

VIII

MEMORIAS ASOCIATIVAS

Francisco Javier López Aligué; M. Isabel Acevedo Sotoca
Universidad de Extremadura

1. INTRODUCCIÓN

Hoy en día, nadie se asombra cuando oye hablar de unidades de memoria de masa (discos, cintas, etc.) con capacidades de gigabytes, crecientes de año en año. Incluso es posible almacenar varios millones de bits en un solo chip de memoria integrada. La única pregunta, si acaso, que suele hacerse es la referida a los tiempos: tiempo de escritura, de lectura, valores máximos y mínimos, etc.

De hecho, solemos aceptar sin dudas que la capacidad de la memoria de que estamos tratando se mide por el número de biestables, o registros, que contiene, de forma que cada ocho biestables supondrían un byte de información almacenada. De la misma forma, para la lectura y/o escritura de datos de la CPU desde/hacia los circuitos de memoria, nos conformamos con algoritmos de seguridad muy sencillos: el bit de paridad para las memorias electrónicas (RAM y ROM) y los CRC (Códigos de Redundancia Cíclica) y similares, para las unidades de memoria de masa (discos, cintas, etc.).

Sin embargo, cuando la información está profundamente distorsionada, bien sea por ruidos de transmisión o por perturbaciones en el propio origen, los sistemas clásicos de memoria no pueden ser utilizados directamente, puesto que lo almacenado carecería de significado y, por tanto, de utilidad. En estos casos, es preciso recurrir a sistemas capaces de efectuar un procesamiento inteligente de los datos. Ocurre algo similar cuando se pretenden realizar tratamientos especiales de la información: el reconocimiento de patrones, la solución de problemas, y tantos otros,

requieren el uso de sistemas mucho más sofisticados, basados por regla general en lo que, hoy en día, conocemos por Inteligencia Artificial.

Muchas veces, no se trata solamente de almacenar un dato, sino de buscar un concepto asociado a ese dato. Como ejemplo paradigmático, fijémonos en la memoria humana: nunca solemos recordar las cosas exactamente como las vimos, ni siquiera solemos necesitar apreciar todos los detalles (ni mucho menos) de aquello que tenemos ante nuestros ojos. Por regla general, las impresiones visuales o auditivas (u olfativas, táctiles, etc.) que recibimos, nos evocan algo que permite identificarlas con facilidad: su concepto, u otra imagen familiar. Se trata, por tanto, de permitir una gran variedad de configuraciones de las señales que recibimos (distintos tipos, colores o tamaños de un cierto objeto) o de elevada contaminación de las mismas (imágenes distorsionadas, parciales, con ruido) de forma que, a pesar de todos sus defectos, somos capaces de identificarlas correctamente.

Una memoria convencional guarda y recupera la información mediante la dirección del registro encargado de almacenarla. Se trata de las llamadas «memorias accedidas por direcciones» (MAD). El método alternativo que a veces se emplea es el de las «memorias accedidas por contenido» (CAM), en las que se compara el dato de entrada con todos los contenidos en la memoria. El caso que aquí vamos a estudiar no debe, bajo ningún concepto, ser confundido con los dos anteriores: en las memorias asociativas, de todo tipo (aunque aquí las vayamos a englobar bajo el denominador común de BAM, Memorias Bipolares Asociativas), los datos de entrada no forman parte obligatoria de los contenidos almacenados en ella, pero son la clave para su búsqueda y localización. Es decir, los datos de entrada son la identificación funcional de los contenidos de la memoria. Esto significa que el proceso de extracción de una información no requiere de un mecanismo de simple comparación, sino que posee sus propios algoritmos de búsqueda que, por cierto, recuerdan mucho a lo que no costaría demasiado admitir como «inteligente»; de hecho, hasta su síntesis neuronal es mucho más propia que para las otras tecnologías.

Podemos resumir que los parámetros que caracterizan a las memorias son los siguientes:

PARÁMETRO	TIPO DE MEMORIA		
	MAD	CAM	BAM
Velocidad	ALTA	BAJA	MEDIA
Seguridad	BAJA	MEDIA	ALTA
Complejidad	BAJA	ALTA	MEDIA
Estructura	HARD	HARD	HARD / SOFT
Otros	Rígida	Compleja	Aprendizaje Procesamiento Inteligencia Artificial

En los siguientes apartados vamos a pasar revista a los aspectos más importantes de los desarrollos habidos hasta el presente en el estudio y diseño de memorias asociativas. En concreto, revisaremos las memorias en las que cada patrón se asocia consigo mismo, Autocorreladores, y en aquellas en que la asociación se hace por parejas de vectores de entrada, Heterocorreladores. Unos y otros tienen aplicaciones prácticas bien distintas y, en ambos casos, de sumo interés, por lo que nos detendremos con cierto detalle en cada una de dichas categorías.

El estudio de dichas memorias se hará desde un planteamiento casi intuitivo, que denominaremos de «primer orden» hasta los desarrollos menos obvios de «orden superior». Cuando ello sea posible o útil, plantearemos algún ejemplo que sirva para aclarar y poner de manifiesto cuanto vayamos diciendo, lo que no quiere decir, ni mucho menos, que siempre sea posible. En particular, tienen una versión práctica inmediata los elementos de primer orden, por lo que en ellos podremos esperar los casos prácticos. En el resto, pueden aparecer como simples «problemas» académicos.

2. AUTOCORRELADORES DE PRIMER ORDEN

La idea básica es que a la entrada de la memoria se presenta una determinada configuración de datos, lo que llamaremos patrón de entrada, que corresponde a alguna configuración, de entre unas cuantas previamente almacenadas, pero que ha sufrido alteraciones y/o mutilaciones. A la salida, esperamos obtener el patrón original perfectamente reconstruido. En este caso, de tanta utilidad en el campo de las comunicaciones, se dice que se trata de una memoria autoasociativa.

Una de las formas más comunes de memoria autoasociativa es aquella que almacena los patrones mediante correlación de sus valores, es decir, se trata de la memoria autoasociativa por correlación o, más brevemente, autocorrelador.

Hebb en 1943 de forma conceptual y Amari en 1972 con un estudio analítico riguroso, son los primeros responsables de la aparición de los autocorreladores, si bien los trabajos de Hopfield en 1982, en los que se hace uso práctico de la re-emergente teoría de las redes neuronales, han marcado el verdadero punto de arranque de los autocorreladores como sistemas de utilidad real.

Para analizar estas memorias, precisamos empezar por un breve análisis teórico: desde el punto de vista matemático, un autocorrelador es una matriz de dimensiones $n \times n$ formada mediante la suma de los productos externos de todos los P vectores a memorizar:

$$A_i \in \{A\} \quad ; \quad i = 1, 2, \dots, P$$

Sin embargo, antes es preciso reconfigurar la estructura de dichos vectores, de forma que partiendo de un conjunto de valores binarios (caso más simple de los valores de un patrón visual, por ejemplo) pasamos a su equivalente bipolar, es decir, hacemos la transformación

$$A_i = (a_1^i, a_2^i, \dots, a_n^i) \rightarrow X_i = (x_1^i, x_2^i, \dots, x_n^i)$$

donde

$$a_j^i = 0 \Rightarrow x_j^i = -1$$

$$a_j^i = 1 \Rightarrow x_j^i = +1$$

Con estos vectores bipolares, podemos ya formar la matriz de autocorrelación tal como antes se indicó, es decir:

$$M = \sum_{i=1}^P X_i^T X_i \quad (X_i^T \text{ es el vector columna traspuesto de } X_i)$$

de forma que sus componentes serán

$$m_{hj} = \sum_{i=1}^P x_h^i x_j^i \quad ; \quad h, j = 1, 2, \dots, n$$

y entonces se dice que la memoria ha almacenado P patrones.

Veamos cómo funciona: supongamos que tenemos la matriz M anterior y que recibimos una configuración de entrada B_j que debemos identificar con alguna de las A_i anteriores, pero que no puede hacerse inmediatamente porque

$$B_j \neq A_i \quad \forall i = 1, \dots, P.$$

El proceso se inicia calculando siempre el equivalente bipolar de B_j ,

$$Y_j = (y_1^j, y_2^j, \dots, y_n^j)$$

$$b_i^j = 0 \rightarrow y_i^j = -1, \quad i = 1, 2, \dots, n$$

$$b_i^j = 1 \rightarrow y_i^j = 1, \quad i = 1, 2, \dots, n$$

a continuación, se obtiene el vector transformado mediante la aplicación de la matriz M al vector Y_j^T , traspuesto de Y_j ,

$$Y_j' = M \cdot Y_j^T$$

donde sus componentes vendrán dadas por

$$y_h^j = \sum_{i=1}^n m_{hi} y_i^j$$

que tendrán un valor comprendido entre +n y -n. Para obtener el vector binario correspondiente, se emplea la función umbral de dos parámetros

$$b_h^{\text{nuevo}} = F\left(y_h^j, b_h^{\text{anterior}}\right) = F\left(\sum_{i=1}^n m_{hi} y_i^j, b_h^{\text{anterior}}\right)$$

definida como

$$F(x,y) = \begin{array}{lll} 1 & \text{si} & x > 0 \\ y & \text{si} & x = 0 \\ 0 & \text{si} & x < 0 \end{array}$$

Con esto, obtenemos un nuevo vector B^j . Si éste ya coincide con alguno de los A_i , el proceso se termina. En caso contrario, se repiten los pasos anteriores tantas veces como sea necesario, hasta conseguir una salida fija, momento en que se dice que el sistema se ha estabilizado.

2.1. Un ejemplo

Supongamos un caso muy sencillo en el que disponemos de tres vectores a almacenar; y sus correspondientes bipolares:

$$A_1 = (1 \ 0 \ 0 \ 0) \Rightarrow X_1 = (1 \ -1 \ -1 \ -1)$$

$$A_2 = (0 \ 1 \ 1 \ 0) \Rightarrow X_2 = (-1 \ 1 \ 1 \ -1)$$

$$A_3 = (1 \ 0 \ 1 \ 1) \Rightarrow X_3 = (1 \ -1 \ 1 \ 1)$$

realizando las oportunas operaciones, llegamos a

$$M = \begin{pmatrix} 3 & -3 & -1 & 1 \\ -3 & 3 & 1 & -1 \\ -1 & 1 & 3 & 1 \\ 1 & -1 & 1 & 3 \end{pmatrix}$$

Supongamos ahora que presentamos el vector $B_j = (1, 1, 1, 0)$. El primer paso consiste en hallar su equivalente bipolar $Y_j = (1, 1, 1, -1)$. Para ello, tendremos la ecuación

$$Y_j = MY_j^T = (-2, 2, 2, -2)$$

con lo que la función umbral nos lleva a

$$B_j^1 = (0, 1, 1, 0) \equiv A_2$$

que ya coincide con uno de los previamente almacenados y con cuya identificación finaliza el proceso.

2.2. Síntesis neuronal

Una vez visto el funcionamiento de los autocorreladores, llega el momento de intentar sintetizarlo. En principio, es evidente que su aparente sencillez permitirá construirlo mediante el uso de cualquier lenguaje de programación sobre ordenadores personales. Sin embargo, cuando el número y tamaño de los vectores a almacenar llegue a alcanzar un tamaño considerable, es más que probable que la programación imponga restricciones excesivas en cuanto a los tiempos de respuesta, hasta el punto de no poder utilizarla. En este caso tan real, la solución consiste en utilizar Redes Neuronales.

En efecto, la característica que identifica a las Neuronas artificiales es la función umbral de la suma ponderada de sus entradas. En forma analítica,

$$y_i = F \left(\sum_{j=1}^n m_{ij} x_j, q \right)$$

donde q es el umbral escogido, que en nuestro caso es 0. Por tanto, resulta sencillo construir el correspondiente autocorrelador donde los términos m_{ij} corresponden a los pesos sinápticos de las entradas que, en un principio, responden a los valores de las entradas para pasar, a continuación, a ser las respuestas de las propias neuronas. En la figura 1 puede verse el esquema organizativo de uno de tales

autocorreladores, propuesto por Hopfield. En este caso, Hopfield presentó una versión reducida en la que se prohibían las conexiones recurrentes, es decir, en la que $m_{ij} = 0 \forall i = 1, 2, \dots, n$. En realidad, el cálculo de los coeficientes sinápticos se reduce exclusivamente a la mitad, pues es evidente que tal y como está definida la matriz M , siempre se cumplirá que

$$m_{ij} = m_{ji} \quad \forall i, j = 1, 2, \dots, n$$

Como puede verse, una vez diseñado el autocorrelador, quedan fijados los valores de los pesos de conexión sináptica, por lo que se dice que se trata de una Red Neuronal sin aprendizaje. El conjunto funciona como una memoria PROM (o REPROM si nos es posible volver a cambiarlos) de forma que parece natural admitir que si en los circuitos convencionales la memoria está localizada en los biestables, aquí lo está en las conexiones entre neuronas (las sinapsis).

Dado que los valores de los pesos son siempre positivos (o nulos), los términos negativos de la matriz se obtienen a partir de los correspondientes inversores colocados a las salidas de las neuronas, lo que da a la red el particular aspecto de la figura 1.

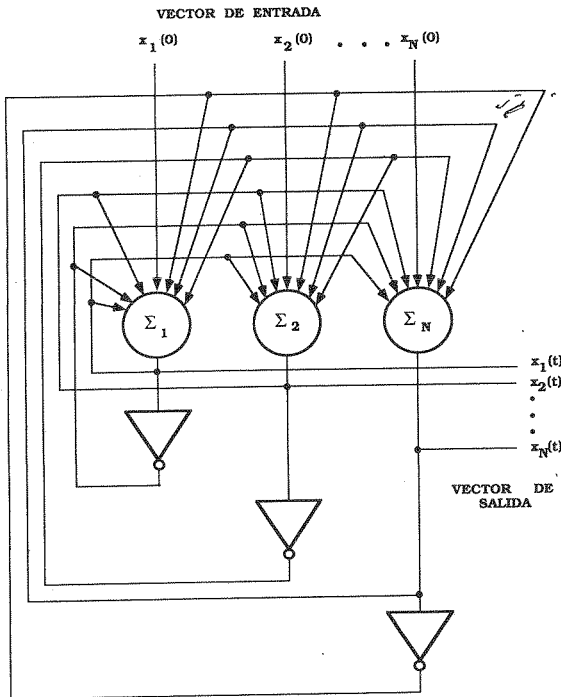


Figura 1. Síntesis neuronal de un Autocorrelador.

2.3. Estabilidad y capacidad

Qué duda cabe de que los autocorreladores suponen una atractiva idea cuando se piensa en identificación y reconstrucción de señales, imágenes, etc. Sin embargo, surge de inmediato la duda sobre la seguridad de su funcionamiento: ¿Cómo podemos saber que ante una determinada entrada la salida, tras ciertas admisibles y lógicas variaciones, llegará a una respuesta permanente?. Este problema es el que se conoce como *estabilidad*, es decir, la garantía de que al cabo de un tiempo de cómputo, se obtendrá una salida que permanecerá constantemente disponible.

Es claro que en las memorias convencionales, accesibles por direccionamiento, el tiempo de estabilización es el que marquen los dispositivos constitutivos (del orden de decenas de nsecs.) y prácticamente igual para todos los registros direccionables. Aquí, por el contrario, dependerá de lo «distorsionada» que esté la entrada.

En cuanto a la seguridad de su resultado, el método más popular está basado en la función energética de Liapunov, propuesto inicialmente por Hopfield. La idea básica consiste en definir una función E de energía relacionada con el autocorrelador y demostrar, para todo tipo de actuación de éste, que tal energía disminuye, de forma que al final se llega a un estado de energía mínima y, por tanto, punto final de la trayectoria. Esta función E se define a través de un vector Y_i , que se usa como identificador del estado de la red en un determinado instante, de la forma

$$E = -Y_i M Y_i^T$$

donde Y^T es el vector traspuesto de Y. Hopfield propuso que el cambio de las salidas y_k^i (cada una de las componentes del vector Y_i de las neuronas se hiciera asíncronamente, es decir, una detrás de otra. Así, el problema se reduce a demostrar que *siempre* $\Delta E < 0$ para cualquier alteración de la forma

$$y_k^i \rightarrow y_k^i + \Delta y_k^i$$

La demostración no es difícil, puesto que podemos comprobar que

$$\Delta E = E\left(y_k^i + \Delta y_k^i\right) - E\left(y_k^i\right) = -2\Delta y_k^i \sum_{j=1}^n m_{kj} y_j^i - m_{kk} \left(\Delta y_k^i\right)^2$$

que se consigue admitiendo que $m_{kj} = m_{jk}$. Evidentemente, m_{kk} es positivo y, obviamente, $(\Delta y_k^i)^2$ también. Queda, por lo tanto, analizar el término de la sumatoria, que puede hacerse separando sus dos componentes y estudiando todos los casos posibles:

1.- $\Delta y_k^i = 0$ (no hay variación de la entrada). Entonces $\Delta E = 0$

y el sistema permanece en el mismo estado en que se hallaba, es decir, ha alcanzado un punto de equilibrio.

2.- $\Delta y_k^i > 0$. Esto sólo puede ocurrir cuando y_k^i pasa de valor -1 a valor +1, es decir, $\Delta y_k^i = 2$.

Para que esto ocurra, la neurona y_k ha tenido que recibir una aportación positiva de todas sus entradas (funcionamiento típico), es decir,

$$\sum_{j=1}^n m_{kj} y_j^i > 0,$$

luego

$$\Delta y_k^i > 0 \Rightarrow \sum_{j=1}^n m_{kj} y_j^i > 0$$

de donde podemos deducir inmediatamente que $\Delta E < 0$.

3.- $\Delta y_k^i < 0$. Ahora nos encontramos en el caso opuesto al anterior. Ello significa que el valor de la salida y_k ha pasado de +1 a -1, con lo que $\Delta y_k^i = -2$. Pero además, para que así ocurra, la suma de las entradas ha de ser inferior al umbral de disparo, que en nuestro caso es 0, luego

$$\sum_{j=1}^N m_{kj} y_j^i < 0$$

y, por lo tanto,

$$\Delta y_k^i < 0 \Rightarrow \sum_{j=1}^N m_{kj} y_j^i < 0 \Rightarrow \Delta E < 0$$

Es evidente que lo dicho para una variable puede extenderse a todo el resto, de forma que en el supuesto de funcionamiento síncrono, las neuronas cambian su estado al unísono, también podemos concluir que el estado del autocorrelador irá evolucionando por estados de energía decreciente hasta llegar al nivel mínimo, en el que permanecerá indefinidamente, dando por finalizado el proceso de reconocimiento y reconstrucción del primitivo patrón de entrada.

El otro problema que se plantea, es el típico de la *Capacidad* de nuestra memoria. En el caso de las memorias convencionales, no hay ninguna duda, tantos registros suponen igual número de bytes almacenables. Aquí, por el contrario, la respuesta es mucho más difícil. Amari hizo un cálculo estadístico según el cual, el número «K» de patrones que pueden memorizarse en una red de N neuronas, es decir, guardarse con la absoluta seguridad de que pueden ser recuperados (identificación positiva de los patrones) es

$$K = \frac{N}{2 \log N + \log \log N} \quad (\log: \text{logaritmos naturales})$$

que, con mayor exactitud, debe leerse como sigue: a medida que aumenta el número de patrones K que deseemos almacenar, la dimensión «N» de la red deberá crecer de forma exponencial, hasta el límite en el que N habrá de aumentar un infinito para cada nuevo patrón, es decir,

$$\lim_{N \rightarrow \infty} \frac{K}{N} = \lim_{N \rightarrow \infty} \frac{1}{2 \log N + \log \log N} = 0$$

A este mismo resultado llegaron McEliece y VenKatesh entre otros, mientras Hopfield proponía, para sus redes en las que $m_{kk} = 0$, una relación lineal del tipo $K = 0.15 N$.

2.4. Autocorreladores de orden superior

El funcionamiento de los autocorreladores estudiado hasta aquí responde a lo que suele entenderse como de «Primer Orden», dado que la asociación de los distintos vectores corresponde a un producto interior simple. En ciertas ocasiones, se presentan situaciones no lineales en las que, como en el caso de los discriminantes, se precisa de la definición de hiperplanos. Para estos casos, es preciso recurrir a las correlaciones de segundo orden y/o superior. Se trata, por lo tanto, de sintetizar una red con un número de conexiones mucho más elevado.

En lo que a nosotros respecta, vamos a exponer las de segundo orden, de forma que su extensión hasta el orden N-1 es inmediato (obsérvese que el máximo orden alcanzable es el de componentes de los vectores menos uno). Tomando como punto de partida el autocorrelador de primer orden, podemos definir el de segundo orden a través de sus componentes de la nueva matriz M:

$$m_{ijk} = \sum_{h=1}^n x_i^h x_j^h x_k^h ; \quad i, j, k = 1, 2, \dots, n$$

Esta matriz, tiene tres ejes y representa la creación de $n^3 - n^2$ conexiones nuevas con respecto a la matriz de segundo orden. En cuanto al método de obtención de las respuestas que se obtienen frente a una entrada exterior cualquiera, seguirá el mismo camino que en el caso de primer orden, es decir:

$$y_k^h = \sum_{i=1}^n \sum_{j=1}^n m_{ijk} y_i^h y_j^h ; \quad k=1,2,\dots,n$$

si queremos utilizar una expresión similar también para el vector B, podemos identificar inmediatamente el resultado siguiente

$$b_h^{\text{nuevo}} = F\left(\sum_{i=1}^n \sum_{j=1}^n m_{ijk} y_j^h + \sum_{i=1}^n \sum_{j=1}^n m_{ijk} y_i^h y_j^h , b^{\text{anterior}} \right)$$

Ahora queda el aspecto relativo a la estabilidad de estas memorias. El método de los valores decrecientes de energía se desarrolla a través de la nueva función

$$E_g = - \sum_{h=1}^n (Z \bullet Z_h)^{g+1}$$

siendo «g» el orden del autocorrelador, y «•» el signo del producto de operadores. Ahora debemos demostrar que cualquier variación supone siempre una disminución del valor de la energía. Un método ampliamente aceptado es el propuesto por Chen en 1988, mediante el cual es posible demostrar que todos los autocorreladores de orden superior son estables bajo funcionamiento *asíncrono*, mientras que los de orden g-impar lo son bajo funcionamiento *síncrono*.

Para ello, consideremos la energía total descompuesta en sus dos sumandos:

$$E = E_1 + E_2$$

correspondientes a las contribuciones de primero y segundo orden respectivamente, puesto que no hay que olvidar que las de orden «g» incluyen a las de orden «g-1» para todo valor de «g».

Con ello, tendremos que

$$\begin{aligned} E &= - \sum_{h=1}^P (Z \bullet Z_h)^2 - \sum_{h=1}^P (Z \bullet Z_h)^3 \\ &= - \sum_{j=1}^n \sum_{k=1}^n m_{jk} y_j y_k - \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n m_{ijk} y_i y_j y_k \end{aligned}$$

a partir de aquí, basta con seguir el mismo camino anterior para demostrar los asertos que decíamos al principio.

El punto restante, es el relativo a la capacidad de almacenamiento de estas memorias. Como siempre, nos encontramos ante problemas matemáticos de muy difícil solución. Pero, también como siempre, podemos encontrar en la bibliografía, trabajos de los autores más destacados en estos temas, en los que se presentan algunas soluciones a nuestras preguntas. Para esta situación en concreto, Venkatesh llegó a la expresión

$$P = \frac{N^g - 1}{2N! \log N}$$

Merece la pena observar que, a medida que aumenta la dimensión, disminuye la capacidad relativa, de forma que el límite también en estas situaciones es de cero. Un resultado muy importante que invita a reflexionar.

3. LAS MEMORIAS HETEROASOCIATIVAS

3.1. Heterocorreladores de primer orden: Las «BAM»

Contrariamente a lo que pueda parecer, los autocorreladores no son las memorias asociativas más populares. En realidad no son más que un caso particular de los heterocorreladores, en los que se construyen asociaciones de patrones por parejas. Aunque no fue el primero en utilizarlas, Kosko ocupa en las memorias heteroasociativas, que él llamó bidireccionales, el mismo lugar preeminente que Hopfield en el campo de las autoasociativas.

En lo que sigue, podemos admitir que un patrón convencional, como los caracteres 7x5 de las impresoras de agujas, no son más que empaquetamientos especiales de un vector de 35 componentes: el resultado de leer ordenadamente, sin solución de continuidad, las siete filas de cinco componentes (puntos) cada una. Ahora, se trata de asociar los vectores de dos conjuntos {A} y {B} equidimensionales

$$\{A\} = \{A_1, A_2, \dots, A_N\}$$

$$\{B\} = \{B_1, B_2, \dots, B_N\}$$

formado cada uno de dichos vectores por p y q componentes respectivamente,

$$A_i = (a_1^i, a_2^i, \dots, a_p^i)$$

$$B_j = (b_1^j, b_2^j, \dots, b_q^j)$$

de manera que se genera un conjunto de N pares (A_i, B_i) :

$$\{(A_1, B_1), (A_2, B_2), \dots, (A_N, B_N)\}$$

A continuación, podemos pasar a construir la matriz de correlación de forma que en este caso será

$$M = \sum_{i=1}^N X_i^T Y_i$$

donde X_i e Y_i son las versiones bipolares de los vectores A_i y B_i , con la misma estrategia de los autocorreladores, a cuyo caso se reducirán haciendo, simplemente $\{B_i\} \equiv \{A_i\}$, es decir,

$$Y_i \equiv X_i.$$

El nombre de BAM (*Bidirectional Associative Memory*) dado por Kosko a los heterocorreladores, obedece al particular mecanismo de funcionamiento de éstos, asimilable (hasta cierto punto) al de los autocorreladores: a la «entrada» del sistema se presenta un vector desconocido $A_a = (a_1^a, \dots, a_p^a)$ diferente, por regla general, de los del grupo $\{A_i\}$. Generamos su equivalente bipolar X_a que, evidentemente, no estará incluido en $\{X_i\}$. Con la matriz M, obtenemos el vector de salida Y_a :

$$Y_a \equiv X_a M.$$

del que se obtiene el binario $B_a = F(Y_a)$. Ahora, puede ocurrir que el vector B_a esté entre los del conjunto $\{B_i\}$ o no. En caso afirmativo, el proceso habrá terminado, mientras que en caso negativo (lo más general), debemos volver a aplicar la matriz M, pero ahora en sentido contrario, es decir, generando un nuevo vector

$$X_a^1 = Y_a M^T$$

Esto, expresado en sus componentes, queda de la forma

$$x_1^{\text{nuevo}} = F\left(\sum_{j=1}^q m_{ij} y_j, x_i^{\text{anterior}}\right), \quad \forall i = 1, 2, \dots, p$$

y, similarmente,

$$y_j^{\text{nuevo}} = F\left(\sum_{i=1}^p m_{ji} x_i, y_j^{\text{anterior}}\right), \quad \forall j = 1, 2, \dots, q$$

para las variables de la otra capa. El proceso se repite de forma indefinida, alternando los «sentidos» de aplicación de la matriz de correlación (M y M^T sucesivamente). Obsérvese que para el autocorrelador se obtenía la serie de vectores

$$Y_j \rightarrow Y_j^1 \rightarrow Y_j^2 \rightarrow \dots \rightarrow Y_j^r$$

hasta llegar al resultado final Y_j^r , que ya coincidía con alguno de los memorizados, permaneciendo estable. Ahora, la serie es doble,

$$X_a \rightarrow Y_a \rightarrow X_a^1 \rightarrow Y_a^1 \rightarrow \dots \rightarrow X_a^r \rightarrow Y_a^r$$

donde deberá cumplirse que $(X_a^r, Y_a^r) \in \{(X_1, Y_1), \dots, (X_N, Y_N)\}$

3.2. Un ejemplo

Si bien bastaría con utilizar cualquier par de conjuntos de vectores cualesquiera, no hemos podido resistir la tentación de presentar un caso muy práctico relacionado con la codificación de caracteres, poniendo además de manifiesto su eficaz rechazo al ruido. Se trata del ejemplo propuesto por Kosko en 1987, en un trabajo clásico donde, por primera vez, se estudiaba la aplicación de las BAM a procesado de señales y su síntesis mediante redes de neuronas. Para ello, se tomaron como vectores del conjunto {A} las letras M, S y G, dibujadas sobre una cuadrícula de 10 x 14 puntos, y como vectores del conjunto {B} las letras V, E y N, con un dibujo de 9x12 puntos. La asociación se hizo con las parejas (M, V), (S, E) y (G, N). De esta forma, las dimensiones de las capas era, respectivamente, de 140 y 108 neuronas. Obviamente, la matriz M resultante era de dimensión 140 x 108, lo cual resulta excesivo para desarrollarlo aquí con detalle. Sin embargo, resulta interesante comprobar su inmunidad frente al ruido, para lo cual presentaremos aquí el resultado obtenido por el propio Kosko.

Se presentó una letra S con una distorsión del 40%, obteniéndose a la salida la respuesta 1 de la figura 2. Cada vez que se llevaba a cabo una aplicación de la matriz M , se iban obteniendo los resultados (en la parte de los vectores $\{Y_j\}$) que se muestran

en la citada figura 2, hasta que al final del paso 11 se ha regenerado completamente la letra E, asociada con la S. Evidentemente, esta S es la salida $[X_j]$ que se obtiene de la otra capa de neuronas, de forma que la secuencia de salidas sigue un camino evolutivo similar al que aparece en la figura 2.

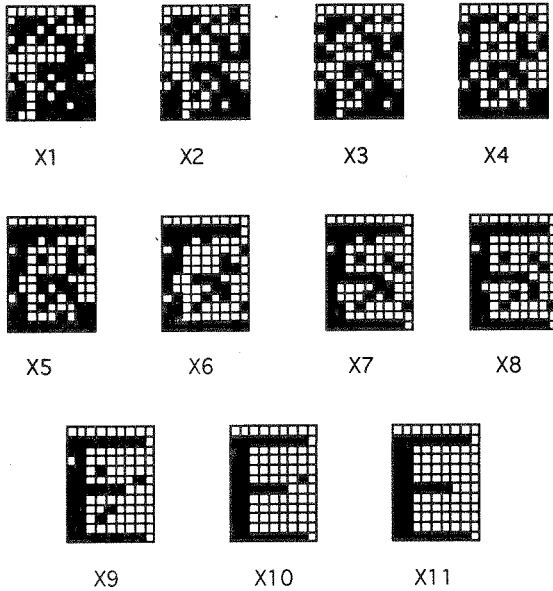


Figura 2. Los 11 pasos de reconocimiento del par S - E [Kosko, 1987].

3.3. Síntesis neuronal

Todas las razones expuestas en el caso de los autocorreladores son, nuevamente, aplicables aquí. La diferencia es que ahora hay que generar dos conjuntos de vectores, los X_a y los Y_a con la única matriz M , lo que indica que la síntesis se realiza mediante dos capas de neuronas, una que tiene por pesos sinápticos los valores en columna de M y la otra los valores en filas, es decir, tal y como se definieron,

$$y_j^a = F\left(\sum_{i=1}^q m_{ij} x_i^a\right), \quad j=1,2,\dots,p$$

$$x_k^a = F\left(\sum_{h=1}^p m_{hk} y_h^a\right), \quad k=1,2,\dots,q$$

En la figura 3 puede verse el resultado final, en el que pueden apreciarse las evidentes similitudes con el autocorrelador de la figura 1, excepción hecha de los inversores de las salidas, que se han omitido por sencillez.

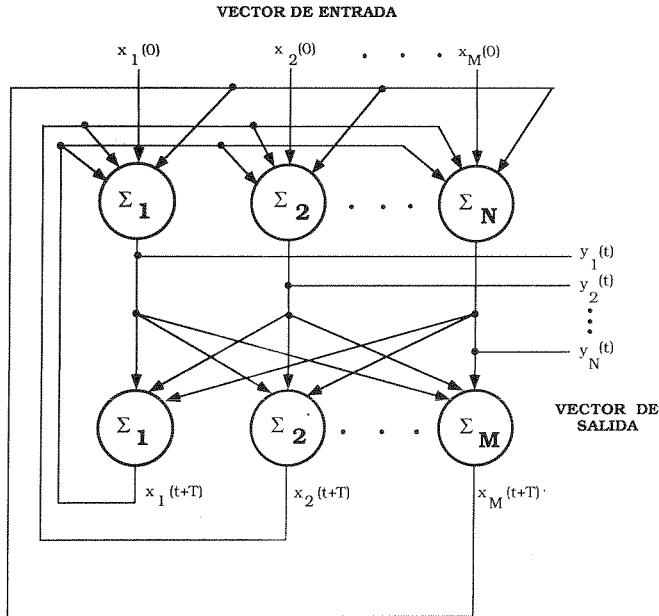


Figura 3. Síntesis neuronal de un Autocorrelador (BAM).

Aquí son también de aplicación todas las consideraciones relativas a la simulación de los pesos m_{ij} , hechas para el autocorrelador. Esto nos lleva a darnos cuenta de que ni el total de memoria ocupada ni el tiempo de CPU en la simulación serán el doble para una BAM que para un autocorrelador. En efecto, una vez fijados los valores m_{ij} de los contactos sinápticos para una de las capas, quedan automáticamente fijados los de la otra por cuanto se trata exactamente de los mismos. La única diferencia reside que antes, para cada neurona «j» se leían en fila (m_{1j} , m_{2j} , ... m_{ij} , ...) y ahora deberán emplearse en columna (m_{j1} , m_{j2} , ... m_{ji} , ...), para lo que bastará diseñar el programa adecuado.

Este hecho resulta fundamental para el ahorro del tiempo de ejecución, por cuanto la tabla de pesos es única, aunque la reducción de tiempo se ve disminuida por la búsqueda inversa de pesos.

Un detalle que, desde el punto de vista de las Redes Neuronales, resulta muy importante, es la falta de intrarealimentación, es decir, en una BAM se han perdido los lazos que conectan las salidas de una capa con las entradas a la propia capa. Esto significa que aún quedan muchas funciones muy complejas que sintetizar mediante redes neuronales.

3.4. Estabilidad y capacidad

El método a seguir, como no podía ser de otra forma, es absolutamente similar al seguido con los autocorreladores. La única diferencia se encuentra en la definición del parámetro de energía, que ahora es

$$E' = -XMY^T = - \sum_{i=1}^p x_i \sum_{j=1}^q m_{ij} y_j$$

Cualquier variación de la energía puede calcularse observando que corresponderá a un cambio en el vector Y, dada la forma en que hemos definido la función. Si quisiéramos estudiar los cambios energéticos producidos por las alteraciones en el vector X, deberíamos recurrir a la definición

$$E' = -YM^T X^T = - \sum_{j=1}^q y_j \sum_{i=1}^p m_{ij} x_i$$

aunque, como puede comprobarse, dada la evidente similitud de ambas expresiones, el resultado sería el mismo. Retomando la anterior formulación, y con elementales operaciones, puede comprobarse que

$$\Delta E = -XM(\Delta Y)^T = - \sum_{j=1}^q \Delta y_j \sum_{i=1}^p m_{ij} x_i$$

a la vista de esta expresión, qué duda cabe de que nos encontramos, como era de esperar, en el mismo caso anterior. Para no ser reiterativos, resumiremos los resultados:

- 1.- $\Delta y_j = 0$, entonces evidentemente, $\Delta E = 0$.
- 2.- $\Delta y_j > 0$ (resulta ser $\Delta y_j = 2$) lo que supone $y_j^{\text{viejo}} = -1$ e $y_j^{\text{nuevo}} = +1$, para lo que se exige que $\sum_{i=1}^p m_{ij} x_i > 0$.
- 3.- $\Delta y_j < 0$ (como antes, $\Delta y_j = -2$) lo que supone $y_j^{\text{viejo}} = 1$ e $y_j^{\text{nuevo}} = -1$, para lo que se exige que $\sum_{i=1}^p m_{ij} x_i < 0$.

En definitiva, también en este caso podemos estar tranquilos de que el sistema evolucionará siempre hacia un punto de energía mínima, puesto que, como antes, la suma de los productos sigue siendo positiva, llevando a una evolución negativa de $-\Delta E$.

A partir de lo dicho, parece lo más conveniente extender el concepto de Energía Local a una función en la que participen las variaciones de ambos tipos de patrones, y así, siguiendo a Simpson [10], se acepta como tal definición la de

$$E = -XMY^T - YM^T X^T = -2 \sum_{h=1}^N (X \bullet X_h) (Y \bullet Y_h)$$

que más adelante resultará de utilidad.

En cuanto a la capacidad de una BAM, parece que hoy en día se acepta como válido el resultado alcanzado por Hecht-Nielsen, según el cual, si queremos asegurar que las parejas de patrones almacenadas serán siempre recordadas, debe cumplirse que su número será

$$N = \frac{r}{2 \log r}$$

donde $r = \min(p, q)$.

En relación con la estabilidad y la capacidad, se presenta un problema que hasta ahora estaba latente: aún sabiendo que frente a cualquier entrada la red evolucionará hacia un mínimo de energía, lo que no podemos asegurar es que ese mínimo corresponda, precisamente, a alguno de los patrones almacenados. Es decir, si tenemos una cierta pareja (A_i, B_i) , a la que asociamos una energía $E_i = -X_i M Y_i^T$, la situación llega al extremo de que no podemos asegurar que ante la entrada A_i (la mismísima entrada A_i empleada para definir a una de las asociaciones prototipo) se obtenga nunca la salida B_i .

Puede ocurrir que se obtenga, al cabo de las necesarias transiciones para llegar al punto estable, una salida C_i , es decir, su equivalente bipolar Z_i , para lo que bastará con que

$$-X_i M Z_i^T < -X_i M Y_i^T, \quad \forall Y_i$$

es decir, la BAM evoluciona espontáneamente hacia mínimos locales de energía, de forma independiente de como nosotros lo hayamos previsto.

Evidentemente, se trata de un grave problema que puede llegar a inutilizar por completo toda la memoria asociativa. Como soluciones, hoy en día se han planteado algunas como las basadas en extensiones de los patrones de entrada (Wang) o en mecanismos adaptivos que los autores están tratando de poner a punto.

3.5. BAMs de orden superior

Una de las grandes ventajas de las memorias de correlación consiste, sin duda alguna, en su rigor repetitivo a la hora de ampliar el campo de acción. Así, de la misma forma que sobre los autocorreladores hemos presentado los de orden superior como una simple extensión de aquellos, podremos ampliar los heterocorreladores con bastante sencillez.

Es evidente que el paso a las de segundo orden, abre el camino para órdenes aún mayores. En efecto, la regla que aquí podamos adoptar, deberá poder hacerse extensiva inmediatamente para órdenes tres y superiores. Para ello, debemos empezar por darnos cuenta de que no se trata de asociar tripletas de patrones, sino de aumentar el grado de asociación entre los mismos pares de vectores de entrada de los que se ocupaba la de primer orden.

Así las cosas, admitiendo que las dos redes vienen identificadas por **Plano X** y **Plano Y**, las entradas a cada uno de ellos, los patrones que ahora definen la nueva matriz deberán ser las tríadas (X_i, X_j, Y_k) y (X_i, Y_j, Y_k) . De esta forma, los elementos de las matrices, serán de dos tipos:

$$m_{ijk} = \sum_{h=1}^N x_i^h x_j^h y_k^h, \quad \forall i, j = 1, 2, \dots, p; \quad \forall k = 1, 2, \dots, q$$

para el primer tipo de asociación, y

$$w_{klj} = \sum_{h=1}^N y_k^h x_l^h x_j^h, \quad \forall k, l = 1, 2, \dots, q; \quad \forall j = 1, 2, \dots, p$$

para el segundo. Si deseáramos proseguir en el orden, la BAM de tercer orden se haría correlacionando tripletas de un tipo con unidades del otro, es decir, definiendo elementos matriciales de la forma

$$m_{ijkl} = \sum_{h=1}^N x_i^h x_j^h y_k^h y_l^h, \quad \forall i, j, k = 1, 2, \dots, p; \quad \forall l = 1, 2, \dots, q$$

para la primera asociación, y

$$w_{jkli} = \sum_{h=1}^N y_j^h y_k^h y_l^h x_i^h, \quad \forall j, k, l = 1, 2, \dots, q; \quad \forall i = 1, 2, \dots, p$$

para la segunda. En éstas deja verse la ausencia de las asociaciones por pares de elementos de cada uno de los conjuntos, es decir, dos del X_i con dos del Y_j . La razón radica en el hecho de que de esta forma sólo se consigue duplicar la BAM de primer orden sin efectuar nuevas asociaciones que pudieran llevarnos a mejores resultados en la recuperación de datos. Se trataría, en definitiva, de la doble aplicación de la BAM original, lo que no supone ningún adelanto respecto de aquella.

Ahora, sólo queda resolver el problema de la recuperación de información, es decir, debemos generar una regla totalmente genérica para que pueda extenderse a los casos de órdenes superiores. Siguiendo una metodología tensorial clásica, pero teniendo cuidado con las matrices que debemos emplear en cada caso, podemos decir que:

$$x_j^{\text{nuevo}} = F\left(\sum_{k=1}^q y_k \sum_{l=1}^q y_l w_{klj} x_j^{\text{anterior}}\right); \quad \forall j = 1, 2, \dots, p$$

y, de forma similar,

$$y_k^{\text{nuevo}} = F\left(\sum_{i=1}^p x_i \sum_{j=1}^p x_j m_{ijk}, y_k^{\text{anterior}}\right); \quad \forall k = 1, 2, \dots, q$$

esta forma de recuperar los patrones de cada capa es indicativa de cómo deberá hacerse en todos los casos, a lo que sólo podemos añadir que la de primer orden no es más que un caso particular suyo.

En lo que respecta a la **Convergencia y Estabilidad** de estas memorias, podemos partir, tal como Chen y otros han propuesto, del parámetro de energía de las BAM de primer orden, haciendo una extensión inmediata suya de la forma

$$E_n = -(E_{nY} + E_{nX})$$

$$= -\sum_{h=1}^m (X \bullet X_h)^n (Y \bullet Y_h) - \sum_{h=1}^m (X \bullet X_h)(Y \bullet Y_h)^n$$

Dada su estructura de separación de ambos contribuyentes energéticos, la variación del parámetro queda definida por

$$\Delta E_n = \Delta E_{nY} + \Delta E_{nX}$$

que para el caso de $n=2$, es relativamente fácil de comprobar. Su extensión a órdenes mayores puede encontrarse en el trabajo de Chen, y no resulta ya de interés para nosotros.

4. HETEROCORRELADORES

4.1. Auto-heterocorreladores (IBAM)

Una característica de las BAM, que parece (al menos en principio) bastante razonable, es la que consiste en el recuerdo automático de los negativos. Es decir, si decidimos almacenar la pareja (A_i, B_i) , y si llamamos A_i^C y B_i^C a sus complementarios binarios, la BAM asociará inmediatamente la pareja (A_i^C, B_i^C) .

Para estos casos, la solución evidente consiste en aumentar la dimensión del espacio de los vectores, por el simple procedimiento de convertir la BAM en un autocorrelador a base de generar como vectores el resultado de concatenar los de las parejas anteriores. En efecto, supongamos que antes habíamos deseado asociar A_i con B_i . Ahora construimos un auto-correlador con el prototipo $C_i = (A_i / B_i)$, es decir,

$$C_i = (c_1^i, \dots, c_r^i) = (a_1^i, \dots, a_p^i, b_1^i, \dots, b_q^i)$$

Merece la pena observar la matriz de autocorrelación resultante.

En la figura 4 puede verse su aspecto convenientemente esquematizado. En realidad, está compuesta por cuatro matrices de correlación: la matriz M_A corresponde a un autocorrelador del conjunto $\{A_i\}$ y la M_B al autocorrelador del conjunto $\{B_i\}$, mientras que M_C y M_D son el resultado de la heterocorrelación entre $\{A_i\}$ y $\{B_i\}$, en su forma directa y traspuesta respectivamente, es decir $M_D = M_C^T$. Recordando que

$$X_i = (x_1^i, \dots, x_p^i), \quad X_i \in \{-1, +1\}^p$$

$$Y_i = (y_1^i, \dots, y_q^i), \quad Y_i \in \{-1, +1\}^q$$

las cuatro matrices se definen a través de las siguientes ecuaciones:

$$M_A = \sum_{i=1}^N X_i^T X_i, \quad a_{jk} = \sum_{i=1}^N x_j^i x_k^i, \quad j, k = 1, 2, \dots, p$$

$$M_B = \sum_{i=1}^N Y_i^T Y_i, \quad b_{jk} = \sum_{i=1}^N y_j^i y_k^i, \quad j, k = 1, 2, \dots, q$$

$$M_C = \sum_{i=1}^N X_i^T Y_i, \quad c_{jk} = \sum_{i=1}^N x_j^i y_k^i, \quad j=1,2,\dots,p; \quad k=1,2,\dots,q$$

$$M_D = M_C^T$$

De esta forma, hemos conseguido aumentar la capacidad de la memoria, puesto que ahora los complementarios de los vectores de partida pueden asociarse de la manera que más nos interese. Para ello, sin embargo, no ha sido preciso aumentar el número de neuronas, sino (y ahí reside su importancia) su conexionado. En efecto, las matrices M_C y M_D se sintetizan mediante el esquema convencional de la BAM, propuesto en la figura 3, mientras que las A y B corresponden, como se deduce de la expresión de sus componentes a_{jk} y b_{jk} , a la estructura del autocorrelador, donde la primera se sintetiza con las «p» neuronas de la primera capa y la M_B con las «q» de la segunda. Se trata, por tanto, de añadir las intra-conexiones a las interconexiones de una BAM, lo que responde a la idea que se nos planteaba anteriormente. Además, por la razón del denso conexionado (inter e intra), suelen denominarse como IBAM. De todas formas, esto ha venido a corroborar algo que suele aceptarse sin discusión: la capacidad de memoria de una red neuronal reside, para un determinado número de neuronas, en la riqueza de la conectividad existente.

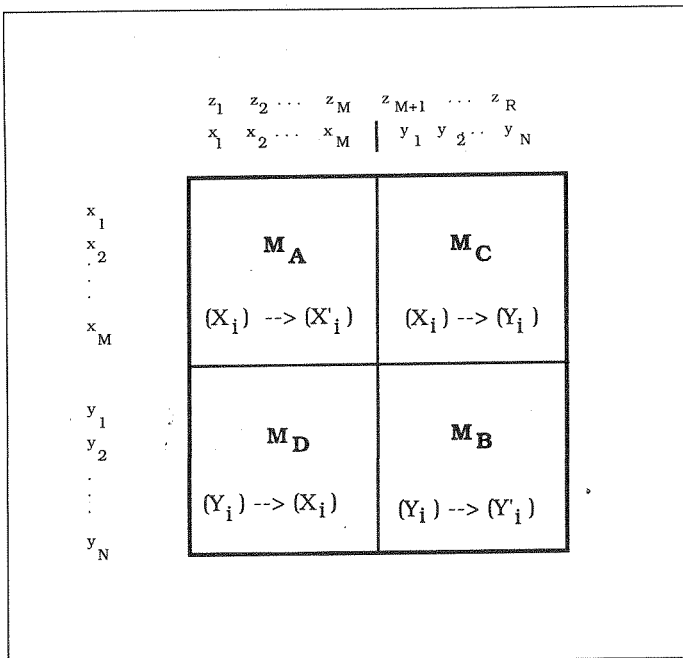
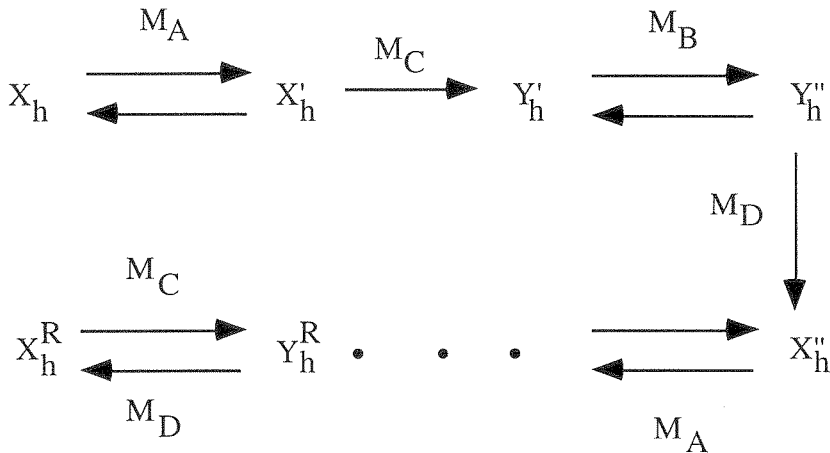


Figura 4. Matriz de la memoria IBAM.

El proceso de recuperación de información de una IBAM resulta algo más complejo que en los sistemas anteriores. En efecto: supongamos que recibimos una entrada A_h no coincidente con ninguno de los prototipos A_i . El mecanismo de reconocimiento y recuperación empezará por aplicar el equivalente bipolar, X_h , de la entrada a la primera capa de neuronas para obtener la respuesta del autocorrelador M_A , que denominaremos X'_h . Ahora utilizamos la estructura BAM de conectividad entre ambas capas, a través de M_C , que nos dará, Y'_h . A continuación pasamos al proceso de autocorrelación M_B , para obtener Y''_h , que mediante la BAM inversa, definida por M_D , nos ofrece el resultado X''_h .

El proceso sigue indefinidamente hasta que se alcanza un resultado estable, con lo que se da por finalizado, siendo Y^R_h la salida que el sistema nos genera como correspondiente a la entrada primitiva, y tal que $Y^R_h \in \{Y_i\}$, evidentemente. De forma esquemática, el mecanismo de lectura de información es



Esto significa que tenemos cuatro formas de recuperar información que se deben corresponder con otras tantas expresiones analíticas. Sin embargo, su formulación no es compleja toda vez que ya la hemos desarrollado para las BAM clásicas. En efecto, podemos establecer unas reglas para cada nivel de interacción, y así nos encontraremos con:

1.- Vector Y_i desde el plano de vectores X_j :

$$y_i^{\text{nuevo}} = F\left(\sum_{j=1}^p c_{ji} x_j, y_i^{\text{anterior}}\right), \quad \forall i = 1, 2, \dots, q$$

2.- Vector Y_i dentro de su propio plano de vectores Y_k :

$$y_i^{\text{nuevo}} = F\left(\sum_{j=1}^q b_{kj} y_k, y_i^{\text{anterior}}\right), \quad \forall i=1,2,\dots,q$$

3.- Vector X_j desde el plano de vectores Y_i :

$$x_j^{\text{nuevo}} = F\left(\sum_{i=1}^q c_{ij} y_i, x_j^{\text{anterior}}\right), \quad \forall i=1,2,\dots,p$$

4.- Vector X_j desde su propio plano de vectores X_k :

$$x_j^{\text{nuevo}} = F\left(\sum_{k=1}^p a_{kj} x_k, x_j^{\text{anterior}}\right), \quad \forall j=1,2,\dots,p$$

4.2. Estabilidad

Resulta un ejercicio notable el extender el concepto de parámetro de energía cinética de los anteriores esquemas de memorias hasta el caso que ahora nos ocupa. En efecto, aquí debemos considerar que cada uno de los procesos pone en juego su propia energía. De esta forma, la función a emplear es:

$$E = -XAX^T - XCY^T - YBY^T - YC^T X^T$$

lo cual implica que habrá que calcular las variaciones que se producen para cada uno de los dos vectores posibles, es decir, tendremos los casos

$$\Delta E_X = -(\Delta X)AX^T - (\Delta X)CY^T - YBY^T - YC^T(\Delta X)^T$$

y

$$\Delta E_Y = -XAX^T - XC(\Delta Y)^T - (\Delta Y)BY^T - (\Delta Y)C^T X^T$$

para los que hay que demostrar su constante descenso. Sin embargo, podemos simplificar de forma notable ese cálculo, puesto que los procesos no son simultáneos, y así, por ejemplo, fijándonos sólo en las variaciones de los vectores X (el caso es idéntico para los Y), podemos darnos cuenta de que hay dos formas de producirse su variación: en la intraconexión (plano X sobre plano X), y en la interconexión (plano Y sobre plano X), con lo cual llegamos a que

$$\Delta E_X^{\text{intra}} = -(\Delta X)AX^T \quad \text{y que} \quad \Delta E_X^{\text{inter}} = -(\Delta X)CY^T$$

para cada una de ambas situaciones. Su cálculo analítico es simple a la vista de las definiciones adoptadas para las distintas matrices, resultando ser

$$\Delta E_X^{\text{intra}} = - \sum_{j=1}^p (\Delta x_j) \sum_{k=1}^p x_k a_{jk} \leq 0$$

$$\Delta E_X^{\text{inter}} = - \sum_{j=1}^p (\Delta x_j) \sum_{k=1}^q y_k c_{jk} \leq 0$$

para asegurar su valor menor que cero nos hemos apoyado, según recordamos, en lo dicho para las BAM.

Dado que lo que es válido para los vectores X lo es también para los Y, queda claro que el resultado total es el de una variación negativa en el proceso de reconocimiento, o nula en el caso de haber alcanzado la estabilidad.

4.3. Síntesis neuronal

La síntesis neuronal, más simplificada aún que las anteriores por razones de claridad, es la representada en la figura 5, donde se aprecian los cuatro buses de conexionado posibles entre las dos capas. No merece la pena insistir en el tema por cuanto se trata de una extensión que, en la realidad, traerá mucha más complicación a la hora de tratar de implementarla, que en su estudio teórico.

4.4. IBAM de orden superior

A pesar de tratarse de la memoria asociativa más complicada, su análisis no lo es en proporción puesto que toda su estructura, desde el mismo momento de su definición, está basada en las de las dos anteriores, dado que lo que hemos venido tratando es de generar una memoria dotada del diagrama de conectividad más amplio que se nos pueda ocurrir.

En esa línea, podemos imaginar que una memoria real (¿biológica quizás?) será un modelo perfecto de hasta donde puede llegarse en la construcción de sistemas asociativos. Así, resulta fácil imaginar que un caso de altas «prestaciones» estará constituido como una gran IBAM en la que las interacciones se realicen mediante autocorreladores de órdenes muy elevados, y las interconexiones lo sean entre distintas BAM de órdenes, asimismo, muy elevados.

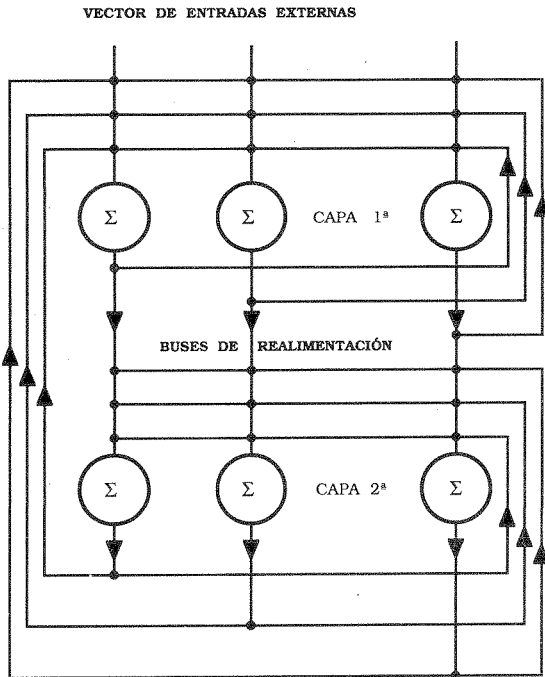


Figura 5. Estructura simplificada de una IBAM.

Si siguiendo en esa línea de complejidad creciente, podemos pensar, sin salirnos de la teoría aquí expuesta, que podríamos encontrar con IBAM de orden superior formando parte de otras IBAM de órdenes aún mayores (o por lo menos, iguales), según un esquema que puede repetirse hasta el infinito, limitándose, exclusivamente, por razón de las disponibilidades materiales a las que estemos sujetos. Queda a la imaginación de cada uno el punto de capacidad y eficacia que puede llegar a alcanzarse.

5. REALIZACIÓN PRÁCTICA

5.1. Un desarrollo práctico

La construcción práctica de una memoria asociativa ha de pasar, obviamente, por la de una Red Neuronal con la que sintetizarla. Para ello, hay que empezar por escoger un sistema microordenador capaz de multiproceso, es decir, un equipo en el que varias CPU puedan compartir los recursos de memoria y de entrada/salida generales. Aunque no es obligatorio, parece lo más adecuado el funcionamiento

pseudo-síncrono, consistente en implementar cada una de las capas de neuronas de la BAM en una placa de CPU, de forma que los valores de salidas se actualicen independientemente en una placa de la otra. De esta forma, se permite cierta flexibilidad a la hora de efectuar el obligado sincronismo total. Esta idea pertenece ya a las primeras realizaciones sugeridas por el propio Hopfield.

En nuestro caso, además, hemos recurrido a un sistema flexible dotado de un Sistema Operativo en tiempo real con posibilidad de multiproceso, que dispusiera de un lenguaje ensamblador fácilmente manejable, con vistas a generar rápidas y eficaces subrutinas con las que acelerar la programación general, bien en C bien, incluso, en ensamblador. Como solución, hemos escogido un equipo con bus VME, sistema operativo multitarea/multiprocesador convencional que emplea CPU tipo MC68040 con controlador de memoria virtual. Con este sistema, la inclusión de varias CPU en paralelo es casi inmediato, de forma que el único inconveniente posible estaría en las contenciones para los accesos a los recursos generales de memoria y de entrada/salida.

Para este problema, el equipo escogido dispone de buses adicionales privados que permiten, mediante la adecuada distribución de las memorias realizar accesos simultáneos en casos de interés. Por lo demás, el conjunto resultante es totalmente clásico y queda constituido tal como indica la figura 6, en la que solamente hemos presentado una CPU, que puede ser el caso práctico para definiciones no excesivas de la dimensión de los vectores.

A pesar de que el funcionamiento de la BAM es independiente del tipo y origen de los vectores que se manejan, puesto que al fin y al cabo se trata solamente de conjuntos binarios ordenados, siempre resulta conveniente disponer de algún método directo de comprobación. Así, y tal como aparece en la figura 6, la entrada y salida se realiza a través de una unidad de video (con sus correspondientes cámara y monitor) conectada al bus VME. De esta forma, y mediante los programas de acoplo adecuados, los vectores responden a caracteres gráficos fácilmente reconocibles. En el caso particular construido, la retícula utilizada ha sido de hasta 100 x 100, lo que supone un total máximo de 10.000 neuronas por capa, siendo iguales las dimensiones de todos los vectores tratados, con lo que además, la matriz de correlación es cuadrada.

En el diagrama de funcionamiento se describen los pasos fundamentales de la programación software de todo el sistema, contemplando la ejecución del proceso de la BAM. A este respecto, merece la pena destacar que el punto 4.4 es el que admite más adaptaciones posteriores. En efecto, si el proceso se realiza normalmente, tras una serie de pasos (trayectoria) en los que la energía disminuye, se alcanzará el punto estable final. Tal y como se demostró, si se produjera algún cambio positivo de la energía, ello sería indicativo de que algo no funciona bien, y el proceso deberá

interrumpirse para recuperar el proceso normal. La dificultad sólo debe presentarse a la salida, es decir, cuando se alcance el mínimo de energía. En este momento, hay que detectar qué vector de los $\{A_i\}$ posibles se ha alcanzado, puesto que si no es así, quiere decir que nuestra asociación inicial no era correcta.

Para este caso, tal y como ya indicamos en su momento, habrá que disponer de los algoritmos de corrección adecuados que nos permitan alterar la matriz de forma conveniente hasta llegar a algún resultado satisfactorio. Los autores están diseñando en la actualidad una estrategia adaptativa que convierte el proceso en un mecanismo autoadaptativo con lo que se alcanza lo que podríamos llamar una estructura dinámica de reconocimiento, haciendo uso de las propiedades de aprendizaje de las redes neuronales. Es un camino abierto en el que aparecen constantemente nuevas soluciones.

5.2. Diagrama de funcionamiento de una BAM

- 1.- **Simulación de la Red Neuronal:**
 - Distribución de conexionado
 - Rutinas funcionales
 - Asignación memorias E/S de datos

- 2.- **Programación del sistema de Video**
 - Especificación Entrada/Salida de vectores

- 3.- **Especificación de la asociación de pares de vectores:**
 - Entrada (bajo control del usuario) de pares de caracteres
 - Cálculo de la matriz
 - Distribución de los pesos sinápticos

- 4.- **Entrada desconocida:**
 - 4.1.- Cálculo de la energía
 - 4.2.- Aplicación $x \rightarrow y$ de la Red (1ª Capa)
 - 4.3.- Aplicación $y \rightarrow x$ de la Red (2ª Capa)
 - 4.4.- Cálculo de la nueva energía:
 - 4.4.1.- Si ha disminuido pasar al punto 4.2
 - 4.4.2.- Si ha aumentado generar error
 - 4.4.3.- Si permanece constante, salida final

AGRADECIMIENTO

El autor desea hacer constar su agradecimiento a la C.I.C.Y.T. por su ayuda del programa T.I.C. 92-0254 en cuyo desarrollo se han llevado a cabo los trabajos que dieron lugar al estudio expuesto en este capítulo.

REFERENCIAS

- Amari, Sun-Ichi y Maginu, Kenjiro, «Statistical Neurodynamics of Associative Memory», *Neural Networks*, Vol. 1, N. 1, (1988), 63-73.
- Chen, H., Lee, Y., Sun, G., Lee, H., Maxwell, T. y Giles, C., «Higher Order Correlation Model for Associative Memory», en Denker, J. (Ed.), *AIP Conference Proceedings 151: Neural Networks for Computing*, 1986, 86-99.
- Fukushima, K., «A Neural Network Model for Selective Attention in Visual Pattern Recognition and associative recall», *Applied Optics*, Vol. 26, N. 23, noviembre, (1987), 4985-4992.
- Grossberg, S. y Carpenter, G., «The ART of Adaptive Pattern Recognition by a Self-Organizing Neural Network», *IEEE Computer.*, Vol. 21, N. 3, marzo, (1988).
- Hebb, D.O., *The Organization of behaviour*, Nueva York, John Wiley, 1949.
- Hopfield, J., Feinstein, D. y Palmer, R., «Unlearning has a stabilizing effect in collective memories», *Nature*, Vol. 304, (1983), 158-159.
- Kohonen, T., *Self-Organization and Associative Memory*, Berlin Heidelberg, New York, Springer Verlag, 1988²
- Kosko, B., «Constructing an Associative Memory», *BYTE*, Vol. 12, N. 9, septiembre, (1987):
- McEliece, R.J., Posner, E.C., Rodemich, E.R. y Venkatesh, S.S., «The Capacity of the Hopfield Associative Memory», *IEEE Transactions on Information Theory*, Vol. 33, julio, (1987).
- Simpson, P.K., «Higher-Ordered and Intraconnected Bidirectional Associative Memories», *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 20, N. 3, mayo/junio, (1990), 637-653.
- Verleysen, M., Sirletti, B. y Jespers, P.A., «Neural Networks for High-Storage Content-Addressable Memory: VLSI Circuit and Learning Algorithm», *IEEE Journal of Solid-State Circuits*. Vol. 24, N. 3, (1989).
- Wang, Y.-F., Cruz, J.B. y Mulligan, J.H., «Two Coding Strategies for Bidirectional Associative Memory», *IEEE Transactions on Neural Networks*, Vol. 1, N. 1, marzo, (1990), 81-92.
- Xu, X. y Tsai, W.T., «Constructing Associative memories Using Neural Networks», *Neural Networks*, Vol. 3, (1990), 301-309.