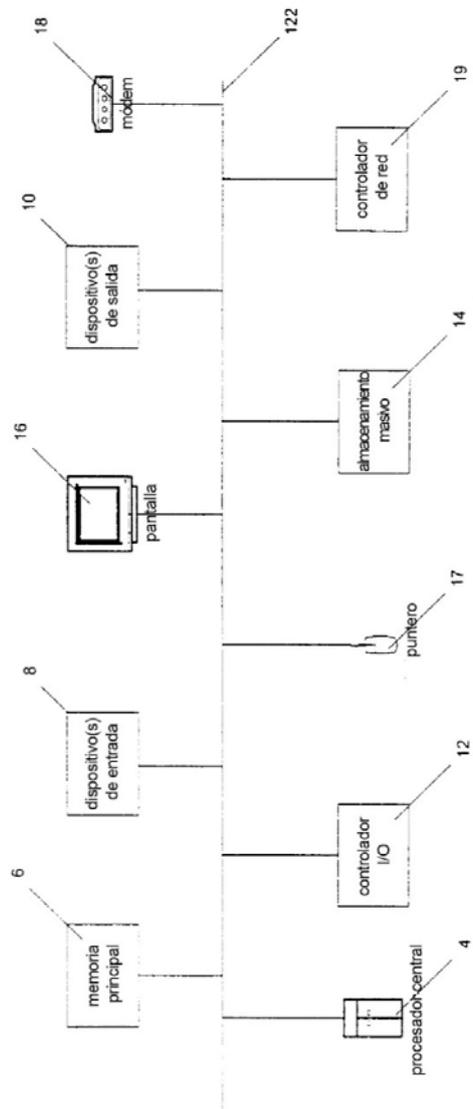


FIGURA 1

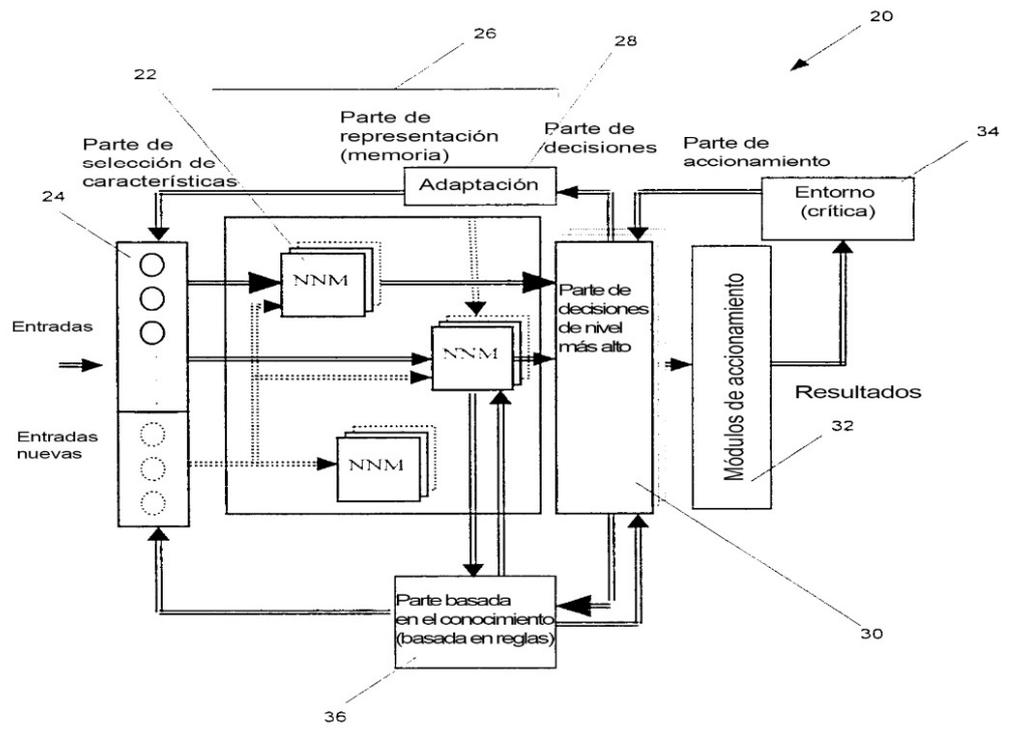


FIGURA 2

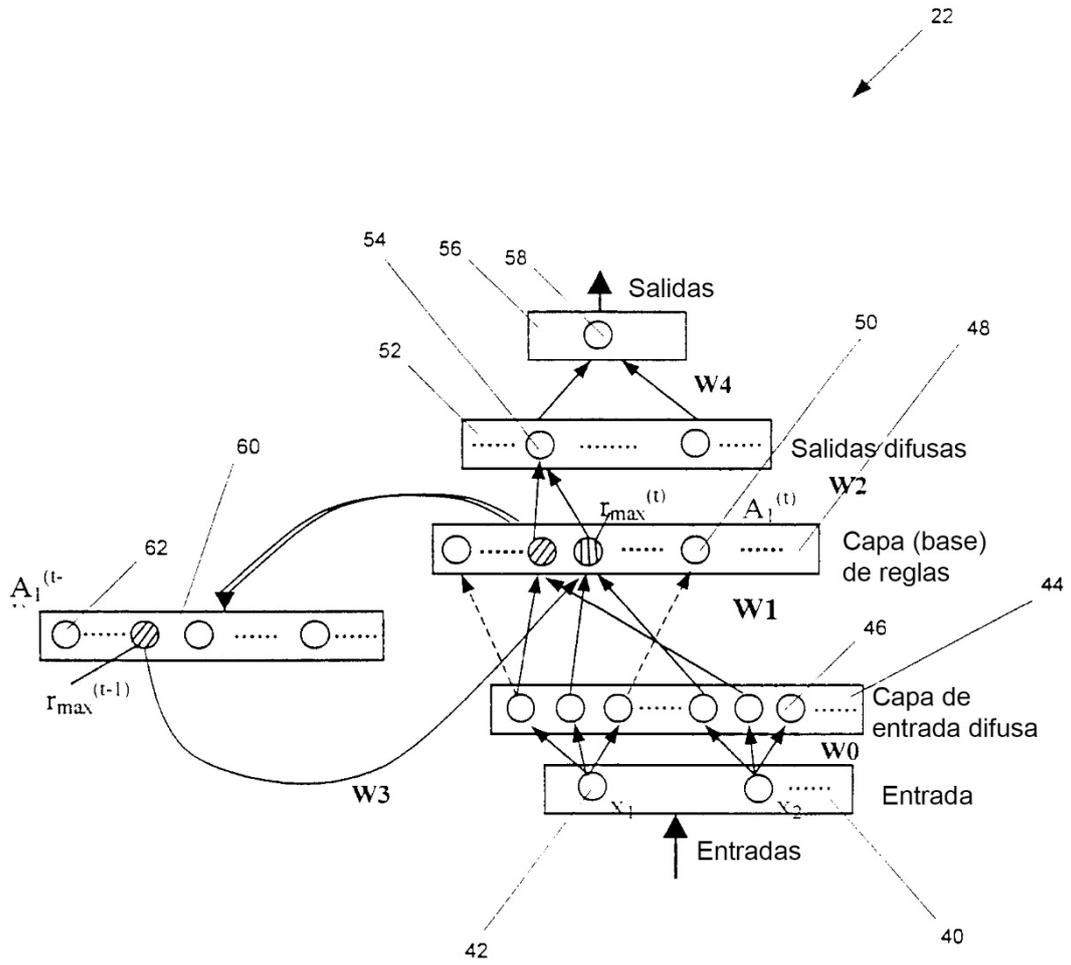


FIGURA 3

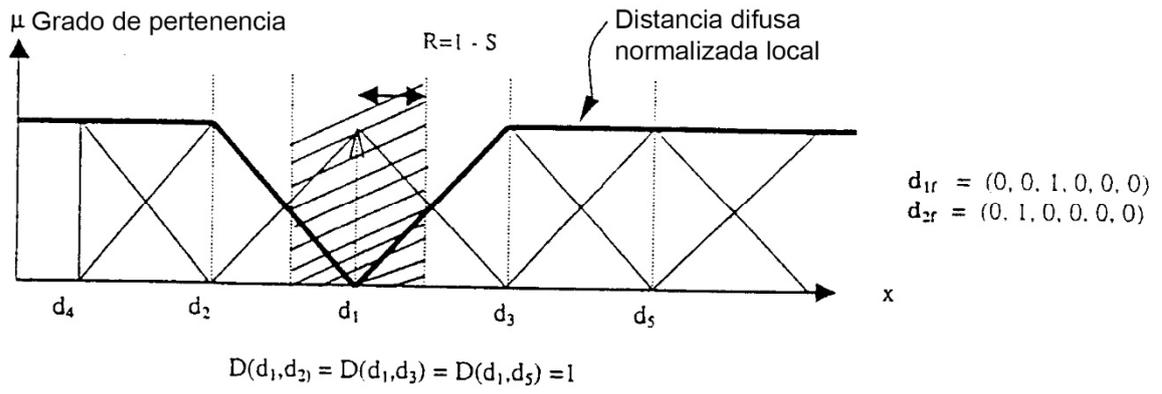


FIGURA 4

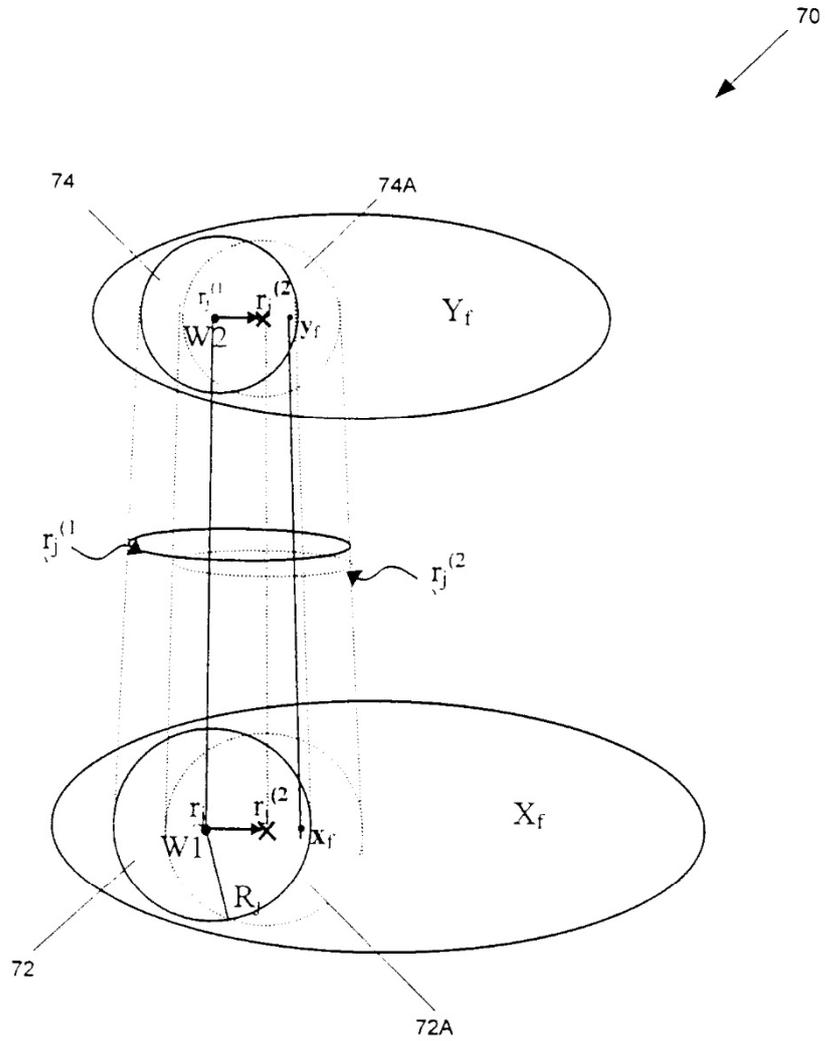


FIGURA 5

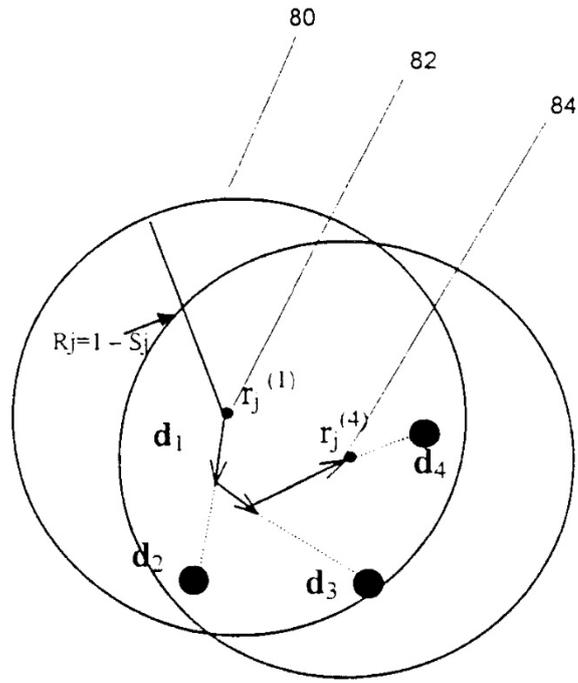


FIGURA 6

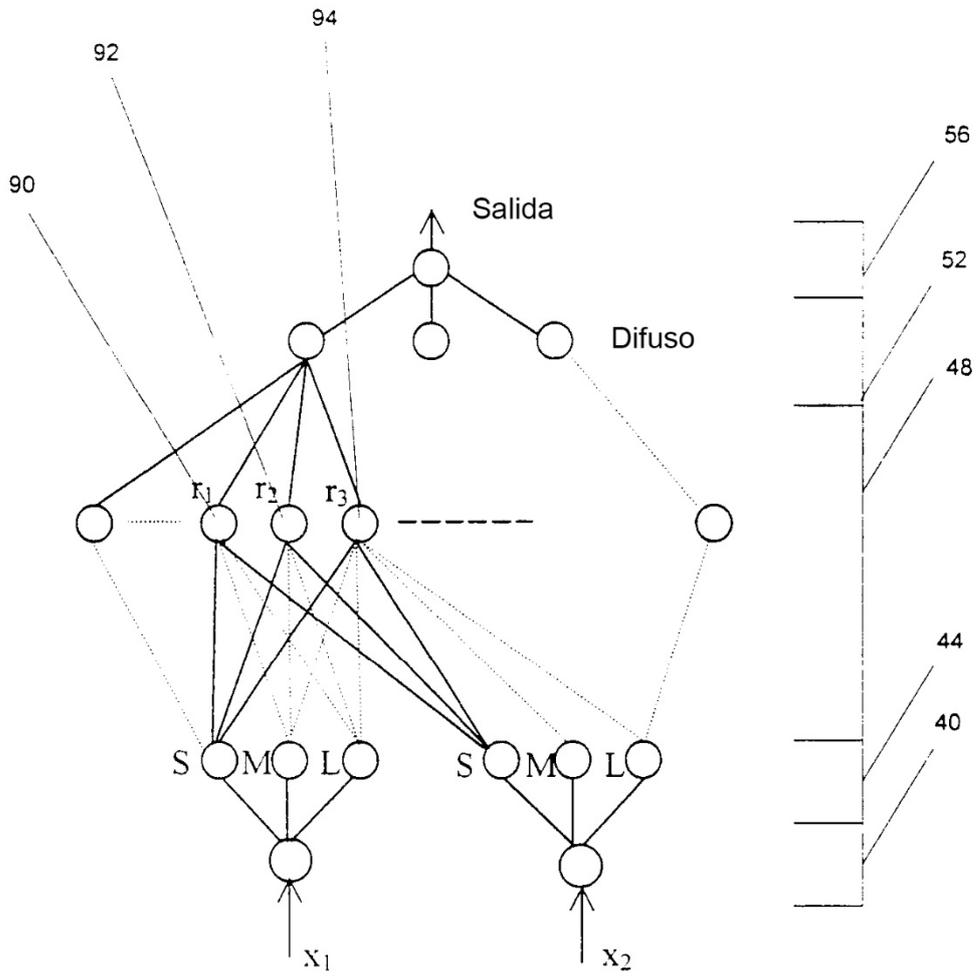


FIGURA 7

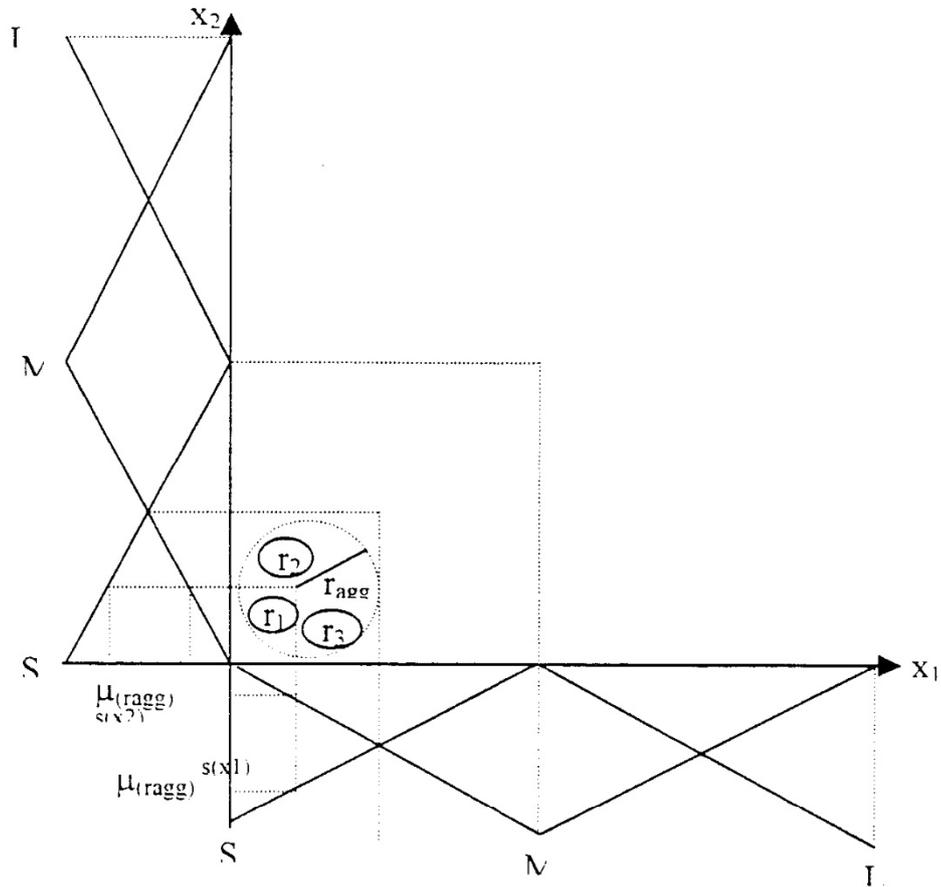


FIGURA 8

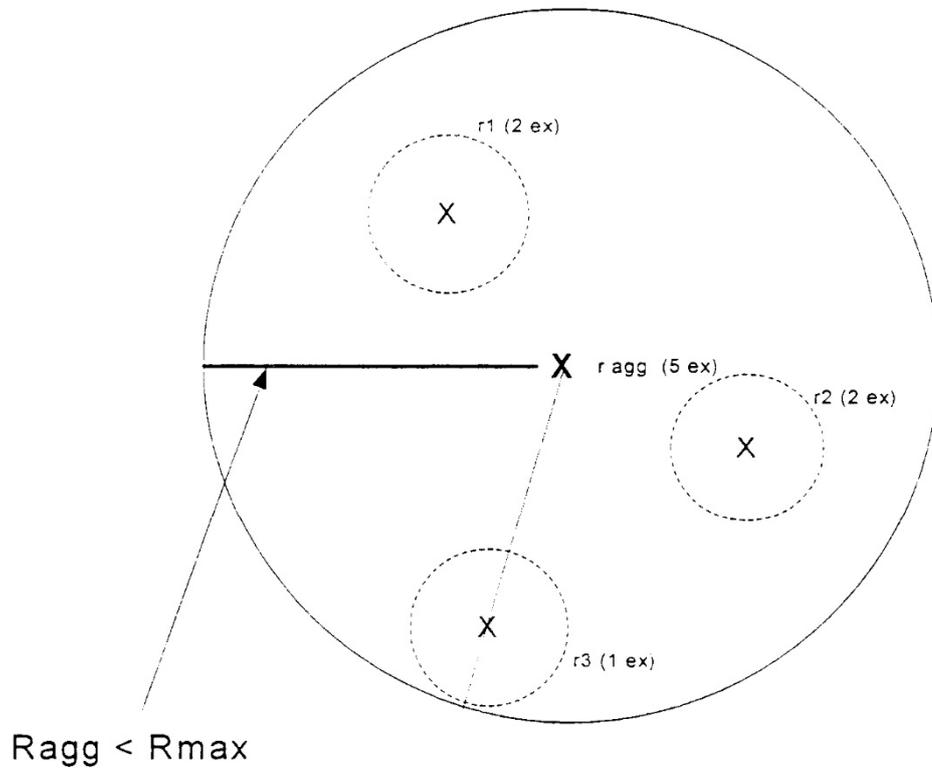


FIGURA 9

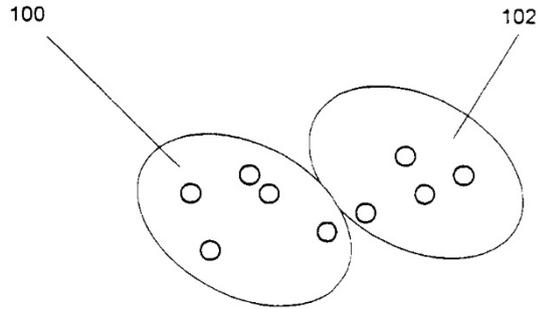


FIGURA 10

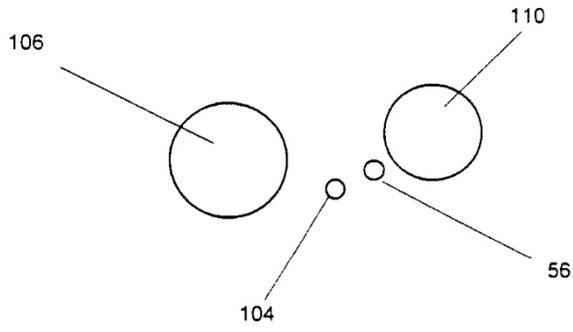


FIGURA 11

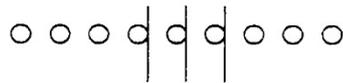


FIGURA 12



FIGURA 13

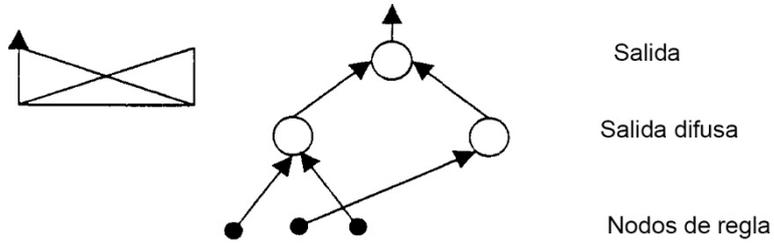


FIGURA 14

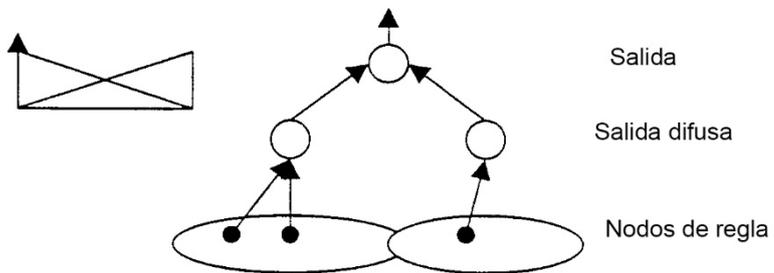


FIGURA 15

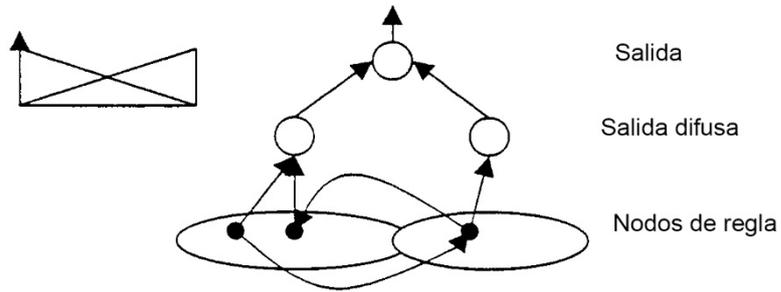


FIGURA 16

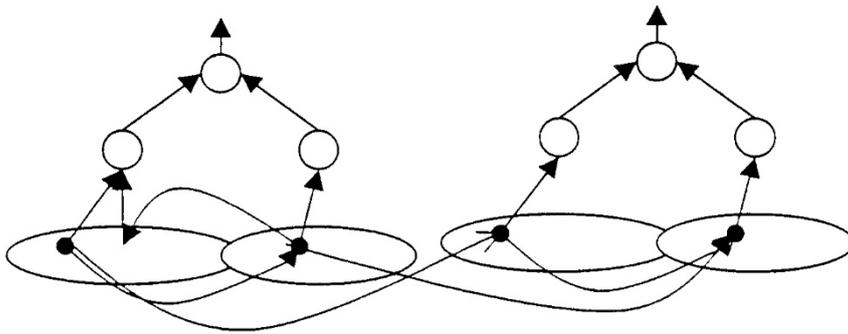


FIGURA 17

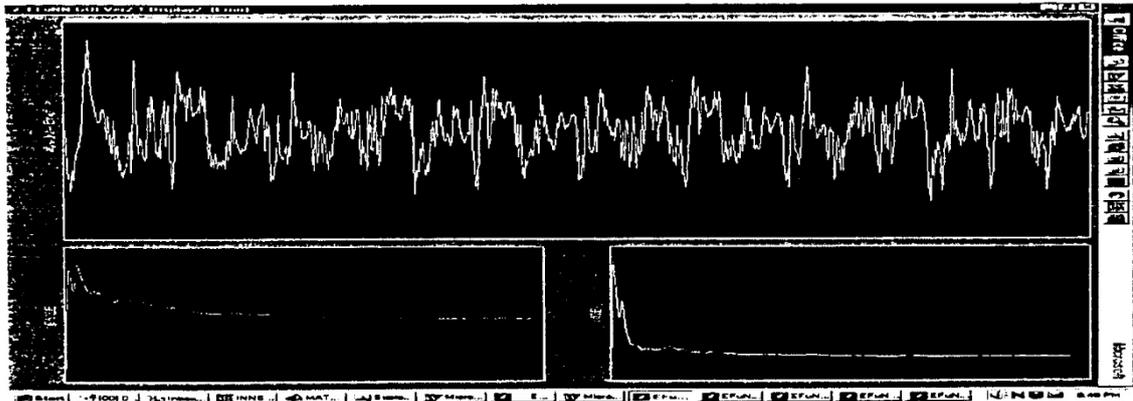
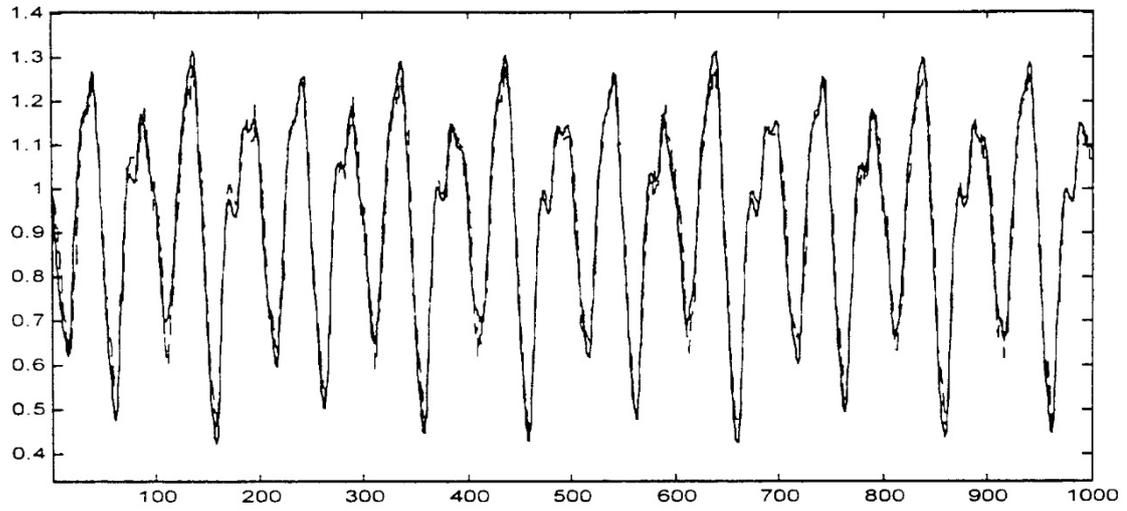


FIGURA 18A

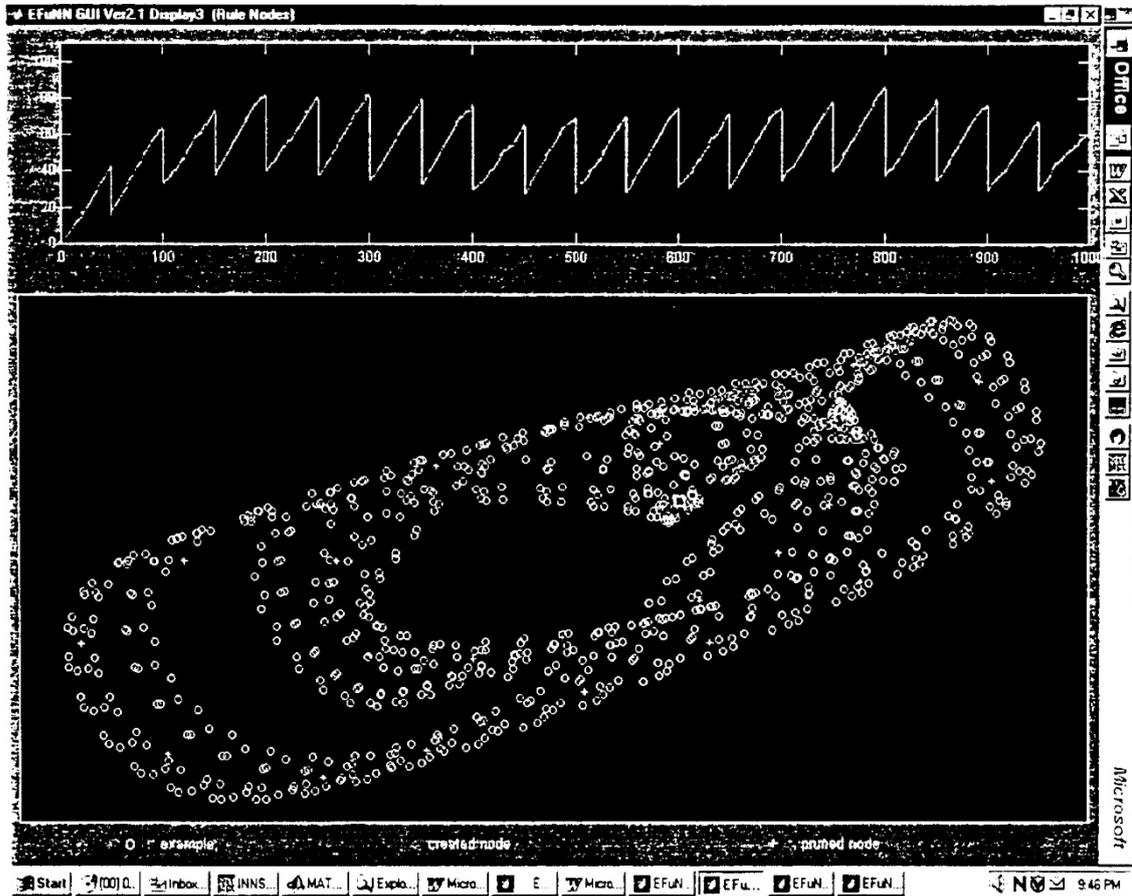


FIGURA 18B

## ES 2 643 448 T3

Regla 1: si [x1 es (3 0,658) Y [x2 es (4 0,884)] Y [x3 es (4 0,822)] Y

[x4 es (4 0,722)] [Radio del campo receptivo R1 = 0,086]

entonces [y es (4 0,747) [ejemplos de formación alojados Nex(r1) = 6]

Regla 2: si [x1 es (3 0,511)] Y [x2 es (4 0,774)] Y [x3 es (4 0,852)] Y

[x4 es (4 0,825)] [Radio del campo receptivo R2 = 0,179]

entonces [y es (3 0,913)][ejemplos de formación alojados Nex(r2)=2]

.....

Regla 16: si [x1 es (2 0,532)] Y [x2 es (2 0,810)] Y [x3 es (3 0,783)] Y

[x4 es (4 0,928)] [Radio del campo receptivo R16 = 0,073)

entonces [y es (5 0,516)] [ejemplos de formación alojados Nex(r16)=12]

Nota: Los valores difusos se indican con números de la siguiente manera: 1- muy pequeño, 2- pequeño, 3- medio, 4- grande, 5- muy grande: las ponderaciones antecedentes y consiguientes se redondean al tercer dígito después del punto decimal; los valores menores a 0,5 serán ignorados, ya que 0,5 se utiliza como umbral T1 y T2 para la extracción de reglas)

FIGURA 19

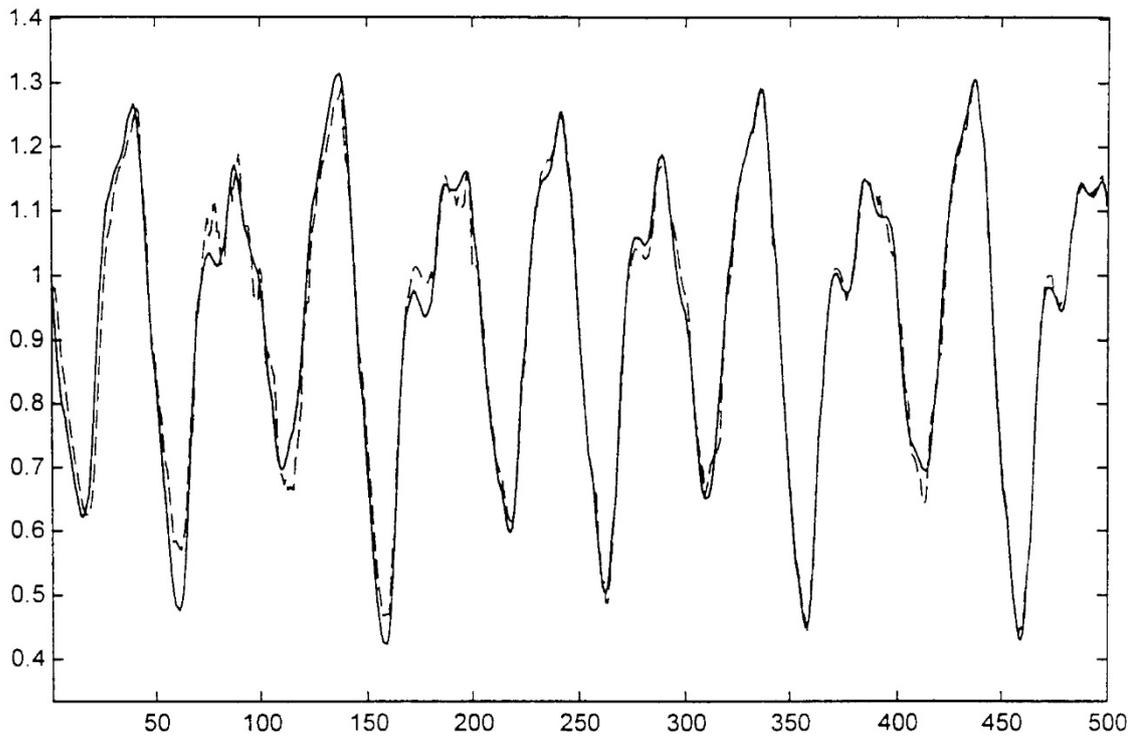


FIGURA 20

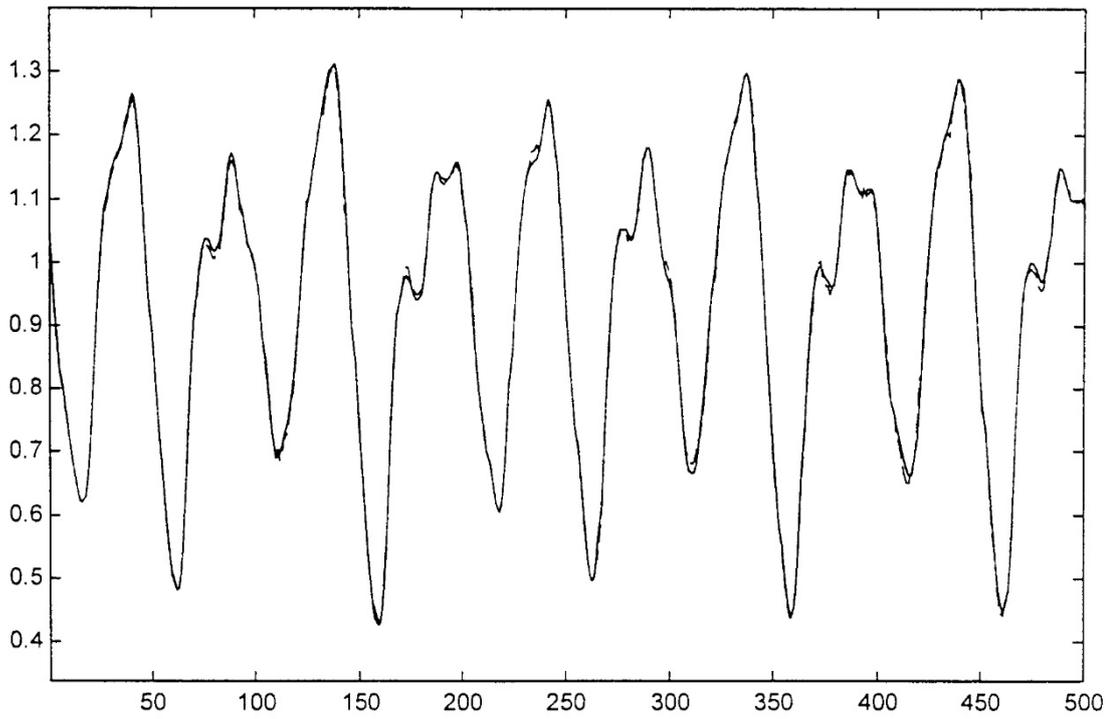


FIGURA 21

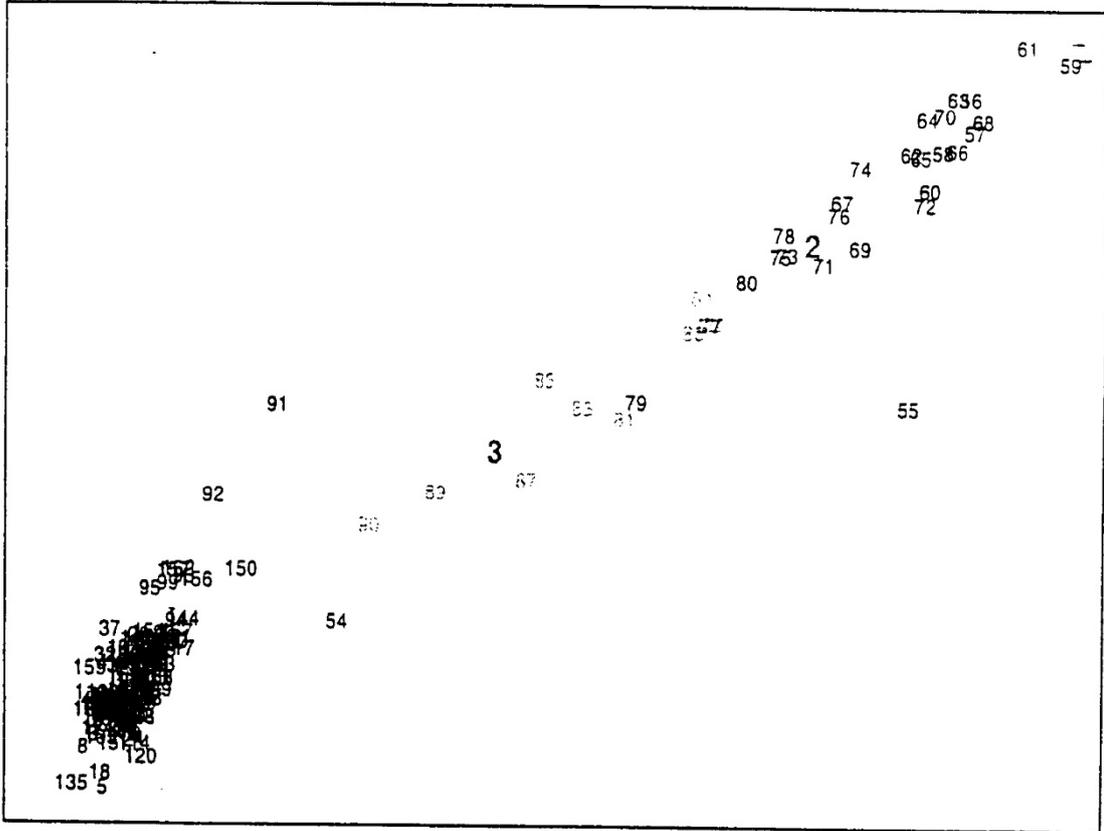


FIGURA 22

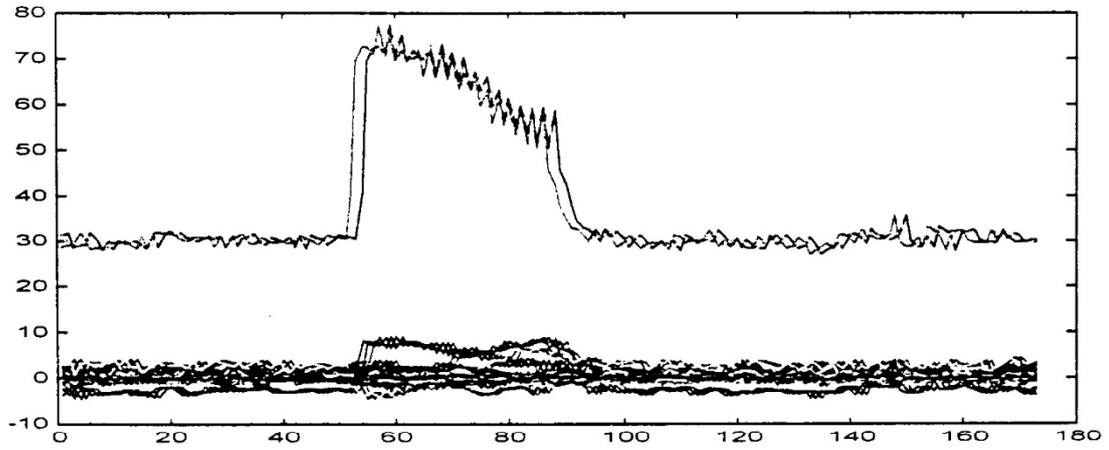


FIGURA 23

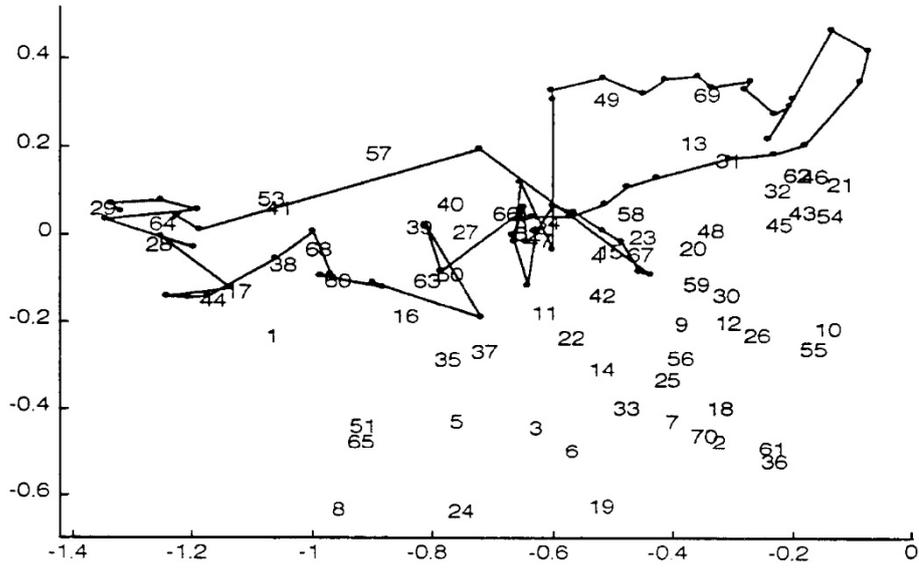


FIGURA 24

<b>Modelo</b>	<b>Valores de parámetro</b>	<b>Número de centros (nodos de regla en EFuNN)</b>	<b>LNDEI en línea después de aprender 3000 ejemplos</b>
RAN (Platt)	$\varepsilon = 0,01$	113	0,373
RAN-GRD (Rosipal)	$\varepsilon = 0,01$	50	0,165
RAN-P-GQRD (Rosipal)	$\varepsilon = 0,02$	31	0,160
System-su	$E = 0,05; R_{max} = 0,2$	91	0,115
System-dp	$E = 0,05; R_{max} = 0,2$	93	0,113

FIGURA 25

<b>Método \ Error</b>	<b>Precisión de clasificación de prueba global (fuera de línea)</b>	<b>Precisión de clasificación de prueba local (en línea)</b>
LVQ	57%	N/A
System-s/u	77%	86,6%

FIGURA 26

```
R1: SI -----C--C-C-TCC-G--CTC-GT-C--GGTGAGTG--GGC---C---G-GG-C--CC-  
  Entonces [EI]  
  Campo receptivo =0,216  
  Radio máx. =0,628  
  Ejemplos de formación alojados =26/1000
```

FIGURA 27

```
R1: SI [g1](2 0.9) y [g3](2 0.9) y [g5](2 0.7) y [g6](2 0.7) y [g8](1 0.8) y [g9](2 0.7)  
  {radio del campo receptivo = 0.109 }  
  ENTONCES clasificar TODO de leucemia {ejemplos de formación alojados = 47/72}
```

Nota: regla extraída de un sistema formado en datos de expresión genética de micromatriz: [g1] (2 0.9) significa que el grado de pertenencia al que el valor de expresión del gen 1 pertenece a la función de pertenencia “Alta” es 0.9.

FIGURA 28