

RESEACH ON COMPUTING SCIENCE

Introducción a las  
**Memorias  
Asociativas**

---

**Cornelio Yáñez Márquez  
Juan Luis Díaz de León Santiago**

**Instituto Politécnico Nacional  
Centro de Investigación en Computación**





# **Introducción a las Memorias Asociativas**

---

# **Research on computing science**

---

**Edition**

**Ignacio García Aráoz**

**Cover Design**

**Paula León Alvarado**

---

**Volumen 6**

---

# **Introducción a las Memorias Asociativas**

**Cornelio Yáñez Márquez**

**Juan Luis Díaz de León Santiago**

**Instituto Politécnico Nacional  
Centro de Investigación en Computación  
México 2003**





**ISBN: 970-36-0116-2**

---

Copyright © by Instituto Politécnico Nacional

Published by Instituto Politécnico Nacional  
Centro de Investigación en Computación  
Av. Juan de Dios Batíz s/n Esq. M. Othón de Mendizábal  
Unidad Profesional "Adolfo López Mateos", Zacatenco  
07738, México D.F., MÉXICO.

<http://www.ipn.mx>  
<http://www.cic.ipn.mx>

The editors and Publisher of this book have made their best effort in preparing this book, but make no warranty of any kind, expressed or implied, with regard to the information contained in this book.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of Instituto Politecnico Nacional. Except for personal or classroom use provided that copies bear the full citation notice provided on the first page of each paper

**Printing: 500**  
**Printed in Mexico**

# Índice general

|  |             |
|--|-------------|
| <b>Prefacio</b>  | <b>VII</b>  |
| <b>Agradecimientos</b>   | <b>XI</b>   |
| <b>Simbología</b>  | <b>XIII</b> |
| <b>1. Introducción.</b>  | <b>1</b>    |
| <b>2. Lernmatrix.</b>  | <b>9</b>    |
| 2.1. Contexto. . . . .   | 9           |
| 2.2. El trabajo de Steinbuch. . . . .                            | 9           |
| 2.3. Consideraciones finales del capítulo. . . . .               | 18          |
| <b>3. Correlograph.</b>  | <b>21</b>   |
| 3.1. Contexto. . . . .   | 21          |
| 3.2. El trabajo de Willshaw, Bunceman & Longuet-Higgins. . . . . | 21          |
| 3.3. Consideraciones finales del capítulo. . . . .               | 30          |
| <b>4. Linear Associator.</b>                                     | <b>31</b>   |
| 4.1. Contexto. . . . .   | 31          |
| 4.2. El trabajo de Anderson y Kohonen. . . . .                   | 32          |
| 4.3. Consideraciones finales del capítulo. . . . .               | 39          |
| <b>5. Amari.</b>   | <b>41</b>   |
| 5.1. Contexto. . . . .   | 41          |
| 5.2. El trabajo de Amari. . . . .                                | 42          |
| 5.3. Consideraciones finales del capítulo. . . . .               | 51          |
| <b>6. Memoria Asociativa Hopfield.</b>                           | <b>53</b>   |
| 6.1. Contexto. . . . .   | 53          |

|  |            |
|--|------------|
| 6.2. El trabajo de Hopfield. . . . .   | 54         |
| 6.3. Consideraciones finales del capítulo. . . . .   | 65         |
| <b>7. ADAM, BAM y SDM</b>  | <b>67</b>  |
| 7.1. Advanced Distributed Associative Memory . . . . .   | 67         |
| 7.2. Bidirectional Associative Memory . . . . .  | 70         |
| 7.3. Sparse Distributed Memory . . . . .   | 73         |
| 7.4. Consideraciones finales del capítulo. . . . .   | 77         |
| <b>8. Memorias Asociativas Morfológicas</b>  | <b>79</b>  |
| 8.1. Contexto. . . . .   | 79         |
| 8.2. Memorias Morfológicas Heteroasociativas. . . . .  | 80         |
| 8.2.1. Memorias heteroasociativas <i>max</i> . . . . .   | 83         |
| 8.2.2. Memorias heteroasociativas <i>min</i> . . . . .   | 89         |
| 8.3. Memorias Morfológicas Autoasociativas. . . . .  | 92         |
| 8.3.1. Memorias autoasociativas <i>max</i> . . . . .   | 93         |
| 8.3.2. Memorias autoasociativas <i>min</i> . . . . .   | 96         |
| 8.4. Consideraciones finales del capítulo. . . . .   | 99         |
| <b>9. Memorias Asociativas <math>\alpha\beta</math></b>  | <b>101</b> |
| 9.1. Herramientas Matemáticas . . . . .  | 102        |
| 9.1.1. Operaciones Binarias $\alpha$ y $\beta$ . . . . .   | 102        |
| 9.1.2. Operaciones Matriciales $\Psi_\alpha$ , $\mathfrak{M}_\alpha$ , $\Psi_\beta$ y $\mathfrak{M}_\beta$ . . . . . | 103        |
| 9.1.3. Relaciones de Orden y Tipos de Ruido . . . . .  | 105        |
| 9.2. El modelo de las Memorias Asociativas $\alpha\beta$ . . . . .   | 107        |
| 9.2.1. Memorias Heteroasociativas $\alpha\beta$ . . . . .  | 107        |
| 9.2.2. Memorias Autoasociativas $\alpha\beta$ . . . . .  | 118        |
| 9.3. Densidad Aritmética . . . . .   | 122        |
| 9.4. Consideraciones finales del capítulo. . . . .   | 123        |
| <b>A. Pruebas de convergencia.</b>   | <b>125</b> |
| A.1. Contexto. . . . .   | 125        |
| A.2. Los desarrollos. . . . .  | 128        |
| A.2.1. Hopfield . . . . .  | 128        |
| A.2.2. McEliece et. al. . . . .  | 132        |
| A.3. Consideraciones finales del apéndice. . . . .   | 136        |



# Prefacio

*Es mejor equivocarse por actuar y experimentar, que salvarse del error al precio de no hacer nada.*

Lema del GRAI<sup>1</sup>.

El propósito de los autores al presentar esta obra, es poner en manos del amable lector un trabajo conciso, serio y profesional que cubre los principales hechos históricos y los modelos más importantes de memorias asociativas que han surgido, a nivel mundial, desde la década de los sesenta.

Las memorias asociativas han mantenido su presencia en los escenarios de la investigación científica durante varias décadas, y en la actualidad es uno de los temas relevantes que aparecen en publicaciones institucionales, libros y revistas de reconocido nivel y prestigio internacional; además, las memorias asociativas son tema de interés en diferentes eventos nacionales e internacionales, en congresos mundiales y en reuniones de científicos e ingenieros.

A medida que se avanza en las investigaciones se obtienen resultados notables (impresionantes, algunos de ellos), y a su vez esos nuevos modelos generan conocimientos frescos que sirven de materia prima para que otros investigadores, a veces en laboratorios alejados de los primeros, realicen nuevos avances y descubrimientos; así es el fascinante mundo de la investigación en memorias asociativas.

Como se menciona en la cita inicial y en el pie de página, los autores somos fundadores del GRAI, grupo cuyo líder es uno de los autores de esta

---

<sup>1</sup> Los autores son fundadores del GRAI: Grupo de Robótica y Análisis de Imágenes, sito en el Centro de Investigación en Computación del Instituto Politécnico Nacional.

obra, y que se ha dado a la tarea de dar a conocer en México los novedosos métodos de la morfología matemática, entre otros temas de interés. Al aplicar esta rama de la ciencia a las memorias asociativas, surgieron las memorias asociativas morfológicas, que han impulsado de manera ostensible el interés en el área de las memorias asociativas.

El libro está conformado por nueve capítulos y un apéndice. El capítulo 1 contiene la imprescindible introducción, donde se contextualiza históricamente el desarrollo de las memorias asociativas, se plantea su funcionamiento y se introduce al lector en el lenguaje y la notación propias de esta área del saber humano.

Los capítulos 2 y 3 contienen dos de los trabajos pioneros más importantes en este campo, que constituyen antecedentes cruciales para el desarrollo de los modelos actuales de memorias asociativas: la *Lernmatrix*, ideada y desarrollada por el científico alemán Karl Steinbuch en 1961, y un trabajo pionero, notable por su poderosa sencillez: el *Correlograph*, dispositivo que fue desarrollado en Inglaterra por los científicos Willshaw, Buneman & Longuet-Higgins en 1969.

Los siguientes dos capítulos, el 4 y el 5, exhiben la trascendental producción de la generación 1972: el *Linear Associator*, un modelo de memoria asociativa desarrollada de manera independiente por dos notables científicos, James A. Anderson y Teuvo Kohonen, y el bosquejo de algunas de las ideas que aparecen en los trabajos del preclaro y prolífico teórico japonés Shun-ichi Amari, quien ha desplegado una febril actividad, trabajando en la *University of Tokyo* y en el Laboratory For Mathematical Neuroscience del Brain Science Institute de Japón (RIKEN), en busca de la creación de nuevos conceptos y la formación de recursos humanos de alto nivel en las áreas de redes neuronales y memorias asociativas.

El capítulo 6 aborda uno de los trabajos clásicos en memorias asociativas, la Memoria Asociativa Hopfield, cuyo autor se ha convertido en algo así como una leyenda viviente. Las razones de esta peculiar situación obedecen a que, entre otras cosas, algunos de los conspicuos autores científicos que publican regularmente en las áreas de redes neuronales, física y memorias asociativas han declarado, de manera muy seria que este personaje, el físico norteamericano John Hopfield, es el responsable de haber revivido el interés, a principios de la década de los ochenta del siglo XX, en las redes neuronales y en particular en las memorias asociativas, después de sufrir un

estancamiento de más de una década. En el apéndice A se presentan dos demostraciones de un teorema relacionado íntimamente con el trabajo de Hopfield.

El capítulo 7 tiene como propósito incluir tres modelos de memoria asociativa que no son esenciales para entender el desarrollo de los modelos de memorias asociativas, pero que comparten una característica común: son extensiones de modelos conocidos. El ADAM de Austin se puede considerar como una extensión de las redes asociativas de Willshaw; la BAM de Kosko, que surgió en pleno auge de los intentos realizados por algunos grupos de investigación con miras a minimizar una de las desventajas de la memoria Hopfield, se puede considerar como una extensión ingeniosa de esta última; y finalmente, la SDM de Kanerva está basada en la estructura de una RAM usual, y puede considerarse como una generalización de los conceptos involucrados en una RAM

Entre la miríada de aportaciones e innovaciones en el campo de las memorias asociativas, después del espectacular éxito de la memoria Hopfield de 1982, no sucedió nada realmente trascendente hasta 1998, cuando aparecieron las memorias asociativas morfológicas. La diferencia fundamental entre estas memorias y las memorias asociativas clásicas como el *Linear Associator* y la Memoria Hopfield, es que mientras éstas basan su operación en la suma y multiplicación usuales, las memorias morfológicas se basan en las operaciones morfológicas de dilatación y erosión. En otras palabras, mientras que las memorias asociativas clásicas echan mano de la suma de productos, las memorias asociativas morfológicas hacen uso de máximos o mínimos de sumas.

Los capítulos 8 y 9 constituyen un colofón de lujo, pues en éstos se explican los orígenes, conceptos teóricos y funcionamiento de las memorias asociativas morfológicas de Ritter & Díaz de León y de las memorias asociativas  $\alpha\beta$  de Yáñez & Díaz de León. Ambos modelos representan importantes vetas de investigación en temas que se encuentran en la frontera del conocimiento en el área de las memorias asociativas.

Con esta obra, los autores hemos contribuido con lo que dicta el lema del IPN: "La Técnica al Servicio de la Patria".

**Cornelio Yáñez Márquez**

**Juan Luis Díaz de León Santiago**





# Agradecimientos

Los autores agradecen el apoyo de las siguientes instituciones:

- Secretaría Académica del Instituto Politécnico Nacional;
- COFAA: Comisión de Operación y Fomento de Actividades Académicas del Instituto Politécnico Nacional;
- CONACyT: Consejo Nacional de Ciencia y Tecnología; y
- Sistema Nacional de Investigadores.

Esta obra es un producto del Proyecto de Investigación:

**Memorias Asociativas Morfológicas**

El proyecto fue auspiciado por la Coordinación General de Posgrado e Investigación del Instituto Politécnico Nacional.

Clave del Proyecto: **990327**



# Simbología

|  |  |
|--|--|
| memorias $\alpha\beta$                         | las nuevas memorias asociativas, producto de esta tesis          |
| $\alpha, \beta$                                | operadores originales en que se basan las memorias $\alpha\beta$ |
| $M$  | memoria asociativa, memoria asociativa morfológica <i>max</i>    |
| $W$  | memoria asociativa morfológica <i>min</i>                        |
| $m_{ij}$                                       | $ij$ -ésima componente de la memoria asociativa $M$              |
| $B$  | conjunto al que pertenecen las componentes $m_{ij}$              |
| $\in$  | pertenencia de un elemento a un conjunto                         |
| $x$  | vector columna que corresponde a un patrón de entrada            |
| $y$  | vector columna que corresponde a un patrón de salida             |
| $A$  | conjunto al que pertenecen las componentes de $x$ y $y$          |
| $(x, y)$                                       | asociación de un patrón de entrada con uno de salida             |
| $(x^k, y^k)$                                   | asociación de la $k$ -ésima pareja de patrones                   |
| $\{(x^\mu, y^\mu) \mid \mu = 1, 2, \dots, p\}$ | conjunto fundamental   |
| $\tilde{x}^k$                                  | versión alterada del patrón fundamental $x^k$                    |
| $x_j^\mu$                                      | $j$ -ésima componente de un vector columna $x^\mu$               |
| $(x^\mu)^t$                                    | transpuesto del vector $x^\mu$                                   |
| $n$  | dimensión de los patrones de entrada                             |
| $m$  | dimensión de los patrones de salida                              |
| $p$  | número de parejas del conjunto fundamental                       |
| $\Delta m_{ij}$                                | incremento en $m_{ij}$   |
| $\vee$   | operador máximo  |
| $\wedge$                                       | operador mínimo  |
| $\cdot$  | producto usual entre vectores o matrices                         |
| $\times$                                       | producto cruz (entre conjuntos)                                  |
| $\delta_{ij}$                                  | delta de Kronecker (afecta a los índices $i$ y $j$ )             |
| $\nabla$                                       | producto máximo (entre matrices)                                 |
| $\triangle$                                    | producto mínimo (entre matrices)                                 |

|                             |  |
|-----------------------------|--|
| $\alpha(x, y)$              | operación binaria $\alpha$ con argumentos $x$ y $y$                                    |
| $\beta(x, y)$               | operación binaria $\beta$ con argumentos $x$ y $y$                                     |
| $P = [p_{ij}]_{m \times r}$ | matriz $P$ de dimensiones $m \times r$ y componente $ij$ -ésima $p_{ij}$               |
| $\forall$                   | cuantificador universal  |
| $\exists$                   | cuantificador existencial  |
| $\Psi_{\alpha}$             | operador $\alpha$ max  |
| $\Psi_{\beta}$              | operador $\beta$ max   |
| $\mathfrak{M}_{\alpha}$     | operador $\alpha$ min  |
| $\mathfrak{M}_{\beta}$      | operador $\beta$ min   |
| $\boxtimes$                 | símbolo que representa a las dos operaciones $\Psi_{\alpha}$ y $\mathfrak{M}_{\alpha}$ |
| $V$                         | memorias $\alpha\beta$ tipo $V$  |
| $\Lambda$                   | memorias $\alpha\beta$ tipo $\Lambda$  |
| $\otimes$                   | operación para calcular la distancia de Hamming  |
| $\oplus$                    | operación lógica or exclusiva  |

# Capítulo 1

## Introducción.

El tema de las memorias asociativas ha estado vigente, desde hace varios lustros, dentro de algunas áreas de investigación en diversas latitudes del orbe. El propósito fundamental de una memoria asociativa es recuperar correctamente patrones completos a partir de patrones de entrada, los cuales pueden estar alterados con ruido aditivo, sustractivo o combinado: ésta es la característica más atractiva de las memorias asociativas.

Notables investigadores han abordado el problema de generar modelos de memorias asociativas (Steinbuch, 1961; Steinbuch & Frank, 1961; Willshaw, Buncman & Louguet-Higgins, 1969; Amari, 1972; Anderson, 1972; Kohonen, 1972; Nakano, 1972; Kohonen & Ruohonen, 1973; Kohonen, 1974; Little & Shaw, 1975; Anderson, Silverstein, Ritz & Jones, 1977; Amari, 1977; Hopfield, 1982; Hopfield, 1984; Austin, 1987; Kanerva, 1988; Kolen, & Pollack, 1991; Buhmann, 1995; Kinsler, 1995; Bandyopadhyay, & Datta, 1996; Alexander & Morton, 1997; Austin, Buckle, Kennedy, Moulds, Pack & Turner, 1997; Ritter, Sussner & Diaz-de-Leon, 1998; Ritter, Diaz-de-Leon & Sussner, 1999), y han logrado resultados de importancia tal, que algunos de los trabajos pioneros (décadas 1970 y 1980) se han convertido en auténticos clásicos.

La capacidad de aprendizaje y almacenamiento, la eficiencia en la respuesta o recuperación de patrones, la rapidez y la inmunidad al ruido, son tópicos de interés entre los investigadores que se dedican a estudiar estos modelos con el fin de proponer variaciones y generalizaciones que, a la postre, se traduzcan en nuevos modelos de memorias asociativas con ventajas claras sobre los modelos conocidos y que, además, sean propicios para su aplicación directa en problemas reales.

Así, se han propuesto esquemas alternativos ante los clásicos (o variaciones de éstos), cuyos autores aseguran que sus modelos de memorias asociativas hacen gala de gran capacidad de almacenamiento (Chen & Honavar, 1995; Graham & Willshaw, 1995; Imada & Araki, 1995; Adcodato & Taylor, 1996; Storkey 1997; Bosch & Kurfess, 1998; Jagota, Narasimhan & Regan, 1998); exhiben más rapidez que otras (Kinser, 1995); prueban ser eficientes en cuanto a la recuperación de patrones (Hely, Willshaw & Hayes, 1997; Sommer & Palm, 1998) o se han implementado en hardware (Kennedy, Austin & Cass, 1995; Krikelis, & Weems, 1997; Stright, Coffield, & Brooks, 1998).

Existen equipos de investigación que han dirigido sus esfuerzos a incorporar las bondades de las técnicas alcatóricas en los esquemas para memorias asociativas (Kanerva, 1988; Browne, & Aleksander, 1996; Villanueva, & Figueroa, 1998). Las indefectibles aplicaciones de las memorias asociativas también han dado lugar a relevantes proyectos y trabajos de investigación (Herrmann, & Sodini, 1992; Schwenk & Milgram, 1995; Gori, Lastrucci & Soda 1996; Silver, Glover & Stonham, 1996; Jørgensen 1997; Hattori & Hagiwara, 2000).

En esta vorágine de aportaciones e innovaciones, la aparición, desarrollo, aplicaciones y consolidación de las memorias asociativas morfológicas representó un salto cualitativo en la concepción y desarrollo de las memorias asociativas, por las características propias de las operaciones morfológicas, que permitieron ventajas de las memorias asociativas morfológicas respecto de los modelos conocidos (Ritter, Sussner & Diaz-de-Leon, 1998; Ritter, Diaz-de-Leon & Sussner, 1999); con objeto de proporcionar una muestra de la magnitud de este salto cualitativo, podemos mencionar que al funcionar en uno de los posibles modos (autoasociativo), las memorias asociativas morfológicas tienen respuesta perfecta y capacidad infinita de aprendizaje y almacenamiento (Díaz-de-León & Yáñez, 1999). Un hecho que puede ser importante en el contexto de las potencialidades de la tecnología de punta a niveles de seguridad nacional, es que investigadores del *Air Force Research Lab.* de los Estados Unidos implementaron las memorias asociativas morfológicas en hardware VLSI (Stright, Coffield, & Brooks, 1998).

Las memorias asociativas morfológicas inspiraron la aparición de las memorias asociativas  $\alpha\beta$  (alfa-beta), las cuales exhiben ventajas claras sobre las primeras, entre las que destaca una menor densidad aritmética (Yáñez, 2002; Yáñez & Díaz de León, 2003). Hoy por hoy, las memorias asociativas  $\alpha\beta$  constituyen un modelo que se encuentra en la frontera del conocimiento en este importante tema.

El propósito de este capítulo es plantear de manera diáfana y concisa el problema inherente al funcionamiento de las memorias asociativas. Por su naturaleza, este problema se escinde en dos fases claramente distinguibles:

1. Fase de *aprendizaje* (generación de la memoria asociativa)
2. Fase de *recuperación* (operación de la memoria asociativa)

Para estar en condiciones de realizar el planteamiento del problema, es preciso previamente proporcionar los conceptos básicos, las notaciones y la nomenclatura relacionados con el diseño y funcionamiento de las memorias asociativas.

Los conceptos básicos son conocidos tres décadas ha, y se presentan como originalmente fueron establecidos en las referencias (Kohonen, 1972, 1977, 1987, 1989; Anderson, 1972; Anderson & Bower, 1977; Anderson & Rosenfeld, 1990; Hassoun, 1993, 1995, 1997).

Respecto de las notaciones y nomenclatura que se utiliza en esta obra, en algunos casos se han realizado adaptaciones de la literatura incluida en la bibliografía, y en otros, para efectos de sencillez y facilidad de manipulación, se establecen arbitrariamente, en la inteligencia de que invariablemente se opta por aquellas notaciones que mejor reflejan, a juicio de los autores, los conceptos allí representados. La notación y nomenclatura que aquí se presenta es consistente a lo largo de todo el libro.

Como se mencionó arriba, el propósito fundamental de una memoria asociativa es recuperar patrones completos a partir de patrones de entrada que pueden estar alterados con ruido aditivo, sustractivo o combinado. De acuerdo con esta afirmación, una *memoria asociativa*  $M$  puede formularse como un sistema de entrada y salida, idea que se esquematiza a continuación:

$$\mathbf{x} \longrightarrow \boxed{M} \longrightarrow \mathbf{y}$$

El *patrón de entrada* está representado por un vector columna denotado por  $\mathbf{x}$  y el *patrón de salida*, por el vector columna denotado por  $\mathbf{y}$ .

Cada uno de los patrones de entrada forma una *asociación* con el correspondiente patrón de salida. La notación para una asociación es similar

a la de una pareja ordenada: por ejemplo, los patrones  $\mathbf{x}$  y  $\mathbf{y}$  del esquema forman la asociación  $(\mathbf{x}, \mathbf{y})$ .

Para facilitar la manipulación algebraica de los patrones de entrada y de salida, los denotaremos con las mismas letras negrillas.  $\mathbf{x}$  y  $\mathbf{y}$ , agregándoles números naturales como superíndices para efectos de discriminación simbólica. Por ejemplo, a un patrón de entrada  $\mathbf{x}^1$  le corresponderá un patrón de salida  $\mathbf{y}^1$ , y ambos formarán la asociación  $(\mathbf{x}^1, \mathbf{y}^1)$ ; del mismo modo, para un número entero positivo  $k$  específico, la asociación correspondiente será  $(\mathbf{x}^k, \mathbf{y}^k)$ .

La memoria asociativa  $M$  se representa mediante una matriz cuya componente  $ij$ -ésima es  $m_{ij}$  (Palm, Schwenker, Sommer & Strey, 1997); la matriz  $M$  se genera a partir de un conjunto finito de asociaciones conocidas de antemano: éste es el *conjunto fundamental de asociaciones*, o simplemente *conjunto fundamental*. Se denota por  $p$  la cardinalidad del conjunto fundamental ( $p$  es un número entero positivo).

Si  $\mu$  es un índice, el conjunto fundamental se representa de la siguiente manera:

$$\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mu = 1, 2, \dots, p\}$$

A los patrones que conforman las asociaciones del conjunto fundamental, se les llama *patrones fundamentales*.

La naturaleza del conjunto fundamental proporciona un importante criterio para clasificar las memorias asociativas. Si se cumple que  $\mathbf{x}^\mu = \mathbf{y}^\mu \forall \mu \in \{1, 2, \dots, p\}$ , se dice que la memoria es *autoasociativa*; de otro modo, la memoria es *heteroasociativa* (Kohonen, 1972). Para una memoria heteroasociativa se puede afirmar lo siguiente:  $\exists \mu \in \{1, 2, \dots, p\}$  para el que se cumple que  $\mathbf{x}^\mu \neq \mathbf{y}^\mu$ .

Es posible que los patrones fundamentales sean alterados con diferentes tipos de ruido. Para diferenciar un patrón alterado del correspondiente patrón fundamental, usaremos la tilde en la parte superior; así, el patrón  $\tilde{\mathbf{x}}^k$  es una versión alterada del patrón fundamental  $\mathbf{x}^k$ , y el tipo de alteración que representa  $\tilde{\mathbf{x}}^k$  se evidenciará en el contexto específico donde se use.

Si al presentarle a la memoria  $M$  un patrón alterado  $\tilde{x}^\omega$  como entrada ( $\omega \in \{1, 2, \dots, p\}$ ),  $M$  responde con el correspondiente patrón fundamental de salida  $y^\omega$ , se dice que la recuperación es *perfecta*. Una *memoria perfecta* es aquella que realiza recuperaciones perfectas para todos los patrones fundamentales.

Naturalmente, también los patrones de salida pueden ser alterados: por ejemplo, si  $y^3$  es un patrón fundamental, entonces  $\tilde{y}^3$  representa una versión alterada de  $y^3$ .

Abundemos en la caracterización de los patrones de entrada, de salida y de la matriz  $M$ .

Primeramente se requiere la especificación de dos conjuntos a los que llamaremos arbitrariamente  $A$  y  $B$ . La importancia de estos dos conjuntos radica en que las componentes de los vectores columna que representan a los patrones, tanto de entrada como de salida, serán elementos del conjunto  $A$ , y las entradas de la matriz  $M$  serán elementos del conjunto  $B$ .

No hay requisitos previos ni limitaciones respecto de la elección de estos dos conjuntos, por lo que no necesariamente deben ser diferentes o poseer características especiales. Esto significa que el número de posibilidades para escoger  $A$  y  $B$  es infinito; a continuación se ejemplifican algunas de ellas:

- $A = B = \mathbb{R}$ , donde  $\mathbb{R}$  es el símbolo que representa al conjunto de los números reales.
- $A = \mathbb{R}$  y  $B = \{0, 1\}$ .
- $A = B = \{0, 1\}$ .
- $A = B = \{-1, 1\}$ .
- $A = \mathbb{R}$  y  $B = \{-1, 1\}$ .
- $A = \mathbb{Z}$  y  $B = \{-1, 1\}$ , donde  $\mathbb{Z}$  es el conjunto de los números enteros.
- $A \subset \mathbb{Z}$  y  $B \subset \mathbb{Z}$

Cada uno de los modelos de memorias asociativas que se incluyen en esta colección, posee sus propias especificaciones para los conjuntos  $A$  y  $B$ , de acuerdo con las necesidades del creador del modelo en cuestión.

Ya que se tienen especificados los conjuntos  $A$  y  $B$ , es necesario establecer las dimensiones de los patrones, tanto de entrada como de salida.

Sean  $m, n$  números enteros positivos. Se denota por  $n$  la dimensión de los patrones de entrada, y por  $m$  la dimensión de los patrones de salida:



claramente, nada impide que los valores de  $m$  y de  $n$  sean iguales. Aún más, uno de los requisitos que debe cumplir una memoria autoasociativa es que la dimensión de los patrones de entrada sea igual a la dimensión de los patrones de salida; por otro lado, si en una memoria sucede que  $m \neq n$ , es evidente que la memoria debe ser heteroasociativa.

Cada vector columna que representa a un patrón de entrada tiene  $n$  componentes cuyos valores pertenecen al conjunto  $A$ , y cada vector columna que representa a un patrón de salida posee  $m$  componentes cuyos valores pertenecen al conjunto  $A$ . Es decir:

$$\mathbf{x}^\mu \in A^n \text{ y } \mathbf{y}^\mu \in A^m \quad \forall \mu \in \{1, 2, \dots, p\}$$

La  $j$ -ésima componente de un vector columna se indica con la misma letra del vector, pero sin negrilla, colocando a  $j$  como subíndice ( $j \in \{1, 2, \dots, n\}$  o  $j \in \{1, 2, \dots, m\}$  según corresponda). La  $j$ -ésima componente de un vector columna  $\mathbf{x}^\mu$  se representa por

$$x_j^\mu$$

Ejemplos:

- La  $i$ -ésima componente del vector columna  $\mathbf{x}^\mu$  se representa por  $x_i^\mu$
- La tercera componente del vector columna  $\mathbf{x}^5$  se representa por  $x_3^5$
- La  $j$ -ésima componente del vector columna  $\mathbf{y}^\mu$  se representa por  $y_j^\mu$
- La  $l$ -ésima componente del vector columna  $\mathbf{y}^\omega$  se representa por  $y_l^\omega$

Al usar el superíndice  $t$  para indicar el transpuesto de un vector, se obtienen las siguientes expresiones para los vectores columna que representan a los patrones fundamentales de entrada y de salida, respectivamente:

$$\mathbf{x}^\mu = (x_1^\mu, x_2^\mu, \dots, x_n^\mu)^t = \begin{pmatrix} x_1^\mu \\ x_2^\mu \\ \vdots \\ x_n^\mu \end{pmatrix} \in A^n$$

$$\mathbf{y}^\mu = (y_1^\mu, y_2^\mu, \dots, y_m^\mu)^t = \begin{pmatrix} y_1^\mu \\ y_2^\mu \\ \vdots \\ y_m^\mu \end{pmatrix} \in A^m$$

Con lo anterior, es ya posible presentar el planteamiento del *problema general de las memorias asociativas*:

1. Fase de aprendizaje. Encontrar los operadores adecuados y una manera de generar una matriz  $\mathbf{M}$  que almacene las  $p$  asociaciones del conjunto fundamental  $\{(\mathbf{x}^1, \mathbf{y}^1), (\mathbf{x}^2, \mathbf{y}^2), \dots, (\mathbf{x}^p, \mathbf{y}^p)\}$ , donde  $\mathbf{x}^\mu \in A^n$  y  $\mathbf{y}^\mu \in A^m \quad \forall \mu \in \{1, 2, \dots, p\}$ . Si  $\exists \mu \in \{1, 2, \dots, p\}$  tal que  $\mathbf{x}^\mu \neq \mathbf{y}^\mu$ , la memoria será *heteroasociativa*; si  $m = n$  y  $\mathbf{x}^\mu = \mathbf{y}^\mu \quad \forall \mu \in \{1, 2, \dots, p\}$ , la memoria será *autoasociativa*.
2. Fase de recuperación. Hallar los operadores adecuados y las condiciones suficientes para obtener el patrón fundamental de salida  $\mathbf{y}^\mu$ , cuando se opera la memoria  $\mathbf{M}$  con el patrón fundamental de entrada  $\mathbf{x}^\mu$ ; lo anterior para todos los elementos del conjunto fundamental y para ambos modos: autoasociativo y heteroasociativo. Exhibir y caracterizar, además, el ruido que puede soportar la memoria en el patrón de entrada  $\tilde{\mathbf{x}}^\omega$ , para entregar una salida perfecta  $\mathbf{y}^\omega$ .



## Capítulo 2

# Lernmatrix.

### 2.1. Contexto.

Un análisis cuidadoso de la bibliografía que se ha generado a través de los años en relación con el tema de las memorias asociativas, nos permite percatarnos de que antes de la década de los setenta brilló por su ausencia la producción de trabajos en el área. Notables excepciones fueron la *Lernmatrix*, surgida en los albores de la década de los sesenta (Steinbuch, 1961; Steinbuch & Frank, 1961) y que constituye el tema central de este capítulo, y el *correlograph* (capítulo 2), cuyo advenimiento ocurrió en las postrimerías de esa misma década (Willshaw, Buncman & Longuet-Higgins, 1969).

En los inicios de la siguiente década llegaría la generación 1972 de científicos que produjeron trabajos claves en el devenir de las memorias asociativas: Anderson, Kohonen, Amari y Nakano.

### 2.2. El trabajo de Steinbuch.

Karl Steinbuch fue uno de los primeros investigadores en desarrollar un método para codificar información en arreglos cuadrículados conocidos como *crossbar* (Simpson, 1990). La importancia de la *Lernmatrix* (Steinbuch, 1961; Steinbuch & Frank, 1961) se evidencia en una afirmación que hace Kohonen en su artículo de 1972, donde apunta que las matrices de correlación, base fundamental de su innovador trabajo, vinieron a sustituir a la *Lernmatrix* de Steinbuch.

La *Lernmatrix* es una memoria heteroasociativa que puede funcionar como un clasificador de patrones binarios si se escogen adecuadamente los patrones de salida: es un sistema de entrada y salida que al operar acepta como entrada un patrón binario  $\mathbf{x}^\mu \in A^n$ ,  $A = \{0, 1\}$  y produce como salida la clase  $\mathbf{y}^\mu \in A^p$  que le corresponde (de entre  $p$  clases diferentes), codificada ésta con un método simple, a saber: para representar la clase  $k \in \{1, 2, \dots, p\}$ , se asignan a las componentes del vector de salida  $\mathbf{y}^\mu$  los siguientes valores:  $y_k^\mu = 1$ , y  $y_j^\mu = 0$  para  $j = 1, 2, \dots, k-1, k+1, \dots, p$ .

En la siguiente figura se esquematiza la fase de aprendizaje para la *Lernmatrix* de Steinbuch, al incorporar la pareja de patrones de entrenamiento  $(\mathbf{x}^\mu, \mathbf{y}^\mu) \in A^n \times A^p$ .

|           |           |           |         |           |         |           |  |
|-----------|-----------|-----------|---------|-----------|---------|-----------|--|
|           | $x_1^\mu$ | $x_2^\mu$ | $\dots$ | $x_j^\mu$ | $\dots$ | $x_n^\mu$ |  |
| $y_1^\mu$ | $m_{11}$  | $m_{12}$  | $\dots$ | $m_{1j}$  | $\dots$ | $m_{1n}$  |  |
| $y_2^\mu$ | $m_{21}$  | $m_{22}$  | $\dots$ | $m_{2j}$  | $\dots$ | $m_{2n}$  |  |
| $\vdots$  | $\vdots$  | $\vdots$  |         | $\vdots$  |         | $\vdots$  |  |
| $y_i^\mu$ | $m_{i1}$  | $m_{i2}$  | $\dots$ | $m_{ij}$  | $\dots$ | $m_{in}$  |  |
| $\vdots$  | $\vdots$  | $\vdots$  |         | $\vdots$  |         | $\vdots$  |  |
| $y_p^\mu$ | $m_{p1}$  | $m_{p2}$  | $\dots$ | $m_{pj}$  | $\dots$ | $m_{pn}$  |  |

(2.1)

Cada uno de los componentes  $m_{ij}$  de  $M$ , la *Lernmatrix* de Steinbuch, tiene valor cero al inicio, y se actualiza de acuerdo con la regla  $m_{ij} + \Delta m_{ij}$ , donde:

$$\Delta m_{ij} = \begin{cases} +\varepsilon & \text{si } x_i^\mu = 1 = y_j^\mu \\ -\varepsilon & \text{si } x_i^\mu = 0 \text{ y } y_j^\mu = 1 \\ 0 & \text{en otro caso} \end{cases} \quad (2.2)$$

siendo  $\varepsilon$  una constante positiva escogida previamente.

La fase de recuperación consiste en encontrar la clase a la que pertenece un vector de entrada  $\mathbf{x}^\omega \in A^n$  dado. Encontrar la clase significa obtener las coordenadas del vector  $\mathbf{y}^\omega \in A^p$  que le corresponde al patrón  $\mathbf{x}^\omega$ : en virtud

del método de construcción de los vectores  $y^h$  la clase debería obtenerse sin ambigüedad.

La  $i$ -ésima coordenada  $y_i^\omega$  del vector de clase  $y^\omega \in A^p$  se obtiene como lo indica la siguiente expresión, donde  $\vee$  es el operador *máximo*:

$$y_i^\omega = \begin{cases} 1 & \text{si } \sum_{j=1}^n m_{ij} \cdot x_j^\omega = \vee_{h=1}^p \left[ \sum_{j=1}^n m_{hj} \cdot x_j^\omega \right] \\ 0 & \text{en otro caso} \end{cases} \quad (2.3)$$

### Ejemplo 1.-

Sean  $p = 3$ ,  $n = 5$  y  $\varepsilon > 0$ . Es decir, se tienen 3 clases de patrones de dimensión 5; las primeras tres asociaciones se presentan a continuación:

$$x^1 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \quad y^1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}; \quad x^2 = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \quad y^2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix};$$

$$x^3 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \quad y^3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Para iniciar la creación (fase de aprendizaje) de la *Lernmatrix*, se asigna el valor cero a todos los elementos  $m_{ij}$  y se realizan las operaciones de la crossbar 2.1 y la expresión 2.2 con la primera pareja de asociaciones:

|             | $x_1^1 = 1$   | $x_2^1 = 0$    | $x_3^1 = 1$   | $x_4^1 = 0$    | $x_5^1 = 1$   |
|-------------|---------------|----------------|---------------|----------------|---------------|
| $y_1^1 = 1$ | $\varepsilon$ | $-\varepsilon$ | $\varepsilon$ | $-\varepsilon$ | $\varepsilon$ |
| $y_2^1 = 0$ | 0             | 0              | 0             | 0              | 0             |
| $y_3^1 = 0$ | 0             | 0              | 0             | 0              | 0             |

La segunda pareja de asociaciones provoca los siguientes cambios en la matriz:

|             | $x_1^2 = 1$   | $x_2^2 = 1$    | $x_3^2 = 0$    | $x_4^2 = 0$    | $x_5^2 = 1$   |
|-------------|---------------|----------------|----------------|----------------|---------------|
| $y_1^2 = 0$ | $\varepsilon$ | $-\varepsilon$ | $\varepsilon$  | $-\varepsilon$ | $\varepsilon$ |
| $y_2^2 = 1$ | $\varepsilon$ | $\varepsilon$  | $-\varepsilon$ | $-\varepsilon$ | $\varepsilon$ |
| $y_3^2 = 0$ | 0             | 0              | 0              | 0              | 0             |

Y finalmente con la tercera pareja la matriz toma el siguiente aspecto:

|             | $x_1^3 = 1$   | $x_2^3 = 0$    | $x_3^3 = 1$    | $x_4^3 = 1$    | $x_5^3 = 0$    |
|-------------|---------------|----------------|----------------|----------------|----------------|
| $y_1^3 = 0$ | $\varepsilon$ | $-\varepsilon$ | $\varepsilon$  | $-\varepsilon$ | $\varepsilon$  |
| $y_2^3 = 0$ | $\varepsilon$ | $\varepsilon$  | $-\varepsilon$ | $-\varepsilon$ | $\varepsilon$  |
| $y_3^3 = 1$ | $\varepsilon$ | $-\varepsilon$ | $\varepsilon$  | $\varepsilon$  | $-\varepsilon$ |

(2.4)

Es decir,

$$M = \begin{pmatrix} \varepsilon & -\varepsilon & \varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & \varepsilon & -\varepsilon \end{pmatrix} \quad (2.5)$$

La fase de recuperación consiste en presentarle a la matriz  $M$  uno de los patrones de entrada y realizar las operaciones indicadas en la expresión 2.3; a la salida se espera obtener la clase a la que pertenece el vector de entrada.

Iniciemos con el patrón  $x^1$ .

$$M \cdot x^1 = \begin{pmatrix} \varepsilon & -\varepsilon & \varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & \varepsilon & -\varepsilon \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 3\varepsilon \\ \varepsilon \\ \varepsilon \end{pmatrix}$$

Se observa que:  $\sum_{j=1}^5 m_{1j}x_j^1 = 3\varepsilon$ ;  $\sum_{j=1}^5 m_{2j}x_j^1 = \varepsilon$  y  $\sum_{j=1}^5 m_{3j}x_j^1 = \varepsilon$ . Por ello,  $\sum_{j=1}^5 m_{1j}x_j^1 = V_{h=1}^p \left[ \sum_{j=1}^5 m_{hj}x_j^1 \right]$  y de acuerdo con la expresión 2.3, se tiene que  $y_1^1 = 1$  y  $y_2^1 = 0 = y_3^1$ . Por lo tanto, el vector que representa a la clase es:

$$\mathbf{y}^1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Recuperemos la clase a la que pertenece el patrón de entrada  $\mathbf{x}^2$ .

$$\mathbf{M} \cdot \mathbf{x}^2 = \begin{pmatrix} \varepsilon & -\varepsilon & \varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & \varepsilon & -\varepsilon \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \varepsilon \\ 3\varepsilon \\ -\varepsilon \end{pmatrix}$$

En este caso se observa que:  $\sum_{j=1}^5 m_{1j}x_j^2 = \varepsilon$ ;  $\sum_{j=1}^5 m_{2j}x_j^2 = 3\varepsilon$  y  $\sum_{j=1}^5 m_{3j}x_j^2 = -\varepsilon$ . Por ello,  $\sum_{j=1}^5 m_{2j}x_j^2 = \bigvee_{h=1}^p \left[ \sum_{j=1}^5 m_{hj}x_j^2 \right]$  y de acuerdo con la expresión 2.3, se tiene que  $y_2^2 = 1$  y  $y_1^2 = 0 = y_3^2$ . Por lo tanto, el vector que representa a la clase es:

$$\mathbf{y}^2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

finalmente recuperemos la clase a la que pertenece el patrón de entrada  $\mathbf{x}^3$ .

$$\mathbf{M} \cdot \mathbf{x}^3 = \begin{pmatrix} \varepsilon & -\varepsilon & \varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & \varepsilon & -\varepsilon \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \varepsilon \\ -\varepsilon \\ 3\varepsilon \end{pmatrix}$$

En este caso se observa que:  $\sum_{j=1}^5 m_{1j}x_j^3 = \varepsilon$ ;  $\sum_{j=1}^5 m_{2j}x_j^3 = -\varepsilon$  y  $\sum_{j=1}^5 m_{3j}x_j^3 = 3\varepsilon$ . Por ello,  $\sum_{j=1}^5 m_{3j}x_j^3 = \bigvee_{h=1}^p \left[ \sum_{j=1}^5 m_{hj}x_j^3 \right]$  y de acuerdo con la expresión 2.3, se tiene que  $y_3^3 = 1$  y  $y_1^3 = 0 = y_2^3$ . Por lo tanto, el vector que representa a la clase es:



$$y^3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

### Ejemplo 2.-

Surge una cuestión interesante: ¿qué pasa si hay más vectores de entrada que clases?.

Dado que sólo existen tres clases diferentes, un cuarto patrón debe pertenecer necesariamente a una de esas tres clases.

Asignemos la clase  $y^1$  al vector de entrada  $x^4$  dado por:

$$x^4 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$

Eso significa que debemos modificar la matriz 2.5 llevando a cabo las operaciones de la expresión 2.2 en la crossbar 2.4:

|             | $x_1^4 = 0$   | $x_2^4 = 1$    | $x_3^4 = 0$    | $x_4^4 = 1$    | $x_5^4 = 1$    |
|-------------|---------------|----------------|----------------|----------------|----------------|
| $y_1^1 = 1$ | 0             | 0              | 0              | 0              | $2\varepsilon$ |
| $y_2^1 = 0$ | $\varepsilon$ | $\varepsilon$  | $-\varepsilon$ | $-\varepsilon$ | $\varepsilon$  |
| $y_3^1 = 0$ | $\varepsilon$ | $-\varepsilon$ | $\varepsilon$  | $\varepsilon$  | $-\varepsilon$ |

Ahora la nueva *Lernmatrix*, que se representará con el símbolo  $M_{4patrones}$ , es:

$$M_{4patrones} = \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & \varepsilon & -\varepsilon \end{pmatrix}$$

Recuperemos la clase para cada uno de los 4 vectores de entrada, de acuerdo con la expresión 2.3:

$$\begin{aligned} \mathbf{M}_{4patrones} \cdot \mathbf{x}^1 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & \varepsilon & -\varepsilon \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 2\varepsilon \\ \varepsilon \\ \varepsilon \end{pmatrix} \rightarrow \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \rightarrow \text{clase } y^1 \end{aligned}$$

$$\begin{aligned} \mathbf{M}_{4patrones} \cdot \mathbf{x}^2 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & \varepsilon & -\varepsilon \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 2\varepsilon \\ 3\varepsilon \\ -\varepsilon \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \rightarrow \text{clase } y^2 \end{aligned}$$

$$\begin{aligned} \mathbf{M}_{4patrones} \cdot \mathbf{x}^3 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & \varepsilon & -\varepsilon \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ -\varepsilon \\ 2\varepsilon \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \rightarrow \text{clase } y^3 \end{aligned}$$

$$\begin{aligned} \mathbf{M}_{1patrones} \cdot \mathbf{x}^1 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & \varepsilon & -\varepsilon \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 2\varepsilon \\ \varepsilon \\ -\varepsilon \end{pmatrix} \rightarrow \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \rightarrow \text{clase } y^1 \end{aligned}$$

Se observa que la recuperación se realizó de manera exacta para los cuatro patrones, a pesar de que hay más patrones que clases. Hasta aquí la *Lernmatrix* luce como un clasificador preciso. Cuando se aumenta el número de patrones, ocurre el fenómeno llamado saturación.

### Ejemplo 3.-

Ejemplifiquemos el fenómeno de *saturación* en la *Lernmatrix* de Steinbuch, y para ello modifiquemos la matriz  $\mathbf{M}_{1patrones}$  con la pareja:

$$\mathbf{x}^5 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \quad \mathbf{y}^3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

A partir de la matriz:

$$\mathbf{M}_{1patrones} = \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & \varepsilon & -\varepsilon \end{pmatrix}$$

y de la pareja  $(\mathbf{x}^5, \mathbf{y}^3)$  obtenemos:

|             | $x_1^5 = 0$   | $x_2^5 = 0$     | $x_3^5 = 1$    | $x_4^5 = 0$    | $x_5^5 = 1$    |
|-------------|---------------|-----------------|----------------|----------------|----------------|
| $y_1^3 = 0$ | 0             | 0               | 0              | 0              | $2\varepsilon$ |
| $y_2^3 = 0$ | $\varepsilon$ | $\varepsilon$   | $-\varepsilon$ | $-\varepsilon$ | $\varepsilon$  |
| $y_3^3 = 1$ | 0             | $-2\varepsilon$ | $2\varepsilon$ | 0              | 0              |

La nueva *Lernmatrix*, que se representará con  $M_{5patrones}$  es:

$$M_{5patrones} = \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ 0 & -2\varepsilon & 2\varepsilon & 0 & 0 \end{pmatrix}$$

Al intentar recuperar la clase para cada uno de los cinco patrones de entrada, se obtiene:

$$\begin{aligned} M_{5patrones} \cdot x^1 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ 0 & -2\varepsilon & 2\varepsilon & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 2\varepsilon \\ \varepsilon \\ 2\varepsilon \end{pmatrix} \rightarrow \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \text{ ¿clase } y^1 \text{ o } y^3? \end{aligned}$$

$$\begin{aligned} M_{5patrones} \cdot x^2 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ 0 & -2\varepsilon & 2\varepsilon & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 2\varepsilon \\ 3\varepsilon \\ -2\varepsilon \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \rightarrow \text{clase } y^2 \end{aligned}$$

$$\begin{aligned} M_{5patrones} \cdot x^3 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ 0 & -2\varepsilon & 2\varepsilon & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ -\varepsilon \\ 2\varepsilon \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \rightarrow \text{clase } y^3 \end{aligned}$$

$$\begin{aligned} M_{5patrones} \cdot x^4 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ 0 & -2\varepsilon & 2\varepsilon & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 2\varepsilon \\ \varepsilon \\ -2\varepsilon \end{pmatrix} \rightarrow \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \rightarrow \text{clase } y^1 \end{aligned}$$

$$\begin{aligned} M_{5patrones} \cdot x^5 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 2\varepsilon \\ \varepsilon & \varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ 0 & -2\varepsilon & 2\varepsilon & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 2\varepsilon \\ 0 \\ 2\varepsilon \end{pmatrix} \rightarrow \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \rightarrow \text{¿clase } y^1 \text{ o } y^3? \end{aligned}$$

El fenómeno de saturación en todo su esplendor.

### 2.3. Consideraciones finales del capítulo.

Steinbuch es un buen ejemplo de la manera en que los trabajos pioneros en un campo específico del quehacer humano tienen trascendencia con el paso

de los años... y de los siglos. Las recientes declaraciones de Robert Hecht-Nielsen, el notable científico creador de las redes neuronales conocidas como *counterpropagation*, lo constatan: Hecht-Nielsen es el director científico de un millonario proyecto de la DARPA (Defense Advanced Research Projects Agency) de los Estados Unidos, cuyo objetivo es el diseño y construcción de un novedoso tipo de redes neuronales conocidas como redes cortrónicas; estas redes, según pregonaba el mencionado investigador, tienen como base la Lernmatrix del alemán Steinbuch ([www.gcn.com](http://www.gcn.com)).



## Capítulo 3

# Correlograph.

### 3.1. Contexto.

Como se mencionaba al inicio del capítulo anterior, uno de los trabajos pioneros en memorias asociativas surgidos en la década de los sesenta, además de la *Lernmatrix* (Steinbuch, 1961; Steinbuch & Frank, 1961), es el *Correlograph*, cuyo advenimiento acació en las postrimerías de esa década (Willshaw, Buneman & Longuet-Higgins, 1969).

### 3.2. El trabajo de Willshaw, Buneman & Longuet-Higgins.

El *correlograph* es un dispositivo óptico elemental capaz de funcionar como una memoria asociativa (Willshaw, Buneman & Longuet-Higgins, 1969). En palabras de los autores “el sistema es tan simple, que podría ser construido en cualquier laboratorio escolar de física elemental”.

El *correlograph* consta de tres pantallas opacas y una fuente de luz; el aparato genera *correlogramas* de puntos luminosos en una de las pantallas (la cual se perfora precisamente en esos puntos), a partir de pares de patrones de huecos realizados en las otras dos pantallas (fase de *aprendizaje*). Cada correlograma así generado se usa entonces para obtener cada uno de los patrones de entrada partiendo de su patrón asociado (fase de *recuperación*).

Con el fin de ilustrar la fase de aprendizaje del *correlograph*, consideremos la pareja de patrones de huecos formados en las pantallas A y B de la figura 1 (por simplicidad los llamaremos patrón A y patrón B).



La fase de aprendizaje consta de dos pasos:

1. Se usa la fuente de luz FL para iluminar desde atrás la pantalla A. Los rayos de luz que pasan por los huecos del patrón A, continúan su trayectoria rectilínea a través de los huecos del patrón B, hasta chocar con la pantalla C, para formar un correlograma de  $mn$  puntos, considerando que el número de huecos de los patrones A y B son  $m$  y  $n$ , respectivamente. Para el caso de la figura 05.1,  $m = 2$  y  $n = 3$ , por lo que el correlograma consta de 6 puntos.
2. Se realizan las  $mn$  perforaciones en la pantalla C que corresponden a los puntos del correlograma.

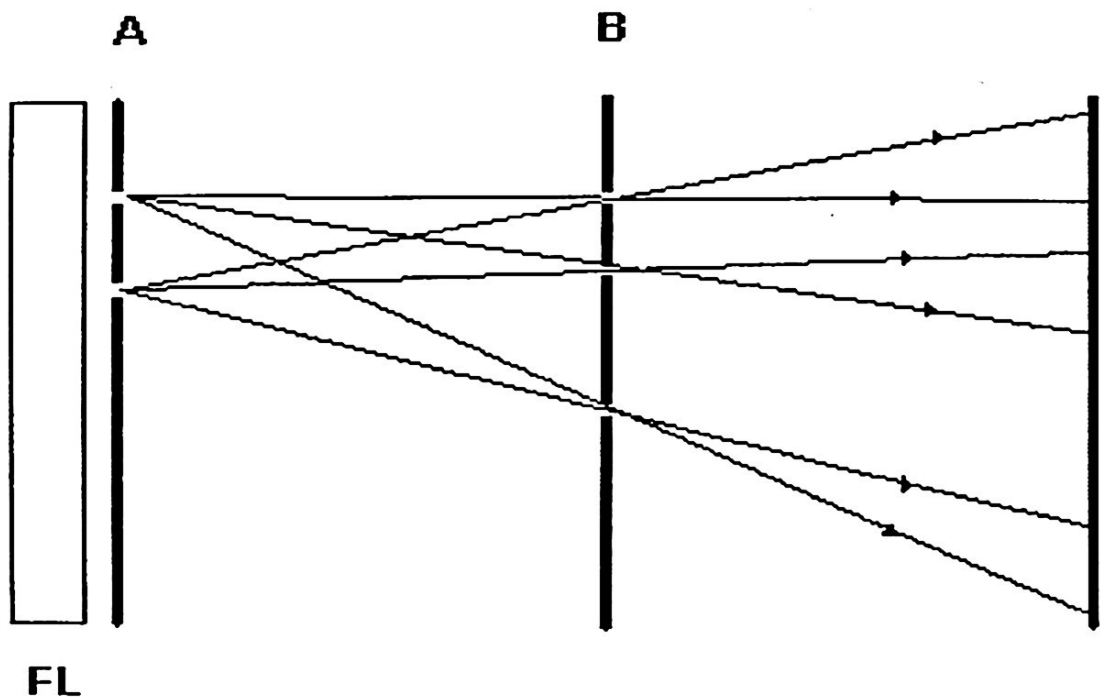


Figura 1

### 3.2. EL TRABAJO DE WILLSHAW, BUNEMAN & LONGUET-HIGGINS.23

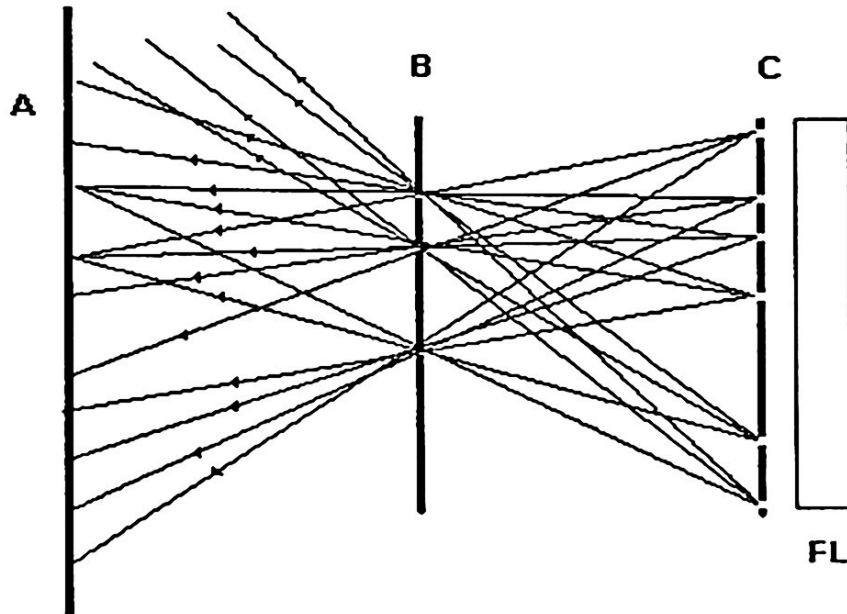


Figura 2

La fase de recuperación, ilustrada en la figura 2, consiste en presentarle al correlograma el patrón B, para obtener el patrón asociado A.

Esta fase también consta de dos pasos:

1. Se coloca una pantalla opaca, sin huecos, en la posición de A, y la fuente de luz FL se ubica delante de la pantalla C, que contiene el correlograma. Los rayos de luz pasan por los huecos del correlograma, continúan su trayectoria rectilínea a través de los huecos del patrón B, y finalmente inciden en la pantalla opaca A. El número de puntos de luz que inciden en la pantalla A es  $mn^2$ , que es el producto de  $mn$  y  $n$ , números de puntos del correlograma C y del patrón B, respectivamente. Para el caso del ejemplo de la figura 2, son 18 puntos los que aparecen en la pantalla A. Es necesario realizar alguna operación que elimine los  $mn^2 - m$  puntos espúrcos (16 en el ejemplo) para que queden únicamente los  $m$  puntos (2 para el ejemplo) que pertenecen al patrón A original de la figura 1.
2. Se realiza una operación de umbralizado para eliminar los  $mn^2 - m$  puntos con menor luminancia y dejar los  $m$  puntos más brillantes: éstos son los puntos que eran huecos en el patrón original A.

Es posible almacenar varios conjuntos de patrones y reconstruirlos exitosamente usando el mismo correlograph. Si se almacenan demasiadas asociaciones, habrá demasiados puntos espúreos, y aparecerán errores; esto es, la reconstrucción de  $A$  será ruidosa.

A pesar de las evidentes diferencias entre el *correlograph* y la *Lernmatrix*, los autores Willshaw, Buneman & Longuet-Higgins tomaron como punto de partida este dispositivo óptico para crear un ente al que llamaron *red asociativa*, la cual no es otra cosa que un modelo de memoria asociativa tipo *crossbar* parecida a la *Lernmatrix*, pero con una regla de aprendizaje diferente a la regla de Steinbuch 2.2 (ver capítulo 2) y con una característica adicional: esta memoria no se comporta sólo como un clasificador, sino como una memoria que asocia patrones de entrada con patrones de salida (Willshaw, Buneman & Longuet-Higgins, 1969; Kohonen, 1989).

En la *red asociativa* los puntos de la pantalla B se convierten en  $n$  líneas paralelas verticales (componentes  $x_j^\mu$ ), y los puntos de la pantalla A se transforman en  $m$  líneas paralelas horizontales (componentes  $y_i^\mu$ ); los puntos de C corresponden ahora a las  $mn$  intersecciones entre las líneas  $x_j^\mu$  y  $y_i^\mu$ . Lo anterior significa que la red asociativa es un sistema de entrada y salida que al operar acepta como entrada un patrón binario  $x^\mu \in A^n$ ,  $A = \{0, 1\}$  y produce como salida un patrón  $y^\mu \in A^m$ .

En la siguiente figura se esquematiza la *fase de aprendizaje* para la red asociativa, al incorporar la pareja de patrones de entrenamiento  $(x^\mu, y^\mu) \in A^n \times A^m$ .

|           |           |           |         |           |         |           |  |
|-----------|-----------|-----------|---------|-----------|---------|-----------|--|
|           | $x_1^\mu$ | $x_2^\mu$ | $\dots$ | $x_j^\mu$ | $\dots$ | $x_n^\mu$ |  |
| $y_1^\mu$ | $m_{11}$  | $m_{12}$  | $\dots$ | $m_{1j}$  | $\dots$ | $m_{1n}$  |  |
| $y_2^\mu$ | $m_{21}$  | $m_{22}$  | $\dots$ | $m_{2j}$  | $\dots$ | $m_{2n}$  |  |
| $\vdots$  | $\vdots$  | $\vdots$  |         | $\vdots$  |         | $\vdots$  |  |
| $y_i^\mu$ | $m_{i1}$  | $m_{i2}$  | $\dots$ | $m_{ij}$  | $\dots$ | $m_{in}$  |  |
| $\vdots$  | $\vdots$  | $\vdots$  |         | $\vdots$  |         | $\vdots$  |  |
| $y_m^\mu$ | $m_{m1}$  | $m_{m2}$  | $\dots$ | $m_{mj}$  | $\dots$ | $m_{mn}$  |  |

(3.1)

Cada uno de los componentes  $m_{ij}$  de la red asociativa  $M$  tiene valor cero al inicio, y se actualiza de acuerdo con la regla:

### 3.2. EL TRABAJO DE WILLSHAW, BUNEMAN & LONGUET-HIGGINS.25

$$m_{ij} = \begin{cases} 1 & \text{si } y_i^\mu = 1 = x_j^\mu \\ \text{valor anterior} & \text{en otro caso} \end{cases} \quad (3.2)$$

En la *fase de recuperación* se le presenta a la red asociativa M un vector de entrada  $x^\omega \in A^n$ . Se realiza el producto de la matriz M por el vector  $x^\omega$  y se ejecuta una operación de umbralizado, de acuerdo con la siguiente expresión:

$$y_i^\omega = \begin{cases} 1 & \text{si } \sum_{j=1}^n m_{ij} \cdot x_j^\omega \geq u \\ 0 & \text{en otro caso} \end{cases} \quad (3.3)$$

donde  $u$  es el valor de umbral. Una estimación aproximada del valor de umbral  $u$  se puede lograr con la ayuda de un número indicador mencionado en el artículo de Willshaw *et al.* de 1969:  $\log_2 n$ .

Ejemplo.

Vale la pena presentar el mismo ejemplo que aparece en (Willshaw, Buneman & Longuet-Higgins, 1969). Son 4 parejas de patrones de dimensión 8; es decir,  $p = 4$  y  $m = n = 8$  de acuerdo con la sección 2.

$$\begin{array}{cc} \mathbf{x}^1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} & \mathbf{y}^1 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} & \mathbf{x}^2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} & \mathbf{y}^2 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} \\ \\ \mathbf{x}^3 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} & \mathbf{y}^3 = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} & \mathbf{x}^4 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} & \mathbf{y}^4 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \end{array}$$

Para iniciar la fase de aprendizaje, es decir, la creación de la red asociativa, se asigna el valor cero a todos los elementos  $m_{ij}$  del arreglo 3.1 y se realizan las operaciones de la expresión 3.2 con la primera pareja de asociaciones  $(x^1, y^1)$ .

Obsérvese que los cambios provocados en la red asociativa inicial se deben a las componentes 4, 6 y 7 del vector de salida  $y^1$ , y a las componentes 1, 2 y 3 del vector de entrada  $x^1$ , por lo que los elementos de la red asociativa que toman el valor 1 son:  $m_{41}$ ,  $m_{42}$ ,  $m_{43}$ ,  $m_{61}$ ,  $m_{62}$ ,  $m_{63}$ ,  $m_{71}$ ,  $m_{72}$  y  $m_{73}$ :

|             | $x_1^1$ | $x_2^1$ | $x_3^1$ | $x_4^1$ | $x_5^1$ | $x_6^1$ | $x_7^1$ | $x_8^1$ |
|-------------|---------|---------|---------|---------|---------|---------|---------|---------|
|             | 1       | 1       | 1       | 0       | 0       | 0       | 0       | 0       |
| $y_1^1 = 0$ | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       |
| $y_2^1 = 0$ | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       |
| $y_3^1 = 0$ | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       |
| $y_4^1 = 1$ | 1       | 1       | 1       | 0       | 0       | 0       | 0       | 0       |
| $y_5^1 = 0$ | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       |
| $y_6^1 = 1$ | 1       | 1       | 1       | 0       | 0       | 0       | 0       | 0       |
| $y_7^1 = 1$ | 1       | 1       | 1       | 0       | 0       | 0       | 0       | 0       |
| $y_8^1 = 0$ | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       |

Al observar la segunda pareja de asociaciones  $(x^2, y^2)$ , podemos percatarnos de que el patrón de salida  $y^2$  tiene valor 1 en las entradas 1, 5 y 7, mientras que el patrón de entrada  $x^2$  los tiene en las componentes 2, 5 y 8. Lo anterior significa que, de acuerdo con la expresión 3.2, los elementos de la red asociativa que deben tomar el valor 1 son:  $m_{12}$ ,  $m_{15}$ ,  $m_{18}$ ,  $m_{52}$ ,  $m_{55}$ ,  $m_{58}$ ,  $m_{72}$ ,  $m_{75}$  y  $m_{78}$ ; en todos los casos, el elemento  $m_{ij}$  correspondiente cambia su valor de cero a uno, excepto en  $m_{72}$  que ya tenía valor 1, por lo que se queda con su valor anterior, de acuerdo con la regla de aprendizaje de Willshaw, Buneman & Longuet-Higgins:

### 3.2. EL TRABAJO DE WILLSHAW, BUNEMAN & LONGUET-HIGGINS.27

|             | $x_1^2$ | $x_2^2$ | $x_3^2$ | $x_4^2$ | $x_5^2$ | $x_6^2$ | $x_7^2$ | $x_8^2$ |
|-------------|---------|---------|---------|---------|---------|---------|---------|---------|
|             | 0       | 1       | 0       | 0       | 1       | 0       | 0       | 1       |
| $y_1^2 = 1$ | 0       | 1       | 0       | 0       | 1       | 0       | 0       | 1       |
| $y_2^2 = 0$ | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       |
| $y_3^2 = 0$ | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       |
| $y_4^2 = 0$ | 1       | 1       | 1       | 0       | 0       | 0       | 0       | 0       |
| $y_5^2 = 1$ | 0       | 1       | 0       | 0       | 1       | 0       | 0       | 1       |
| $y_6^2 = 0$ | 1       | 1       | 1       | 0       | 0       | 0       | 0       | 0       |
| $y_7^2 = 1$ | 1       | 1       | 1       | 0       | 1       | 0       | 0       | 1       |
| $y_8^2 = 0$ | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       |

Continuemos razonando de manera similar, para afirmar que, de acuerdo con los valores de las componentes de los vectores que forman la tercera asociación ( $x^3, y^3$ ), los elementos de la red asociativa que deben tomar el valor 1 son:  $m_{22}$ ,  $m_{24}$ ,  $m_{26}$ ,  $m_{32}$ ,  $m_{34}$ ,  $m_{36}$ ,  $m_{62}$ ,  $m_{64}$  y  $m_{66}$ ; en todos los casos, el elemento  $m_{ij}$  cambia su valor de cero a uno, excepto en  $m_{62}$  que ya tenía valor 1, por lo que se queda con su valor anterior, de acuerdo con la regla 3.2:

|             | $x_1^3$ | $x_2^3$ | $x_3^3$ | $x_4^3$ | $x_5^3$ | $x_6^3$ | $x_7^3$ | $x_8^3$ |
|-------------|---------|---------|---------|---------|---------|---------|---------|---------|
|             | 0       | 1       | 0       | 1       | 0       | 1       | 0       | 0       |
| $y_1^3 = 0$ | 0       | 1       | 0       | 0       | 1       | 0       | 0       | 1       |
| $y_2^3 = 1$ | 0       | 1       | 0       | 1       | 0       | 1       | 0       | 0       |
| $y_3^3 = 1$ | 0       | 1       | 0       | 1       | 0       | 1       | 0       | 0       |
| $y_4^3 = 0$ | 1       | 1       | 1       | 0       | 0       | 0       | 0       | 0       |
| $y_5^3 = 0$ | 0       | 1       | 0       | 0       | 1       | 0       | 0       | 1       |
| $y_6^3 = 1$ | 1       | 1       | 1       | 1       | 0       | 1       | 0       | 0       |
| $y_7^3 = 0$ | 1       | 1       | 1       | 0       | 1       | 0       | 0       | 1       |
| $y_8^3 = 0$ | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       |

Por efectos de operar con la última pareja de patrones, la asociación ( $x^4, y^4$ ), los elementos de la red asociativa que deben tomar el valor 1 son:  $m_{31}$ ,  $m_{33}$ ,  $m_{37}$ ,  $m_{41}$ ,  $m_{43}$ ,  $m_{47}$ ,  $m_{81}$ ,  $m_{83}$  y  $m_{87}$ , de los cuales dos se quedan con su valor anterior ( $m_{41}$  y  $m_{43}$ ):

|             | $x_1^1$ | $x_2^1$ | $x_3^1$ | $x_4^1$ | $x_5^1$ | $x_6^1$ | $x_7^1$ | $x_8^1$ |
|-------------|---------|---------|---------|---------|---------|---------|---------|---------|
|             | 1       | 0       | 1       | 0       | 0       | 0       | 1       | 0       |
| $y_1^1 = 0$ | 0       | 1       | 0       | 0       | 1       | 0       | 0       | 1       |
| $y_2^1 = 0$ | 0       | 1       | 0       | 1       | 0       | 1       | 0       | 0       |
| $y_3^1 = 1$ | 1       | 1       | 1       | 1       | 0       | 1       | 1       | 0       |
| $y_4^1 = 1$ | 1       | 1       | 1       | 0       | 0       | 0       | 1       | 0       |
| $y_5^1 = 0$ | 0       | 1       | 0       | 0       | 1       | 0       | 0       | 1       |
| $y_6^1 = 0$ | 1       | 1       | 1       | 1       | 0       | 1       | 0       | 0       |
| $y_7^1 = 0$ | 1       | 1       | 1       | 0       | 1       | 0       | 0       | 1       |
| $y_8^1 = 1$ | 1       | 0       | 1       | 0       | 0       | 0       | 1       | 0       |

De manera que la red asociativa finalmente queda así:

$$M = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Analícemos ahora la fase de recuperación. Para ello, primeramente realicemos el producto matricial entre  $M$  y  $x^1$ , de acuerdo con la expresión 3.3:

$$M \cdot x^1 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 3 \\ 3 \\ 1 \\ 3 \\ 3 \\ 2 \end{pmatrix}$$

Al tomar en cuenta la sugerencia de Willshaw *et al.* para calcular indicador que permite obtener el umbral, el valor de  $u$  se puede aproximar por  $\log_2 8 = 3$ .

### 3.2. EL TRABAJO DE WILLSHAW, BUNEMAN & LONGUET-HIGGINS.29

Al aplicar la umbralización de la expresión 3.3 al vector anterior con  $u = 3$ , resulta que las componentes 1, 2, 5 y 8 son menores que el umbral, por lo que se quedan con el valor cero; por otro lado, las componentes 3, 4, 6 y 7 poseen un valor mayor o igual que el umbral, por lo que toman el valor uno:

$$\begin{pmatrix} 1 \\ 1 \\ 3 \\ 3 \\ 1 \\ 3 \\ 3 \\ 2 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \neq y^1$$

Veamos qué sucede al aplicar la umbralización con  $u = 3$  después de operar la red asociativa M con cada uno de los patrones  $x^2$ ,  $x^3$  y  $x^4$ .

$$M \cdot x^2 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \\ 1 \\ 1 \\ 3 \\ 1 \\ 3 \\ 0 \end{pmatrix} \rightarrow \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} = y^2$$

$$M \cdot x^3 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 3 \\ 3 \\ 1 \\ 1 \\ 3 \\ 1 \\ 0 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = y^3$$



$$M \cdot x^1 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 3 \\ 3 \\ 0 \\ 2 \\ 2 \\ 3 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = y^1$$

Se observa que, aunque la recuperación no es perfecta, al menos la mayoría de los vectores fueron recuperados de manera perfecta.

### 3.3. Consideraciones finales del capítulo.

En este capítulo hemos presentado el *correlograph*, un dispositivo óptico elemental capaz de asociar patrones, que fue diseñado, desarrollado y aplicado por una terna de preclaros científicos ingleses en 1969: Willshaw Buneman & Longuet-Higgins; además, hemos presentado y ejemplificado la *red asociativa*, una memoria asociativa que se derivó del *correlograph* y que, junto con la *Lernmatrix* de Steinbuch, constituyen los entes más longevos con que se cuenta entre los trabajos pioneros en el tema de las memorias asociativas.

Anderson y Rosenfeld, los editores del compendio *Neurocomputing: Foundations of Research* (Anderson & Rosenfeld (Eds.), 1990), afirman que "El *correlograph* es un dispositivo que asocia patrones, y no es un clasificador como el perceptrón; este es un punto muy importante." (El perceptrón es un clasificador de patrones ópticos, inventado y desarrollado en 1958 por el psicólogo F. Rosenblatt en el Cornell Aeronautical Laboratory, en un intento por entender la memoria humana y los procesos cognitivos y de aprendizaje).

Respecto de la *red asociativa*, los mismos editores Anderson y Rosenfeld, afirman que "es excepcionalmente fácil implementar este modelo con circuitos integrados" debido a que "la red asociativa es muy parecida a un dispositivo puramente digital" y su estructura permite que "el uso de técnicas VLSI para realizar la implementación sea potencialmente muy alto".

## Capítulo 4

# Linear Associator.

### 4.1. Contexto.

El siguiente acontecimiento trascendente después de los trabajos pioneros de la década de los sesenta, la *Lernmatrix* (Steinbuch, 1961; Steinbuch & Frank, 1961) y el *Correlograph* (Willshaw, Buneman & Longuet-Higgins, 1969), sucedió a principios de la década de los setenta, en 1972.

El año de 1972 fue testigo de una febril actividad de los investigadores pioneros en memorias asociativas. Para corroborar esta aserción, baste dar una vistazo a los items bibliográficos enlistados en la Introducción, donde se puede observar que cuatro de los primeros trabajos sobre modelos matemáticos para memorias asociativas se publicaron en revistas internacionales precisamente en las ediciones de ese año.

Apoyado por la *UCLA*, James A. Anderson prendió la mecha con su *Interactive Memory* (Anderson, 1972); en abril, Teuvo Kohonen, a la sazón profesor de la *Helsinki University of Technology*, presentó ante el mundo sus *Correlation Matrix Memories* (Kohonen, 1972); tres meses después, Kaoru Nakano de la *University of Tokyo*, dio a conocer su *Associatron* (Nakano, 1972); y en el ocaso del año, Shun-Ichi Amari, profesor de la *University of Tokyo* y uno de los más prolíficos escritores de artículos científicos hasta nuestros días, publicó un trabajo teórico donde continuaba con sus investigaciones sobre las *Self-Organizing Nets of Threshold Elements* (Amari, 1972), tópico cuyo estudio había iniciado un par de años atrás. Este trabajo de Amari constituye una valiosa muestra del estilo teórico característico de este personaje y representa, históricamente, un importante antecedente para

la creación de lo que una década más tarde se convertiría en el modelo de memoria asociativa por antonomasia: la memoria Hopfield.

Los trabajos de Anderson y Kohonen, y en cierta medida el de Nakano, dieron lugar al modelo que actualmente se conoce con el nombre genérico de *Linear Associator*, descrito precisamente en este capítulo.

## 4.2. El trabajo de Anderson y Kohonen.

Después de la incursión en los terrenos de Steinbuch y Willshaw, Buncman & Longuet-Higgins (ver contenidos de los capítulos 2 y 3), en este capítulo presentaremos los fundamentos del modelo que dio lugar al *Linear Associator*, el cual, como se había mencionado antes, tiene su origen en los trabajos pioneros de Anderson, Kohonen y Nakano publicados durante año 1972.

Es pertinente mencionar un hecho curioso, que se ha presentado en personajes dedicados a otras ramas de la ciencia: James A. Anderson y Teuvo Kohonen obtuvieron resultados *asombrosamente similares* a pesar de que trabajaron independientemente, alejados, y sin tener noticia uno del otro, hasta tiempo después de que aparecieron los artículos; además, estos autores tienen formaciones profesionales totalmente diferentes: Anderson es neurofisiólogo (estadunidense) y Kohonen es físico e ingeniero eléctrico (finlandés) (Anderson & Rosenfeld, 1990; Kohonen, 1989).

Penetremos en los recovecos del *Linear Associator*, y para ello consideremos el conjunto fundamental  $\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mu = 1, 2, \dots, p\}$  con

$$\mathbf{x}^\mu = (x_1^\mu, x_2^\mu, \dots, x_n^\mu)^t = \begin{pmatrix} x_1^\mu \\ x_2^\mu \\ \vdots \\ x_n^\mu \end{pmatrix} \in A^n \quad \text{y}$$

$$\mathbf{y}^\mu = (y_1^\mu, y_2^\mu, \dots, y_m^\mu)^t = \begin{pmatrix} y_1^\mu \\ y_2^\mu \\ \vdots \\ y_m^\mu \end{pmatrix} \in A^m$$

La *fase de aprendizaje* consiste de dos etapas:

1.- Para cada una de las  $p$  asociaciones  $(\mathbf{x}^\mu, \mathbf{y}^\mu)$  se encuentra la matriz  $\mathbf{y}^\mu \cdot (\mathbf{x}^\mu)^t$  de dimensiones  $m \times n$

$$\mathbf{y}^\mu \cdot (\mathbf{x}^\mu)^t = \begin{pmatrix} y_1^\mu \\ y_2^\mu \\ \vdots \\ y_m^\mu \end{pmatrix} \cdot (x_1^\mu, x_2^\mu, \dots, x_n^\mu)$$

$$\mathbf{y}^\mu \cdot (\mathbf{x}^\mu)^t = \begin{pmatrix} y_1^\mu x_1^\mu & y_1^\mu x_2^\mu & \dots & y_1^\mu x_j^\mu & \dots & y_1^\mu x_n^\mu \\ y_2^\mu x_1^\mu & y_2^\mu x_2^\mu & \dots & y_2^\mu x_j^\mu & \dots & y_2^\mu x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ y_i^\mu x_1^\mu & y_i^\mu x_2^\mu & \dots & y_i^\mu x_j^\mu & \dots & y_i^\mu x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ y_m^\mu x_1^\mu & y_m^\mu x_2^\mu & \dots & y_m^\mu x_j^\mu & \dots & y_m^\mu x_n^\mu \end{pmatrix} \quad (4.1)$$

2.- Se suman la  $p$  matrices para obtener la memoria

$$\mathbf{M} = \sum_{\mu=1}^p \mathbf{y}^\mu \cdot (\mathbf{x}^\mu)^t = [m_{ij}]_{m \times n} \quad (4.2)$$

de manera que la  $ij$ -ésima componente de la memoria  $\mathbf{M}$  se expresa así:

$$m_{ij} = \sum_{\mu=1}^p y_i^\mu x_j^\mu \quad (4.3)$$

La *fase de recuperación* consiste en presentarle a la memoria un patrón de entrada  $\mathbf{x}^\omega$ , donde  $\omega \in \{1, 2, \dots, p\}$  y realizar la operación

$$\mathbf{M} \cdot \mathbf{x}^\omega = \left[ \sum_{\mu=1}^p \mathbf{y}^\mu \cdot (\mathbf{x}^\mu)^t \right] \cdot \mathbf{x}^\omega \quad (4.4)$$

Al desarrollar la sumatoria de la expresión 4.4, se tiene:

$$\begin{aligned} M \cdot x^\omega &= \left[ y^1 \cdot (x^1)^t + y^2 \cdot (x^2)^t + \dots + y^\omega \cdot (x^\omega)^t + \dots + y^p \cdot (x^p)^t \right] \cdot x^\omega \\ M \cdot x^\omega &= \left[ y^1 \cdot (x^1)^t \right] \cdot x^\omega + \left[ y^2 \cdot (x^2)^t \right] \cdot x^\omega + \dots + \left[ y^\omega \cdot (x^\omega)^t \right] \cdot x^\omega + \dots \\ &\quad + \left[ y^p \cdot (x^p)^t \right] \cdot x^\omega \\ M \cdot x^\omega &= y^1 \cdot \left[ (x^1)^t \cdot x^\omega \right] + y^2 \cdot \left[ (x^2)^t \cdot x^\omega \right] + \dots + y^\omega \cdot \left[ (x^\omega)^t \cdot x^\omega \right] + \dots \\ &\quad + y^p \cdot \left[ (x^p)^t \cdot x^\omega \right] \end{aligned}$$

Y esto último se puede escribir de la siguiente manera, usando de nuevo la notación sumatoria:

$$M \cdot x^\omega = y^\omega \cdot \left[ (x^\omega)^t \cdot x^\omega \right] + \sum_{\mu \neq \omega} y^\mu \cdot \left[ (x^\mu)^t \cdot x^\omega \right] \quad (4.5)$$

La forma de la expresión 4.5 nos permite investigar las condiciones que deben cumplir para que el método de recuperación propuesto dé como resultado salidas perfectas. Para que la expresión anterior arroje como resultado al patrón  $y^\omega$ , es preciso que se cumplan dos igualdades:

- a)  $\left[ (x^\omega)^t \cdot x^\omega \right] = 1$
- b)  $\left[ (x^\mu)^t \cdot x^\omega \right] = 0$  siempre que  $\mu \neq \omega$

Dado que  $\omega$  se escogió arbitrariamente, las dos igualdades se deben cumplir  $\forall \omega \in \{1, 2, \dots, p\}$ ; lo cual indica que los vectores de entrada deben ser ortonormales. Esta condición de ortonormalidad se puede resumir en la siguiente expresión:

$$(x^\mu)^t \cdot x^\omega = \delta_{\mu\omega} = \begin{cases} 1 & \text{si } \mu = \omega \\ 0 & \text{si } \mu \neq \omega \end{cases} \quad (4.6)$$

donde  $\delta_{\mu\omega}$  es la conocida *delta de Kronecker* (Moore, 1968).

Si se cumple la condición que se manifiesta en la expresión 4.6, entonces la recuperación es perfecta; es decir, la expresión 4.5 toma la forma  $\mathbf{M} \cdot \mathbf{x}^\omega = \mathbf{y}^\omega$ .

Sin embargo, si los vectores de entrada no son ortonormales, suceden dos cosas:

- el factor  $[(\mathbf{x}^\omega)^t \cdot \mathbf{x}^\omega]$  no es 1
- el término  $\sum_{\mu \neq \omega} \mathbf{y}^\mu \cdot [(\mathbf{x}^\mu)^t \cdot \mathbf{x}^\omega]$  no es 0

Este último término, llamado *cross-talk*, representa el ruido producido por la interacción entre los patrones de entrada, y tiene como consecuencia inmediata que la recuperación no es perfecta, excepto si el número de patrones almacenados es pequeño comparado con la dimensión  $n$  de los vectores de entrada. Algunos investigadores afirman que ese número pequeño de patrones debe estar entre  $0.1n$  y  $0.2n$  (Anderson & Rosenfeld, 1990; Hassoun, 1995; Ritter, Sussner & Díaz-de-Lcón, 1998).

### Ejemplo.

Ejemplifiquemos el funcionamiento del *Linear Associator* con tres parejas de patrones ( $p = 3$ ). Los patrones de entrada son de dimensión 3 y los de salida, de dimensión 5, lo cual significa que  $n = 3$  y  $m = 5$ ; además, los tres vectores de entrada con la expresión 4.6, es decir, son vectores ortonormales:

$$\mathbf{x}^1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad \mathbf{y}^1 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}; \quad \mathbf{x}^2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \quad \mathbf{y}^2 = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix};$$

$$\mathbf{x}^3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad \mathbf{y}^3 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

Calculemos los términos  $\mathbf{y}^\mu \cdot (\mathbf{x}^\mu)^t$  usando la expresión 4.1:

$$\mathbf{y}^1 \cdot (\mathbf{x}^1)^t = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \cdot (1, 0, 0) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

$$\mathbf{y}^2 \cdot (\mathbf{x}^2)^t = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \cdot (0.1.0) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

$$\mathbf{y}^3 \cdot (\mathbf{x}^3)^t = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \cdot (0.0.1) = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

Se obtiene la memoria  $\mathbf{M}$  a partir de las expresiones 4.2 y 4.3:

$$\mathbf{M} = \sum_{\mu=1}^3 \mathbf{y}^{\mu} \cdot (\mathbf{x}^{\mu})^t = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\mathbf{M} = \sum_{\mu=1}^3 \mathbf{y}^{\mu} \cdot (\mathbf{x}^{\mu})^t = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \quad (4.7)$$

Se usa la expresión 4.4 para recuperar los patrones de salida. Obsérvese que las tres recuperaciones son perfectas, porque se cumple la condición de ortonormalidad en los patrones de entrada.

$$\mathbf{M} \cdot \mathbf{x}^1 = \left[ \sum_{\mu=1}^3 \mathbf{y}^{\mu} \cdot (\mathbf{x}^{\mu})^t \right] \cdot \mathbf{x}^1 = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \mathbf{y}^1$$

$$M \cdot x^2 = \left[ \sum_{\mu=1}^3 y^{\mu} \cdot (x^{\mu})^t \right] \cdot x^2 = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = y^2$$

$$M \cdot x^3 = \left[ \sum_{\mu=1}^3 y^{\mu} \cdot (x^{\mu})^t \right] \cdot x^3 = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = y^3$$

A continuación veremos los devastadores efectos que tiene, en la fase de recuperación, un término de *cross-talk* diferente de cero; para ello, se añadirá una nueva asociación al conjunto fundamental, con la propiedad de que el patrón de entrada de la nueva pareja no es ortonormal a los tres patrones originales.

Ahora el valor de  $p$  se incrementa a 4 con la pareja siguiente:

$$x^4 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \quad y^4 = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

El término  $y^4 \cdot (x^4)^t$  resulta ser:

$$y^4 \cdot (x^4)^t = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{pmatrix} \cdot (1, 0, 1) = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix} \quad (4.8)$$



La nueva memoria, denotada por  $M_{4parejas}$  se calcula a partir de las expresiones ?? y 4.8:

$$\begin{aligned} M_{4parejas} &= M + y^4 \cdot (x^4)^t \\ &= \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 2 \\ 2 & 1 & 1 \end{pmatrix} \end{aligned}$$

Intentemos recuperar los cuatro patrones:

$$\begin{aligned} M_{4parejas} \cdot x^1 &= \left[ \sum_{\mu=1}^4 y^\mu \cdot (x^\mu)^t \right] \cdot x^1 \\ &= \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 2 \\ 2 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 2 \end{pmatrix} \neq y^1 \end{aligned}$$

$$\begin{aligned} M_{4parejas} \cdot x^2 &= \left[ \sum_{\mu=1}^4 y^\mu \cdot (x^\mu)^t \right] \cdot x^2 \\ &= \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 2 \\ 2 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = y^2 \end{aligned}$$

$$\begin{aligned}
M_{1parejas} \cdot x^3 &= \left[ \sum_{\mu=1}^4 y^{\mu} \cdot (x^{\mu})^t \right] \cdot x^3 \\
&= \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 2 \\ 2 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 2 \\ 1 \end{pmatrix} \neq y^3
\end{aligned}$$

$$\begin{aligned}
M_{1parejas} \cdot x^4 &= \left[ \sum_{\mu=1}^4 y^{\mu} \cdot (x^{\mu})^t \right] \cdot x^4 \\
&= \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 2 \\ 2 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 2 \\ 3 \\ 3 \end{pmatrix} \neq y^4
\end{aligned}$$

Y así se termina el encanto del *Linear Associator*.

### 4.3. Consideraciones finales del capítulo.

En este capítulo hemos presentado el *Linear Associator*, un modelo pionero de memoria asociativa, producto del trabajo de dos grandes científicos en este campo: el neurofisiólogo estadounidense James A. Anderson con su *Interactive Memory* (Anderson, 1972), y el físico e ingeniero eléctrico finlandés Teuvo Kohonen, con sus *Correlation Matrix Memories* (Kohonen, 1972).

Anderson ha continuado activo, hasta la fecha, en los campos de las memorias asociativas y de las redes neuronales; él y Edward Rosenfeld se dieron a la tarea de editar uno de los compendios de Neurocomputación más importantes de todos los tiempos: *Neurocomputing: Foundations of Research* (Anderson & Rosenfeld (Eds.), 1990), cuya segunda parte *Neurocomputing: Directions for Research* (en colaboración con Andras Pellionisz) apareció en 1993.

Por su parte, Kohonen se ha convertido en un prolífico autor de artículos y libros relacionados con la generación constante de nuevos conceptos e inventos, de gran trascendencia en los ámbitos científico, ingenieril e industrial (Kohonen, T., 1972; Kohonen, T., 1974; Kohonen, T., 1987; Kohonen, T., 1989; Kohonen, T., 1997). Así, este ilustre personaje ha dado a conocer la DAM (distributed associative memory), los OAM (optimal associative mappings), el LSM (learning subspace method), los SOMs (self-organizing feature maps), la LVQ (learning vector quantization), el RHA (redundant hash addressing) y el ASSOM (Adaptive-Subspace SOM).

Recientemente, Kohonen ha desarrollado una nueva arquitectura de SOM para minería de datos, el WEBSOM, que permite organizar hasta siete millones de documentos en un millón de redes neuronales

*[websom.hut.fi/websom/](http://websom.hut.fi/websom/)*

# Capítulo 5

## Amari.

### 5.1. Contexto.

No está por demás enfatizar el papel clave que jugó el año de 1972 para el área de las memorias asociativas, pues como se indicó en capítulos anteriores, este año fue testigo de una febril actividad de los investigadores pioneros en memorias asociativas.

Uno de los cuatro grandes de la generación 1972, además de James A. Anderson con su *Interactive Memory* (Anderson, 1972). Teuvo Kohonen y sus *Correlation Matrix Memories* (Kohonen, 1972) y Kaoru Nakano y su *Associatron* (Nakano, 1972), es, como ya se mencionó en el capítulo 4. Shunichi Amari, profesor de la *University of Tokyo* y uno de los más prolíficos escritores de artículos científicos hasta nuestros días.

En 1972 Amari publicó un trabajo teórico donde continuaba con sus investigaciones sobre las *Self-Organizing Nets of Threshold Elements* (Amari, 1972), tópico cuyo estudio había iniciado un par de años atrás. Este trabajo de Amari constituye una valiosa muestra del estilo teórico característico de este personaje y representa, históricamente, un importante antecedente para la creación de lo que una década más tarde se convertiría en el modelo de memoria asociativa por antonomasia: la memoria Hopfield.

Bosquejos de algunas de las ideas que aparecen en los trabajos de Amari son el tema central de este capítulo.

## 5.2. El trabajo de Amari.

Mientras que el *Linear Associator* resume la esencia de los trabajos pioneros de Steinbuch, Anderson, Kohonen y Nakano (ver capítulos 1 a 4), el trabajo de Amari apunta en otra dirección: propone métodos y enfoques teóricos novedosos para algunos conceptos conocidos, y pretende establecer nuevas teorías sobre cierto tipo de procesos, entre los que destaca el tema de nuestro interés, la existencia y funcionamiento de las memorias asociativas de patrones.

A diferencia de la *Lernmatrix*, del *correlograph* y del *Linear Associator*, en tanto que modelos factibles de simularse o implementarse casi de inmediato, las *Self-Organizing Nets of Threshold Elements* son estudiadas por Shun-ichi Amari desde un punto de vista teórico (Amari, 1972). Aún más, cinco años después volvió a la carga tomando como base estos resultados, para proponer una teoría neuronal de la asociación y la formación de conceptos (Amari, 1977).

El artículo de 1972 incluye un buen número de definiciones, tres lemas, ocho teoremas y un corolario, parte de los cuales tienen que ver con memorias asociativas. La información que se presenta a continuación está tomada libremente del contenido de este artículo, con las adaptaciones pertinentes y adecuadas en notación y estilo.

Con un enfoque teórico, el autor desarrolla los siguientes *temas*:

1. Una corta descripción de las ya conocidas (en 1972) redes de elementos de umbral, las transiciones de estados y estados de equilibrio
2. Definiciones de los números de estabilidad de la transición de estados y los números de estabilidad de los estados de equilibrio
3. Condiciones de convergencia para los números de estabilidad de los estados de equilibrio
4. Los fundamentos de la auto-organización de las redes de elementos de umbral
5. Las redes auto-organizativas de elementos de umbral como memorias asociativas

Parafraseando al autor, "una red auto-organizativa de elementos de umbral tiene la propiedad de que su estructura varía gradualmente, dependiendo de los patrones aplicados a la red, como estímulos externos. Cuando los patrones de estímulo se aplican repetidamente a la red, se espera que ésta aprenda de los estímulos y fije algunos de ellos como estados de equilibrio estables, por medio de auto-organización. Se puede decir que la red *recuerda* esos patrones. Una vez que un patrón es recordado como un estado de equilibrio estable, será recuperado y reproducido correctamente cuando un patrón vecino se da como estímulo."

La información para el *tema 1*, cuyo resumen se presenta a continuación, procede de 12 fuentes a las que el autor hace referencia, entre ellas tres trabajos previos de él mismo, un artículo de Krohn & Rhodes de 1965 titulado precisamente *Nets of threshold elements*, y la edición original (1969) del famoso libro *Perceptrons* (Minsky & Papert, 1988).

Un elemento de umbral de  $n$  entradas se especifica por  $n$  cantidades  $w_1, w_2, \dots, w_n$  llamados pesos, y una cantidad  $h$  llamada umbral. El elemento se denota por  $E(w_1, w_2, \dots, w_n; h)$  o más brevemente  $E(w_i; h)$ . Consideremos  $n$  variables de entrada  $x_1, x_2, \dots, x_n$  cuyo dominio es el conjunto  $\{-1, 1\}$ . La salida  $y$  es 1 cuando la suma pesada  $\sum_i w_i x_i$  excede  $h$ ; de otro modo, la salida es  $-1$ . Por lo tanto, la relación de entrada-salida del elemento se describe por la siguiente expresión:

$$y = \operatorname{sgn} \left( \sum_i w_i x_i - h \right)$$

donde

$$\operatorname{sgn}(u) = \begin{cases} 1 & \text{si } u > 0 \\ -1 & \text{si } u < 0 \end{cases}$$

Amari asume, "con el fin de evitar discusiones innecesariamente complicadas", que el caso de igualdad nunca ocurre. Además, es posible hacer una simplificación adicional: si  $c$  es una constante positiva, los elementos de umbral  $E(w_i; h)$  y  $E(cw_i; ch)$  tienen las mismas características, por lo que

es válido considerar que  $|w_i| \leq 1$  y que además los pesos están normalizados de manera que el valor máximo de todos los  $|w_i|$  es precisamente 1.

Sea una red compuesta de  $n$  elementos de umbral  $E_1, E_2, \dots, E_n$  donde  $E_j = E(w_{j1}, w_{j2}, \dots, w_{jn}; h)$ . Estos elementos están conectados de manera tal que la salida de  $E_i$ , denotada por  $y_i$ , está conectada con las  $i$ -ésimas entradas de todos los elementos con un retraso de una unidad de tiempo; es decir, la salida  $y_i$  se multiplica por  $w_{ji}$  cuando se opera con  $E_j$ .

La red trabaja de modo síncrono en intervalos fijos de tiempo y está especificada por la matriz de pesos  $(w_{ji})_{n \times n}$  y el vector de umbrales  $(h_i)_n$ . El vector columna

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}$$

donde  $y_i$  es la salida actual del elemento  $E_i$ , se llama el estado actual de la red.

El estado siguiente  $\mathbf{y}'$  está determinado unívocamente por el estado actual  $\mathbf{y}$ , debido a que la salida  $y'_i$  del elemento  $E_i$  en el tiempo siguiente está determinada por

$$y'_i = \text{sgn} \left( \sum_j w_{ij} y_j - h_i \right)$$

Si se denota el operador de transición de estados por  $T$ , el estado siguiente  $\mathbf{y}'$  se expresa así:

$$\mathbf{y}' = T\mathbf{y}$$

Un estado de equilibrio es un estado invariante bajo  $T$ ; es decir,  $\mathbf{y}_e$  es un estado de equilibrio si  $\mathbf{y}_e = T\mathbf{y}_e$ .

Un conjunto ordenado de  $m$  estados  $B = \{y_1, y_2, \dots, y_m\}$  para los cuales se cumple que  $y_{i+1} = Ty_i$  donde  $i = 1, 2, \dots, (m - 1)$  se llama *secuencia de transiciones de estados* de longitud  $m$ .

Una secuencia de transiciones de estados  $C = \{y_1, y_2, \dots, y_r\}$  para la cual se cumple que  $Ty_r = y_1$ , se llama *ciclo* de longitud  $r$ .

Dado que el número de estados en una red es finito, la secuencia de estados  $y_0, Ty_0, T^2y_0, T^3y_0, T^4y_0, \dots$  donde  $y_0$  es arbitrario, se comporta de dos posibles maneras:

- converge a un estado de equilibrio,
- o bien, cae en un ciclo dentro de un número finito de transiciones de estado

Los estados de equilibrio y los ciclos son patrones que la red puede *retener persistentemente* sin entradas externas.

Los temas 2 a 5 son las aportaciones originales de Amari.

En lo que sigue, y tomando como base la información de los párrafos anteriores, presentaremos algunos comentarios únicamente sobre los contenidos que están relacionados directamente con la capacidad que exhiben las redes de elementos de umbral para actuar como memorias asociativas. Es decir, no habrá comentarios sobre las definiciones, lemas y teoremas relacionados con la recuperación de secuencias y la formación de conceptos, tópicos importantes en el artículo, pero ajenos al tema de nuestro interés.

Con objeto de definir y usar los números de estabilidad mencionados en los temas 2 y 3, Amari echó mano de dos conceptos: el de distancia de Hamming entre dos estados  $\mathbf{x}$  y  $\mathbf{y}$ , expresada como  $dis(\mathbf{x}, \mathbf{y}) = \frac{1}{2} \sum_i |x_i - y_i|$ , y el de  $k$ -vecindad de  $\mathbf{x}$ , cuya expresión es  $N(\mathbf{x}, k) = \{\mathbf{y} \mid dis(\mathbf{x}, \mathbf{y}) \leq k\}$ .

A partir de estos conceptos, definió las  $n$  funciones

$$u_i(\mathbf{x}, \mathbf{y}) = x_i \left( \sum_j w_{ij} y_j - h_i \right)$$

Es claro que  $u_i(\mathbf{x}, \mathbf{y})$  es positiva cuando la  $i$ -ésima componente de



$$Ty = \sum_j w_{ij} y_j - h_i$$

coincide con la de  $x$ . y negativa de otra manera.

La combinación de estas funciones y los operadores  $\min(k)$ , propuestos también por Amari. hace posible llegar a la definición de los números de estabilidad.

Para  $n$  números reales  $r_1, r_2, \dots, r_n$ , el operador  $\min(k)$  denota el  $(k+1)$ -ésimo número más pequeño de todos los  $r_i$ ; es decir, si los  $r_i$  se ordenan de acuerdo con su magnitud de manera que  $r_{i_1} \leq r_{i_2} \leq r_{i_3} \leq \dots \leq r_{i_n}$ , el operador se define así:

$$\min(k) \{r_i\} = \begin{cases} r_{i_{k+1}} & \text{si } 0 \leq k < n \\ r_{i_n} & \text{si } k \geq n \end{cases}$$

El primer resultado del artículo, que ilustra el papel de las funciones  $u_i(x, y)$  y del operador  $\min(k)$ , afirma que una condición necesaria y suficiente para que un estado  $y$  caiga dentro de la  $k$ -vecindad de  $x$  después de una transición simple de estados, es decir  $Ty \in N(x, k)$ , es  $\min(k) \{u_i(x, y)\} > 0$ .

Un caso particular del resultado anterior es de gran trascendencia: veamos: al hacer  $k = 0$ ,  $Ty \in N(x, k)$  se convierte en  $Ty \in N(x, 0)$  y esto significa ni más ni menos que  $Ty = Tx$ .

Si además de  $k = 0$  hacemos  $y = x$ , entonces obtenemos el primer teorema:

$$\text{El estado } y \text{ es de equilibrio} \leftrightarrow \min_i \{u_i(y, y)\} > 0$$

Con la información y notación anteriores es posible presentar la primera definición central de este artículo, el grado de estabilidad de la transición de estados.

Sea  $k$  un número no negativo: el número de  $k$ -estabilidad  $s(\mathbf{y}, k)$  de la transición de estado  $\mathbf{y} \rightarrow T\mathbf{y}$  se define como la parte entera de la cantidad  $\frac{1}{2} \min(k) \{u_i(\mathbf{y}, T\mathbf{y})\}$ : si  $[n]$  representa la parte entera de  $n$ , la definición del número de  $k$ -estabilidad es:

$$s(\mathbf{y}, k) = \left[ \frac{1}{2} \min(k) \{u_i(\mathbf{y}, T\mathbf{y})\} \right]$$

El segundo resultado, que ilustra el significado del número de  $k$ -estabilidad  $s(\mathbf{y}, k)$ , afirma que si  $\mathbf{z}$  es un estado que pertenece a la  $s(\mathbf{y}, k)$ -vecindad de  $\mathbf{y}$ , entonces  $T\mathbf{z}$  pertenece a la  $k$ -vecindad de  $T\mathbf{y}$ ; simbólicamente:

$$[\mathbf{z} \in N(\mathbf{y}, s(\mathbf{y}, k))] \rightarrow [T\mathbf{z} \in N(T\mathbf{y}, k)]$$

En particular,  $[\mathbf{z} \in N(\mathbf{y}, s(\mathbf{y}, 0))] \rightarrow [T\mathbf{z} = T\mathbf{y}]$ .

Los números de estabilidad de un estado de equilibrio  $\mathbf{y}$  se definen mediante el número de  $k$ -estabilidad  $s(\mathbf{y}, k)$ ; el primer número de estabilidad  $s_1(\mathbf{y})$  de  $\mathbf{y}$  es definido por  $s_1(\mathbf{y}) = s(\mathbf{y}, 0) = \left[ \frac{1}{2} \min(0) \{u_i(\mathbf{y}, \mathbf{y})\} \right]$ , y los demás se definen recursivamente:  $s_j(\mathbf{y}) = s(\mathbf{y}, s_{j-1}(\mathbf{y}))$ .

La secuencia  $s_1(\mathbf{y}), s_2(\mathbf{y}), \dots$  es monótona no decreciente y converge al límite  $s(\mathbf{y})$  en un número finito de términos. Al límite  $s(\mathbf{y})$  se le llama *número de estabilidad del estado de equilibrio  $\mathbf{y}$* .

Un estado de equilibrio  $\mathbf{y}$  es *estable* si  $s(\mathbf{y}) > 0$  e *inestable* si  $s(\mathbf{y}) = 0$ . El estado de equilibrio  $\mathbf{y}$  será inestable si y sólo si  $s_1(\mathbf{y}) = 0$ .

El  $k$ -ésimo *dominio de estabilidad* de un estado de equilibrio  $\mathbf{y}$  es la  $s_k(\mathbf{y})$ -vecindad  $N(\mathbf{y}, s_k(\mathbf{y}))$  y se denota por  $D_k(\mathbf{y})$ ; es decir,  $D_k(\mathbf{y}) = N(\mathbf{y}, s_k(\mathbf{y}))$ .

Dos casos particulares importantes son  $D_0(\mathbf{y}) = N(\mathbf{y}, 0) = \{\mathbf{y}\}$  y  $D(\mathbf{y}) = N(\mathbf{y}, s(\mathbf{y}))$ ; a  $D(\mathbf{y})$  se le llama simplemente *dominio de estabilidad* de  $\mathbf{y}$ .

Además, debido a que la secuencia  $s_1(\mathbf{y}), s_2(\mathbf{y}), \dots$  es monótona no decreciente y converge a  $s(\mathbf{y})$ , se cumple que

$$D_0(\mathbf{y}) \subset D_1(\mathbf{y}) \subset D_2(\mathbf{y}) \subset \dots \subseteq D(\mathbf{y})$$

A continuación se presenta el enunciado completo del teorema 2 del artículo. La importancia de este teorema radica en que establece las condiciones para que se dé la convergencia de la red a los estados de equilibrio, en un número finito de pasos. Un resultado similar es la piedra angular del famoso artículo de J. Hopfield, que marcó el nacimiento de la memoria Hopfield, una década después.

**Teorema 2.-** Sea  $\mathbf{y}$  un estado de equilibrio. Entonces, la red arriba al estado  $\mathbf{y}$  después de un número finito de transiciones de estado, si su estado inicial pertenece a  $D(\mathbf{y})$ . Más específicamente, la red arriba al estado  $\mathbf{y}$  en  $k$  transiciones si el estado inicial pertenece a  $D_k(\mathbf{y})$ . ♦

En la presentación del *tema 4*, los fundamentos de la auto-organización de las redes de elementos de umbral, Amari asume que los patrones se aplican a la red como estímulos externos, y que un patrón se representa como un vector columna  $n$ -dimensional cuyas componentes son  $\pm 1$ , donde  $n$  es el número de elementos de la red.

Se asume también que la red se fuerza a llegar al estado  $\mathbf{x}$  cuando se aplica el patrón estímulo  $\mathbf{x}$ . Sea  $\mathbf{x}(t)$  el patrón estímulo aplicado en el tiempo  $t$  ( $t = 1, 2, \dots$ ).

Al auto-organizarse, los pesos de la red cambian al recibir los patrones de estímulo, de acuerdo con la siguiente expresión para el incremento de los pesos en el tiempo  $t$ :

$$\Delta w_{ij}(t) = -aw_{ij}(t) + bx_i(t)x_j(t)$$

donde  $a$  y  $b$  son constantes positivas pequeñas.

Así llegamos al *tema 5*, las redes auto-organizativas de elementos de umbral como memorias asociativas.

Se considera que los pesos iniciales de la red auto-organizativa están dados por:

$$w_{ij}^0 = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases}$$

Sin pérdida de generalidad, se considera además que  $h_i = 0, \forall i$ .

El cumplimiento de estas condiciones conlleva al hecho de que todos los estados están inicialmente en equilibrio; esto se concluye de la definición de  $u_i(\mathbf{x}, \mathbf{y})$  y del primer teorema:

$$u_i(\mathbf{x}, \mathbf{x}) = x_i \left( \sum_j w_{ij} x_j - h_i \right) = x_i \sum_j w_{ij}^0 x_j = 1$$

y de aquí resulta que  $\min_i \{u_i(\mathbf{x}, \mathbf{x})\} > 0$ .

De acuerdo con el primer teorema, todos los estados  $\mathbf{x}$  están inicialmente en equilibrio. Por consiguiente, la red puede retener persistentemente cualquier estado inicial, pero sus números de estabilidad son cero (porque  $s_1(\mathbf{x}) = s(\mathbf{x}, 0) = [\frac{1}{2} \min(0) \{u_i(\mathbf{x}, \mathbf{x})\}] = 0$ ) y todos los estados son inestables.

El asumir que  $h_i = 0, \forall i$  tiene como consecuencia inmediata que si se cumple  $T\mathbf{x} = \mathbf{y}$  entonces también se cumple  $T(-\mathbf{x}) = (-\mathbf{y})$ . Dado que el comportamiento de  $(-\mathbf{x})$  está determinado por el comportamiento de  $\mathbf{x}$ , si  $\mathbf{x}$  es estable, entonces  $(-\mathbf{x})$  también será estable.

Se muestra la auto-organización de la red para el caso en que un solo patrón con ruido se aplica repetidamente. Sea  $\tilde{\mathbf{x}}$  un patrón aleatorio cuyas componentes se obtienen cambiando el signo de cada componente de  $\mathbf{x}$  en forma independiente con probabilidad  $p$ , es decir:

$$\tilde{x}_i = \begin{cases} x_i & \text{con probabilidad } 1 - p \\ -x_i & \text{con probabilidad } p \end{cases}$$

Se propone el parámetro  $\sigma^2 = 4p(1 - p)$  como una cantidad que representa el ruido del patrón  $\tilde{\mathbf{x}}$ . Cuando  $p = 0$ , no hay ruido, y cuando  $p = \frac{1}{2}$ ,  $\sigma^2 = 1$ . En este último caso,  $\tilde{\mathbf{x}}$  no incluye información original del patrón  $\mathbf{x}$ .

Anari presenta el comportamiento de las redes auto-organizativas de elementos de umbral como memorias asociativas en los teoremas 4 y 5 de su artículo.

El teorema 4 exhibe el caso en que la red aprende un único patrón de estímulo, y el teorema 5 trata el caso más general, cuando  $m$  patrones se aplican repetidamente con cierta frecuencia a la red, y las condiciones para que la red pueda recuperar los patrones asociativamente.

**Teorema 4.-** Cuando un patrón  $\mathbf{x}$  se aplica repetidamente bajo distorsiones de ruido de intensidad  $\sigma^2$ , la red se auto-organiza de manera que se cumple lo siguiente:

· El patrón  $\mathbf{x}$  es almacenado en la red como un estado de equilibrio y su número de estabilidad es

$$s(\mathbf{x}) = \left[ \frac{1 - \sigma^2}{2} n + \frac{\sigma^2}{2} \right]$$

· Aquellos estados  $\mathbf{y}$  que satisfacen la relación

$$|\cos(\mathbf{x}, \mathbf{y})| < \frac{\sigma^2}{n(1 - \sigma^2)}$$

permanecen como estados de equilibrio inestable con número de estabilidad 0. Todos los demás patrones cambian a  $\mathbf{x}$  en una transición simple.♦

En la última expresión,  $\cos(\mathbf{x}, \mathbf{y})$  es el coseno del ángulo entre los patrones  $\mathbf{x}$  y  $\mathbf{y}$ , el cual está dado por  $\cos(\mathbf{x}, \mathbf{y}) = \frac{1}{n} \mathbf{x} \cdot \mathbf{y}$ , y  $\mathbf{x} \cdot \mathbf{y}$  es el producto interno.

Se puede decir que la red recuerda a  $\mathbf{x}$  correctamente como un estado de equilibrio, para valores de  $n$  suficientemente grandes. Consideremos el importante caso en el que se presenta a la red un patrón  $\mathbf{y}$  que cumple con

$$|\cos(\mathbf{x}, \mathbf{y})| \geq \frac{\sigma^2}{n(1 - \sigma^2)}$$

después de que el aprendizaje es completado. La red es puesta en el estado  $\mathbf{y}$  con la llegada del patrón externo, pero inmediatamente cambia

al estado  $\mathbf{x}$  y permanece en él. En otras palabras, la memoria asociativa recupera el patrón  $\mathbf{x}$  a partir de un patrón ruidoso de entrada  $\mathbf{y}$ .

Se omitirá el enunciado del teorema 5. debido a que su aportación no va más allá de lo exhibido por el *Linear Associator*: si los patrones aprendidos son ortogonales, la recuperación será perfecta.

En su artículo de 1977, Amari toma como base los resultados del artículo de 1972, y propone una teoría neuronal de la asociación y la formación de conceptos (Amari, 1977).

En un estilo teórico aún más riguroso, el autor expone su idea de que los componentes básicos de una red no son elementos simples. sino pequeños grupos de neuronas conectadas entre sí: Amari los llama *neuron pools*; muestra que un *neuron pool* puede actuar como un elemento de dos estados.

Respecto del tema de memorias asociativas, Amari toma las ideas del artículo de 1972 para definir los sistemas asociativos de manera tal que un patrón de entrada  $\mathbf{x}$  da lugar a un patrón de salida  $\mathbf{y}$  a través de una transformación  $T$ : es decir,  $\mathbf{y} = T\mathbf{x}$ .

Enfatiza la importancia de las relaciones de ortogonalidad para lograr salidas perfectas en la memoria asociativa, pues en el caso ortonormal desaparece el término de *cross-talk*, como ya se había mencionado al presentar el *Linear Associator*.

### 5.3. Consideraciones finales del capítulo.

En este capítulo hemos presentado el bosquejo de algunas de las ideas que aparecen en los trabajos del notable y prolífico teórico japonés, ingeniero matemático Shun-ichi Amari.

El Profesor Amari ha escrito más de 300 artículos internacionales en varias decenas de revistas de gran prestigio, ha participado en al menos 5 series de libros especializados incluyendo *Neural Networks: Research and Applications* (Pergamon Press), ha obtenido una docena de premios internacionales de investigación científica y es autor, entre otros muchos, del libro *Mathematical Theory of Nerve Nets*.

Actualmente es el Director of Brain Information Processing Group, RIKEN Frontier Research Program, en Japón ([www.islab.brain.riken.go.jp/~amari/](http://www.islab.brain.riken.go.jp/~amari/)).



## Capítulo 6

# Memoria Asociativa Hopfield.

### 6.1. Contexto.

Si 1972 fue el año de los pioneros en el área de las memorias asociativas, 1982 fue el año de John J. Hopfield. Su artículo de ese año (Hopfield, 1982), publicado por la prestigiosa y respetada *National Academy of Sciences* (en sus *Proceedings*), impactó positivamente y trajo a la palestra internacional su memoria asociativa.

Las autorizadas voces de los editores del compendio *Neurocomputing* (Anderson. & Rosenfeld (Eds.), 1990) aseguran que la era moderna de las memorias asociativas (y de las redes neuronales) nace a raíz de la publicación del artículo de Hopfield; afirman que el éxito de este artículo se debe en gran parte a que, además de tener un estilo claro, coherente y sofisticado, fue escrito por el distinguido y reconocido físico John J. Hopfield, “en cuyas manos la teoría se convierte en algo legítimo y respetable”.

La formación como físico del autor queda de manifiesto cuando declara que en los sistemas físicos constituidos por un gran número de elementos simples, las interacciones entre estos elementos dan lugar a fenómenos colectivos (las orientaciones de los dominios en sistemas magnéticos y los patrones de vórtices en sistemas de fluidos ejemplifican esta afirmación).

A partir de estas consideraciones, Hopfield se pregunta si la interacción de elementos simples de procesamiento similares a las *neuronas*, cuyo modelo simplificado se conocía desde hacía cuatro décadas (McCulloch & Pitts,



1943), da lugar a la aparición de propiedades computacionales colectivas, tales como la estabilidad de memorias: acto seguido, el autor afirma que en efecto, su artículo de 1982 demuestra que este tipo de propiedades computacionales aparecen espontáneamente.

Este capítulo está dedicado a la, hoy famosa, Memoria Asociativa Hopfield.

## 6.2. El trabajo de Hopfield.

En el legendario artículo (Hopfield, 1982) se considera un sistema físico descrito por un vector de estado  $\mathbf{x}$  cuyas coordenadas son  $(x_1, x_2, \dots, x_n)$ , se considera además que el sistema tiene puntos límite localmente estables  $\mathbf{x}^a, \mathbf{x}^b, \dots$ . Entonces, si el sistema es activado en un estado suficientemente cercano a cualquier punto límite localmente estable, digamos en  $\mathbf{x} = \mathbf{x}^a + \Delta$ , al transcurrir el tiempo el estado del sistema cambiará hasta llegar a que se cumpla  $\mathbf{x} \approx \mathbf{x}^a$ .

El punto de arranque  $\mathbf{x}^a + \Delta$  representa un conocimiento parcial del estado estable  $\mathbf{x}^a$ , y a partir de ahí el sistema genera la información correcta  $\mathbf{x}^a$ . Hopfield declara categóricamente que un sistema físico que se comporte de esta manera y que, además, sea susceptible de que *cualquier* conjunto predeterminado de estados pueda ser asignado como el conjunto de estados localmente estables, es útil como memoria asociativa.

Es preciso hacer notar la semejanza de estos puntos límite localmente estables de Hopfield, con los estados de equilibrio que pueden retener persistentemente, sin entradas externas, las redes de elementos de umbral de Amari, al actuar como memorias asociativas (ver capítulo anterior).

En el modelo que originalmente propuso Hopfield, cada neurona  $x_i$  tiene dos posibles estados, a la manera de las neuronas de McCulloch-Pitts:  $x_i = 0$  y  $x_i = 1$ ; sin embargo, en la sección *Studies of the collective behaviors of the model* del mismo artículo, el autor hace la relevante observación de que, para un nivel dado de exactitud en la recuperación de patrones, la capacidad de almacenamiento de información de la memoria se puede incrementar por un factor de 2, si se escogen como posibles estados de las neuronas los valores  $x_i = -1$  y  $x_i = 1$  en lugar de los valores originales  $x_i = 0$  y  $x_i = 1$ .

Al utilizar el conjunto  $\{1, -1\}$  y el valor de umbral cero, la fase de aprendizaje para la memoria Hopfield será similar, en cierta forma, a la fase de aprendizaje del *Linear Associator*.

La intensidad de la fuerza de conexión de la neurona  $x_i$  a la neurona  $x_j$  se representa por el valor de  $m_{ij}$ , y se considera que hay simetría, es decir,  $m_{ij} = m_{ji}$ . Si  $x_i$  no está conectada con  $x_j$ , entonces  $m_{ij} = 0$ ; en particular, no hay conexiones recurrentes de una neurona a sí misma, lo cual significa que  $m_{ii} = 0, \forall i$ . El estado instantáneo del sistema está completamente especificado por el vector columna de dimensión  $n$  cuyas coordenadas son los valores de las  $n$  neuronas.

La memoria Hopfield es autoasociativa, simétrica, con ceros en la diagonal principal.

En virtud de que la memoria es autoasociativa, el conjunto fundamental para la memoria Hopfield es  $\{\mathbf{x}^\mu, \mathbf{x}^\mu \mid \mu = 1, 2, \dots, p\}$  con

$$\mathbf{x}^\mu = \begin{pmatrix} x_1^\mu \\ x_2^\mu \\ \vdots \\ x_n^\mu \end{pmatrix} \in A^n \quad \text{y} \quad A = \{1, -1\}$$

La fase de aprendizaje para la memoria Hopfield es similar a la fase de aprendizaje del *Linear Associator*, con una diferencia relacionada con la diagonal principal en ceros, como se muestra en la siguiente regla para obtener la  $ij$ -ésima componente de la memoria Hopfield  $M$ :

$$m_{ij} = \begin{cases} \sum_{\mu=1}^p x_i^\mu x_j^\mu & \text{si } i \neq j \\ 0 & \text{si } i = j \end{cases} \quad (6.1)$$

Operativamente, el resultado de la expresión anterior se puede obtener en tres etapas:

1.- Para cada una de las  $p$  asociaciones  $(\mathbf{x}^\mu, \mathbf{x}^\mu)$  se encuentra la matriz  $\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t$  de dimensiones  $n \times n$ , la cual tiene unos en su diagonal principal:

$$\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t = \begin{pmatrix} x_1^\mu \\ x_2^\mu \\ \vdots \\ x_n^\mu \end{pmatrix} \cdot (x_1^\mu, x_2^\mu, \dots, x_n^\mu)$$

$$\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t = \begin{pmatrix} x_1^\mu x_1^\mu & x_1^\mu x_2^\mu & \dots & x_1^\mu x_i^\mu & \dots & x_1^\mu x_n^\mu \\ x_2^\mu x_1^\mu & x_2^\mu x_2^\mu & \dots & x_2^\mu x_i^\mu & \dots & x_2^\mu x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ x_i^\mu x_1^\mu & x_i^\mu x_2^\mu & \dots & x_i^\mu x_i^\mu & \dots & x_i^\mu x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ x_n^\mu x_1^\mu & x_n^\mu x_2^\mu & \dots & x_n^\mu x_i^\mu & \dots & x_n^\mu x_n^\mu \end{pmatrix} \quad (6.2)$$

2.- A cada una de las  $p$  matrices  $\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t$  se le resta la matriz identidad  $\mathbf{I}$  de dimensiones  $n \times n$ , con el fin de lograr ceros en la diagonal principal:

$$\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t - \mathbf{I} = \begin{pmatrix} 0 & x_1^\mu x_2^\mu & \dots & x_1^\mu x_i^\mu & \dots & x_1^\mu x_n^\mu \\ x_2^\mu x_1^\mu & 0 & \dots & x_2^\mu x_i^\mu & \dots & x_2^\mu x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ x_i^\mu x_1^\mu & x_i^\mu x_2^\mu & \dots & 0 & \dots & x_i^\mu x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ x_n^\mu x_1^\mu & x_n^\mu x_2^\mu & \dots & x_n^\mu x_i^\mu & \dots & 0 \end{pmatrix} \quad (6.3)$$

3.- Se suman la  $p$  matrices  $\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t - \mathbf{I}$  para finalmente obtener la memoria  $\mathbf{M}$  :

$$\mathbf{M} = \sum_{\mu=1}^p [\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t - \mathbf{I}] = [m_{ij}]_{n \times n} \quad (6.4)$$

La forma en que se lleva a cabo la *fase de recuperación* para la memoria Hopfield cambia drásticamente respecto de lo que sucede en el *Linear Associator*. Al presentar un patrón de entrada  $\tilde{\mathbf{x}}$  a la memoria Hopfield, ésta cambiará su estado con el tiempo, de modo que cada neurona  $x_i$  ajuste su valor de acuerdo con el resultado que arroje la comparación de la cantidad

$$\sum_{j=1}^n m_{ij} x_j$$

con un valor de umbral cuyo valor es normalmente cero.

Representemos el estado de la memoria Hopfield en el tiempo  $t$  por  $\mathbf{x}(t)$ : entonces  $x_i(t)$  representa el valor de la neurona  $x_i$  en el tiempo  $t$  y  $x_i(t+1)$  el valor de  $x_i$  en el tiempo siguiente ( $t+1$ ).

Dado un vector columna de entrada  $\tilde{\mathbf{x}}$ , la fase de recuperación consta de tres pasos:

1. Para  $t = 0$ , se hace  $\mathbf{x}(t) = \tilde{\mathbf{x}}$ ; es decir,  $x_i(0) = \tilde{x}_i, \forall i \in \{1, 2, 3, \dots, n\}$
2.  $\forall i \in \{1, 2, 3, \dots, n\}$  se calcula  $x_i(t+1)$  de acuerdo con la condición siguiente:

$$x_i(t+1) = \begin{cases} 1 & \text{si } \sum_{j=1}^n m_{ij}x_j(t) > 0 \\ x_i(t) & \text{si } \sum_{j=1}^n m_{ij}x_j(t) = 0 \\ -1 & \text{si } \sum_{j=1}^n m_{ij}x_j(t) < 0 \end{cases} \quad (6.5)$$

3. Se compara  $x_i(t+1)$  con  $x_i(t) \forall i \in \{1, 2, 3, \dots, n\}$ . Si  $\mathbf{x}(t+1) = \mathbf{x}(t)$  el proceso termina y el vector recuperado es  $\mathbf{x}(0) = \tilde{\mathbf{x}}$ . De otro modo, el proceso continúa de la siguiente manera: los pasos 2 y 3 se iteran tantas veces como sea necesario hasta llegar a un valor  $t = \tau$  para el cual  $x_i(\tau+1) = x_i(\tau) \forall i \in \{1, 2, 3, \dots, n\}$ : el proceso termina y el patrón recuperado es  $\mathbf{x}(\tau)$ .

El proceso de convergencia descrito en el paso 3 de la fase de recuperación, indica que el sistema llega a un punto límite localmente estable en el tiempo  $\tau$ .

La existencia de  $\tau$  está garantizada a través de la demostración que hace Hopfield de que existen puntos límite localmente estables en su modelo de memoria asociativa: para ello, define  $E$  de la siguiente manera, tomando en cuenta la condición de que  $m_{ii} = 0, \forall i$ :

$$E = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n m_{ij}x_i x_j \quad (6.6)$$

y acto seguido demuestra que  $E$  es una función de  $x_i$  monótona decreciente.

Además de la demostración sugerida por Hopfield, los investigadores McElicee *et. al.* esbozan otro camino para demostrar la convergencia de la expresión 6.6, en un importante trabajo donde analizan la capacidad de la memoria asociativa Hopfield (McElicee, Posner, Rodemich, & Venkatesh, 1987). Los desarrollos de la sugerencia de Hopfield y del esbozo de demostración de McElicee *et. al.* se presentan en el apéndice de esta obra.

El riguroso análisis de McElicee *et. al.* arroja como resultado principal que la memoria Hopfield es capaz de recuperar en forma perfecta  $\frac{n}{4 \log n}$  patrones donde  $n$  es la dimensión de los patrones.

Si se suaviza la restricción de que *todos* los patrones sean recuperados en forma perfecta, y se cambia por la condición de que la *mayoría* lo sean, entonces la capacidad se duplica: ahora la memoria Hopfield es capaz de recuperar  $\frac{n}{2 \log n}$  patrones, permitiendo leves errores.

En el artículo original de 1982, Hopfield había estimado empíricamente que su memoria tenía una capacidad de recuperar  $0,15n$  patrones, y en (Abu-Mostafa & St. Jacques, 1985) se había establecido formalmente que una cota superior para el número de vectores de estado arbitrarios estables en una memoria Hopfield es  $n$ .

Es pertinente apuntar que, como ya se ha mencionado antes en el capítulo 6, el teorema 2 del artículo (Amari, 1972) es un importante antecedente para este resultado de Hopfield. Los resultados son equivalentes pero con una presentación ligeramente diferente: Amari presenta el teorema de la convergencia de las redes de elementos de umbral en términos de estados de equilibrio que son alcanzados por la red en un número finito de transiciones de estado.

### Ejemplo.

Para ejemplificar la memoria Hopfield, usaremos patrones de dimensión  $n = 4$ ; iniciemos con tres vectores ( $p = 3$ ).

$$\mathbf{x}^1 = \begin{pmatrix} 1 \\ -1 \\ -1 \\ 1 \end{pmatrix}; \quad \mathbf{x}^2 = \begin{pmatrix} -1 \\ 1 \\ -1 \\ 1 \end{pmatrix}; \quad \mathbf{x}^3 = \begin{pmatrix} -1 \\ -1 \\ 1 \\ -1 \end{pmatrix}$$

Calculemos, de acuerdo con las expresiones 6.1, 6.2 y 6.3, las matrices  $\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t - \mathbf{I}$ :

$$\begin{aligned} \mathbf{x}^1 \cdot (\mathbf{x}^1)^t - \mathbf{I} &= \begin{pmatrix} 1 \\ -1 \\ -1 \\ 1 \end{pmatrix} \cdot (1, -1, -1, 1) - \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} 0 & -1 & -1 & 1 \\ -1 & 0 & 1 & -1 \\ -1 & 1 & 0 & -1 \\ 1 & -1 & -1 & 0 \end{pmatrix} \end{aligned}$$

$$\begin{aligned} \mathbf{x}^2 \cdot (\mathbf{x}^2)^t - \mathbf{I} &= \begin{pmatrix} -1 \\ 1 \\ -1 \\ 1 \end{pmatrix} \cdot (-1, 1, -1, 1) - \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} 0 & -1 & 1 & -1 \\ -1 & 0 & -1 & 1 \\ 1 & -1 & 0 & -1 \\ -1 & 1 & -1 & 0 \end{pmatrix} \end{aligned}$$

$$\begin{aligned} \mathbf{x}^3 \cdot (\mathbf{x}^3)^t - \mathbf{I} &= \begin{pmatrix} -1 \\ -1 \\ 1 \\ -1 \end{pmatrix} \cdot (-1, -1, 1, -1) - \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} 0 & 1 & -1 & 1 \\ 1 & 0 & -1 & 1 \\ -1 & -1 & 0 & -1 \\ 1 & 1 & -1 & 0 \end{pmatrix} \end{aligned}$$

La memoria Hopfield  $M$  se obtiene a partir de la suma de las matrices, como lo indica la expresión 6.4 que corresponde al tercer paso de la fase de aprendizaje:

$$\begin{aligned}
 M &= \sum_{\mu=1}^3 \mathbf{x}^{\mu} \cdot (\mathbf{x}^{\mu})^t \\
 &= \begin{pmatrix} 0 & -1 & -1 & 1 \\ -1 & 0 & 1 & -1 \\ -1 & 1 & 0 & -1 \\ 1 & -1 & -1 & 0 \end{pmatrix} + \begin{pmatrix} 0 & -1 & 1 & -1 \\ -1 & 0 & -1 & 1 \\ 1 & -1 & 0 & -1 \\ -1 & 1 & -1 & 0 \end{pmatrix} \\
 &\quad + \begin{pmatrix} 0 & 1 & -1 & 1 \\ 1 & 0 & -1 & 1 \\ -1 & -1 & 0 & -1 \\ 1 & 1 & -1 & 0 \end{pmatrix} \\
 M &= \begin{pmatrix} 0 & -1 & -1 & 1 \\ -1 & 0 & -1 & 1 \\ -1 & -1 & 0 & -3 \\ 1 & 1 & -1 & 0 \end{pmatrix}
 \end{aligned}$$

Con la obtención de la memoria  $M$  concluye la fase de aprendizaje.

Ahora se le presentará el patrón  $\tilde{\mathbf{x}} = \mathbf{x}^3$  a la memoria  $M$  para llevar cabo la fase de recuperación.

El primer paso consiste en realizar la asignación  $\mathbf{x}(0) = \tilde{\mathbf{x}}$ . Entonces se tiene:

$$\mathbf{x}(0) = \begin{pmatrix} -1 \\ -1 \\ 1 \\ -1 \end{pmatrix}$$

Se requiere calcular la cantidad

$$\sum_{j=1}^n m_{ij} x_j(0)$$

$\forall i$  antes de aplicar la condición del paso 2; esto se logra operando el producto de la matriz  $M$  por el vector  $x(0)$ :

$$\begin{aligned} M \cdot x(0) &= \begin{pmatrix} 0 & -1 & -1 & 1 \\ -1 & 0 & -1 & 1 \\ -1 & -1 & 0 & -3 \\ 1 & 1 & -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} -1 \\ -1 \\ 1 \\ -1 \end{pmatrix} \\ &= \begin{pmatrix} 0 \cdot (-1) + (-1) \cdot (-1) + (-1) \cdot (1) + 1 \cdot (-1) \\ (-1) \cdot (-1) + (0) \cdot (-1) + (-1) \cdot (1) + 1 \cdot (-1) \\ (-1) \cdot (-1) + (-1) \cdot (-1) + (0) \cdot (1) + (-3) \cdot (-1) \\ 1 \cdot (-1) + (1) \cdot (-1) + (-1) \cdot (1) + (0) \cdot (-1) \end{pmatrix} \end{aligned}$$

$$M \cdot x(0) = \begin{pmatrix} \sum_{j=1}^3 m_{1j}x_j(0) \\ \sum_{j=1}^3 m_{2j}x_j(0) \\ \sum_{j=1}^3 m_{3j}x_j(0) \\ \sum_{j=1}^3 m_{4j}x_j(0) \end{pmatrix} = \begin{pmatrix} -1 \\ -1 \\ 5 \\ -3 \end{pmatrix}$$

Es decir:  $\sum_{j=1}^3 m_{1j}x_j(0) = -1$ ,  $\sum_{j=1}^3 m_{2j}x_j(0) = -1$ ,  $\sum_{j=1}^3 m_{3j}x_j(0) = 5$  y

$$\sum_{j=1}^3 m_{4j}x_j(0) = -3$$

Ahora apliquemos el paso 2 a cada una de las coordenadas:

$$x_1(1) = -1 \text{ porque } \sum_{j=1}^3 m_{1j}x_j(0) = -1 < 0$$

$$x_2(1) = -1 \text{ porque } \sum_{j=1}^3 m_{2j}x_j(0) = -1 < 0$$

$$x_3(1) = 1 \text{ porque } \sum_{j=1}^3 m_{3j}x_j(0) = 5 > 0$$

$$x_4(1) = -1 \text{ porque } \sum_{j=1}^3 m_{4j}x_j(0) = -3 < 0$$



El paso 3 indica que se realicen las comparaciones de  $x_i(t+1)$  con  $x_i(t) \forall i \in \{1, 2, 3, \dots, n\}$ ; veamos:

$$\begin{aligned}x_1(1) &= -1 = x_1(0) \\x_2(1) &= -1 = x_2(0) \\x_3(1) &= 1 = x_3(0) \\x_4(1) &= -1 = x_4(0)\end{aligned}$$

Lo anterior significa que  $\mathbf{x}(1) = \mathbf{x}(0)$ . Por lo tanto, el patrón recuperado es el mismo  $\mathbf{x}(0) = \tilde{\mathbf{x}} = \mathbf{x}^3$ .

Analicemos lo que sucede en la fase de recuperación al presentarle el patrón  $\tilde{\mathbf{x}} = \mathbf{x}^1$  a la memoria  $M$ . Primer paso:

$$\mathbf{x}(0) = \tilde{\mathbf{x}} = \mathbf{x}^1 = \begin{pmatrix} 1 \\ -1 \\ -1 \\ 1 \end{pmatrix}$$

Igual que antes, previamente a la aplicación de la condición del paso 2 se requiere calcular la cantidad  $\sum_{j=1}^n m_{ij}x_j(0)$ ,  $\forall i$ .

$$\begin{aligned}M \cdot \mathbf{x}(0) &= \begin{pmatrix} 0 & -1 & -1 & 1 \\ -1 & 0 & -1 & 1 \\ -1 & -1 & 0 & -3 \\ 1 & 1 & -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ -1 \\ -1 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 0 \cdot 1 + (-1) \cdot (-1) + (-1) \cdot (-1) + 1 \cdot 1 \\ (-1) \cdot (1) + (0) \cdot (-1) + (-1) \cdot (-1) + 1 \cdot 1 \\ (-1) \cdot (1) + (-1) \cdot (-1) + (0) \cdot (-1) + (-3) \cdot (1) \\ 1 \cdot 1 + (1) \cdot (-1) + (-1) \cdot (-1) + (0) \cdot (1) \end{pmatrix}\end{aligned}$$

$$\mathbf{M} \cdot \mathbf{x}(0) = \begin{pmatrix} \sum_{j=1}^3 m_{1j}x_j(0) \\ \sum_{j=1}^3 m_{2j}x_j(0) \\ \sum_{j=1}^3 m_{3j}x_j(0) \\ \sum_{j=1}^3 m_{4j}x_j(0) \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \\ -3 \\ 1 \end{pmatrix}$$

Paso 2 en cada una de las coordenadas:

$$x_1(1) = 1 \text{ porque } \sum_{j=1}^3 m_{1j}x_j(0) = 3 > 0$$

$$x_2(1) = 1 \text{ porque } \sum_{j=1}^3 m_{2j}x_j(0) = 1 > 0$$

$$x_3(1) = -1 \text{ porque } \sum_{j=1}^3 m_{3j}x_j(0) = -3 < 0$$

$$x_4(1) = 1 \text{ porque } \sum_{j=1}^3 m_{4j}x_j(0) = 1 > 0$$

El paso 3 indica que se realicen las comparaciones de  $x_i(t+1)$  con  $x_i(t) \forall i \in \{1, 2, 3, \dots, n\}$ ; veamos:

$$x_1(1) = 1 = x_1(0)$$

$$x_2(1) = 1 \neq x_2(0)$$

$$x_3(1) = -1 = x_3(0)$$

$$x_4(1) = 1 = x_4(0)$$

Para la coordenada 2 no se cumple la igualdad, y esto significa que  $\mathbf{x}(1) \neq \mathbf{x}(0)$  y que  $\tau \neq 0$ : se deben iterar los pasos 2 y 3.

Producto de la matriz  $\mathbf{M}$  por el vector  $\mathbf{x}(1)$ :

$$\begin{aligned}
\mathbf{M} \cdot \mathbf{x}(1) &= \begin{pmatrix} 0 & -1 & -1 & 1 \\ -1 & 0 & -1 & 1 \\ -1 & -1 & 0 & -3 \\ 1 & 1 & -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ -1 \\ 1 \end{pmatrix} \\
&= \begin{pmatrix} 0 \cdot 1 + (-1) \cdot (1) + (-1) \cdot (-1) + 1 \cdot 1 \\ (-1) \cdot (1) + (0) \cdot (1) + (-1) \cdot (-1) + 1 \cdot 1 \\ (-1) \cdot (1) + (-1) \cdot (1) + (0) \cdot (-1) + (-3) \cdot (1) \\ 1 \cdot 1 + (1) \cdot (1) + (-1) \cdot (-1) + (0) \cdot (1) \end{pmatrix} \\
\mathbf{M} \cdot \mathbf{x}(1) &= \begin{pmatrix} \sum_{j=1}^3 m_{1j}x_j(1) \\ \sum_{j=1}^3 m_{2j}x_j(1) \\ \sum_{j=1}^3 m_{3j}x_j(1) \\ \sum_{j=1}^3 m_{4j}x_j(1) \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ -5 \\ 1 \end{pmatrix}
\end{aligned}$$

Al aplicar el paso 2 a cada una de las coordenadas, tenemos:

$$x_1(2) = 1 \text{ porque } \sum_{j=1}^3 m_{1j}x_j(1) = 1 > 0$$

$$x_2(2) = 1 \text{ porque } \sum_{j=1}^3 m_{2j}x_j(1) = 1 > 0$$

$$x_3(2) = -1 \text{ porque } \sum_{j=1}^3 m_{3j}x_j(1) = -5 < 0$$

$$x_4(2) = 1 \text{ porque } \sum_{j=1}^3 m_{4j}x_j(1) = 1 > 0$$

Apliquemos el paso 3  $\forall i \in \{1, 2, 3, \dots, n\}$ :

$$x_1(2) = 1 = x_1(1)$$

$$x_2(2) = 1 = x_2(1)$$

$$x_3(2) = -1 = x_3(1)$$

$$x_4(2) = 1 = x_4(1)$$

Lo anterior significa que  $\mathbf{x}(2) = \mathbf{x}(1)$ . Por lo tanto, el patrón recuperado es:

$$\mathbf{x}(1) = \begin{pmatrix} 1 \\ 1 \\ -1 \\ 1 \end{pmatrix} \neq \tilde{\mathbf{x}}$$

### 6.3. Consideraciones finales del capítulo.

En este capítulo hemos presentado uno de los trabajos clásicos en memorias asociativas: La Memoria Asociativa Hopfield, cuyo autor es el físico norteamericano John Hopfield quien es responsable, según voces autorizadas, de haber revivido el interés, a principios de los ochenta del siglo XX, en las redes neuronales y en particular en las memorias asociativas, después de un período de estancamiento de más de una década.

Dos años después de haber publicado su famoso artículo (Hopfield, 1982), el autor publica un nuevo artículo donde describe las memorias asociativas con valores reales (Hopfield, 1984). Para entonces, ya se había desatado un furor por las memorias asociativas y las redes neuronales entre los científicos e ingenieros involucrados con este tipo de temas.

En la actualidad, el Profr. Hopfield dirige exitosamente el Hopfield Group en Princeton University ([www.hopfield.net/~group/](http://www.hopfield.net/~group/)). Entre los interesantes temas que actualmente investiga el grupo, se encuentra la implementación neuronal de la percepción de olores (Hopfield, J. J., Odor Space and Olfactory Processing: Collective Algorithms and Neural Implementation, PNAS (1999)).



## Capítulo 7

# ADAM, BAM y SDM

Este capítulo tiene como propósito incluir tres modelos de memoria asociativa que no son esenciales para entender el desarrollo de los modelos de memorias asociativas, pero que comparten una característica común: son extensiones de modelos conocidos.

El ADAM (Advanced Distributed Associative Memory) de Austin se puede considerar como una extensión de las redes asociativas de Willshaw; la BAM (Bidirectional Associative Memory) de Kosko, que surgió en pleno auge de los intentos realizados por algunos grupos de investigación con miras a minimizar una de las desventajas de la memoria Hopfield, se puede considerar como una extensión ingeniosa de ésta última; y finalmente, la SDM (Sparse Distributed Memory) de Kanerva está basada en la estructura de una RAM usual, y puede considerarse como una generalización de los conceptos involucrados en una RAM

### 7.1. Advanced Distributed Associative Memory

El modelo de memoria asociativa ADAM (*Advanced Distributed Associative Memory*) fue propuesto por Jim Austin de la *University of York*, como tema de tesis doctoral, en 1986. Austin creó ADAM con miras a mejorar la capacidad de las *redes asociativas* de Willshaw *et. al.*; para ello, diseñó un arreglo de dos *redes asociativas* a las que agregó un sistema de *códigos de clase*, como se describirá a continuación.

El desarrollo de esta parte está basado en la libre adaptación de la información contenida en las referencias (Austin, 1987). (Kennedy, Austin,

& Cass. 1995), (Austin, Buckle, Kennedy, Moulds, Pack. & Turner, 1997) (Turner. & Austin. 1997).

Para describir el modelo, consideremos  $A = \{0, 1\}$  y el conjunto fundamental

$$\{(x^\mu, y^\mu) \mid x^\mu \in A^n \text{ y } y^\mu \in A^m, \text{ donde } \mu = 1, 2, \dots, p\}$$

ADAM es un sistema de entrada y salida que al operar acepta como entrada un patrón binario  $x^\mu \in A^n$ , y produce como salida un patrón  $y^\mu \in A^m$ .

En ambas fases, aprendizaje y recuperación, para cada pareja de patrones de entrada y salida se requiere un código de clase, el cual es un patrón  $z^\mu \in A^k$ , donde  $k$  es un número entero positivo; cada código de clase contiene  $r$  componentes con valor 1 y  $k - r$  componentes con valor 0, donde  $r$  es un número entero positivo fijo con la condición  $r < k$  (esto significa que habrá  $\frac{k!}{r!(k-r)!}$  diferentes códigos de clase disponibles).

Antes de iniciar las fases de aprendizaje y recuperación, es preciso llevar a cabo dos pasos previos:

1. Se escoge el valor  $r < k$  para crear los códigos de clase
2. Se crean dos matrices nulas (llenas de valores 0):  $\mathbf{P} = [p_{ij}]_{k \times n}$  y  $\mathbf{Q} = [q_{ij}]_{m \times k}$

En la fase de aprendizaje de ADAM, para cada  $\mu = 1, 2, \dots, p$  se realiza lo siguiente:

1. Se propone un código de clase  $z^\mu$  (el cual contiene  $r$  componentes con valor 1)
2. Se actualiza la matriz  $\mathbf{P}$  de acuerdo con el siguiente esquema:

|           |           |           |         |           |         |           |
|-----------|-----------|-----------|---------|-----------|---------|-----------|
|           | $x_1^\mu$ | $x_2^\mu$ | $\dots$ | $x_j^\mu$ | $\dots$ | $x_n^\mu$ |
| $z_1^\mu$ | $p_{11}$  | $p_{12}$  | $\dots$ | $p_{1j}$  | $\dots$ | $p_{1n}$  |
| $z_2^\mu$ | $p_{21}$  | $p_{22}$  | $\dots$ | $p_{2j}$  | $\dots$ | $p_{2n}$  |
| $\vdots$  | $\vdots$  | $\vdots$  |         | $\vdots$  |         | $\vdots$  |
| $z_i^\mu$ | $p_{i1}$  | $p_{i2}$  | $\dots$ | $p_{ij}$  | $\dots$ | $p_{in}$  |
| $\vdots$  | $\vdots$  | $\vdots$  |         | $\vdots$  |         | $\vdots$  |
| $z_k^\mu$ | $p_{k1}$  | $p_{k2}$  | $\dots$ | $p_{kj}$  | $\dots$ | $p_{kn}$  |

donde la regla para actualizar los componentes  $p_{ij}$  es:

$$p_{ij} = \begin{cases} 1 & \text{si } z_i^\mu = 1 = x_j^\mu \\ \text{valor anterior} & \text{en otro caso} \end{cases}$$

3. Se actualiza la matriz  $Q$  de acuerdo con el siguiente esquema:

|           |           |           |         |           |         |           |
|-----------|-----------|-----------|---------|-----------|---------|-----------|
|           | $z_1^\mu$ | $z_2^\mu$ | $\dots$ | $z_j^\mu$ | $\dots$ | $z_k^\mu$ |
| $y_1^\mu$ | $q_{11}$  | $q_{12}$  | $\dots$ | $q_{1j}$  | $\dots$ | $q_{1k}$  |
| $y_2^\mu$ | $q_{21}$  | $q_{22}$  | $\dots$ | $q_{2j}$  | $\dots$ | $q_{2k}$  |
| $\vdots$  | $\vdots$  | $\vdots$  |         | $\vdots$  |         | $\vdots$  |
| $y_i^\mu$ | $q_{i1}$  | $q_{i2}$  | $\dots$ | $q_{ij}$  | $\dots$ | $q_{ik}$  |
| $\vdots$  | $\vdots$  | $\vdots$  |         | $\vdots$  |         | $\vdots$  |
| $y_m^\mu$ | $q_{m1}$  | $q_{m2}$  | $\dots$ | $q_{mj}$  | $\dots$ | $q_{mk}$  |

donde la regla para actualizar los componentes  $q_{ij}$  es:

$$q_{ij} = \begin{cases} 1 & \text{si } y_i^\mu = 1 = z_j^\mu \\ \text{valor anterior} & \text{en otro caso} \end{cases}$$

Al final de la fase de aprendizaje se tendrán las dos redes asociativas de Willshaw  $P$  y  $Q$ .

En la fase de recuperación se considera un patrón de entrada  $\tilde{x}^\omega \in A^n$ , donde  $\omega \in \{1, 2, \dots, p\}$  (si el ruido es cero, se cumple que  $\tilde{x}^\omega = x^\omega$  pertenece al conjunto fundamental de entrada) y se llevan a cabo las siguientes acciones:



1. Se realiza la operación  $\mathbf{P} \cdot \tilde{\mathbf{x}}^\omega$ ; es decir,

$$(\mathbf{P} \cdot \tilde{\mathbf{x}}^\omega)_i = \sum_{j=1}^n p_{ij} \tilde{x}_j^\omega, \forall i \in \{1, 2, \dots, k\}$$

2. Se calcula el correspondiente código de clase  $\mathbf{z}^\omega$  a través de la umbralización:

$$z_i^\omega = \begin{cases} 1 & \text{si } \sum_{j=1}^n p_{ij} \tilde{x}_j^\omega = \bigvee_{h=1}^k \left[ \sum_{j=1}^n p_{hj} \tilde{x}_j^\omega \right] \\ 0 & \text{en otro caso} \end{cases}$$

3. Ya conocido el código de clase  $\mathbf{z}^\omega$ , se realiza la operación  $\mathbf{Q} \cdot \mathbf{z}^\omega$ , decir,

$$(\mathbf{Q} \cdot \mathbf{z}^\omega)_i = \sum_{j=1}^k q_{ij} z_j, \forall i \in \{1, 2, \dots, m\}$$

Se espera que el vector  $\mathbf{Q} \cdot \mathbf{z}^\omega$  sea precisamente el patrón fundamental de salida  $\mathbf{y}^\omega$ .

## 7.2. Bidirectional Associative Memory

Pasada la euforia que despertó la aparición de la memoria Hopfield los albores de la década de los ochenta, y como parte de las productivas consecuencias que vivió el campo de las memorias asociativas, los investigadores hurgaron en la posibilidad de buscar maneras de subsanar la clara desventaja de la autoasociatividad; es decir, se inició la búsqueda de modelos heteroasociativos, que dieron lugar a la creación de memorias que aceptasen conjuntos fundamentales como el siguiente:

$$\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mathbf{x}^\mu \in A^n \text{ y } \mathbf{y}^\mu \in A^m, \text{ donde } \mu = 1, 2, \dots, p\}$$

Bart Kosko, investigador de la *University of Southern California*, propuso una respuesta: la hoy famosa BAM (*Bidirectional Associative Memory*)

Similarmente a como Austin ensambló dos redes asociativas de Willshaw para diseñar su ADAM, Kosko ideó un ingenioso arreglo de dos memorias

Hopfield, y demostró que este diseño, al que llamó BAM, es capaz de asociar patrones de manera heteroasociativa.

A pesar de que en el bagaje editorial y en internet pululan ensayos, escritos, monografías y opiniones sobre la BAM de Kosko, el contenido de esta sección se basa en sólo tres fuentes: el artículo original (Kosko, 1988), un libro de texto (Kosko, 1992) y una excelente obra colectiva sobre el tema de las memorias asociativas (Hassoun, 1993).

Al igual que Hopfield descubrió que su memoria asociativa funcionaba mejor con patrones bipolares que con patrones binarios, Kosko mostró que la BAM se comporta mejor con patrones bipolares que como lo hace con patrones binarios, y por ello diseñó la BAM para que aceptase el conjunto fundamental

$$\{(x^\mu, y^\mu) \mid x^\mu \in A^n \text{ y } y^\mu \in A^m, \text{ donde } \mu = 1, 2, \dots, p\}, \text{ donde } A = \{-1, 1\}$$

Es decir, la BAM es un sistema de entrada y salida que al operar acepta como entrada un patrón bipolar  $x^\mu \in A^n$ , y produce como salida un patrón  $y^\mu \in A^m$ .

La fase de aprendizaje para la BAM genera dos matrices diferentes, M y W, pero en cada caso es similar a la fase de aprendizaje de la memoria Hopfield y del *Linear Associator*, recordando que en la memoria Hopfield hay una diferencia relacionada con la diagonal principal (en ceros); la BAM no presenta esta restricción.

1. Para cada una de las  $p$  asociaciones  $(x^\mu, y^\mu)$  se encuentra la matriz  $y^\mu \cdot (x^\mu)^t$  de dimensiones  $m \times n$ :

$$y^\mu \cdot (x^\mu)^t = \begin{pmatrix} y_1^\mu \\ y_2^\mu \\ \vdots \\ y_n^\mu \end{pmatrix} \cdot (x_1^\mu, x_2^\mu, \dots, x_n^\mu) = \begin{pmatrix} y_1^\mu x_1^\mu & y_1^\mu x_2^\mu & \dots & y_1^\mu x_i^\mu & \dots & y_1^\mu x_n^\mu \\ y_2^\mu x_1^\mu & y_2^\mu x_2^\mu & \dots & y_2^\mu x_i^\mu & \dots & y_2^\mu x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ y_i^\mu x_1^\mu & y_i^\mu x_2^\mu & \dots & y_i^\mu x_i^\mu & \dots & y_i^\mu x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ y_n^\mu x_1^\mu & y_n^\mu x_2^\mu & \dots & y_n^\mu x_i^\mu & \dots & y_n^\mu x_n^\mu \end{pmatrix}$$

2. Se suman la  $p$  matrices  $y^\mu \cdot (x^\mu)^t$  para obtener la matriz  $M$  :

$$M = \sum_{\mu=1}^p y^\mu \cdot (x^\mu)^t = [m_{ij}]_{m \times n}$$

3. La matriz  $W$  de  $n \times m$  se define como la transpuesta de  $M$ ; es decir:

$$\begin{aligned} W &= (M)^t \\ &= \left[ \sum_{\mu=1}^p y^\mu \cdot (x^\mu)^t \right]^t \\ &= \sum_{\mu=1}^p [y^\mu \cdot (x^\mu)^t]^t \\ &= \sum_{\mu=1}^p (x^\mu) \cdot (y^\mu)^t \\ &= [w_{ij}]_{n \times m} \end{aligned}$$

El paso 3 es la aportación de Kosko: la ingeniosa idea de que la matriz  $W$  sea la transpuesta de  $M$ .

La fase de recuperación de la BAM, para cada uno de los dos tipos de patrones  $x^\mu, y^\mu$  es idéntica a la fase de recuperación de la memoria Hopfield. En la BAM se requiere verificar convergencia en ambos casos, para  $x^\mu$  y para  $y^\mu$ .

Representemos el estado de la BAM en el tiempo  $t$  por  $[x(t), y(t)]$ ; entonces  $[x_i(t), y_i(t)]$  representa los valores de  $x_i$  y  $y_i$  en el tiempo  $t$ , y  $[x_i(t+1), y_i(t+1)]$  los valores en el tiempo siguiente ( $t+1$ ).

Dado un par de vectores columna de entrada  $(\tilde{x}, \tilde{y})$ , la fase de recuperación consta de cuatro pasos:

1. Para  $t = 0$ , se hace  $x(t) = \tilde{x}$  y  $y(t) = \tilde{y}$ ; es decir,  $x_i(0) = \tilde{x}_i, \forall i \in \{1, 2, 3, \dots, n\}$  y  $y_i(0) = \tilde{y}_i, \forall i \in \{1, 2, 3, \dots, m\}$

2.  $\forall i \in \{1, 2, 3, \dots, m\}$  se calcula  $y_i(t+1)$  de acuerdo con la condición siguiente:

$$y_i(t+1) = \begin{cases} 1 & \text{si } \sum_{j=1}^n m_{ij}x_j(t) > 0 \\ y_i(t) & \text{si } \sum_{j=1}^n m_{ij}x_j(t) = 0 \\ -1 & \text{si } \sum_{j=1}^n m_{ij}x_j(t) < 0 \end{cases}$$

3.  $\forall i \in \{1, 2, 3, \dots, n\}$  se calcula  $x_i(t+1)$  de acuerdo con la condición siguiente:

$$x_i(t+1) = \begin{cases} 1 & \text{si } \sum_{j=1}^m w_{ij}y_j(t) > 0 \\ x_i(t) & \text{si } \sum_{j=1}^m w_{ij}y_j(t) = 0 \\ -1 & \text{si } \sum_{j=1}^m w_{ij}y_j(t) < 0 \end{cases}$$

4. Se realizan las siguientes comparaciones:  $x_i(t+1)$  con  $x_i(t) \forall i \in \{1, 2, 3, \dots, n\}$  y  $y_i(t+1)$  con  $y_i(t) \forall i \in \{1, 2, 3, \dots, m\}$ . Si se cumplen ambas condiciones  $\mathbf{x}(t+1) = \mathbf{x}(t)$  y  $\mathbf{y}(t+1) = \mathbf{y}(t)$ , el proceso termina y la pareja de vectores recuperados es  $[\mathbf{x}(0), \mathbf{y}(0)] = (\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ . De otro modo, el proceso continúa de la siguiente manera: los pasos 2, 3 y 4 se iteran tantas veces como sea necesario hasta llegar a un valor  $t = \tau$  para el cual se cumplan las dos condiciones:  $x_i(\tau+1) = x_i(\tau) \forall i \in \{1, 2, 3, \dots, n\}$  y  $y_i(\tau+1) = y_i(\tau) \forall i \in \{1, 2, 3, \dots, m\}$ ; el proceso termina y la pareja de patrones recuperados es  $[\mathbf{x}(\tau), \mathbf{y}(\tau)]$ .

Se espera que la pareja recuperada  $[\mathbf{x}(\tau), \mathbf{y}(\tau)]$  sea una de las parejas  $(\mathbf{x}^\mu, \mathbf{y}^\mu)$  del conjunto fundamental.

### 7.3. Sparse Distributed Memory

En su búsqueda de un modelo para la memoria humana, el finlandés Pentti Kanerva desarrolló un tipo de memoria asociativa a la que llamó *Sparse Distributed Memory* (SDM).

La SDM puede considerarse como una variante de la RAM usual, como quedará de manifiesto en los siguientes párrafos, en donde se ha adaptado notación e información contenidas en diversas fuentes (Kanerva, 1988; Flynn, Kanerva & Bhadkamkar, 1989; Simpson, 1990; Zboril, 1997; Ludermir Carvalho, Braga, & Souto, 1999).

Para describir la SDM, consideremos  $A = \{0, 1\}$  y el conjunto fundamental

$$\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mathbf{x}^\mu \in A^n \text{ y } \mathbf{y}^\mu \in A^m, \text{ donde } \mu = 1, 2, \dots, p\}$$

Esta memoria requiere de dos pasos previos a las fases de aprendizaje y recuperación:

1. Se escoge  $k \in \mathbb{N}$ , tal que  $n \ll k < 2^n$
2. Se crea la matriz aleatoria  $\mathbf{Q}_{k \times n}$  cuyas entradas  $q_{ij}$  son elementos de  $A$  escogidos aleatoriamente

La fase de aprendizaje consiste de dos etapas:

1. En esta etapa primera, para cada una de las  $p$  asociaciones  $(\mathbf{x}^\mu, \mathbf{y}^\mu)$  llevan a cabo los siguientes cuatro pasos:
  - a) Se obtiene el vector de Hamming, el cual es un vector columna  $\mathbf{h}^\mu$  de dimensión  $k$  que resulta de  $\mathbf{Q}_{k \times n} \otimes \mathbf{x}^\mu = \mathbf{h}^\mu$ ; la operación  $\otimes$  está definida de modo que el valor de la  $i$ -ésima componente

$$h_i^\mu = H(\mathbf{Q}_i, \mathbf{x}^\mu) = \sum_{j=1}^n q_{ij} \oplus x_j^\mu$$

donde  $H(\mathbf{Q}_i, \mathbf{x}^\mu)$  es la distancia Hamming entre el renglón  $\mathbf{Q}_i$  de la matriz  $\mathbf{Q}_{k \times n}$  y el vector columna de entrada  $\mathbf{x}^\mu$ . La distancia Hamming se calcula sumando todos los ítems para los cuales la operación OR exclusiva  $\oplus$  tiene valor lógico 1. Los valores almacenados en las entradas del vector  $\mathbf{h}^\mu$  son números enteros entre 1 y  $n$ ; es decir,  $1 \leq h_i^\mu \leq n, \forall i \forall \mu$

- b) Se escoge el radio de Kanerva  $\rho$ , el cual es un número entero tal que  $1 \leq \rho \leq \frac{n}{2}$

- c) A partir del vector de Hamming  $\mathbf{h}^\mu$  y el radio de Kanerva  $\rho$ , se obtiene el vector columna de transición interna  $\mathbf{b}^\mu$ , de dimensión  $k$  y con entradas en el conjunto  $A$ , cuya  $i$ -ésima componente se calcula por la expresión

$$b_i^\mu = \begin{cases} 1 & \text{si } h_i^\mu \leq n - 2\rho \\ 0 & \text{de otro modo} \end{cases}$$

- d) Se operan los vectores  $\mathbf{y}^\mu$  y  $(\mathbf{b}^\mu)^t$  para obtener una matriz  $\mathbf{M}^\mu$  de dimensiones  $m \times k$ , de modo que ocurra lo siguiente: el valor de la  $ij$ -ésima entrada  $m_{ij}^\mu$  debe ser 1 si se cumple que  $b_j^\mu = 1$  y  $y_i^\mu = 1$ , y debe ser  $-1$  si se cumple que  $b_j^\mu = 1$  y  $y_i^\mu = 0$ ; además,  $m_{ij}^\mu$  debe ser 0 en otro caso. La expresión con que se logran obtener estos valores para la  $ij$ -ésima entrada es:

$$m_{ij}^\mu = (2y_i^\mu b_j^\mu - 1) b_j^\mu$$

2. En esta segunda etapa se suman la  $p$  matrices  $\mathbf{M}^\mu$  para obtener la memoria

$$\mathbf{M} = \sum_{\mu=1}^p \mathbf{M}^\mu = [m_{ij}]_{k \times m}$$

en donde la  $ij$ -ésima componente  $m_{ij}$  se expresa así:

$$m_{ij} = \sum_{\mu=1}^p (2y_i^\mu b_j^\mu - 1) b_j^\mu$$

En la **fase de recuperación**, que también consta de dos etapas, se considera un patrón de entrada  $\mathbf{x}^\omega$ , donde  $\omega \in \{1, 2, \dots, p\}$  (si el ruido es cero, se cumple que  $\tilde{\mathbf{x}}^\omega = \mathbf{x}^\omega$  pertenece al conjunto fundamental de entrada).

1. Esta primera etapa es en parte similar a la etapa 1 de la fase de aprendizaje y consiste de dos pasos, dado el vector columna de entrada  $\mathbf{x}^\omega$ :

- a) Se realiza la operación  $\mathbf{Q}_{k \times n} \otimes \tilde{\mathbf{x}}^\omega = \mathbf{h}^\omega$  para obtener el vector de Hamming  $\mathbf{h}^\omega$ , de manera que:

$$h_i^\omega = H(\mathbf{Q}_i \cdot \tilde{\mathbf{x}}^\omega) = \sum_{j=1}^n q_{ij} \oplus \tilde{x}_j^\omega$$

donde  $H(Q_i, \tilde{x}^\omega)$  es la distancia Hamming entre el renglón  $Q_i$  de la matriz  $Q_{k \times n}$  y el vector  $\tilde{x}^\omega$ . La distancia Hamming se calcula sumando todos los items para los cuales la operación OR exclusiva  $\oplus$  tiene valor lógico 1. Los valores almacenados en las entradas del vector  $h^\omega$  son números enteros entre 1 y  $n$ ; es decir,  $1 \leq h_i^\omega \leq n$ ,  $\forall i$

- b) A partir del vector de Hamming  $h^\omega$  y del radio de Kanerva  $\rho$  escogido en la fase de aprendizaje, se obtiene el primer vector columna de transición interna  $b^\omega$ , de dimensión  $k$  y con entradas en el conjunto  $A$ , cuya  $i$ -ésima componente se calcula por la expresión  $b_i^\omega = \phi(h^\omega, \rho)$ , donde la función umbral  $\phi$  de dos parámetros  $u$  y  $v$  está definida por:

$$\phi(u, v) = \begin{cases} 1 & \text{si } u \geq n - 2v \\ 0 & \text{de otro modo} \end{cases}$$

2. Esta segunda etapa consta de tres pasos, a saber:

- a) Se calculan los  $m$  umbrales que se representan en el vector umbral  $\theta^\omega$  de dimensión  $m$ , cuya  $i$ -ésima componente se expresa de la siguiente manera:

$$\theta_i^\omega = \frac{1}{2} \sum_{j=1}^k m_{ij}$$

- b) Se obtiene el segundo vector de transición interna  $t^\omega$  de dimensión  $m$  cuya  $i$ -ésima componente se calcula así:

$$t_i^\omega = \sum_{j=1}^k m_{ij} b_j^\omega$$

- c) Finalmente se obtiene el vector columna  $\tilde{y}^\omega$  de dimensión  $m$ , que es la salida correspondiente al vector de entrada  $\tilde{x}^\omega$ . Para calcular el valor de la  $i$ -ésima componente  $\tilde{y}_i^\omega$  se hace uso del vector umbral  $\theta^\omega$  y del segundo vector de transición  $t^\omega$ , por medio de la siguiente expresión:

$$\tilde{y}_i^\omega = \begin{cases} 1 & \text{si } t_i^\omega \geq \theta_i^\omega \\ 0 & \text{de otro modo} \end{cases}$$

Se espera que el vector  $\tilde{y}^w$  sea precisamente el patrón fundamental de salida  $y^w$ .

#### 7.4. Consideraciones finales del capítulo.

Actualmente, Austin es Profesor de la Universidad de York y Director del Advanced Computer Architecture Group: ha trabajado con un grupo de colegas y alumnos en un superconjunto de ADAM, el cual ha sido bautizado con el acrónimo de AURA (Advanced Uncertain Reasoning Architecture).

Los avances del grupo de Austin se pueden consultar en

*[www.cs.york.ac.uk/arch/nn/aura.html](http://www.cs.york.ac.uk/arch/nn/aura.html)*

Respecto de la BAM, en el año de 1998, Hongchi Shi propuso un modelo general de memorias asociativas bidireccionales en su artículo A General Model for Bidirectional Associative Memories, *TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS*, AUGUST 1998.

Actualmente Kanerva es el principal investigador del Stochastic Pattern Computing Group en el Swedish Institute of Computer Science, y su trabajo se enfoca al tema representación distribuida:

*[www.sics.se](http://www.sics.se)*





## Capítulo 8

# Memorias Asociativas Morfológicas

### 8.1. Contexto.

Los trabajos de Hopfield causaron excitación en el mundo de las memorias asociativas y las redes neuronales, de modo tal que muchos de los científicos que habían estado al margen, gracias a Hopfield se interesaron en estos temas y se generó una actividad importante. En la segunda mitad de la década de los ochenta del siglo XX aparecieron investigadores que tomaron los modelos clásicos, los modificaron o extendieron y dieron lugar a nuevos tipos de memorias asociativas, cuya importancia radica en que son consecuencia del trabajo de los grandes hombres involucrados en la concepción y desarrollo de los modelos hasta ahora descritos en esta obra.

Entre la miríada de aportaciones e innovaciones en el campo de las memorias asociativas, después del espectacular éxito de la memoria Hopfield de 1982, no sucedió nada realmente trascendente hasta 1998, cuando aparecieron las memorias asociativas morfológicas.

La diferencia fundamental entre estas memorias y las memorias asociativas clásicas como el *Linear Associator* y la Memoria Hopfield, es que mientras éstas basan su operación en la suma y multiplicación usuales, las memorias morfológicas se basan en las operaciones morfológicas de dilatación y erosión. En otras palabras, mientras que las memorias asociativas clásicas echan mano de la suma de productos, las memorias asociativas morfológicas hacen uso de máximos o mínimos de sumas.

## 8.2. Memorias Morfológicas Heteroasociativas.

Los modelos de memorias asociativas hasta aquí presentados en este libro funcionan con base en las operaciones usuales de suma y multiplicación. A este tipo de memorias asociativas, representadas principalmente por el *Linear Associator* y la memoria Hopfield, se les llama *clásicas* para contrastar su funcionamiento con las *memorias asociativas morfológicas*.

La diferencia fundamental entre las memorias asociativas clásicas y las morfológicas, radica en que los fundamentos operacionales de las *memorias asociativas morfológicas*, se encuentran en las operaciones morfológicas de dilatación y erosión. Estas memorias rompieron el esquema utilizado a través de los años en los modelos de memorias asociativas clásicas, que utilizan operaciones convencionales entre vectores y matrices para la fase de aprendizaje, y *suma de productos* para recuperar patrones.

Las memorias asociativas morfológicas, por otro lado, usan *máximos o mínimos de sumas* para la fase de aprendizaje, y *máximos o mínimos de sumas* para la fase de recuperación (Ritter, Sussner & Diaz-de-Leon, 1998; Ritter, Diaz-de-Leon & Sussner, 1999).

La información que se expone a continuación constituye un resumen ilustrativo del diseño, operación y características de las memorias asociativas morfológicas y está tomada libremente del contenido de estas dos referencias (las cuales contienen, en total, 17 teoremas y cuatro corolarios) y de (Díaz-de-León & Yáñez, 1999).

Hay dos tipos de memorias asociativas morfológicas: las memorias *max*, simbolizadas con *M*, y las memorias *min*, las cuales se simbolizan con *W*: en cada uno de los dos tipos, las memorias pueden funcionar en ambos modos *heteroasociativo* y *autoasociativo*. Nótese, de entrada, esta ventaja que poseen las memorias asociativas morfológicas respecto de la memoria Hopfield, la cual se diseñó para funcionar sólo como memoria autoasociativa.

El conjunto fundamental será  $\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mu = 1, 2, \dots, p\}$  con

$$\mathbf{x}^\mu = \begin{pmatrix} x_1^\mu \\ x_2^\mu \\ \vdots \\ x_n^\mu \end{pmatrix} \in A^n \quad \text{y} \quad \mathbf{y}^\mu = \begin{pmatrix} y_1^\mu \\ y_2^\mu \\ \vdots \\ y_m^\mu \end{pmatrix} \in A^m$$

Antes de iniciar, es necesario definir dos nuevas operaciones entre matrices en términos de las operaciones  $+$ ,  $\vee$  y  $\wedge$ , con el fin de poder expresar las fases de aprendizaje y recuperación de las memorias asociativas morfológicas.

Sea  $D$  una matriz  $[d_{ij}]_{m \times p}$  y  $H$  una matriz  $[h_{ij}]_{p \times n}$  cuyos términos son números enteros.

El *producto máximo* de  $D$  y  $H$ , denotado por  $C = D \nabla H$ , da como resultado una matriz  $C = [c_{ij}]_{m \times n}$  y se define de la siguiente manera:

$$c_{ij} = \bigvee_{k=1}^p (d_{ik} + h_{kj}) \quad (8.1)$$

Existen dos casos particulares de suma importancia en la aplicación del *producto máximo*.

El primer caso es cuando  $D$  es una matriz de dimensiones  $m \times n$  y  $H$  es un vector columna de dimensión  $n$ . Al aplicar la expresión 8.1 para calcular el producto máximo  $C = D \nabla H$  cuando  $D$  es una matriz  $[d_{ij}]_{m \times n}$  y  $H$  es un vector columna de dimensión  $n$   $[h_i]_n$  se obtiene un vector columna  $C$  de dimensión  $m$ , cuya  $i$ -ésima componente es:

$$c_i = \bigvee_{j=1}^n (d_{ij} + h_j) \quad (8.2)$$

El segundo caso es cuando se realiza el *producto máximo* entre un vector columna de dimensión  $m$  por un vector fila de dimensión  $n$ ; en ambos vectores se cumple que  $p = 1$ .

Al aplicar la expresión 8.1 para calcular  $C = D \nabla H$  siendo  $D$  un vector columna de dimensión  $m$  y  $H$  un vector fila de dimensión  $n$ , se obtiene una matriz  $C = [c_{ij}]_{m \times n}$ :

$$c_{ij} = \bigvee_{k=1}^1 (d_{ik} + h_{kj}) = (d_{i1} + h_{1j})$$

expresión en la que se puede eliminar el subíndice 1, para quedar finalmente que la componente  $c_{ij}$  de la matriz  $[c_{ij}]_{m \times n}$  que resulta del *producto máximo* entre un vector columna  $D$  de dimensión  $m$  por un vector fila  $H$  de dimensión  $n$  es:

$$c_{ij} = (d_i + h_j) \quad (8.3)$$

Similarmente, el *producto mínimo*  $C = D \Delta H$  se define así:

$$c_{ij} = \bigwedge_{k=1}^p (d_{ik} + h_{kj}) \quad (8.4)$$

Al aplicar la expresión 8.4 para calcular el producto mínimo  $C = D \Delta H$  cuando  $D$  es una matriz  $[d_{ij}]_{m \times n}$  y  $H$  es un vector columna de dimensión  $n$   $[h_i]_n$ , se obtiene un vector columna  $C$  de dimensión  $m$ , cuya  $i$ -ésima componente es:

$$c_i = \bigwedge_{j=1}^n (d_{ij} + h_j) \quad (8.5)$$

Y la componente  $c_{ij}$  del *producto mínimo* entre un vector columna  $D$  de dimensión  $m$  por un vector fila  $H$  de dimensión  $n$  es:

$$c_{ij} = \bigwedge_{k=1}^1 (d_{ik} + h_{kj}) = (d_{i1} + h_{j1}) = (d_i + h_j) \quad (8.6)$$

Las expresiones anteriores contienen a los operadores máximo  $\vee$  y mínimo  $\wedge$ , los cuales están íntimamente ligados con los conceptos de las dos operaciones básicas de la morfología matemática: *dilatación* y *erosión*, respectivamente (Serra, 1992; Haralick, Sternberg, & Zhuang, 1987; Ritter, Sussner & Diaz-de-Leon, 1998); de ahí el nombre de memorias asociativas *morfológicas*.

8.2.1. Memorias heteroasociativas *max*

Las memorias morfológicas heteroasociativas *max* utilizan el *producto mínimo* para ambas fases: aprendizaje y recuperación de patrones. Además, en la fase de aprendizaje juega un papel relevante el operador *máximo*; de ahí el nombre de memorias *max*.

La *fase de aprendizaje* consiste de dos etapas:

1. Aplicando la expresión 8.6, para cada una de las  $p$  asociaciones  $(\mathbf{x}^\mu, \mathbf{y}^\mu)$  se encuentra la matriz  $\mathbf{y}^\mu \Delta (-\mathbf{x}^\mu)^t$  de dimensiones  $m \times n$ , donde  $(-\mathbf{x}^\mu)^t = (-x_1^\mu, -x_2^\mu, \dots, -x_n^\mu)$ :

$$\mathbf{y}^\mu \Delta (-\mathbf{x}^\mu)^t = \begin{pmatrix} y_1^\mu \\ y_2^\mu \\ \vdots \\ y_m^\mu \end{pmatrix} \Delta (-x_1^\mu, -x_2^\mu, \dots, -x_n^\mu)$$

$$\mathbf{y}^\mu \Delta (-\mathbf{x}^\mu)^t = \begin{pmatrix} [y_1^\mu + (-x_1^\mu)] & \dots & [y_1^\mu + (-x_j^\mu)] & \dots & [y_1^\mu + (-x_n^\mu)] \\ [y_2^\mu + (-x_1^\mu)] & \dots & [y_2^\mu + (-x_j^\mu)] & \dots & [y_2^\mu + (-x_n^\mu)] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [y_i^\mu + (-x_1^\mu)] & \dots & [y_i^\mu + (-x_j^\mu)] & \dots & [y_i^\mu + (-x_n^\mu)] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [y_m^\mu + (-x_1^\mu)] & \dots & [y_m^\mu + (-x_j^\mu)] & \dots & [y_m^\mu + (-x_n^\mu)] \end{pmatrix}$$

$$\mathbf{y}^\mu \Delta (-\mathbf{x}^\mu)^t = \begin{pmatrix} y_1^\mu - x_1^\mu & y_1^\mu - x_2^\mu & \dots & y_1^\mu - x_j^\mu & \dots & y_1^\mu - x_n^\mu \\ y_2^\mu - x_1^\mu & y_2^\mu - x_2^\mu & \dots & y_2^\mu - x_j^\mu & \dots & y_2^\mu - x_n^\mu \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ y_i^\mu - x_1^\mu & y_i^\mu - x_2^\mu & \dots & y_i^\mu - x_j^\mu & \dots & y_i^\mu - x_n^\mu \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ y_m^\mu - x_1^\mu & y_m^\mu - x_2^\mu & \dots & y_m^\mu - x_j^\mu & \dots & y_m^\mu - x_n^\mu \end{pmatrix} \quad (8.7)$$

2. Se aplica el operador *máximo*  $\bigvee$  a las  $p$  matrices para obtener la memoria  $M$ .

$$M = \bigvee_{\mu=1}^p [y^\mu \Delta (-x^\mu)^t] = [m_{ij}]_{m \times n} \quad (8.8)$$

de manera que la  $ij$ -ésima componente de la memoria  $M$  se expresa así:

$$m_{ij} = \bigvee_{\mu=1}^p (y_i^\mu - x_j^\mu) \quad (8.9)$$

La *fase de recuperación* consiste en realizar el producto mínimo  $\Delta$  de la memoria  $M$  con un patrón de entrada  $x^\omega$ , donde  $\omega \in \{1, 2, \dots, p\}$ , para obtener un vector columna  $y$  de dimensión  $m$ :

$$y = M \Delta x^\omega \quad (8.10)$$

De acuerdo con la expresión 8.5, la  $i$ -ésima componente del vector  $y$  es:

$$y_i = \bigwedge_{j=1}^n (m_{ij} + x_j^\omega) \quad (8.11)$$

### Ejemplo.-

Este ejemplo es meramente ilustrativo de los tipos de operaciones que se llevan a cabo en la generación y operación de las memorias asociativas morfológicas *max*; no se intenta mostrar, de ningún modo, sus bondades y ventajas, las cuales se tratarán detalladamente en un Informe Técnico de esta misma serie Verde, cuyo título es: "Memorias Asociativas: Clásicas vs. Morfológicas".

Se usarán tres parejas de patrones con  $p = 3$ ,  $n = 5$  y  $m = 3$ .

$$\mathbf{x}^1 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \quad \mathbf{y}^1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}; \quad \mathbf{x}^2 = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \quad \mathbf{y}^2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}; \quad \mathbf{x}^3 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

$$\mathbf{y}^3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Para llevar a cabo el primer paso de la fase de aprendizaje, se encuentra la matriz  $\mathbf{y}^\mu \Delta (-\mathbf{x}^\mu)^t$  de dimensiones  $3 \times 5$  para cada una de las parejas del conjunto fundamental (expresión 8.7).

$$\mathbf{y}^1 \Delta (\mathbf{x}^1)^t = \begin{pmatrix} 1-1 & 1-0 & 1-1 & 1-0 & 1-1 \\ 0-1 & 0-0 & 0-1 & 0-0 & 0-1 \\ 0-1 & 0-0 & 0-1 & 0-0 & 0-1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ -1 & 0 & -1 & 0 & -1 \\ -1 & 0 & -1 & 0 & -1 \end{pmatrix}$$

$$\mathbf{y}^2 \Delta (\mathbf{x}^2)^t = \begin{pmatrix} 0-1 & 0-1 & 0-0 & 0-0 & 0-1 \\ 1-1 & 1-1 & 1-0 & 1-0 & 1-1 \\ 0-1 & 0-1 & 0-0 & 0-0 & 0-1 \end{pmatrix} = \begin{pmatrix} -1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 1 & 1 & 0 \\ -1 & -1 & 0 & 0 & -1 \end{pmatrix}$$

$$\mathbf{y}^3 \Delta (\mathbf{x}^3)^t = \begin{pmatrix} 0-1 & 0-0 & 0-1 & 0-1 & 0-0 \\ 0-1 & 0-0 & 0-1 & 0-1 & 0-0 \\ 1-1 & 1-0 & 1-1 & 1-1 & 1-0 \end{pmatrix} = \begin{pmatrix} -1 & 0 & -1 & -1 & 0 \\ -1 & 0 & -1 & -1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

El segundo paso de la fase de aprendizaje consiste en aplicar el operador máximo a las tres matrices (expresión 8.8).

$$\mathbf{M} = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ -1 & 0 & -1 & 0 & -1 \\ -1 & 0 & -1 & 0 & -1 \end{pmatrix} \vee \begin{pmatrix} -1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 1 & 1 & 0 \\ -1 & -1 & 0 & 0 & -1 \end{pmatrix}$$

$$\vee \begin{pmatrix} -1 & 0 & -1 & -1 & 0 \\ -1 & 0 & -1 & -1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 0 \vee (-1) \vee (-1) & 1 \vee (-1) \vee 0 & 0 \vee 0 \vee (-1) & 1 \vee 0 \vee (-1) & 0 \vee (-1) \vee 0 \\ (-1) \vee 0 \vee (-1) & 0 \vee 0 \vee 0 & (-1) \vee 1 \vee (-1) & 0 \vee 1 \vee (-1) & (-1) \vee 0 \vee 0 \\ (-1) \vee (-1) \vee 0 & 0 \vee (-1) \vee 1 & (-1) \vee 0 \vee 0 & 0 \vee 0 \vee 0 & (-1) \vee (-1) \vee \end{pmatrix}$$

La memoria M queda así:



$$M = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Para la fase de recuperación de patrones, se realiza el producto mínimo entre la memoria  $M$  y cada uno de los tres patrones de entrada del conjunto fundamental (expresiones 8.10 y 8.11).

$$\begin{aligned} y = M \Delta x^1 &= \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix} \Delta \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \wedge_{j=1}^5 (m_{1j} + x_j^1) \\ \wedge_{j=1}^5 (m_{2j} + x_j^1) \\ \wedge_{j=1}^5 (m_{3j} + x_j^1) \end{pmatrix} \\ &= \begin{pmatrix} (0+1) \wedge (1+0) \wedge (0+1) \wedge (1+0) \wedge (0+1) \\ (0+1) \wedge (0+0) \wedge (1+1) \wedge (1+0) \wedge (0+1) \\ (0+1) \wedge (1+0) \wedge (0+1) \wedge (0+0) \wedge (1+1) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = y^1 \end{aligned}$$

$$\begin{aligned} y = M \Delta x^2 &= \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix} \Delta \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \wedge_{j=1}^5 (m_{1j} + x_j^2) \\ \wedge_{j=1}^5 (m_{2j} + x_j^2) \\ \wedge_{j=1}^5 (m_{3j} + x_j^2) \end{pmatrix} \\ &= \begin{pmatrix} (0+1) \wedge (1+1) \wedge (0+0) \wedge (1+0) \wedge (0+1) \\ (0+1) \wedge (0+1) \wedge (1+0) \wedge (1+0) \wedge (0+1) \\ (0+1) \wedge (1+1) \wedge (0+0) \wedge (0+0) \wedge (1+1) \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = y^2 \end{aligned}$$

$$\begin{aligned} y = M \Delta x^3 &= \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix} \Delta \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \wedge_{j=1}^5 (m_{1j} + x_j^3) \\ \wedge_{j=1}^5 (m_{2j} + x_j^3) \\ \wedge_{j=1}^5 (m_{3j} + x_j^3) \end{pmatrix} \\ &= \begin{pmatrix} (0+1) \wedge (1+0) \wedge (0+1) \wedge (1+1) \wedge (0+0) \\ (0+1) \wedge (0+0) \wedge (1+1) \wedge (1+1) \wedge (0+0) \\ (0+1) \wedge (1+0) \wedge (0+1) \wedge (0+1) \wedge (1+0) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = y^3 \end{aligned}$$

■

Como se menciona en las referencias, la pregunta obvia a plantear es: ¿cuáles son las condiciones necesarias y suficientes para asegurar respuesta perfecta en una memoria morfológica  $max$ ? Para ser más específicos, ¿para

qué pares de vectores  $\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mu = 1, 2, \dots, p\}$  podrá la memoria asociativa descrita por las expresiones 8.7, 8.8, 8.9, 8.10 y 8.11 proporcionar una salida perfecta?. La respuesta a esta interrogante se da en el teorema 1 y su correspondiente corolario 1.1 de (Ritter, Diaz-de-Leon & Sussner, 1999):

**Teorema 1:**  $M \Delta \mathbf{x}^\omega = \mathbf{y}^\omega$  para todo  $\omega = 1, \dots, p$  si y sólo si para cada índice de fila  $i = 1, \dots, m$  existen índices de columna  $j_i^\varepsilon \in \{1, \dots, n\}$  tales que  $m_{ij_i^\varepsilon} = y_i^\omega - x_{j_i^\varepsilon}^\omega$  para todo  $\omega = 1, \dots, p$ .

**Corolario 1.1:**  $M \Delta \mathbf{x}^\omega = \mathbf{y}^\omega$  para todo  $\omega = 1, \dots, p$  si y sólo si para cada índice de fila  $i = 1, \dots, m$  y cada  $\gamma \in \{1, \dots, p\}$  existen índices de columna  $j_i^\gamma \in \{1, \dots, n\}$  tales que

$$x_{j_i^\gamma}^\gamma = \bigwedge_{\varepsilon=1}^p (x_{j_i^\varepsilon}^\varepsilon - y_i^\varepsilon) + y_i^\gamma$$

Con estos dos resultados se garantizan las condiciones formales para la recuperación perfecta en el caso de patrones del conjunto fundamental. En la figura 1 se presenta la respuesta perfecta de una memoria morfológica heteroasociativa *max* al presentársele como entradas los patrones superiores, en donde el conjunto fundamental tenía como salidas los patrones inferiores.

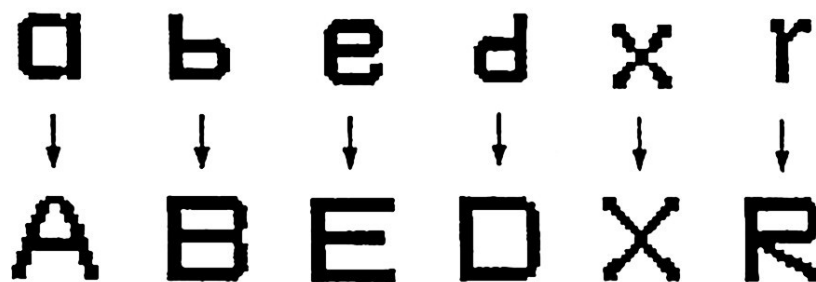


Figura 1

Es importante hacer notar la considerable mejora en la capacidad de almacenamiento de las memorias morfológicas heteroasociativas *max* con respecto a las memorias clásicas, así como el hecho de que la memoria morfológica *no requiere* convergencia; es decir, es capaz de responder en forma perfecta *en un solo paso*.

La segunda pregunta de interés con respecto al desempeño de una memoria asociativa es: ¿qué cantidad, y qué tipo de ruido se puede permitir como

distorsión para los patrones de entrada sin sacrificar la respuesta (salida) perfecta?. Para el caso de las memorias clásicas, es un hecho que aun sin distorsión en la entrada, no es plausible esperar una respuesta perfecta. Para las memorias morfológicas heteroasociativas *max.* en cambio, es permisible la existencia de ruidos sin perder la propiedad de respuesta perfecta. La forma del ruido permisible está dada por el teorema 5 de la misma referencia (Ritter, Diaz-de-Leon & Sussner, 1999):

**Teorema 5:** Para  $\gamma = 1, \dots, p$  sea  $\tilde{x}^\gamma$  una versión distorsionada del patrón  $x^\gamma$ . Entonces  $M \Delta \tilde{x}^\gamma = y^\gamma$  si y sólo si se cumple que:

$$\tilde{x}_j^\gamma \geq x_j^\gamma \wedge \bigvee_{i=1}^m \left( \bigwedge_{\varepsilon \neq \gamma} [y_i^\gamma - y_i^\varepsilon + x_i^\varepsilon] \right) \quad \forall j = 1, \dots, n$$

y para cada índice renglón  $i \in \{1, \dots, m\}$  existe un índice columna  $j_i \in \{1, \dots, n\}$  tal que:

$$\tilde{x}_{j_i}^\gamma = x_{j_i}^\gamma \wedge \left( \bigwedge_{\varepsilon \neq \gamma} [y_i^\gamma - y_i^\varepsilon + x_{j_i}^\varepsilon] \right)$$

La figura 2 ilustra la recuperación perfecta de patrones alterados con 20% de ruido aditivo, por una memoria morfológica heteroasociativa *max.* Como es posible apreciar, estas memorias son bastante robustas al ruido aditivo.

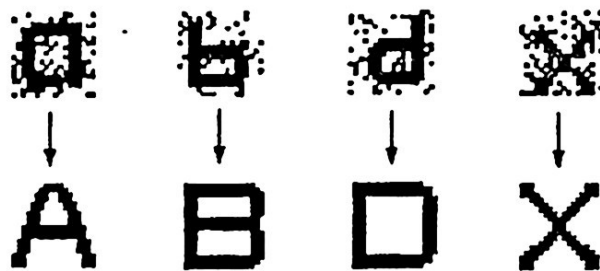


Figura 2

8.2.2. Memorias heteroasociativas *min*

Las memorias morfológicas heteroasociativas *min* utilizan el *producto máximo* para ambas fases: aprendizaje y recuperación de patrones. Además, en la fase de aprendizaje juega un papel relevante el operador *mínimo*; de ahí el nombre de memorias *min*.

La fase de aprendizaje consiste de dos etapas:

1. Aplicando la expresión 8.3, para cada una de las  $p$  asociaciones  $(x^\mu, y^\mu)$  se encuentra la matriz  $y^\mu \nabla (-x^\mu)^t$  de dimensiones  $m \times n$ , donde  $(-x^\mu)^t = (-x_1^\mu, -x_2^\mu, \dots, -x_n^\mu)$ :

$$y^\mu \nabla (-x^\mu)^t = \begin{pmatrix} y_1^\mu \\ y_2^\mu \\ \vdots \\ y_m^\mu \end{pmatrix} \nabla (-x_1^\mu, -x_2^\mu, \dots, -x_n^\mu)$$

$$y^\mu \nabla (-x^\mu)^t = \begin{pmatrix} [y_1^\mu + (-x_1^\mu)] & \dots & [y_1^\mu + (-x_j^\mu)] & \dots & [y_1^\mu + (-x_n^\mu)] \\ [y_2^\mu + (-x_1^\mu)] & \dots & [y_2^\mu + (-x_j^\mu)] & \dots & [y_2^\mu + (-x_n^\mu)] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [y_i^\mu + (-x_1^\mu)] & \dots & [y_i^\mu + (-x_j^\mu)] & \dots & [y_i^\mu + (-x_n^\mu)] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [y_m^\mu + (-x_1^\mu)] & \dots & [y_m^\mu + (-x_j^\mu)] & \dots & [y_m^\mu + (-x_n^\mu)] \end{pmatrix}$$

$$y^\mu \nabla (-x^\mu)^t = \begin{pmatrix} y_1^\mu - x_1^\mu & y_1^\mu - x_2^\mu & \dots & y_1^\mu - x_j^\mu & \dots & y_1^\mu - x_n^\mu \\ y_2^\mu - x_1^\mu & y_2^\mu - x_2^\mu & \dots & y_2^\mu - x_j^\mu & \dots & y_2^\mu - x_n^\mu \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ y_i^\mu - x_1^\mu & y_i^\mu - x_2^\mu & \dots & y_i^\mu - x_j^\mu & \dots & y_i^\mu - x_n^\mu \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ y_m^\mu - x_1^\mu & y_m^\mu - x_2^\mu & \dots & y_m^\mu - x_j^\mu & \dots & y_m^\mu - x_n^\mu \end{pmatrix} \quad (8.12)$$

2. Se aplica el operador mínimo  $\wedge$  a las  $p$  matrices para obtener la memoria  $W$ .

$$W = \bigwedge_{\mu=1}^p [y^\mu \nabla (-x^\mu)^t] = [w_{ij}]_{m \times n} \quad (8.13)$$

de manera que la  $ij$ -ésima componente de la memoria  $W$  se expresa así:

$$w_{ij} = \bigwedge_{\mu=1}^p (y_i^\mu - x_j^\mu) \quad (8.14)$$

La fase de recuperación consiste en realizar el producto máximo  $\nabla$  de la memoria  $W$  con un patrón de entrada  $x^\omega$ , donde  $\omega \in \{1, 2, \dots, p\}$ , para obtener un vector columna  $y$  de dimensión  $m$ :

$$y = W \nabla x^\omega \quad (8.15)$$

La  $i$ -ésima componente del vector  $y$ , de acuerdo con la expresión 8.2 es:

$$y_i = \bigvee_{j=1}^n (w_{ij} + x_j^\omega) \quad (8.16)$$

La referencia (Ritter, Diaz-de-Leon & Sussner, 1999) también proporciona respuestas a las dos preguntas:

- ¿cuáles son las condiciones necesarias y suficientes para asegurar respuesta perfecta en una memoria morfológica heteroasociativa *min*?
- ¿para qué pares de vectores  $\{(x^\mu, y^\mu) \mid \mu = 1, 2, \dots, p\}$  podrá la memoria asociativa descrita por las expresiones 8.12, 8.13, 8.14, 8.15 y 8.16 proporcionar una salida perfecta?

Las respuestas se encuentran en el teorema 2 y su correspondiente corolario 2.1:

**Teorema 2:**  $\mathbf{W} \nabla \mathbf{x}^\omega = \mathbf{y}^\omega$  para todo  $\omega = 1, \dots, p$  si y sólo si para cada índice de fila  $i = 1, \dots, m$  existen índices de columna  $j_i^\omega \in \{1, \dots, n\}$  tales que  $w_{ij_i^\omega} = y_i^\omega - x_{j_i^\omega}^\omega$  para todo  $\omega = 1, \dots, p$ .

**Corolario 2.1:**  $\mathbf{W} \nabla \mathbf{x}^\omega = \mathbf{y}^\omega$  para todo  $\omega = 1, \dots, p$  si y sólo si para cada índice de fila  $i = 1, \dots, m$  y cada  $\gamma \in \{1, \dots, p\}$  existen índices de columna  $j_i^\gamma \in \{1, \dots, n\}$  tales que

$$x_{j_i^\gamma}^\gamma = \bigvee_{\varepsilon=1}^p (x_{j_i^\gamma}^\varepsilon - y_i^\varepsilon) + y_i^\gamma$$

Las memorias morfológicas heteroasociativas *min* muestran experimentalmente un comportamiento similar al de las memorias morfológicas heteroasociativas *max*, cuando se diseñan y se operan en conjuntos fundamentales *sin ruido*; los experimentos son análogos al mostrado en la figura 1, con patrones del mismo tipo y tamaño.

Sin embargo, las memorias morfológicas heteroasociativas *min* responden al ruido de una manera totalmente diferente a como lo hacen las memorias morfológicas heteroasociativas *max*, como se desprende del teorema 6 de (Ritter, Diaz-de-Leon & Sussner, 1999), el cual es la respuesta a la interrogante ¿qué cantidad, y qué tipo de ruido se puede permitir como distorsión para los patrones de entrada en una memoria asociativa morfológica  $\mathbf{W}$  sin sacrificar la respuesta perfecta?

**Teorema 6:** Para  $\gamma = 1, \dots, p$  sea  $\tilde{\mathbf{x}}^\gamma$  una versión distorsionada del patrón  $\mathbf{x}^\gamma$ . Entonces  $\mathbf{W} \nabla \tilde{\mathbf{x}}^\gamma = \mathbf{y}^\gamma$  si y sólo si se cumple que:

$$\tilde{x}_j^\gamma \leq x_j^\gamma \vee \bigwedge_{i=1}^m \left( \bigvee_{\varepsilon \neq \gamma} [y_i^\gamma - y_i^\varepsilon + x_i^\varepsilon] \right) \quad \forall j = 1, \dots, n$$

y para cada índice renglón  $i \in \{1, \dots, m\}$  existe un índice columna  $j_i \in \{1, \dots, n\}$  tal que:

$$\tilde{x}_{j_i}^\gamma = x_{j_i}^\gamma \vee \left( \bigvee_{\varepsilon \neq \gamma} [y_i^\gamma - y_i^\varepsilon + x_{j_i}^\varepsilon] \right)$$

La contraparte de la figura 2 es la figura 3, donde se ilustra la respuesta de las memorias asociativas morfológicas  $W$  ante patrones de entrada alterados con 20% de ruido sustractivo. A diferencia de las memorias asociativas morfológicas tipo  $M$ , las memorias asociativas morfológicas  $min$  son robustas a ruido sustractivo: en eso difieren principalmente.

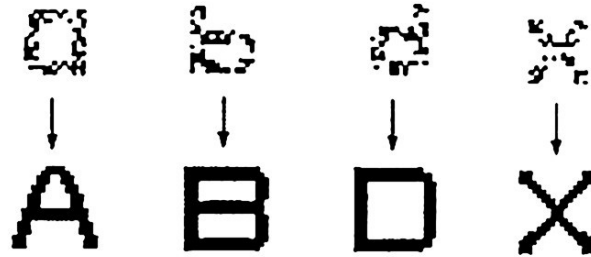


Figura 3

Similarmente a como lo hacen las memorias asociativas morfológicas tipo  $M$ , las memorias asociativas morfológicas tipo  $W$  son capaces de responder en forma perfecta en un solo paso, lo cual significa que *no requiere convergencia*.

### 8.3. Memorias Morfológicas Autoasociativas.

Hay dos tipos de memorias asociativas morfológicas: las memorias *max*, simbolizadas con  $M$ , y las memorias *min*, las cuales se simbolizan con  $W$ ; en cada uno de los dos tipos, las memorias pueden funcionar en ambos modos *heteroasociativo* y *autoasociativo*. Nótese, de entrada, esta ventaja que poseen las memorias asociativas morfológicas respecto de la memoria Hopfield, la cual se diseñó para funcionar sólo como memoria autoasociativa.

El conjunto fundamental será  $\{(x^\mu, y^\mu) \mid \mu = 1, 2, \dots, p\}$  con

$$x^\mu = \begin{pmatrix} x_1^\mu \\ x_2^\mu \\ \vdots \\ x_n^\mu \end{pmatrix} \in A^n \quad \text{y} \quad y^\mu = \begin{pmatrix} y_1^\mu \\ y_2^\mu \\ \vdots \\ y_m^\mu \end{pmatrix} \in A^m$$

Las expresiones anteriores son útiles para ambos modos de operación de las memorias morfológicas: *heteroasociativo* y *autoasociativo*. Dado que en

este número se describirán las memorias morfológicas autoasociativas. el conjunto fundamental estará constituido por parejas de patrones de entrada y salida idénticos: el conjunto fundamental será  $\{(\mathbf{x}^\mu, \mathbf{x}^\mu) \mid \mu = 1, 2, \dots, p\}$  con

$$\mathbf{x}^\mu = \begin{pmatrix} x_1^\mu \\ x_2^\mu \\ \vdots \\ x_n^\mu \end{pmatrix} \in A^n$$

Las expresiones 8.1, 8.2, 8.3, 8.4, 8.5 y 8.6 se adaptarán considerando esta particularidad de las memorias autoasociativas. al igual como se hizo con la memoria asociativa Hopfield la cual, por su propia naturaleza, tiene la fuerte restricción de que *sólo* actúa en modo autoasociativo.

### 8.3.1. Memorias autoasociativas max

Las memorias morfológicas autoasociativas *max* utilizan el *producto mínimo* para ambas fases: aprendizaje y recuperación de patrones. Además, en la fase de aprendizaje juega un papel relevante el operador *máximo*; de ahí el nombre de memorias *max*.

La *fase de aprendizaje* consiste de dos etapas:

1. Aplicando la expresión 8.6, para cada una de las  $p$  asociaciones  $(\mathbf{x}^\mu, \mathbf{x}^\mu)$  se encuentra la matriz  $\mathbf{x}^\mu \Delta (-\mathbf{x}^\mu)^t$  de dimensiones  $n \times n$ , donde  $(-\mathbf{x}^\mu)^t = (-x_1^\mu, -x_2^\mu, \dots, -x_n^\mu)$ :

$$\mathbf{x}^\mu \Delta (-\mathbf{x}^\mu)^t = \begin{pmatrix} x_1^\mu \\ x_2^\mu \\ \vdots \\ x_n^\mu \end{pmatrix} \Delta (-x_1^\mu, -x_2^\mu, \dots, -x_n^\mu)$$



$$\mathbf{x}^\mu \Delta (-\mathbf{x}^\mu)^t = \begin{pmatrix} [x_1^\mu + (-x_1^\mu)] & \cdots & [x_1^\mu + (-x_j^\mu)] & \cdots & [x_1^\mu + (-x_n^\mu)] \\ [x_2^\mu + (-x_1^\mu)] & \cdots & [x_2^\mu + (-x_j^\mu)] & \cdots & [x_2^\mu + (-x_n^\mu)] \\ \vdots & & \vdots & & \vdots \\ [x_i^\mu + (-x_1^\mu)] & \cdots & [x_i^\mu + (-x_j^\mu)] & \cdots & [x_i^\mu + (-x_n^\mu)] \\ \vdots & & \vdots & & \vdots \\ [x_n^\mu + (-x_1^\mu)] & \cdots & [x_n^\mu + (-x_j^\mu)] & \cdots & [x_n^\mu + (-x_n^\mu)] \end{pmatrix}$$

$$\mathbf{x}^\mu \Delta (-\mathbf{x}^\mu)^t = \begin{pmatrix} 0 & x_1^\mu - x_2^\mu & \cdots & x_1^\mu - x_j^\mu & \cdots & x_1^\mu - x_n^\mu \\ x_2^\mu - x_1^\mu & 0 & \cdots & x_2^\mu - x_j^\mu & \cdots & x_2^\mu - x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ x_i^\mu - x_1^\mu & x_i^\mu - x_2^\mu & \cdots & 0 & \cdots & x_i^\mu - x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ x_n^\mu - x_1^\mu & x_n^\mu - x_2^\mu & \cdots & x_n^\mu - x_j^\mu & \cdots & 0 \end{pmatrix} \quad (8.17)$$

2. Se aplica el operador *máximo*  $\bigvee$  a las  $p$  matrices para obtener la memoria  $\mathbf{M}$ .

$$\mathbf{M} = \bigvee_{\mu=1}^p [\mathbf{x}^\mu \Delta (-\mathbf{x}^\mu)^t] = [m_{ij}]_{n \times n} \quad (8.18)$$

de manera que la  $ij$ -ésima componente de la memoria  $\mathbf{M}$  se expresa así:

$$m_{ij} = \bigvee_{\mu=1}^p (x_i^\mu - x_j^\mu) \quad (8.19)$$

La *fase de recuperación* consiste en realizar el producto mínimo  $\Delta$  de la memoria  $\mathbf{M}$  con un patrón de entrada  $\mathbf{x}^\omega$ , donde  $\omega \in \{1, 2, \dots, p\}$ , para obtener un vector columna  $\mathbf{x}$  de dimensión  $n$ :

$$\mathbf{x} = \mathbf{M} \Delta \mathbf{x}^\omega \quad (8.20)$$

De acuerdo con la expresión 8.5, la  $i$ -ésima componente del vector  $\mathbf{x}$  es:

$$x_i = \bigwedge_{j=1}^n (m_{ij} + x_j^\omega) \quad (8.21)$$

En la descripción de las memorias morfológicas heteroasociativas (número anterior de esta colección) se planteó una pregunta ¿cuáles son las condiciones necesarias y suficientes para asegurar respuesta perfecta en una memoria morfológica *max*? Para ser más específicos, ¿para qué pares de vectores  $\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mu = 1, 2, \dots, p\}$  podrá una memoria morfológica heteroasociativa proporcionar una salida perfecta?. Como se mencionó, la respuesta a esta interrogante se da en el teorema 1 de (Ritter, Diaz-de-Leon & Sussner, 1999):

**Teorema 1:**  $\mathbf{M} \Delta \mathbf{x}^\omega = \mathbf{y}^\omega$  para todo  $\omega = 1, \dots, p$  si y sólo si para cada índice de fila  $i = 1, \dots, m$  existen índices de columna  $j_i^\omega \in \{1, \dots, n\}$  tales que  $m_{ij_i^\omega} = y_i^\omega - x_{j_i^\omega}^\omega$  para todo  $\omega = 1, \dots, p$ .

Al observar la matriz 8.17 de la fase de aprendizaje para las memorias morfológicas autoasociativas, vemos que los elementos de la diagonal principal son ceros, es decir, para cada índice de columna  $i$  existe el índice de fila  $i$  para el cual  $m_{ii} = x_i - x_i = 0$

Lo anterior garantiza que para las memorias morfológicas autoasociativas siempre se cumplen las condiciones del teorema, por lo que estas memorias poseen una fascinante propiedad adicional respecto de las heteroasociativas: tienen *capacidad máxima* de aprendizaje (cuando se permite que los valores de los patrones sean números reales, la capacidad es infinita), según muestra el teorema 4a de (Ritter, Sussner & Diaz-de-Leon, 1998):

**Teorema 4a:**  $\mathbf{M} \Delta \mathbf{x}^\omega = \mathbf{x}^\omega$  para todo  $\omega = 1, \dots, p$ .

Los autores usaron un conjunto fundamental formado por los doce patrones de la figura 1 para crear una memoria Hopfield y una memoria morfológica *max* autoasociativa; ésta última recuperó en forma perfecta todos los patrones del conjunto fundamental.



Figura 8.1: Figura 2

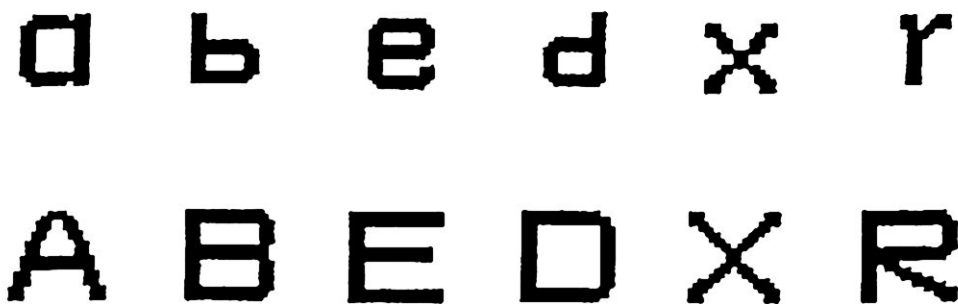


Figura 1

Por otro lado, cuando se le presentó cualquiera de los patrones de entrada del conjunto fundamental a memoria Hopfield, ésta arrojó como salida el patrón de la figura 2:

Como sucede con las memorias morfológicas heteroasociativas *max*, es importante hacer notar la considerable mejora en la capacidad de almacenamiento de las memorias morfológicas autoasociativas *max* con respecto a las memorias clásicas, así como el hecho de que la memoria morfológica *no requiere* convergencia; es decir, es capaz de responder en forma perfecta *en un solo paso*.

### 8.3.2. Memorias autoasociativas *min*

Las memorias morfológicas autoasociativas *min* utilizan el *producto máximo* para ambas fases: aprendizaje y recuperación de patrones. Además, en la fase de aprendizaje juega un papel relevante el operador *mínimo*: de ahí el nombre de memorias *min*.

La *fase de aprendizaje* consiste de dos etapas:

1. Aplicando la expresión 8.3, para cada una de las  $p$  asociaciones  $(\mathbf{x}^\mu, \mathbf{x}^\mu)$  se encuentra la matriz  $\mathbf{x}^\mu \nabla (-\mathbf{x}^\mu)^t$  de dimensiones  $n \times n$ , donde  $(-\mathbf{x}^\mu)^t = (-x_1^\mu, -x_2^\mu, \dots, -x_n^\mu)$ :

$$\mathbf{x}^\mu \nabla (-\mathbf{x}^\mu)^t = \begin{pmatrix} x_1^\mu \\ x_2^\mu \\ \vdots \\ x_n^\mu \end{pmatrix} \nabla (-x_1^\mu, -x_2^\mu, \dots, -x_n^\mu)$$

$$\mathbf{x}^\mu \nabla (-\mathbf{x}^\mu)^t = \begin{pmatrix} [x_1^\mu + (-x_1^\mu)] & \dots & [x_1^\mu + (-x_j^\mu)] & \dots & [x_1^\mu + (-x_n^\mu)] \\ [x_2^\mu + (-x_1^\mu)] & \dots & [x_2^\mu + (-x_j^\mu)] & \dots & [x_2^\mu + (-x_n^\mu)] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [x_i^\mu + (-x_1^\mu)] & \dots & [x_i^\mu + (-x_j^\mu)] & \dots & [x_i^\mu + (-x_n^\mu)] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [x_n^\mu + (-x_1^\mu)] & \dots & [x_n^\mu + (-x_j^\mu)] & \dots & [x_n^\mu + (-x_n^\mu)] \end{pmatrix}$$

$$\mathbf{x}^\mu \nabla (-\mathbf{x}^\mu)^t = \begin{pmatrix} 0 & x_1^\mu - x_2^\mu & \dots & x_1^\mu - x_j^\mu & \dots & x_1^\mu - x_n^\mu \\ x_2^\mu - x_1^\mu & 0 & \dots & x_2^\mu - x_j^\mu & \dots & x_2^\mu - x_n^\mu \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_i^\mu - x_1^\mu & x_i^\mu - x_2^\mu & \dots & 0 & \dots & x_i^\mu - x_n^\mu \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n^\mu - x_1^\mu & x_n^\mu - x_2^\mu & \dots & x_n^\mu - x_j^\mu & \dots & 0 \end{pmatrix} \quad (8.22)$$

2. Se aplica el operador mínimo  $\bigwedge$  a las  $p$  matrices para obtener la memoria  $W$ .

$$W = \bigwedge_{\mu=1}^p [\mathbf{x}^\mu \nabla (-\mathbf{x}^\mu)^t] = [w_{ij}]_{n \times n} \quad (8.23)$$

de manera que la  $ij$ -ésima componente de la memoria  $W$  se expresa así:

$$w_{ij} = \bigwedge_{\mu=1}^p (x_i^{\mu} - x_j^{\mu}) \quad (8.24)$$

La *fase de recuperación* consiste en realizar el producto máximo  $\nabla$  de la memoria  $W$  con un patrón de entrada  $x^{\omega}$ , donde  $\omega \in \{1, 2, \dots, p\}$ , para obtener un vector columna  $x$  de dimensión  $n$ :

$$x = W \nabla x^{\omega} \quad (8.25)$$

De acuerdo con la expresión 8.2, la  $i$ -ésima componente del vector  $x$  es:

$$x_i = \bigwedge_{j=1}^n (w_{ij} + x_j^{\omega}) \quad (8.26)$$

Similarmente a como lo hacen las memorias asociativas morfológicas tipo  $M$ , las memorias asociativas morfológicas tipo  $W$  son capaces de responder en forma perfecta en un solo paso, lo cual significa que *no requiere convergencia*.

Estas memorias poseen una fascinante propiedad adicional respecto de las heteroasociativas: tienen *capacidad máxima* de aprendizaje (cuando se permite que los valores de los patrones sean números reales, la capacidad es infinita), según muestra el teorema 4b de (Ritter, Sussner & Diaz-de-Leon, 1998):

**Teorema 4b:**  $W \nabla x^{\omega} = x^{\omega}$  para todo  $\omega = 1, \dots, p$ .

El comportamiento de las memorias autoasociativas morfológicas tipo  $W$  es igual al ilustrado en las figuras 1 y 2 para las memorias autoasociativas tipo  $M$ ; se usó el mismo conjunto fundamental de la figura 1 para crear y operar una memoria autoasociativa *min*, la cual respondió de manera perfecta al conjunto de entrada completo.

#### **8.4. Consideraciones finales del capítulo.**

Los problemas abiertos relacionados con las memorias asociativas morfológicas constituyen una rica veta de investigación, donde los científicos mexicanos pueden incursionar, con grandes probabilidades de obtener resultados interesantes para la comunidad internacional que cultiva estos temas.

Actualmente los autores de este modelo de memorias asociativas siguen investigando en éste y en otros temas. El Dr. Gerhard X Ritter es investigador de la University of Florida at Gainesville, mientras que el Dr. Juan Luis Díaz de León Santiago, coautor de esta obra, es el Director del Centro de Investigación en Computación del Instituto Politécnico Nacional



## Capítulo 9

# Memorias Asociativas $\alpha\beta$

En este capítulo se propone un nuevo modelo de memorias asociativas.

Las herramientas matemáticas del nuevo modelo incluyen dos operaciones binarias inventadas ex profeso, cuyos operadores fueron bautizados arbitrariamente con las dos primeras grafías del alfabeto griego:  $\alpha$  y  $\beta$ .

Las nuevas memorias asociativas  $\alpha\beta$  son de dos tipos y cada uno de ellos puede operar en dos modos diferentes. La operación  $\alpha$  es útil en la fase de aprendizaje, mientras que la operación  $\beta$  da sustento a la fase de recuperación de patrones.

Las propiedades algebraicas de las operaciones  $\alpha$  y  $\beta$  permiten que las nuevas memorias  $\alpha\beta$  exhiban características similares a las que son inherentes a las memorias asociativas morfológicas binarias, en cuanto a capacidades de aprendizaje y almacenamiento, tipos y cantidades de ruido a que son robustas, y condiciones suficientes para respuesta perfecta; adicionalmente, es preciso enfatizar que la densidad aritmética de las nuevas memorias asociativas es menor que la correspondiente a las memorias asociativas morfológicas.

La razón para tomar como referencia a las memorias asociativas morfológicas para la creación de las memorias  $\alpha\beta$ , consiste en que los autores de las primeras han mostrado que estas memorias superan en varios aspectos a los modelos conocidos de memorias asociativas hasta los inicios del tercer milenio.

En este capítulo se demuestra de manera teórica que las nuevas memorias, en ambos modos de operación, pueden soportar cantidades impresio-



nantes de ruido, aditivo o sustractivo, para entregar salidas perfectas en la fase de recuperación de patrones.

## 9.1. Herramientas Matemáticas

Esta sección consta de tres partes. En la primera se presentan las dos operaciones binarias originales  $\alpha$  y  $\beta$ , las cuales sirven de base para construir cuatro operaciones matriciales. que son presentadas en la segunda parte; finalmente, en la tercera parte se enfatiza el papel que juegan las relaciones de orden en este trabajo, al definir los diferentes tipos de ruido que pueden alterar un patrón binario dado.

### 9.1.1. Operaciones Binarias $\alpha$ y $\beta$

Los conjuntos  $A$  y  $B$  se definen así:

$$A = \{0, 1\} \quad \text{y} \quad B = \{0, 1, 2\}$$

La operación binaria  $\alpha : A \times A \rightarrow B$  está definida en la siguiente tabla:

| $x$ | $y$ | $\alpha(x, y)$ |
|-----|-----|----------------|
| 0   | 0   | 1              |
| 0   | 1   | 0              |
| 1   | 0   | 2              |
| 1   | 1   | 1              |

La operación binaria  $\alpha$  exhibe algunas propiedades algebraicas, expuestas a continuación. donde  $\vee$  es el operador *máximo* y  $\wedge$  es el operador *mínimo*:

|   |
|---|
| $\alpha(x, x) = 1$  |
| $(x \leq y) \leftrightarrow \alpha(x, y) \leq \alpha(y, x)$   |
| $(x \leq y) \leftrightarrow [\alpha(x, z) \leq \alpha(y, z)]$ |
| $(x \leq y) \leftrightarrow [\alpha(z, x) \geq \alpha(z, y)]$ |
| $\alpha[(x \vee y), z] = \alpha(x, z) \vee \alpha(y, z)$      |
| $\alpha[(x \wedge y), z] = \alpha(x, z) \wedge \alpha(y, z)$  |

La operación binaria  $\beta : B \times A \rightarrow A$  está definida en la siguiente tabla:

| $x$ | $y$ | $\beta(x, y)$ |
|-----|-----|---------------|
| 0   | 0   | 0             |
| 0   | 1   | 0             |
| 1   | 0   | 0             |
| 1   | 1   | 1             |
| 2   | 0   | 1             |
| 2   | 1   | 1             |

Propiedades algebraicas de la operación binaria  $\beta$ :

|   |
|---|
| $\beta(1, x) = x$   |
| $\beta(x, x) = x \quad \forall x \in A$                   |
| $(x \leq y) \rightarrow [\beta(x, z) \leq \beta(y, z)]$   |
| $(x \leq y) \rightarrow [\beta(z, x) \leq \beta(z, y)]$   |
| $\beta[(x \vee y), z] = \beta(x, z) \vee \beta(y, z)$     |
| $\beta[(x \wedge y), z] = \beta(x, z) \wedge \beta(y, z)$ |
| $\beta[x, (y \vee z)] = \beta(x, y) \vee \beta(x, z)$     |
| $\beta[x, (y \wedge z)] = \beta(x, y) \wedge \beta(x, z)$ |

Propiedades de la aplicación combinada de ambas operaciones  $\alpha$  y  $\beta$ :

|                              |
|------------------------------|
| $\beta[\alpha(x, y), y] = x$ |
| $\beta[\alpha(x, y), x] = x$ |
| $\beta[\alpha(x, x), y] = y$ |

Lo anterior significa que  $\beta$  es la inversa de  $\alpha$  por la derecha y por la izquierda.

Los conjuntos  $A$  y  $B$ , las operaciones  $\alpha$  y  $\beta$  junto con los operadores  $\wedge$  (mínimo) y  $\vee$  (máximo) usuales, conforman el sistema algebraico  $(A, B, \alpha, \beta, \wedge, \vee)$  en el que están inmersas las nuevas memorias asociativas  $\alpha\beta$ .

### 9.1.2. Operaciones Matriciales $\Psi_\alpha$ , $\mathfrak{M}_\alpha$ , $\Psi_\beta$ y $\mathfrak{M}_\beta$

Se definen las siguientes cuatro operaciones entre matrices:

- Operación  $\alpha\text{max}$ :  $P_{m \times r} \Psi_\alpha Q_{r \times n} = [f_{ij}^\alpha]_{m \times n}$ ; donde

$$f_{ij}^\alpha = \bigvee_{k=1}^r \alpha(p_{ik}, q_{kj})$$

2. Operación  $\beta\max$ :  $P_{m \times r} \uplus_{\beta} Q_{r \times n} = [f_{ij}^{\beta}]_{m \times n}$ , donde

$$f_{ij}^{\beta} = \bigvee_{k=1}^r \beta(p_{ik}, q_{kj})$$

3. Operación  $\alpha\min$ :  $P_{m \times r} \pitchfork_{\alpha} Q_{r \times n} = [h_{ij}^{\alpha}]_{m \times n}$ , donde

$$h_{ij}^{\alpha} = \bigwedge_{k=1}^r \alpha(p_{ik}, q_{kj})$$

4. Operación  $\beta\min$ :  $P_{m \times r} \pitchfork_{\beta} Q_{r \times n} = [h_{ij}^{\beta}]_{m \times n}$ , donde

$$h_{ij}^{\beta} = \bigwedge_{k=1}^r \beta(p_{ik}, q_{kj})$$

$k$  es un entero positivo que puede tomar valores entre 1 y  $r$  inclusive.

Obsérvese la *dualidad* entre los pares de operaciones  $\uplus_{\alpha}$  y  $\pitchfork_{\alpha}$  por un lado, y entre  $\uplus_{\beta}$  y  $\pitchfork_{\beta}$  por el otro.

Restricciones:

- Ninguna de las cuatro operaciones está definida si  $\exists j, k$  tales que  $q_{kj} = 2$ .
- Las operaciones  $\uplus_{\alpha}$  y  $\pitchfork_{\alpha}$  no están definidas si  $\exists i, j, k$  tales que  $p_{ik} = 2$  o  $q_{kj} = 2$ .

Estas restricciones aparentan ser causa de potenciales problemas que podrían aparecer al usar las operaciones anteriores; sin embargo, las nuevas memorias asociativas están diseñadas de modo que nunca ocurra algún caso prohibido.

**Lema 1.** Sean  $x \in A^n$ ,  $y \in A^m$ ; entonces  $y \uplus_{\alpha} x^t$  es una matriz de dimensiones  $m \times n$ , y además se cumple que:  $y \uplus_{\alpha} x^t = y \pitchfork_{\alpha} x^t$ .

El símbolo  $\boxtimes$  representará a las dos operaciones  $\uplus_{\alpha}$  y  $\pitchfork_{\alpha}$  cuando se opere un vector columna de dimensión  $m$  con un vector fila de dimensión  $n$ :

$$y \uplus_{\alpha} x^t = y \boxtimes x^t = y \pitchfork_{\alpha} x^t$$

La  $ij$ -ésima componente de la matriz  $\mathbf{y} \boxtimes \mathbf{x}'$  está dada por:

$$[\mathbf{y} \boxtimes \mathbf{x}']_{ij} = \alpha(y_i, x_j)$$

Es decir, la  $ij$ -ésima componente de la matriz  $\mathbf{y}'' \boxtimes (\mathbf{x}'')'$  se expresa de la siguiente manera:

$$[\mathbf{y}'' \boxtimes (\mathbf{x}'')']_{ij} = \alpha(y''_i, x''_j)$$

**Lema 2.** Sean  $\mathbf{x} \in A^n$  y  $\mathbf{P}$  una matriz de dimensiones  $m \times n$ . La operación  $\mathbf{P}_{m \times n} \cup_{\beta} \mathbf{x}$  da como resultado un vector columna de dimensión  $m$ , cuya  $i$ -ésima componente tiene la siguiente forma:  $(\mathbf{P}_{m \times n} \cup_{\beta} \mathbf{x})_i = \bigvee_{j=1}^n \beta(p_{ij}, x_j)$

**Lema 3.** Sean  $\mathbf{x} \in A^n$  y  $\mathbf{P}$  una matriz de dimensiones  $m \times n$ . La operación  $\mathbf{P}_{m \times n} \cap_{\beta} \mathbf{x}$  da como resultado un vector columna de dimensión  $m$ , cuya  $i$ -ésima componente tiene la siguiente forma:  $(\mathbf{P}_{m \times n} \cap_{\beta} \mathbf{x})_i = \bigwedge_{j=1}^n \beta(p_{ij}, x_j)$

### 9.1.3. Relaciones de Orden y Tipos de Ruido

La relación de orden usual tiene importancia central al definir operativamente los tipos de ruido que es posible encontrar en los patrones de entrada, y en el papel que juegan los operadores  $\bigvee$  y  $\bigwedge$  en la generación y operación de las memorias  $\alpha\beta$ .

A continuación se enuncian algunos conceptos respecto de la relación de orden entre matrices, considerando a los vectores columna como casos particulares (Moore, 1968; Rosen, 1995). Las componentes de matrices y vectores serán elementos de uno de los conjuntos  $A$  o  $B$ .

El máximo de dos matrices  $P$  y  $Q$  es otra matriz  $M$  que se representa por  $M = P \bigvee Q$ , y cuya  $ij$ -ésima entrada se define como  $m_{ij} = p_{ij} \bigvee q_{ij}$ .

El mínimo de dos matrices  $P$  y  $Q$  es otra matriz  $N$  que se representa por  $N = P \bigwedge Q$ , y cuya  $ij$ -ésima entrada se define como  $n_{ij} = p_{ij} \bigwedge q_{ij}$ .

La notación  $P \leq Q$  indica que la matriz  $P$  es menor o igual que la matriz  $Q$ , y esto se cumple si y sólo si  $p_{ij} \leq q_{ij}, \forall i \forall j$ .

$P < Q$  indica que la matriz  $P$  es estrictamente menor que la matriz  $Q$ . y esto se cumple si y sólo si  $p_{ij} \leq q_{ij}$ ,  $\forall i \forall j$  y  $\exists i_0, j_0$  tales que  $p_{i_0 j_0} < q_{i_0 j_0}$ .

Las anteriores consideraciones tienen relevancia en el contexto de este trabajo, al considerar los diferentes tipos de ruido que pueden distorsionar un patrón de entrada dado.

Sean dos vectores columna  $\mathbf{x}^1 \in A^n$  y  $\mathbf{x}^2 \in A^n$ ; se dice que  $\mathbf{x}^1$  es menor o igual a  $\mathbf{x}^2$  si y sólo si cada una de las componentes del vector  $\mathbf{x}^1$  es menor o igual a la correspondiente componente en el vector  $\mathbf{x}^2$ . Esto se expresa así:

$$\mathbf{x}^1 \leq \mathbf{x}^2 \iff x_i^1 \leq x_i^2 \forall i \in \{1, 2, \dots, n\}$$

Sean dos vectores columna  $\mathbf{x}^1 \in A^n$  y  $\mathbf{x}^2 \in A^n$ ; se dice que  $\mathbf{x}^1$  es menor a  $\mathbf{x}^2$  si y sólo si: cada una de las componentes del vector  $\mathbf{x}^1$  es menor o igual a la correspondiente componente en el vector  $\mathbf{x}^2$ , y existe al menos un valor  $i_0 \in \{1, 2, \dots, n\}$  para el cual se cumple la desigualdad estricta. Simbólicamente, esto se expresa así:

$$\mathbf{x}^1 < \mathbf{x}^2 \iff \left[ \begin{array}{l} x_i^1 \leq x_i^2 \forall i \in \{1, 2, \dots, n\} \\ \text{y } \exists i_0 \in \{1, 2, \dots, n\} \text{ tal que } x_{i_0}^1 < x_{i_0}^2 \end{array} \right]$$

Para el caso de dos vectores columna  $\mathbf{y}^1$  y  $\mathbf{y}^2$  que pertenecen al conjunto  $A^m$  las definiciones anteriores siguen siendo válidas:

$$\mathbf{y}^1 \leq \mathbf{y}^2 \iff y_j^1 \leq y_j^2 \forall j \in \{1, 2, \dots, m\}$$

$$\mathbf{y}^1 < \mathbf{y}^2 \iff \left[ \begin{array}{l} y_j^1 \leq y_j^2 \forall j \in \{1, 2, \dots, m\} \\ \text{y } \exists j_0 \in \{1, 2, \dots, m\} \text{ tal que } y_{j_0}^1 < y_{j_0}^2 \end{array} \right]$$

Ahora, sea  $\mathbf{x} \in A^n$  un patrón fundamental de entrada para una memoria asociativa  $\alpha\beta$ . El patrón  $\mathbf{x}$  puede ser alterado, para dar lugar a un vector  $\tilde{\mathbf{x}}$ , por tres tipos de ruido:

1. Ruido *aditivo*, si  $\mathbf{x} \leq \tilde{\mathbf{x}}$ . Esto significa que todos los posibles cambios en los valores de las coordenadas de  $\mathbf{x}$  para obtener  $\tilde{\mathbf{x}}$  consisten en colocar un valor 1 donde había un valor 0; es decir, la única posibilidad de cambio en las coordenadas de  $\mathbf{x}$  se traduce en:  $\exists i \in \{1, 2, \dots, n\}$  para el que  $x_i = 0$  y  $\tilde{x}_i = 1$ , pero no existe  $j \in \{1, 2, \dots, n\}$  para el que  $x_j = 1$  y  $\tilde{x}_j = 0$ . Además:  $\mathbf{x} \leq \tilde{\mathbf{x}} \rightarrow x_i \leq \tilde{x}_i, \forall i \in \{1, 2, \dots, n\}$

2. Ruido *sustractivo*, si  $\mathbf{x} \geq \tilde{\mathbf{x}}$ . Esto significa que todos los posibles cambios en los valores de las coordenadas de  $\mathbf{x}$  para obtener  $\tilde{\mathbf{x}}$  consisten en colocar un valor 0 donde había un valor 1; es decir, la única posibilidad de cambio en las coordenadas de  $\mathbf{x}$  se traduce en:  $\exists i \in \{1, 2, \dots, n\}$  para el que  $x_i = 1$  y  $\tilde{x}_i = 0$ . pero no existe  $j \in \{1, 2, \dots, n\}$  para el que  $x_j = 0$  y  $\tilde{x}_j = 1$ . Además:  $\mathbf{x} \geq \tilde{\mathbf{x}} \rightarrow x_i \geq \tilde{x}_i, \forall i \in \{1, 2, \dots, n\}$
3. Ruido *combinado o mezclado*. si el ruido es una mezcla de aditivo con sustractivo. En este caso no es posible establecer un orden entre el patrón limpio y el ruidoso, dado que los valores podrán ser cambiados aleatoriamente. sin respetar necesariamente las reglas de los items 1 y 2.

Si el ruido es de 0%, es claro que  $\mathbf{x} = \tilde{\mathbf{x}}$

## 9.2. El modelo de las Memorias Asociativas $\alpha\beta$

En esta sección se presenta la obtención, justificación teórica y uso de las nuevas memorias asociativas basadas en las operaciones binarias originales  $\alpha$  y  $\beta$ .

### 9.2.1. Memorias Heteroasociativas $\alpha\beta$

Se proponen dos tipos de memorias heteroasociativas  $\alpha\beta$ : tipo  $\mathbf{V}$  y tipo  $\mathbf{\Lambda}$ ; se desarrollarán sólo las de tipo  $\mathbf{V}$  ya que las propiedades de las memorias tipo  $\mathbf{\Lambda}$  se obtienen por dualidad.

Se usará el operador  $\boxtimes$  el cual tiene la siguiente forma. para los índices  $\mu \in \{1, 2, \dots, p\}$ ,  $i \in \{1, 2, \dots, m\}$ . y  $j \in \{1, 2, \dots, n\}$ :

$$[\mathbf{y}^\mu \boxtimes (\mathbf{x}^\mu)^t]_{ij} = \alpha(y_i^\mu, x_j^\mu)$$

**Fase de aprendizaje.**

**PASO 1** Para cada  $\mu = 1, 2, \dots, p$ . a partir de la pareja  $(\mathbf{x}^\mu, \mathbf{y}^\mu)$  se construye la matriz

$$[\mathbf{y}^\mu \boxtimes (\mathbf{x}^\mu)^t]_{m \times n} \tag{9.1}$$

**PASO 2** Se aplica el operador binario *máximo*  $\bigvee$  a las matrices obtenidas en el paso 1:

$$\mathbf{V} = \bigvee_{\mu=1}^p [y^{\mu} \boxtimes (x^{\mu})^t] \quad (9.2)$$

La entrada  $ij$ -ésima está dada por la siguiente expresión:

$$\nu_{ij} = \bigvee_{\mu=1}^p \alpha(y_i^{\mu}, x_j^{\mu}) \quad (9.3)$$

Es posible observar que  $\nu_{ij} \in B, \forall i \in \{1, 2, \dots, m\}, \forall j \in \{1, 2, \dots, n\}$

■

**Fase de recuperación.**

**CASO 1: Patrón fundamental** Se presenta un patrón  $\mathbf{x}^{\omega}$ , con  $\omega \in \{1, 2, \dots, p\}$ , a la memoria heteroasociativa  $\alpha\beta$  tipo  $\mathbf{V}$  y se realiza la operación  $\mathbb{M}_{\beta}$ :

$$\mathbf{V} \mathbb{M}_{\beta} \mathbf{x}^{\omega} \quad (9.4)$$

Dado que las dimensiones de la matriz  $\mathbf{V}$  son  $m \times n$  y  $\mathbf{x}^{\omega}$  es un vector columna de dimensión  $n$ , el resultado de la operación anterior debe ser un vector columna de dimensión  $m$ , cuya  $i$ -ésima componente es:

$$(\mathbf{V} \mathbb{M}_{\beta} \mathbf{x}^{\omega})_i = \bigwedge_{j=1}^n \beta(\nu_{ij}, x_j^{\omega}) \quad (9.5)$$

$$(\mathbf{V} \mathbb{M}_{\beta} \mathbf{x}^{\omega})_i = \bigwedge_{j=1}^n \beta \left\{ \left[ \bigvee_{\mu=1}^p \alpha(y_i^{\mu}, x_j^{\mu}) \right], x_j^{\omega} \right\} \quad (9.6)$$

**CASO 2: Patrón alterado** Se presenta un patrón binario  $\tilde{\mathbf{x}}$  (patrón alterado de algún patrón fundamental  $\mathbf{x}^{\omega}$ ) que es un vector columna de dimensión  $n$ , a la memoria heteroasociativa  $\alpha\beta$  tipo  $\mathbf{V}$  y se realiza la operación  $\mathbb{M}_{\beta}$  :

$$\mathbf{V} \mathbb{M}_{\beta} \tilde{\mathbf{x}} \quad (9.7)$$

El resultado de la operación anterior es un vector columna de dimensión  $m$ , cuya  $i$ -ésima componente se expresa de la siguiente manera:

$$(\mathbf{V} \mathfrak{m}_\beta \tilde{\mathbf{x}})_i = \bigwedge_{j=1}^n \beta(\nu_{ij}, \tilde{x}_j) \quad (9.8)$$

$$(\mathbf{V} \mathfrak{m}_\beta \tilde{\mathbf{x}})_i = \bigwedge_{j=1}^n \beta \left\{ \left[ \bigvee_{\mu=1}^p \alpha(y_i^\mu, x_j^\mu) \right], \tilde{x}_j \right\} \quad (9.9)$$

■

**Lema 4.** Sea  $\{(x^\mu, y^\mu) \mid \mu = 1, 2, \dots, p\}$  el conjunto fundamental de una memoria heteroasociativa  $\alpha\beta$  representada por  $\mathbf{V}$ . Si  $\omega$  es un valor arbitrario de índice tal que  $\omega \in \{1, 2, \dots, p\}$  entonces  $\mathbf{V} \mathfrak{m}_\beta \mathbf{x}^\omega \geq \mathbf{y}^\omega$ .

*Demostración.*- Sea  $i \in \{1, 2, \dots, m\}$  arbitraria. La  $i$ -ésima componente del vector  $\mathbf{V} \mathfrak{m}_\beta \mathbf{x}^\omega$  se expresa así:

$$(\mathbf{V} \mathfrak{m}_\beta \mathbf{x}^\omega)_i = \bigwedge_{j=1}^n \beta(\nu_{ij}, x_j^\omega)$$

pero

$$\nu_{ij} = \bigvee_{\mu=1}^p \alpha(y_i^\mu, x_j^\mu)$$

por ello:

$$(\mathbf{V} \mathfrak{m}_\beta \mathbf{x}^\omega)_i = \bigwedge_{j=1}^n \beta \left\{ \left[ \bigvee_{\mu=1}^p \alpha(y_i^\mu, x_j^\mu) \right], x_j^\omega \right\}$$

Por otro lado, por hipótesis  $\omega \in \{1, 2, \dots, p\}$ , y esto significa que:

$$\bigvee_{\mu=1}^p \alpha(y_i^\mu, x_j^\mu) \geq \alpha(y_i^\omega, x_j^\omega)$$

Se realiza la operación binaria  $\beta$ , eligiendo los dos miembros de esta desigualdad como operandos izquierdos, y  $x_j^\omega$  como operando derecho en ambos casos. Dado que  $\beta$  es creciente por la izquierda, se tiene:

$$\beta \left\{ \left[ \bigvee_{\mu=1}^p \alpha(y_i^\mu, x_j^\mu) \right], x_j^\omega \right\} \geq \beta [\alpha(y_i^\omega, x_j^\omega), x_j^\omega]$$



Al tomar el mínimo de ambos miembros respecto del índice  $j$ :

$$\bigwedge_{j=1}^n \beta \left\{ \left[ \bigvee_{\mu=1}^p \alpha(y_i^\mu, x_j^\mu) \right] \cdot x_j^\omega \right\} \geq \bigwedge_{j=1}^n \beta [\alpha(y_i^\omega, x_j^\omega), x_j^\omega]$$

Por transitividad de la desigualdad anterior:

$$(\mathbf{V} \mathfrak{m}_{\beta} \mathbf{x}^\omega)_i \geq \bigwedge_{j=1}^n \beta [\alpha(y_i^\omega, x_j^\omega), x_j^\omega]$$

Además:

$$\beta [\alpha(y_i^\omega, x_j^\omega), x_j^\omega] = y_i^\omega$$

esto es:

$$(\mathbf{V} \mathfrak{m}_{\beta} \mathbf{x}^\omega)_i \geq \bigwedge_{j=1}^n y_i^\omega$$

Pero

$$\bigwedge_{j=1}^n y_i^\omega = y_i^\omega$$

porque  $y_i^\omega$  no depende de  $j$ , es decir:

$$(\mathbf{V} \mathfrak{m}_{\beta} \mathbf{x}^\omega)_i \geq y_i^\omega$$

Dado que  $i$  se escogió de manera arbitraria, se puede afirmar que la expresión anterior es válida para todos los valores de  $i$ , es decir:

$$(\mathbf{V} \mathfrak{m}_{\beta} \mathbf{x}^\omega)_i \geq y_i^\omega \quad \forall i \in \{1, 2, \dots, m\}$$

Por lo tanto:

$$\mathbf{V} \mathfrak{m}_{\beta} \mathbf{x}^\omega \geq \mathbf{y}^\omega \quad \text{Conclusión.}$$

En virtud de que el valor  $\omega$  es arbitrario dentro del conjunto de índices para los patrones del conjunto fundamental, este Lema deja en claro que la desigualdad se cumple para *todas* las parejas de patrones que son elementos del conjunto fundamental, sin imponer condición alguna.

■

**Teorema 1.** Sea  $\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mu = 1, 2, \dots, p\}$  el conjunto fundamental de una memoria heteroasociativa  $\alpha\beta$  representada por  $\mathbf{V}$ . Si  $\omega$  es un valor de índice arbitrario tal que  $\omega \in \{1, 2, \dots, p\}$ , y si además para cada  $i \in \{1, \dots, m\}$  se cumple que  $\exists j = j_0 \in \{1, \dots, n\}$ , el cual depende de  $\omega$  y de  $i$ , tal que

$\nu_{ij_0} = \alpha(y_i^\omega, x_{j_0}^\omega)$ , entonces la recuperación  $\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega$  es perfecta; es decir  $\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega = \mathbf{y}^\omega$ .

Demostración.- Sea  $i \in \{1, 2, \dots, m\}$  arbitraria. La  $i$ -ésima componente del vector  $\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega$  se expresa así:

$$(\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega)_i = \bigwedge_{j=1}^n \beta(\nu_{ij}, x_j^\omega)$$

pero al mismo tiempo, al hacer  $j = j_0$ , se cumple la siguiente desigualdad:  $\bigwedge_{j=1}^n \beta(\nu_{ij}, x_j^\omega) \leq \beta(\nu_{ij_0}, x_{j_0}^\omega)$ , y por transitividad se llega a:

$$(\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega)_i \leq \beta(\nu_{ij_0}, x_{j_0}^\omega)$$

Además, por hipótesis  $\nu_{ij_0} = \alpha(y_i^\omega, x_{j_0}^\omega)$ , es decir:

$$(\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega)_i \leq \beta[\alpha(y_i^\omega, x_{j_0}^\omega), x_{j_0}^\omega]$$

Además, se tiene

$$\beta[\alpha(y_i^\omega, x_{j_0}^\omega), x_{j_0}^\omega] = y_i^\omega$$

por lo que la desigualdad anterior queda así:

$$(\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega)_i \leq y_i^\omega$$

Dado que  $i$  se escogió de manera arbitraria, se puede afirmar que la expresión anterior es válida para todos los valores de  $i$ , por lo que:

$$(\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega)_i \leq y_i^\omega \quad \forall i \in \{1, 2, \dots, m\}$$

La expresión anterior se traduce en la siguiente desigualdad vectorial:

$$\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega \leq \mathbf{y}^\omega$$

Pero al cumplirse la hipótesis del Lema 4, se tiene la desigualdad en el otro sentido

$$\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega \geq \mathbf{y}^\omega$$

Por lo tanto, se llega a la recuperación perfecta del patrón  $\mathbf{y}^\omega$ :

$$\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega = \mathbf{y}^\omega \quad \text{Conclusión.}$$

■

**Teorema 2 (forma equivalente matricial del Teorema 1).** Si para cada asociación  $(\mathbf{x}^\omega, \mathbf{y}^\omega)$  del conjunto fundamental de una memoria heteroasociativa  $\alpha\beta \mathbf{V}$ , se cumple que cada fila de la matriz  $\mathbf{V} - \mathbf{y}^\omega \boxtimes (\mathbf{x}^\omega)^t$  contiene una entrada cero, entonces la memoria  $\mathbf{V}$  recupera el conjunto de patrones de salida fundamentales en forma perfecta.

**Demostración.-** Dado que la tesis es igual, es suficiente encontrar un enunciado que sea lógicamente equivalente a la hipótesis del Teorema 1.

Como  $\omega$  es un valor de índice arbitrario tal que  $\omega \in \{1, 2, \dots, p\}$ , la hipótesis se puede expresar así:

$$\begin{aligned} \forall \omega &\in \{1, 2, \dots, p\} \text{ y cada } i \in \{1, \dots, m\}, \\ \exists j_0 &\in \{1, \dots, n\} \text{ tal que } \nu_{ij_0} = \alpha(y_i^\omega, x_{j_0}^\omega) \end{aligned}$$

Pero las siguientes expresiones son válidas:

$$\begin{aligned} \alpha(y_i^\omega, x_{j_0}^\omega) &= [\mathbf{y}^\omega \boxtimes (\mathbf{x}^\omega)^t]_{ij_0} \\ \nu_{ij_0} &= [\mathbf{V}]_{ij_0} \end{aligned}$$

por ello, la expresión:

$$\nu_{ij_0} = \alpha(y_i^\omega, x_{j_0}^\omega)$$

es equivalente a

$$[\mathbf{V}]_{ij_0} = [\mathbf{y}^\omega \boxtimes (\mathbf{x}^\omega)^t]_{ij_0}$$

que a su vez se puede transformar en

$$[\mathbf{V}]_{ij_0} - [\mathbf{y}^\omega \boxtimes (\mathbf{x}^\omega)^t]_{ij_0} = 0$$

y finalmente en la expresión

$$[\mathbf{V} - \mathbf{y}^\omega \boxtimes (\mathbf{x}^\omega)^t]_{ij_0} = 0$$

Por lo anterior podemos obtener una expresión lógicamente equivalente a la hipótesis del Teorema:

$$\begin{aligned} \forall \omega &\in \{1, 2, \dots, p\} \text{ y cada } i \in \{1, \dots, m\}, \\ \exists j_0 &\in \{1, \dots, n\} \text{ tal que } [\mathbf{V} - \mathbf{y}^\omega \boxtimes (\mathbf{x}^\omega)^t]_{ij_0} = 0 \end{aligned}$$

La última expresión se puede enunciar del siguiente modo equivalente: para todas las asociaciones del conjunto fundamental de la memoria heteroasociativa  $\alpha\beta \mathbf{V}$ , se cumple que cada fila de la matriz  $\mathbf{V} - \mathbf{y}^\omega \boxtimes (\mathbf{x}^\omega)^t$  contiene una entrada cero.



Ha llegado el momento de atacar un problema trascendental en el tema de las memorias asociativas: encontrar las condiciones *suficientes* para que una memoria asociativa (en este caso para la memoria heteroasociativa  $\alpha\beta$  tipo **V**) recupere patrones de salida fundamentales a partir de patrones de entrada distorsionados con ruido, es decir, patrones de entrada no fundamentales. Dentro de esas condiciones suficientes debe incluirse la cantidad y los tipos de ruido a los que la memoria es inmune: aditivo o sustractivo.

**Lema 5.** Sea  $\{(x^\mu, y^\mu) \mid \mu = 1, 2, \dots, p\}$  el conjunto fundamental de una memoria heteroasociativa  $\alpha\beta$  representada por **V**, y sea  $\tilde{x} \in A^n$  un patrón alterado que se presenta a la memoria **V** como entrada. Si  $\exists \omega \in \{1, 2, \dots, p\}$  tal que es posible obtener el patrón  $\tilde{x}$  alterando el patrón fundamental  $x^\omega$  con ruido aditivo, entonces  $\mathbf{V} \circlearrowright \tilde{x} \geq y^\omega$ .

*Demostración.-* Sea  $i \in \{1, 2, \dots, m\}$  arbitraria. La  $i$ -ésima componente del vector  $\mathbf{V} \circlearrowright \tilde{x}$  se expresa así:

$$(\mathbf{V} \circlearrowright \tilde{x})_i = \bigwedge_{j=1}^n \beta(\nu_{ij}, \tilde{x}_j) \quad (9.10)$$

Por otro lado, por hipótesis  $\tilde{x}$  es una alteración con ruido aditivo del patrón fundamental  $x^\omega$ , y se tiene  $x^\omega \leq \tilde{x}$ ; es decir:  $\tilde{x} \geq x^\omega$ , lo cual implica que

$$\tilde{x}_j \geq x_j^\omega, \quad \forall j \in \{1, 2, \dots, n\}$$

Se realiza la operación binaria  $\beta$ , eligiendo los dos miembros de esta desigualdad como operandos derechos, y a la  $ij$ -ésima componente  $\nu_{ij}$  de **V** como operando izquierdo en ambos casos. Dado que  $\beta$  es creciente por la derecha, se tiene:

$$\beta(\nu_{ij}, \tilde{x}_j) \geq \beta(\nu_{ij}, x_j^\omega) \quad \forall j \in \{1, 2, \dots, n\}$$

Al tomar el mínimo de ambos miembros respecto del índice  $j$ :

$$\bigwedge_{j=1}^n \beta(\nu_{ij}, \tilde{x}_j) \geq \bigwedge_{j=1}^n \beta(\nu_{ij}, x_j^\omega)$$

Por transitividad de la desigualdad anterior con la expresión 9.10:

$$(\mathbf{V} \circlearrowright \tilde{x})_i \geq \bigwedge_{j=1}^n \beta(\nu_{ij}, x_j^\omega)$$

Pero

$$\bigwedge_{j=1}^n \beta(\nu_{ij}, x_j^\omega) = (\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega)_i$$

esto es:

$$(\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}})_i \geq (\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega)_i$$

Además, por Lema 4

$$(\mathbf{V} \mathfrak{M}_\beta \mathbf{x}^\omega)_i \geq y_i^\omega$$

y por transitividad con la expresión anterior:

$$(\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}})_i \geq y_i^\omega$$

Dado que  $i$  se escogió de manera arbitraria, se puede afirmar que la expresión anterior es válida para todos los valores de  $i$ , es decir:

$$(\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}})_i \geq y_i^\omega \quad \forall i \in \{1, 2, \dots, m\}$$

Por lo tanto:

$$\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}} \geq \mathbf{y}^\omega \quad \text{Conclusión.}$$

■

**Teorema 3.** Sea  $\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mu = 1, 2, \dots, p\}$  el conjunto fundamental de una memoria heteroasociativa  $\alpha;\beta$  representada por  $\mathbf{V}$ , y sea  $\tilde{\mathbf{x}} \in A^n$  un patrón alterado con ruido aditivo respecto de algún patrón fundamental  $\mathbf{x}^\omega$  con  $\omega \in \{1, 2, \dots, p\}$ . Si se presenta  $\tilde{\mathbf{x}}$  a la memoria  $\mathbf{V}$  como entrada, y si además para cada  $i \in \{1, \dots, m\}$  se cumple la condición de que  $\exists j = j_0 \in \{1, \dots, n\}$ , el cual depende de  $\omega$  y de  $i$  tal que  $\nu_{ij_0} \leq \alpha(y_i^\omega, \tilde{x}_{j_0})$ , entonces la recuperación  $\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}}$  es perfecta; es decir  $\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}} = \mathbf{y}^\omega$ .

*Demostración.*- Sea  $i \in \{1, 2, \dots, m\}$  arbitraria. La  $i$ -ésima componente del vector  $\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}}$  se expresa así:

$$(\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}})_i = \bigwedge_{j=1}^n \beta(\nu_{ij}, \tilde{x}_j)$$

pero al mismo tiempo, al hacer  $j = j_0$ , se cumple la desigualdad:

$$\bigwedge_{j=1}^n \beta(\nu_{ij}, \tilde{x}_j) \leq \beta(\nu_{ij_0}, \tilde{x}_{j_0})$$

y por transitividad se llega a:

$$(\mathbf{V} \mathfrak{m}_\beta \tilde{\mathbf{x}})_i \leq \beta(\nu_{ij_0}, \tilde{x}_{j_0}) \quad (9.11)$$

Por otro lado, por hipótesis  $\nu_{ij_0} \leq \alpha(y_i^\omega, \tilde{x}_{j_0})$ ; se realiza la operación binaria  $\beta$ , eligiendo los dos miembros de esta desigualdad como operandos izquierdos, y  $\tilde{x}_{j_0}$  como operando derecho en ambos casos. Dado que  $\beta$  es creciente por la izquierda (propiedad  $\beta 3$  de la Tabla 3.4), se tiene:

$$\beta(\nu_{ij_0}, \tilde{x}_{j_0}) \leq \beta[\alpha(y_i^\omega, \tilde{x}_{j_0}), \tilde{x}_{j_0}]$$

Por transitividad de esta desigualdad con la expresión 9.11:

$$(\mathbf{V} \mathfrak{m}_\beta \tilde{\mathbf{x}})_i \leq \beta[\alpha(y_i^\omega, \tilde{x}_{j_0}), \tilde{x}_{j_0}]$$

Además:

$$\beta[\alpha(y_i^\omega, \tilde{x}_{j_0}), \tilde{x}_{j_0}] = y_i^\omega$$

esto es:

$$(\mathbf{V} \mathfrak{m}_\beta \tilde{\mathbf{x}})_i \leq y_i^\omega$$

Dado que  $i$  se escogió de manera arbitraria, se puede afirmar que esta expresión es válida para todos los valores de  $i$ , por lo que:

$$(\mathbf{V} \mathfrak{m}_\beta \tilde{\mathbf{x}})_i \leq y_i^\omega \quad \forall i \in \{1, 2, \dots, m\}$$

Esta expresión se traduce en la siguiente desigualdad vectorial:

$$\mathbf{V} \mathfrak{m}_\beta \tilde{\mathbf{x}} \leq \mathbf{y}^\omega$$

Por lo tanto, aplicando el Lema 5, se llega a la recuperación perfecta del patrón  $\mathbf{y}^\omega$ .

$$\mathbf{V} \mathfrak{m}_\beta \tilde{\mathbf{x}} = \mathbf{y}^\omega \quad \text{Conclusión.}$$

■

El Lema 5 y el Teorema 3 indican que las memorias heteroasociativas  $\alpha\beta$  tipo  $\mathbf{V}$  tienen cierta inmunidad al ruido **aditivo**, y especifican las condiciones que deben cumplirse para que la respuesta sea perfecta en presencia de ruido aditivo.

De inmediato surge una interrogante: ¿qué pasa con el ruido **sustractivo**? Las memorias  $\alpha\beta$  tipo  $\mathbf{V}$  son sensibles a ruido sustractivo; una pequeña cantidad de ruido sustractivo puede tener efectos no deseados en la operación

de este tipo de memorias, los cuales son caracterizados por el Teorema siguiente:

**Teorema 4.** Sea  $\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mu = 1, 2, \dots, p\}$  el conjunto fundamental de una memoria heteroasociativa  $\alpha\beta$  representada por  $\mathbf{V}$ , y sea  $\tilde{\mathbf{x}} \in A^n$  un patrón alterado con ruido sustractivo respecto de algún patrón fundamental  $\mathbf{x}^\omega$  con  $\omega \in \{1, 2, \dots, p\}$ . Al presentar  $\tilde{\mathbf{x}}$  a la memoria  $\mathbf{V}$  como entrada se cumple lo siguiente: para cada  $j_0 \in \{1, \dots, n\}$  tal que  $x_{j_0}^\omega$  haya sido alterado para obtener  $\tilde{x}_{j_0}$ , si  $\exists i_0 \in \{1, \dots, m\}$  para el que  $\nu_{i_0 j_0} = 1$ , entonces  $(\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}})_{i_0} = 0$ .

**Demostración.**- Sea  $\tilde{\mathbf{x}}$  una versión distorsionada con ruido sustractivo del patrón fundamental  $\mathbf{x}^\omega$ : es decir,  $\tilde{\mathbf{x}} \leq \mathbf{x}^\omega$ , lo cual implica que  $\tilde{x}_j \leq x_j^\omega \forall j \in \{1, 2, \dots, n\}$ . Dependiendo del porcentaje de ruido sustractivo con que se ha alterado  $\mathbf{x}^\omega$ , puede haber más de un valor de  $j$ , hasta el número de bits con valor 1 en  $\mathbf{x}^\omega$ , para los que se cumple la desigualdad estricta  $\tilde{x}_j < x_j^\omega$ .

Sea  $j_0 \in \{1, \dots, n\}$  un índice para el que se cumple la desigualdad estricta:  $\tilde{x}_{j_0} < x_{j_0}^\omega$ ; es decir, debe cumplirse que  $\tilde{x}_{j_0} = 0$  y  $x_{j_0}^\omega = 1$ . La expresión para la componente  $i$  del vector recuperado es:

$$(\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}})_i = \bigwedge_{j=1}^n \beta(\nu_{ij}, \tilde{x}_j)$$

Reescribamos la expresión anterior para tomar en cuenta explícitamente el valor de  $j_0$ :

$$(\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}})_i = \bigwedge \left\{ \begin{array}{l} \left[ \bigwedge_{j=1}^{j_0-1} \beta(\nu_{ij}, \tilde{x}_j) \right], \\ \left[ \beta(\nu_{ij_0}, \tilde{x}_{j_0}) \right], \\ \left[ \bigwedge_{j=j_0+1}^n \beta(\nu_{ij}, \tilde{x}_j) \right] \end{array} \right\}$$

Es decir:

$$(\mathbf{V} \mathfrak{M}_\beta \tilde{\mathbf{x}})_i = \bigwedge \left\{ \begin{array}{l} \left[ \bigwedge_{j=1}^{j_0-1} \beta(\nu_{ij}, \tilde{x}_j) \right], \\ \left[ \beta(\nu_{ij_0}, 0) \right], \\ \left[ \bigwedge_{j=j_0+1}^n \beta(\nu_{ij}, \tilde{x}_j) \right] \end{array} \right\}$$

Por hipótesis,  $\exists i_0 \in \{1, \dots, m\}$  para el que  $\nu_{i_0 j_0} = 1$ . Analicemos este

caso crítico:

$$(\mathbf{V} \mathfrak{m}_{\beta} \tilde{\mathbf{x}})_{i_0} = \bigwedge \left\{ \begin{array}{l} [\bigwedge_{j=1}^{j_0-1} \beta(\nu_{i_0j}, \tilde{x}_j)] \cdot \\ [\beta(\nu_{i_0j_0}, 0)], \\ [\bigwedge_{j=j_0+1}^n \beta(\nu_{i_0j}, \tilde{x}_j)] \end{array} \right\}$$

$$(\mathbf{V} \mathfrak{m}_{\beta} \tilde{\mathbf{x}})_{i_0} = \bigwedge \left\{ \begin{array}{l} [\bigwedge_{j=1}^{j_0-1} \beta(\nu_{i_0j}, \tilde{x}_j)] \cdot \\ [\beta(1, 0)], \\ [\bigwedge_{j=j_0+1}^n \beta(\nu_{i_0j}, \tilde{x}_j)] \end{array} \right\}$$

Pero  $\beta(1, 0) = 0$ , por lo que la expresión anterior se transforma en:

$$(\mathbf{V} \mathfrak{m}_{\beta} \tilde{\mathbf{x}})_{i_0} = \bigwedge \left\{ \begin{array}{l} [\bigwedge_{j=1}^{j_0-1} \beta(\nu_{i_0j}, \tilde{x}_j)] \cdot \\ 0, \\ [\bigwedge_{j=j_0+1}^n \beta(\nu_{i_0j}, \tilde{x}_j)] \end{array} \right\}$$

Por lo tanto:

$$(\mathbf{V} \mathfrak{m}_{\beta} \tilde{\mathbf{x}})_{i_0} = 0$$

■

*Nota importante:* Sin embargo, dado el  $j_0$  del Teorema 4, si  $\nu_{i_j_0} \neq 1$ ,  $\forall i \in \{1, \dots, m\}$ , entonces el ruido sustractivo en la componente  $\tilde{x}_{j_0}$  no afecta la posible recuperación perfecta del patrón  $\mathbf{y}^\omega$ . Esto significa que las memorias  $\alpha\beta$  tipo  $\mathbf{V}$  son capaces de soportar ciertas cantidades de ruido sustractivo.

Las memorias heteroasociativas  $\alpha\beta$  tipo  $\mathbf{\Lambda}$  se desarrollan por dualidad, partiendo de los resultados obtenidos para las memorias heteroasociativas  $\alpha\beta$  tipo  $\mathbf{V}$ . Para ello, se realizan los siguientes cambios:

- Donde haya un operador  $\bigvee$  colocar un  $\bigwedge$
- Donde haya un operador  $\bigwedge$  colocar un  $\bigvee$
- Usar el operador  $\mathfrak{U}_{\beta}$  en lugar del operador  $\mathfrak{m}_{\beta}$

Mientras que las memorias heteroasociativas  $\alpha\beta$  tipo  $\mathbf{V}$  tienen cierta inmunidad al ruido aditivo y son sensitivas a ruido sustractivo, con las memorias heteroasociativas  $\alpha\beta$  tipo  $\mathbf{\Lambda}$  sucede precisamente lo contrario: son inmunes a cierta cantidad de ruido sustractivo, pero sensitivas a ruido aditivo.



Una pequeña cantidad de ruido aditivo puede tener efectos no deseados en la operación de este tipo de memorias  $\alpha\beta$ ; sin embargo, las memorias  $\alpha\beta$  tipo  $\Lambda$  son capaces de soportar ciertas cantidades de ruido aditivo.

### 9.2.2. Memorias Autoasociativas $\alpha\beta$

Si a una memoria heteroasociativa se le impone la condición de que  $\mathbf{y}^\mu = \mathbf{x}^\mu \forall \mu \in \{1, 2, \dots, p\}$ , entonces deja de ser heteroasociativa y ahora se le denomina memoria *autoasociativa*.

A continuación se enlistan algunas características de las memorias autoasociativas  $\alpha\beta$ :

1. El conjunto fundamental toma la forma  $\{(\mathbf{x}^\mu, \mathbf{x}^\mu) \mid \mu = 1, 2, \dots, p\}$
2. Los patrones fundamentales de entrada y de salida son de la misma dimensión; denotémosla por  $n$ .
3. La memoria es una matriz cuadrada, para ambos tipos,  $\mathbf{V}$  y  $\Lambda$ . Si  $\mathbf{x}^\mu \in A^n$ , entonces  $\mathbf{V} = [\nu_{ij}]_{n \times n}$  y  $\Lambda = [\lambda_{ij}]_{n \times n}$

Al igual como se hizo para las memorias heteroasociativas, se desarrollarán sólo las memorias autoasociativas  $\alpha\beta$  de tipo  $\mathbf{V}$ , ya que las propiedades de las memorias tipo  $\Lambda$  se obtienen por dualidad.

#### Fase de aprendizaje.

**PASO 1** Para cada  $\mu = 1, 2, \dots, p$ , a partir de la pareja  $(\mathbf{x}^\mu, \mathbf{x}^\mu)$  se construye la matriz

$$[\mathbf{x}^\mu \boxtimes (\mathbf{x}^\mu)^t]_{n \times n} \quad (9.12)$$

**PASO 2** Se aplica el operador binario *máximo*  $\bigvee$  a las matrices obtenidas en el paso 1:

$$\mathbf{V} = \bigvee_{\mu=1}^p [\mathbf{x}^\mu \boxtimes (\mathbf{x}^\mu)^t] \quad (9.13)$$

La entrada  $ij$ -ésima de la memoria está dada así:

$$\nu_{ij} = \bigvee_{\mu=1}^p \alpha(x_i^\mu, x_j^\mu) \quad (9.14)$$

Se tiene que  $\nu_{ij} \in B, \forall i \in \{1, 2, \dots, n\}, \forall j \in \{1, 2, \dots, n\}$

Fase de recuperación.

**CASO 1: Patrón fundamental** Se presenta un patrón  $\mathbf{x}^\omega$ , con  $\omega \in \{1, 2, \dots, p\}$ , a la memoria autoasociativa  $\alpha\beta$  tipo **V** y se realiza la operación  $\mathbb{M}_\beta$  :

$$\mathbf{V} \mathbb{M}_\beta \mathbf{x}^\omega \quad (9.15)$$

El resultado de la operación anterior será un vector columna de dimensión  $n$ .

$$(\mathbf{V} \mathbb{M}_\beta \mathbf{x}^\omega)_i = \bigwedge_{j=1}^n \beta(\nu_{ij}, x_j^\omega) \quad (9.16)$$

$$(\mathbf{V} \mathbb{M}_\beta \mathbf{x}^\omega)_i = \bigwedge_{j=1}^n \beta \left\{ \left[ \bigvee_{\mu=1}^p \alpha(x_i^\mu, x_j^\mu) \right], x_j^\omega \right\} \quad (9.17)$$

**CASO 2: Patrón alterado** Se presenta un patrón binario  $\tilde{\mathbf{x}}$  que es un vector columna de dimensión  $n$ , a la memoria autoasociativa  $\alpha\beta$  tipo **V** y se realiza la operación  $\mathbb{M}_\beta$  :

$$\mathbf{V} \mathbb{M}_\beta \tilde{\mathbf{x}} \quad (9.18)$$

Al igual que en el caso 1, el resultado de la operación anterior es un vector columna de dimensión  $n$ , cuya  $i$ -ésima componente se expresa de la siguiente manera:

$$(\mathbf{V} \mathbb{M}_\beta \tilde{\mathbf{x}})_i = \bigwedge_{j=1}^n \beta(\nu_{ij}, \tilde{x}_j) \quad (9.19)$$

$$(\mathbf{V} \mathbb{M}_\beta \tilde{\mathbf{x}})_i = \bigwedge_{j=1}^n \beta \left\{ \left[ \bigvee_{\mu=1}^p \alpha(x_i^\mu, x_j^\mu) \right], \tilde{x}_j \right\} \quad (9.20)$$

■

**Lema 6.** Una memoria autoasociativa  $\alpha\beta$  tipo **V** tiene únicamente unos en su diagonal principal.

**Demostración.-** La  $ij$ -ésima entrada de una memoria autoasociativa  $\alpha\beta$  tipo **V** está dada por

$$\nu_{ij} = \bigvee_{\mu=1}^p \alpha(x_i^\mu, x_j^\mu)$$

Las entradas de la diagonal principal se obtienen de la expresión anterior, haciendo  $i = j$ :

$$\nu_{ii} = \bigvee_{\mu=1}^p \alpha(x_i^\mu, x_i^\mu), \quad \forall i \in \{1, 2, \dots, n\} \quad (9.21)$$

Pero se tiene que

$$\alpha(x_i^\mu, x_i^\mu) = 1$$

por lo que la expresión 9.21 se transforma en:

$$\nu_{ii} = \bigvee_{\mu=1}^p (1) = 1, \quad \forall i \in \{1, 2, \dots, n\}$$

■

**Teorema 5.** Una memoria autoasociativa  $\alpha\beta$  tipo  $V$  recupera de manera perfecta el conjunto fundamental completo; además, tiene máxima capacidad de aprendizaje.

*Demostración.-* Sea  $\omega \in \{1, 2, \dots, p\}$  arbitrario. De acuerdo con el Lema 6, para cada  $i \in \{1, \dots, n\}$  escogida arbitrariamente

$$\nu_{ii} = 1 = \alpha(x_i^\omega, x_i^\omega)$$

Es decir, para  $i \in \{1, \dots, n\}$  escogida arbitrariamente,  $\exists j_0 = i \in \{1, \dots, n\}$  que cumple con:

$$\nu_{ij_0} = \alpha(x_i^\omega, x_{j_0}^\omega)$$

Por lo tanto, de acuerdo con el Teorema 2:

$$V \circlearrowright_{\beta} x^\omega = x^\omega, \quad \forall \omega \in \{1, 2, \dots, p\}$$

Esto significa que la memoria autoasociativa  $\alpha\beta$  tipo  $V$  recupera de manera perfecta el conjunto fundamental completo.

Además, en la demostración de este Teorema, en ningún momento aparece restricción alguna sobre  $p$ , que es la cardinalidad del conjunto fundamental; esto quiere decir que el conjunto fundamental puede crecer tanto como se quiera. La consecuencia directa es que el número de patrones que puede aprender una memoria autoasociativa  $\alpha\beta$  tipo  $V$ , con recuperación perfecta, es máximo.

■

**Teorema 6.** Sea  $\{(x^\mu, x^\mu) \mid \mu = 1, 2, \dots, p\}$  el conjunto fundamental de una memoria autoasociativa  $\alpha\beta$  representada por  $V$ , y sea  $\tilde{x} \in A^n$  un patrón

alterado con ruido aditivo respecto de algún patrón fundamental  $\mathbf{x}^\omega$  con  $\omega \in \{1, 2, \dots, p\}$ . Si se presenta  $\tilde{\mathbf{x}}$  a la memoria  $\mathbf{V}$  como entrada, y si además para cada  $i \in \{1, \dots, n\}$  se cumple la condición de que  $\exists j = j_0 \in \{1, \dots, n\}$ , el cual depende de  $\omega$  y de  $i$  tal que  $\nu_{ij_0} \leq \alpha(x_i^\omega, \tilde{x}_{j_0})$ , entonces la recuperación  $\mathbf{V} \circledast \tilde{\mathbf{x}}$  es perfecta; es decir  $\mathbf{V} \circledast \tilde{\mathbf{x}} = \mathbf{x}^\omega$ .

Demostración.- Por hipótesis se tiene que  $\mathbf{y}^\mu = \mathbf{x}^\mu \forall \mu \in \{1, 2, \dots, p\}$  y, por consiguiente,  $m = n$ . Al establecer estas dos condiciones en el Teorema 4, se obtiene el resultado:  $\mathbf{V} \circledast \tilde{\mathbf{x}} = \mathbf{x}^\omega$ .

■

El Teorema 6 confirma que las memorias autoasociativas  $\alpha\beta$  tipo  $\mathbf{V}$  son inmunes a cierta cantidad de ruido aditivo.

Dado un  $i \in \{1, \dots, n\}$  cualquiera, consideremos las relaciones que hay entre los valores de  $x_i^\omega$  y las coordenadas del vector  $\tilde{\mathbf{x}}$ , con el fin de analizar brevemente cada uno de los casos posibles en la fase de recuperación. Según la hipótesis del Teorema 6 la recuperación del valor  $x_i^\omega$  se garantiza siempre y cuando para este valor  $i$  se pueda encontrar un  $j_0 \in \{1, \dots, n\}$  que cumpla con la desigualdad  $\nu_{ij_0} \leq \alpha(x_i^\omega, \tilde{x}_{j_0})$ . Existen dos casos posibles para el valor de  $x_i^\omega$ :

1. Si  $x_i^\omega = 1$ , es suficiente que alguna de las entradas del patrón  $\tilde{\mathbf{x}}$  sea cero, para garantizar la recuperación del valor  $x_i^\omega$ . Veamos: si existe  $j_0 \in \{1, \dots, n\}$  para el cual  $\tilde{x}_{j_0} = 0$  entonces, de acuerdo con la Tabla 3.1,  $\alpha(x_i^\omega, \tilde{x}_{j_0}) = \alpha(1, 0) = 2$  y esto significa que  $\nu_{ij_0} \leq \alpha(x_i^\omega, \tilde{x}_{j_0})$ , porque el máximo valor posible para  $\nu_{ij_0}$  es precisamente 2, según la misma Tabla.
2. Este caso es más restrictivo. Si  $x_i^\omega = 0$ , no basta con encontrar una entrada cero en el patrón  $\tilde{\mathbf{x}}$ . Al hallar el valor de  $j_0 \in \{1, \dots, n\}$  para el cual  $\tilde{x}_{j_0} = 0$ , se debe pedir como condición adicional que  $\nu_{ij_0} \neq 2$ , porque (según Tabla 3.1)  $\alpha(x_i^\omega, \tilde{x}_{j_0}) = \alpha(0, 0) = 1$  y esto significa que la desigualdad  $\nu_{ij_0} \leq \alpha(x_i^\omega, \tilde{x}_{j_0})$  se da siempre y cuando  $\nu_{ij_0} \neq 2$ . Si se llegase a tener carencia de ceros en las coordenadas del patrón  $\tilde{\mathbf{x}}$ , la condición para recuperar el valor de  $x_i^\omega$ , al tener  $\tilde{x}_{j_0} = 1$ , es más fuerte:  $\nu_{ij_0} = 0$ , porque  $\alpha(0, 1) = 0$ .

Las memorias autoasociativas  $\alpha\beta$  tipo  $\Lambda$  se desarrollan por dualidad, partiendo de los resultados obtenidos para las memorias autoasociativas  $\alpha\beta$

tipo **V**: para ello, se realizan cambios similares a los que se indicaron para las memorias heteroasociativas. También, mientras que las memorias autoasociativas  $\alpha\beta$  tipo **V** son inmunes a cierta cantidad de ruido aditivo pero sensibles a ruido sustractivo, con las memorias autoasociativas  $\alpha\beta$  tipo **A** sucede lo contrario.

### 9.3. Densidad Aritmética

Una colección de operadores lógicos es *funcionalmente completa* si toda proposición compuesta es lógicamente equivalente a una proposición compuesta que involucre sólo a los operadores de la colección (Rosen, 1995).

Es un hecho establecido que los tres operadores lógicos de negación ( $\neg$ ), conjunción ( $\wedge$ ) y disyunción ( $\vee$ ) forman una colección funcionalmente completa de operadores lógicos:

$$\{\neg, \wedge, \vee\}$$

Existen colecciones funcionalmente completas que constan de dos o de un único operador. Uno de estos operadores es el que corresponde a la Tabla de verdad de la disyunción negada: la conectiva lógica *nor*, que denotaremos con el operador  $\downarrow$ .

Sean  $x$  y  $y$  dos variables lógicas booleanas. El operador  $\downarrow$  se define de la siguiente manera:

$$x \downarrow y = \neg(x \vee y) \quad (9.22)$$

Esto significa que la colección  $\{\downarrow\}$  es funcionalmente completa, como lo afirma la Proposición 1.

**Proposición 1.** El operador  $\downarrow$  constituye, por sí mismo, una colección funcionalmente completa  $\{\downarrow\}$ : si  $x$  y  $y$  son variables lógicas booleanas, entonces se cumplen las siguientes equivalencias:

$$\begin{aligned} \neg x &= x \downarrow x \\ x \vee y &= (x \downarrow y) \downarrow (x \downarrow y) \\ x \wedge y &= (x \downarrow x) \downarrow (y \downarrow y) \end{aligned} \quad (9.23)$$

Tanto para las memorias asociativas morfológicas como para las memorias asociativas  $\alpha\beta$  se considera un conjunto fundamental de  $p$  asociaciones, donde los patrones de entrada tienen dimensión  $n$ , y los patrones de salida, dimensión  $m$ .

Al realizar el cálculo del total de operaciones requeridas para ambas fases en las memorias asociativas morfológicas, se llega a los siguientes resultados:

La fase de aprendizaje de una memoria asociativa morfológica requiere de  $28mnp$  operaciones  $\downarrow$  y  $mn(p - 1)$  operaciones de orden.

La fase de recuperación de un patrón de salida en una memoria asociativa morfológica requiere de  $198mn$  operaciones  $\downarrow$  y  $m(n - 1)$  operaciones de orden.

Al realizar el cálculo del total de operaciones requeridas para ambas fases en las memorias asociativas  $\alpha\beta$ , se llega a los siguientes resultados:

La fase de aprendizaje de una memoria asociativa  $\alpha\beta$  requiere de  $28mnp$  operaciones  $\downarrow$  y  $mn(p - 1)$  operaciones de orden.

La fase de recuperación de un patrón de salida en una memoria asociativa  $\alpha\beta$  requiere de  $147mn$  operaciones  $\downarrow$  y  $m(n - 1)$  operaciones de orden.

A diferencia de lo que sucede con las densidades aritméticas de aprendizaje, las cuales son iguales en ambos tipos de memorias asociativas, la densidad aritmética de recuperación es menor para las memorias asociativas  $\alpha\beta$  que la correspondiente a las memorias morfológicas.

**Resultado comparativo de la densidad aritmética:** La fase de recuperación de las memorias asociativas morfológicas requiere de un 34.7% adicional en el número de operaciones lógicas  $\downarrow$ , respecto de lo que requieren las memorias asociativas  $\alpha\beta$ .

## 9.4. Consideraciones finales del capítulo.

Este trabajo tiene como producto un modelo de memoria asociativa que utiliza dos operadores binarios originales que, al combinarlos de ciertas maneras, dan lugar a cuatro operaciones novedosas entre matrices y vectores.

Las memorias asociativas  $\alpha\beta$  tienen al menos una ventaja sobre las morfológicas: la densidad aritmética de las memorias  $\alpha\beta$  es menor. Además, exhiben capacidad máxima de almacenamiento y aprendizaje: la recuperación es perfecta para todo el conjunto fundamental.

Las memorias asociativas  $\alpha\beta$  tipo **V** son robustas a ruido aditivo pero vulnerables ante ruido sustractivo, y las memorias asociativas  $\alpha\beta$  tipo **A** son robustas a ruido sustractivo pero vulnerables ante ruido aditivo.

Las nuevas memorias carecen de problemas de convergencia (son memorias asociativas *one shot*), lo que potencialmente les permite ser más rápidas que las memorias que requieren convergencia.



# Apéndice A

## Pruebas de convergencia.

### A.1. Contexto.

De acuerdo con el capítulo 6, podemos afirmar que la Memoria Asociativa Hopfield es autoasociativa, simétrica, con ceros en la diagonal principal.

En virtud de que la memoria es autoasociativa, el conjunto fundamental para la memoria Hopfield es  $\{(\mathbf{x}^\mu, \mathbf{x}^\mu) \mid \mu = 1, 2, \dots, p\}$  con

$$\mathbf{x}^\mu = \begin{pmatrix} x_1^\mu \\ x_2^\mu \\ \vdots \\ x_n^\mu \end{pmatrix} \in A^n \quad \text{y} \quad A = \{1, -1\}$$

La *fase de aprendizaje* incluye la siguiente regla para obtener la  $ij$ -ésima componente de la memoria Hopfield  $M$ :

$$m_{ij} = \begin{cases} \sum_{\mu=1}^p x_i^\mu x_j^\mu & \text{si } i \neq j \\ 0 & \text{si } i = j \end{cases} \quad (\text{A.1})$$

Operativamente, el resultado de la expresión anterior se puede obtener en tres etapas:



1. Para cada una de las  $p$  asociaciones  $(\mathbf{x}^\mu, \mathbf{x}^\mu)$  se encuentra la matriz  $\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t$  de dimensiones  $n \times n$ , la cual tiene unos en su diagonal principal:

$$\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t = \begin{pmatrix} x_1^\mu \\ x_2^\mu \\ \vdots \\ x_n^\mu \end{pmatrix} \cdot (x_1^\mu, x_2^\mu, \dots, x_n^\mu) = \begin{pmatrix} x_1^\mu x_1^\mu & \cdots & x_1^\mu x_i^\mu & \cdots & x_1^\mu x_n^\mu \\ x_2^\mu x_1^\mu & \cdots & x_2^\mu x_i^\mu & \cdots & x_2^\mu x_n^\mu \\ \vdots & & \vdots & & \vdots \\ x_i^\mu x_1^\mu & \cdots & x_i^\mu x_i^\mu & \cdots & x_i^\mu x_n^\mu \\ \vdots & & \vdots & & \vdots \\ x_n^\mu x_1^\mu & \cdots & x_n^\mu x_i^\mu & \cdots & x_n^\mu x_n^\mu \end{pmatrix}$$

2. A cada una de las  $p$  matrices  $\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t$  se le resta la matriz identidad  $\mathbf{I}$  de dimensiones  $n \times n$ , con el fin de lograr ceros en la diagonal principal:

$$\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t - \mathbf{I} = \begin{pmatrix} 0 & x_1^\mu x_2^\mu & \cdots & x_1^\mu x_i^\mu & \cdots & x_1^\mu x_n^\mu \\ x_2^\mu x_1^\mu & 0 & \cdots & x_2^\mu x_i^\mu & \cdots & x_2^\mu x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ x_i^\mu x_1^\mu & x_i^\mu x_2^\mu & \cdots & 0 & \cdots & x_i^\mu x_n^\mu \\ \vdots & \vdots & & \vdots & & \vdots \\ x_n^\mu x_1^\mu & x_n^\mu x_2^\mu & \cdots & x_n^\mu x_i^\mu & \cdots & 0 \end{pmatrix}$$

3. Se suman la  $p$  matrices  $\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t - \mathbf{I}$  para finalmente obtener la memoria  $\mathbf{M}$ :

$$\mathbf{M} = \sum_{\mu=1}^p [\mathbf{x}^\mu \cdot (\mathbf{x}^\mu)^t - \mathbf{I}] = [m_{ij}]_{n \times n}$$

En la *fase de recuperación* para la memoria Hopfield se le presenta un patrón de entrada  $\tilde{\mathbf{x}}$ , y ésta cambiará su estado con el tiempo, de modo que cada neurona  $x_i$  ajuste su valor de acuerdo con el resultado que arroje la comparación de la cantidad

$$\sum_{j=1}^n m_{ij} x_j$$

con un valor de umbral que es normalmente cero.

Representemos el estado de la memoria Hopfield en el tiempo  $t$  por  $\mathbf{x}(t)$ ; entonces  $x_i(t)$  representa el valor de la neurona  $x_i$  en el tiempo  $t$  y  $x_i(t+1)$  el valor de  $x_i$  en el tiempo siguiente ( $t+1$ ).

Dado un vector columna de entrada  $\tilde{\mathbf{x}}$ , la fase de recuperación consta de tres pasos:

1. Para  $t = 0$ , se hace  $\mathbf{x}(t) = \tilde{\mathbf{x}}$ ; es decir,  $x_i(0) = \tilde{x}_i, \forall i \in \{1, 2, 3, \dots, n\}$
2.  $\forall i \in \{1, 2, 3, \dots, n\}$  se calcula  $x_i(t+1)$  de acuerdo con la condición siguiente:

$$x_i(t+1) = \begin{cases} 1 & \text{si } \sum_{j=1}^n m_{ij}x_j(t) > 0 \\ x_i(t) & \text{si } \sum_{j=1}^n m_{ij}x_j(t) = 0 \\ -1 & \text{si } \sum_{j=1}^n m_{ij}x_j(t) < 0 \end{cases} \quad (\text{A.2})$$

3. Se compara  $x_i(t+1)$  con  $x_i(t) \forall i \in \{1, 2, 3, \dots, n\}$ . Si  $\mathbf{x}(t+1) = \mathbf{x}(t)$  el proceso termina y el vector recuperado es  $\mathbf{x}(0) = \tilde{\mathbf{x}}$ . De otro modo, el proceso continúa de la siguiente manera: los pasos 2 y 3 se iteran tantas veces como sea necesario hasta llegar a un valor  $t = \tau$  para el cual  $x_i(\tau+1) = x_i(\tau) \forall i \in \{1, 2, 3, \dots, n\}$ : el proceso termina y el patrón recuperado es  $\mathbf{x}(\tau)$ .

El proceso de convergencia descrito en el paso 3 de la fase de recuperación, indica que el sistema llega a un punto límite localmente estable en el tiempo  $\tau$ .

La existencia de  $\tau$  está garantizada a través de la demostración que hace Hopfield de que existen puntos límite localmente estables en su modelo de memoria asociativa; para ello, define la energía  $E$  del sistema de la siguiente manera, tomando en cuenta la condición de que  $m_{ii} = 0, \forall i$ :

$$E = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n m_{ij}x_i x_j \quad (\text{A.3})$$

y acto seguido demuestra que  $E$  es una función de  $x_i$  monótona decreciente.

Además de la demostración sugerida por Hopfield, los investigadores McElicee *et. al.* esbozan otro camino para demostrar la convergencia de la expresión A.3, en un importante trabajo donde analizan la capacidad de la memoria asociativa Hopfield (McElicee, Posner, Rodemich, & Venkatesh, 1987).

En este apéndice se desarrolla la sugerencia de Hopfield y también se desarrolla el esbozo de demostración de McElicee *et. al.*

## A.2. Los desarrollos.

### A.2.1. Hopfield

Si la memoria Hopfield  $M$  es simétrica, es decir  $m_{ij} = m_{ji}$ ,  $\forall i \forall j$ , entonces  $E$  es monótona decreciente para el caso bipolar y monótona no creciente para el caso binario.

Consideremos la expresión A.3 para la energía:

$$E = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n m_{ij} x_i x_j$$

Analicemos lo que sucede con  $E$  cuando la coordenada  $k$ -ésima se incrementa en  $\Delta x_k$ , de modo que  $x_k \rightarrow x_k + \Delta x_k$ , para algún  $k \in \{1, 2, \dots, n\}$

$$E + \Delta E = -\frac{1}{2} \left\{ \begin{array}{l} \sum_{i=1}^{k-1} \left[ \begin{array}{l} \sum_{j=1}^{k-1} m_{ij} x_i x_j \\ + m_{ik} x_i (x_k + \Delta x_k) \\ + \sum_{j=k+1}^n m_{ij} x_i x_j \end{array} \right] + \\ \left[ \begin{array}{l} \sum_{j=1}^{k-1} m_{kj} (x_k + \Delta x_k) x_j \\ + m_{kk} (x_k + \Delta x_k) (x_k + \Delta x_k) \\ + \sum_{j=k+1}^n m_{kj} (x_k + \Delta x_k) x_j \end{array} \right] + \\ \sum_{i=k+1}^n \left[ \begin{array}{l} \sum_{j=1}^{k-1} m_{ij} x_i x_j \\ + m_{ik} x_i (x_k + \Delta x_k) \\ + \sum_{j=k+1}^n m_{ij} x_i x_j \end{array} \right] \end{array} \right\}$$

$$E + \Delta E = -\frac{1}{2} \left\{ \begin{array}{l} \sum_{i=1}^{k-1} \left[ \begin{array}{l} \sum_{j=1}^{k-1} m_{ij} x_i x_j + m_{ik} x_i x_k \\ + m_{ik} x_i \Delta x_k \\ + \sum_{j=k+1}^n m_{ij} x_i x_j \end{array} \right] \\ + \sum_{j=1}^{k-1} m_{kj} x_k x_j \\ + \sum_{j=1}^{k-1} m_{kj} \Delta x_k x_j \\ + m_{kk} (x_k + \Delta x_k) (x_k + \Delta x_k) \\ + \sum_{j=k+1}^n m_{kj} x_k x_j + \sum_{j=k+1}^n m_{kj} \Delta x_k x_j \\ + \sum_{i=k+1}^n \left[ \begin{array}{l} \sum_{j=1}^{k-1} m_{ij} x_i x_j + m_{ik} x_i x_k \\ + m_{ik} x_i \Delta x_k + \sum_{j=k+1}^n m_{ij} x_i x_j \end{array} \right] \end{array} \right\}$$

$$E + \Delta E = -\frac{1}{2} \left\{ \begin{aligned} & \sum_{i=1}^{k-1} \left[ \sum_{j=1}^{k-1} m_{ij} x_i x_j + m_{ik} x_i x_k + \sum_{j=k+1}^n m_{ij} x_i x_j \right] \\ & \quad + \Delta x_k \sum_{i=1}^{k-1} m_{ik} x_i \\ & + \left[ \sum_{j=1}^{k-1} m_{kj} x_k x_j + m_{kk} x_k x_k + \sum_{j=k+1}^n m_{kj} x_k x_j \right] \\ & \quad + \Delta x_k \sum_{j=1}^{k-1} m_{kj} x_j + \Delta x_k \sum_{j=k+1}^n m_{kj} x_j \\ & \quad + \Delta x_k m_{kk} (2x_k + \Delta x_k) \\ & + \sum_{i=k+1}^n \left[ \sum_{j=1}^{k-1} m_{ij} x_i x_j + m_{ik} x_i x_k + \sum_{j=k+1}^n m_{ij} x_i x_j \right] \\ & \quad + \Delta x_k \sum_{i=k+1}^n m_{ik} x_i \end{aligned} \right\}$$

$$E + \Delta E = -\frac{1}{2} \left\{ \begin{aligned} & \sum_{i=1}^{k-1} \sum_{j=1}^n m_{ij} x_i x_j + \Delta x_k \sum_{i=1}^{k-1} m_{ik} x_i \\ & \quad + \sum_{j=1}^n m_{kj} x_k x_j + \Delta x_k \sum_{j=1}^{k-1} m_{kj} x_j \\ & + \Delta x_k \sum_{j=k+1}^n m_{kj} x_j + \Delta x_k m_{kk} (2x_k + \Delta x_k) \\ & \quad + \sum_{i=k+1}^n \sum_{j=1}^n m_{ij} x_i x_j + \Delta x_k \sum_{i=k+1}^n m_{ik} x_i \end{aligned} \right\}$$

$$E + \Delta E = -\frac{1}{2} \left\{ \begin{aligned} & \sum_{j=1}^n \sum_{i=1}^{k-1} m_{ij} x_i x_j + \sum_{j=1}^n m_{kj} x_k x_j + \sum_{j=1}^n \sum_{i=k+1}^n m_{ij} x_i x_j \\ & + \Delta x_k \sum_{i=1}^{k-1} m_{ik} x_i + \Delta x_k \sum_{j=1}^{k-1} m_{kj} x_j + \Delta x_k \sum_{j=k+1}^n m_{kj} x_j \\ & \quad + \Delta x_k m_{kk} (2x_k + \Delta x_k) + \Delta x_k \sum_{i=k+1}^n m_{ik} x_i \end{aligned} \right\}$$

$$E + \Delta E = -\frac{1}{2} \left\{ \begin{array}{l} \sum_{j=1}^n \left( \sum_{i=1}^{k-1} m_{ij} x_i x_j + m_{kj} x_k x_j + \sum_{i=k+1}^n m_{ij} x_i x_j \right) \\ + \Delta x_k \left[ \begin{array}{l} \sum_{i=1}^{k-1} m_{ik} x_i + \sum_{j=1}^{k-1} m_{kj} x_j \\ + \sum_{j=k+1}^n m_{kj} x_j + m_{kk} (2x_k + \Delta x_k) \\ + \sum_{i=k+1}^n m_{ik} x_i \end{array} \right] \end{array} \right\}$$

$$E + \Delta E = -\frac{1}{2} \left\{ \begin{array}{l} \sum_{j=1}^n \sum_{i=1}^n m_{ij} x_i x_j \\ + \Delta x_k \left[ \begin{array}{l} \left( \sum_{i=1}^{k-1} m_{ik} x_i + m_{kk} x_k + \sum_{i=k+1}^n m_{ik} x_i \right) \\ + \left( \sum_{j=1}^{k-1} m_{kj} x_j + m_{kk} x_k + \sum_{j=k+1}^n m_{kj} x_j \right) \\ + m_{kk} \Delta x_k \end{array} \right] \end{array} \right\}$$

$$E + \Delta E = -\frac{1}{2} \sum_{j=1}^n \sum_{i=1}^n m_{ij} x_i x_j - \frac{1}{2} \Delta x_k \left[ \sum_{i=1}^n m_{ik} x_i + \sum_{j=1}^n m_{kj} x_j + m_{kk} \Delta x_k \right]$$

Por la expresión A.1 se tiene que  $m_{kk} = 0$ , y por ello  $m_{kk} \Delta x_k = 0$ ; además, de acuerdo con la expresión A.3:  $-\frac{1}{2} \sum_{j=1}^n \sum_{i=1}^n m_{ij} x_i x_j = E$

$$E + \Delta E = E - \frac{1}{2} \Delta x_k \left[ \sum_{i=1}^n m_{ik} x_i + \sum_{j=1}^n m_{kj} x_j \right]$$

Es decir:

$$\Delta E = -\frac{1}{2} \Delta x_k \left[ \sum_{j=1}^n m_{jk} x_j + \sum_{j=1}^n m_{kj} x_j \right]$$

Por hipótesis la memoria es simétrica:  $m_{jk} = m_{kj}$

$$\Delta E = -\frac{1}{2}\Delta x_k \left[ \sum_{j=1}^n m_{kj}x_j + \sum_{j=1}^n m_{kj}x_j \right]$$

$$\Delta E = -\frac{1}{2}\Delta x_k \left( 2 \sum_{j=1}^n m_{kj}x_j \right)$$

$$\Delta E = -\Delta x_k \sum_{j=1}^n m_{kj}x_j$$

Para el caso bipolar, por la expresión A.2 se puede afirmar que  $\Delta x_k$  y  $\sum_{j=1}^n m_{kj}x_j$  tienen signos iguales, por lo que  $\Delta E$  es negativa.

Para el caso binario,  $\Delta x_k$  y  $\sum_{j=1}^n m_{kj}x_j$  no pueden tener signos contrarios.

■

### A.2.2. McEliece et. al.

Se mostrará que el producto interno o correlación

$$C = \mathbf{M} \cdot \mathbf{x} = \sum_{i=1}^n \sum_{j=1}^n m_{ij}x_i x_j \quad (\text{A.4})$$

es no decreciente cuando el estado  $\mathbf{x}$  se mueve a través de una trayectoria del modelo.

La coordenada  $k$ -ésima del vector de prueba  $\mathbf{x}$  cambiará y todas las demás permanecerán sin cambio alguno; es decir,  $x_i(t+1) = x_i(t)$ ,  $\forall i \neq k$ . Entonces, de acuerdo con el algoritmo Hopfield:

$$x_k(t+1) = \begin{cases} 1 & \text{si } \sum_{j=1}^n m_{kj}x_j(t) > 0 \\ x_i(t) & \text{si } \sum_{j=1}^n m_{kj}x_j(t) = 0 \\ -1 & \text{si } \sum_{j=1}^n m_{kj}x_j(t) < 0 \end{cases} \quad (\text{A.5})$$

Sea  $\Delta x_k = x_k(t+1) - x_k(t)$ , esto significa que  $x_k(t+1) = x_k(t) + \Delta x_k$ .  
 Hallemos una expresión para el cambio  $\Delta C = C(t+1) - C(t)$ .

$$\begin{aligned}
 C(t) &= \sum_{i=1}^n \sum_{j=1}^n m_{ij} x_i(t) x_j(t) \\
 C(t+1) &= \sum_{i=1}^{k-1} \left[ \sum_{j=1}^{k-1} m_{ij} x_i(t) x_j(t) + m_{ik} x_i(t) x_k(t+1) \right. \\
 &\quad \left. + \sum_{j=k+1}^n m_{ij} x_i(t) x_j(t) \right] \\
 &\quad + \left[ \sum_{j=1}^{k-1} m_{kj} x_k(t+1) x_j(t) + m_{kk} x_k(t+1) x_k(t+1) \right. \\
 &\quad \left. + \sum_{j=k+1}^n m_{kj} x_k(t+1) x_j(t) \right] \\
 &\quad + \sum_{i=k+1}^n \left[ \sum_{j=1}^{k-1} m_{ij} x_i(t) x_j(t) + m_{ik} x_i(t) x_k(t+1) \right. \\
 &\quad \left. + \sum_{j=k+1}^n m_{ij} x_i(t) x_j(t) \right] \\
 C(t+1) &= \sum_{i=1}^{k-1} \left[ \sum_{j=1}^{k-1} m_{ij} x_i(t) x_j(t) + m_{ik} x_i(t) [x_k(t) + \Delta x_k] \right. \\
 &\quad \left. + \sum_{j=k+1}^n m_{ij} x_i(t) x_j(t) \right] \\
 &\quad + \left[ \sum_{j=1}^{k-1} m_{kj} [x_k(t) + \Delta x_k] x_j(t) \right. \\
 &\quad \left. + m_{kk} [x_k(t) + \Delta x_k] [x_k(t) + \Delta x_k] \right. \\
 &\quad \left. + \sum_{j=k+1}^n m_{kj} [x_k(t) + \Delta x_k] x_j(t) \right] \\
 &\quad + \sum_{i=k+1}^n \left[ \sum_{j=1}^{k-1} m_{ij} x_i(t) x_j(t) + m_{ik} x_i(t) [x_k(t) + \Delta x_k] \right. \\
 &\quad \left. + \sum_{j=k+1}^n m_{ij} x_i(t) x_j(t) \right]
 \end{aligned}$$



$$\begin{aligned}
C(t+1) = & \sum_{i=1}^{k-1} \left[ \begin{array}{l} \sum_{j=1}^{k-1} m_{ij} x_i(t) x_j(t) \\ + m_{ik} x_i(t) x_k(t) \\ + \sum_{j=k+1}^n m_{ij} x_i(t) x_j(t) \end{array} \right] + \sum_{i=1}^{k-1} m_{ik} x_i(t) (\Delta x_k) \\
& + \left[ \begin{array}{l} \sum_{j=1}^{k-1} m_{kj} x_k(t) x_j(t) \\ + m_{kk} x_k(t) x_k(t) \\ + \sum_{j=k+1}^n m_{kj} x_k(t) x_j(t) \end{array} \right] + \sum_{j=1}^{k-1} m_{kj} (\Delta x_k) x_j(t) \\
& + 2m_{kk} x_k(t) \Delta x_k + m_{kk} (\Delta x_k)^2 + \sum_{j=k+1}^n m_{kj} (\Delta x_k) x_j(t) \\
& + \sum_{i=k+1}^n \left[ \begin{array}{l} \sum_{j=1}^{k-1} m_{ij} x_i(t) x_j(t) \\ + m_{ik} x_i(t) x_k(t) \\ + \sum_{j=k+1}^n m_{ij} x_i(t) x_j(t) \end{array} \right] + \sum_{i=k+1}^n m_{ik} x_i(t) (\Delta x_k)
\end{aligned}$$

$$\begin{aligned}
C(t+1) = & \sum_{i=1}^{k-1} \left[ \sum_{j=1}^n m_{ij} x_i(t) x_j(t) \right] + \left[ \sum_{j=1}^n m_{kj} x_k(t) x_j(t) \right] \\
& + \sum_{i=k+1}^n \left[ \sum_{j=1}^n m_{ij} x_i(t) x_j(t) \right] \\
& + \sum_{j=1}^{k-1} m_{kj} (\Delta x_k) x_j(t) + m_{kk} (\Delta x_k) x_k(t) \\
& + \sum_{j=k+1}^n m_{kj} (\Delta x_k) x_j(t) \\
& + \sum_{i=1}^{k-1} m_{ik} x_i(t) (\Delta x_k) + m_{kk} x_k(t) (\Delta x_k) \\
& + \sum_{i=k+1}^n m_{ik} x_i(t) (\Delta x_k) + m_{kk} (\Delta x_k)^2
\end{aligned}$$

$$\begin{aligned}
C(t+1) &= \sum_{i=1}^n \sum_{j=1}^n m_{ij} x_i(t) x_j(t) \\
&\quad + \sum_{j=1}^n m_{kj} (\Delta x_k) x_j \\
&\quad (t) + \sum_{i=1}^n m_{ik} x_i(t) (\Delta x_k) + m_{kk} (\Delta x_k)^2
\end{aligned}$$

$$\begin{aligned}
\Delta C &= C(t+1) - C(t) = \sum_{i=1}^n \sum_{j=1}^n m_{ij} x_i(t) x_j(t) + \sum_{j=1}^n m_{kj} (\Delta x_k) x_j(t) \\
&\quad + \sum_{i=1}^n m_{ik} x_i(t) (\Delta x_k) + m_{kk} (\Delta x_k)^2 - \sum_{i=1}^n \sum_{i=1}^n m_{ij} x_i(t) x_j(t)
\end{aligned}$$

$$\Delta C = \sum_{j=1}^n m_{kj} (\Delta x_k) x_j(t) + \sum_{i=1}^n m_{ik} x_i(t) (\Delta x_k) + m_{kk} (\Delta x_k)^2$$

No se exige que  $m_{kk} = 0$ ; basta con que  $m_{kk} \geq 0$ ,  $1 \leq k \leq n$ .

Entonces

$$\Delta C \geq \sum_{j=1}^n m_{kj} (\Delta x_k) x_j(t) + \sum_{i=1}^n m_{ik} x_i(t) (\Delta x_k)$$

Y por simetría de  $M$ , se tiene que  $m_{kj} = m_{jk}$ ; y con ello

$$\Delta C \geq \sum_{j=1}^n m_{kj} (\Delta x_k) x_j(t) + \sum_{i=1}^n m_{ki} (\Delta x_k) x_i(t) = 2\Delta x_k \sum_{j=1}^n m_{kj} x_j(t)$$

$$\Delta C \geq 2\Delta x_k \sum_{j=1}^n m_{kj} x_j(t)$$

Caso 1:  $x_k(t+1) = x_k(t)$ . entonces  $\Delta x_k = 0$  y claramente  $\Delta C \geq 0$ .

Caso 2:  $x_k(t+1) < x_k(t)$ . entonces  $x_k(t) = +1$  y  $x_k(t+1) = -1$ . De acuerdo con la expresión A.5 se tiene que  $\sum_{j=1}^n m_{kj}x_j(t) < 0$ : por otro lado,  $\Delta x_k = x_k(t+1) - x_k(t) = -1 - 1 = -2$  y esto significa que  $\Delta C \geq 2\Delta x_k \sum_{j=1}^n m_{kj}x_j(t) = 2(-2) \sum_{j=1}^n m_{kj}x_j(t) > 0$ .

Caso 3:  $x_k(t+1) > x_k(t)$ . entonces  $x_k(t) = -1$  y  $x_k(t+1) = +1$ . De acuerdo con la expresión A.5 se tiene que  $\sum_{j=1}^n m_{kj}x_j(t) > 0$ . Por otro lado,  $\Delta x_k = x_k(t+1) - x_k(t) = 1 - (-1) = +2$  y esto significa que  $\Delta C \geq 2\Delta x_k \sum_{j=1}^n m_{kj}x_j(t) = 2(2) \sum_{j=1}^n m_{kj}x_j(t) > 0$ .

■

### A.3. Consideraciones finales del apéndice.

En este apéndice hemos presentado los desarrollos de dos demostraciones para la convergencia de la Memoria Hopfield, que es uno de los trabajos clásicos en memorias asociativas. Su autor, el físico norteamericano John Hopfield es responsable, según voces autorizadas, de haber revivido el interés, a principios de los ochenta del siglo XX, en las redes neuronales y en particular en las memorias asociativas, después de un período de estancamiento de más de una década.

Uno de los desarrollos parte de la sugerencia del propio Hopfield, y el otro, de un importante artículo: (McElice, Posner, Rodemich, & Venkatesh, 1987).

# Bibliografía

- [1] Abu-Mostafa, Y. & St. Jacques. J. (1985). Information capacity of the Hopfield model, *IEEE Transactions on Information Theory*, IT-31, 4, 461-464.
- [2] Adcodato P. J. L. & Taylor J. G. (1996). Autoassociative memory with high storage capacity, In C. von der Malsburg, W. von Seelen. J. C. Vorbrueggen & B. Sendhoff (Eds.), *Lecture Notes in Computer Science*, 1112, (pp. 29-34),. Bochum, Germany: ICANN'96.
- [3] Aleksander. I. & Morton. H. B. (1997). Weightless and other memory-based networks, In Emile Fiesler (Ed.), *Handbook of Neural Computation*, (pp. C1.5:1-C1.5:15). New York: Oxford.
- [4] Amari. S. (1972). Learning patterns and pattern sequences by self-organizing nets of threshold elements, *IEEE Transactions on Computers*, C-21, 11, 1197-1206.
- [5] Amari, S. (1977). Neural theory of association and concept-formation, *Biological Cybernetics*. 26, 175-185.
- [6] Anderson. J. A. (1972). A simple neural network generating an interactive memory, *Mathematical Biosciences*, 14, 197-220.
- [7] Anderson, J. A. & Rosenfeld, E. (Eds.) (1990). *Neurocomputing: Foundations of Research*. Cambridge: MIT Press.
- [8] Anderson, J. A., Silverstein, J., Ritz, S. & Jones, R. (1977). Distinctive features, categorical perception, and probability learning: some applications of a neural model, *Psychological Review*. 84, 413-451.
- [9] Anderson. J. R. & Bower, G. (1977). *Memoria Asociativa*, México: Limusa.

- [10] Austin, J. (1987). ADAM: A Distributed Associative Memory for Scene Analysis. In *Proceedings of First International Conference on Neural Networks*. (pp. 285-295). San Diego: Ed. M Caudhill and C Butler.
- [11] Austin, J., Buckle, S., Kennedy, J., Moulds, A., Pack, R. & Turner, A. (1997). The cellular neural network associative processor, In A. Krikelis & C. C. Weems (Eds.). *Associative Processing and Processors*, (pp. 284-306). Los Alamitos: IEEE Computer Society.
- [12] Bandyopadhyay, S. & Datta, A. K. (1996). A novel neural hetero-associative memory model for pattern recognition. *Pattern Recognition*. 29, 5, 789-795.
- [13] Bosch, H. & Kurfess, F. J. (1998). Information storage capacity of incompletely connected associative memories. *Neural Networks (11)*. 5, 869-876.
- [14] Brown, C. & Aleksander, I. (1996). Learned Probabilistic Prediction in a Weightless Neural Network. London Imperial College Technical Report NSE96\_03.
- [15] Buhmann, J. (1995). Oscillatory associative memories, In M. Arbib (Ed.), *Handbook of Brain Theory & Neural Networks*, (pp. 691-694). Cambridge: MIT Press.
- [16] Chen, C. & Honavar, V. (1995). A neural architecture for content as well as address-based storage and recall: theory and applications, Iowa State University Technical Report TR95-03.
- [17] Díaz-de-León Santiago, J.L. & Yáñez Márquez, C. (1999). Cálculo de momentos morfológicos basado en transformada de distancia, En *Selected Works 1997-1998* (pp. 111-117), ISBN 970-18-3427-5, CIC-IPN, México.
- [18] Díaz de León Santiago, J.L. & Yáñez Márquez, C. (1999). *Generalización del concepto de imagen*. IT-20, Serie Azul, ISBN 970-18-2675-2, CIC-IPN, México.
- [19] Díaz de León Santiago, J.L. & Yáñez Márquez, C. (1999). *Estructuras discretas bajo el concepto generalizado de imagen*. IT-21. Serie Azul, ISBN 970-18-2676-0, CIC-IPN, México.

- [20] Díaz-de-León. J. L. & Yáñez, C. (1999). Memorias Asociativas con Respuesta Perfecta y Capacidad Infinita, *Memoria del TAINA '99*. México, D.F.. 23-38.
- [21] Díaz-de-León Santiago, J.L. Yáñez Márquez, C. & Mosso, J.L. (2000). Telemedicina, *Revista Academia*. No.26, Secretaría Académica del Instituto Politécnico Nacional, México.
- [22] Díaz-de-León Santiago, J.L. & Yáñez-Márquez, C. (2001). *Correlograph de Willshaw, Buneman & Longuet-Higgins*, IT-49. Serie Verde, ISBN 970-18-6689-4. CIC-IPN, México.
- [23] Díaz-de-León Santiago, J.L. & Yáñez-Márquez, C. (2001). *Algunas Aportaciones de S. Amari en Memorias Asociativas*. IT-51, Serie Verde ISBN 970-18-6691-6, CIC-IPN, México.
- [24] Díaz-de-León Santiago, J.L. & Yáñez-Márquez, C. (2001). *Pruebas de Convergencia de la Memoria Hopfield*, IT-53, Serie Verde, ISBN 970-18-6693-2, CIC-IPN, México.
- [25] Díaz-de-León Santiago, J.L. & Yáñez-Márquez, C. (2001). *Model BAM*, IT-55, Serie Verde, ISBN 970-18-6695-9, CIC-IPN, México.
- [26] Díaz-de-León Santiago, J.L. & Yáñez-Márquez. C. (2001). *Memorias Morfológicas Heteroasociativas*. IT-57, Serie Verde, ISBN 970-18-6697 5, CIC-IPN, México.
- [27] Díaz-de-León Santiago, J.L. & Yáñez Márquez, C. (Eds.) (2002) *Reconocimiento de Patrones. Avances y Perspectivas*. Colección RESEARCH ON COMPUTING SCIENCE, Vol. 1, ISBN 970189476-6, CIC-IPN, México.
- [28] Díaz-de-León Santiago, J.L. & Yáñez Márquez, C. (Obra a publicarse en 2003). *Introducción a la morfología matemática de conjuntos*, Colección de Ciencia de la Computación, CIC-IPN-UNAM-FCE, México.
- [29] Flynn, M., Kanerva, P. & Bhadkamkar, N. (1989). Sparse distributed memory: principles and operation, Stanford University Technical Report CSL-TR-89-400.
- [30] Gori, M., Lastrucci, L. & Soda, G. (1996). Autoassociator-based models for speaker verification, *Pattern Recognition Letters*, 17. 3, 241-250.

- [31] Graham, B. & Willshaw, D. (1995). Improving recall from an associative memory. *Biological Cybernetics*, 72, 337-346.
- [32] Haralick, R. M., Sternberg, S. R. & Zhuang, X. (1987). Image analysis using mathematical morphology, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9, 4, 532-550.
- [33] Hassoun, M. H. (Ed.) (1993). *Associative Neural Memories*, New York: Oxford University Press.
- [34] Hassoun, M. H. (1995). *Fundamentals of Artificial Neural Networks*. Cambridge: MIT Press.
- [35] Hassoun, M. H. & Watta, P. B. (1997). Associative memory networks, In Emile Fiesler (Ed.), *Handbook of Neural Computation*, (pp. C1.3:1-C1.3:14). New York: Oxford.
- [36] Hattori, M. & Hagiwara, M. (2000). Associative memory for intelligent control, *Mathematics and Computers in Simulation*, 51, 349-374.
- [37] Hely, T. A., Willshaw, D. J. & Hayes, G. M. (1997). A new approach to Kanerva's sparse distributed memory, *IEEE Transactions on Neural Networks*, 8, 3, 791-794.
- [38] Herrmann, F. P. & Sodini, C. G. (1992). A dynamic associative processor for machine vision applications. *IEEE Micro*, 12, 3, 31-41.
- [39] Hopfield, J.J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79, 2554-2558.
- [40] Hopfield, J.J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons, *Proceedings of the National Academy of Sciences*, 81, 3088-3092.
- [41] Imada, A. & Araki K. (1995). Genetic algorithm enlarges the capacity of associative memory. In L.J.Eshelman(Ed.), *Proceedings of 6th International Conference on Genetic Algorithms*, (pp 413-420). San Francisco: Morgan Kaufmann.
- [42] Jagota, A., Narasimhan, G. & Regan, K. W. (1998). Information Capacity of Binary Weights Associative Memories. *Neurocomputing*, 19(1-3), 35-38.

- [43] Jorgensen, T. M. (1997). Classification of handwritten digits using a RAM neural net architecture, *International Journal of Neural Systems*, 8, 1, 17-26.
- [44] Kanerva, P. (1988). *Sparse Distributed Memory*, Cambridge: MIT Press.
- [45] Kennedy, J. V., Austin, J. & Cass, B. (1995). A hardware implementation of a binary neural image processor, *Proceedings of the IEE Conference on Image Processing and its Applications*, Edinburgh, UK.
- [46] Kinsler, J. M. (1995). Fast analog associative memory. *Proceedings of the SPIE*, 2568, 290-293.
- [47] Kohonen, T. (1972). Correlation matrix memories, *IEEE Transactions on Computers*, C-21, 4, 353-359.
- [48] Kohonen, T. (1974). An adaptive associative memory principle, *IEEE Transactions on Computers*, C-24, 4, 444-445.
- [49] Kohonen, T. (1987). *Content-Addressable Memories*, Berlin: Springer-Verlag.
- [50] Kohonen, T. (1989). *Self-Organization and Associative Memory*. Berlin: Springer-Verlag.
- [51] Kohonen, T. (1997). *Self-Organizing Maps*, Berlin: Springer.
- [52] Kohonen, T. & Ruohonen, M. (1973). Representation of associated data by matrix operators, *IEEE Transactions on Computers*, C-22, 701-702.
- [53] Kolen, J. F., & Pollack, J. B. (1991). Multiassociative memory. *The Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*. 785-789.
- [54] Kosko, B. (1988). Bidirectional associative memories, *IEEE Transactions on Systems, Man, and Cybernetics*, 18, 1, 49-60.
- [55] Kosko, B. (1992). *Neural Networks and Fuzzy Systems*, Englewood Cliffs: Prentice Hall.
- [56] Krikelis, A. & Weems, C. C. (1997). Associative Processing and Processors, In A. Krikelis & C. C. Weems (Eds.), *Associative Processing and Processors*, (pp. 2-9). Los Alamitos: IEEE Computer Society.



- [57] Little. W. & Shaw. G. (1975). A statistical theory of short and long term memory, *Behavioral Biology*, 14, 115-133.
- [58] Ludermir, T. B., Carvalho. A., Braga, A. & Souto, M. C. P. (1999). Weightless neural models: a review of current and past works, *Neural Computing Surveys*, 2. 41-61.
- [59] McCulloch. W. & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity, *Bulletin of Mathematical Biophysics*, 5, 115-133.
- [60] McElicecc. R., Posner, E., Rodemich, E. & Venkatesh, S. (1987). The capacity of the Hopfield associative memory, *IEEE Transactions on Information Theory*, IT-33. 4, 461-482.
- [61] Minsky, M. & Papert, S. (1988). *Perceptrons*. Cambridge: MIT Press.
- [62] Moore. J. (1968). *Elements of linear algebra and matrix theory*, New York: McGraw-Hill.
- [63] Nakano, K. (1972). Associatron-A model of associative memory. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-2, 3, 380-388.
- [64] Palm, G., Schwenker. F., Sommer F. T. & Strey, A. (1997). Neural associative memories, In A. Krikelis & C. C. Weems (Eds.), *Associative Processing and Processors*, (pp. 307-326). Los Alamitos: IEEE Computer Society.
- [65] Ritter. G. X., Sussner, P. & Diaz-de-Leon, J. L. (1998). Morphological associative memories, *IEEE Transactions on Neural Networks*, 9, 281-293.
- [66] Ritter. G. X., Diaz-de-Leon, J. L. & Sussner, P. (1999). Morphological bidirectional associative memories, *Neural Networks*. 12, 851-867.
- [67] Rosen, K. H. (1995) *Discrete Mathematics and its Applications*. New York: McGraw-Hill.
- [68] Sánchez Garfías. F.A., Díaz-de-León Santiago, J.L. & Yáñez Márquez, C. (2002). Lernmatrix de Steinbuch: condiciones necesarias y suficientes para recuperación perfecta de patrones. En Díaz-de-León Santiago, J.L. & Yáñez Márquez. C. (Eds.) *Reconocimiento de Patrones. Avances y Perspectivas* (pp. 437-448). Colección RESEARCH ON COMPUTING SCIENCE, Vol. 1. ISBN 970189476-6, CIC-IPN, México.

- Santiago Montero, R., Díaz-de-León Santiago, J.L. & Yáñez Márquez, C. (2002). Clasificador híbrido de patrones basado en la Lernmatrix de Steinbuch y el Linear Associator de Anderson-Kohonen, En Díaz-de-León Santiago, J.L. & Yáñez Márquez, C. (Eds.) *Reconocimiento de Patrones. Avances y Perspectivas* (pp. 449-460). Colección RESEARCH ON COMPUTING SCIENCE, Vol. 1, ISBN 970189476-6, CIC-IPN, México.
- Schwenk, H. & Milgram, M. (1995). Transformation invariant autoassociation with application to handwritten character recognition, In D. S. Touretzky, G. S. Tesauro & T. K. Leen (Eds.). *Neural Information Processing Systems*, (pp. 991-998). Cambridge: MIT Press.
- Serra, J. (1992). *Image Analysis and Mathematical Morphology, Volume 2: Theoretical Advances*, London: Academic Press.
- Silver, S., Glover, R. & Stonham, T. (1996). Associative memory neural networks for time series prediction, *Proceedings of the Third IEEE ICECS*, 651-654.
- Simpson, P. K. (1990). *Artificial Neural Systems*, New York: Pergamon Press.
- 4] Sommer, F. T. & Palm, G. (1998). Bidirectional retrieval from associative memory, In M. I. Jordan, M. J. Kearns, & S. A. Solla (Eds.), *Advances in Neural Information Processing Systems 10*, (pp. 675-681). Cambridge: MIT Press.
- 5] Sossa-Azucla, J.H., Yáñez-Márquez, C. & Díaz-de-León S., J.L. (1999). *Geometric moments computation based on morphological operations*, IT-23, Serie Azul, ISBN 970-18-2616-6, CIC-IPN, México.
- [76] Sossa-Azucla, J.H., C. & Díaz-de-León S., J.L. (2000). Mathematical Morphology based on Linear Combined Metric Spaces on  $\mathbb{Z}^2$  (Part I): Fast Distance Transforms. *Journal of Mathematical Imaging and Vision*, 12, 137-154.
- 77] Sossa-Azucla, J.H., C. & Díaz-de-León S., J.L. (2000). Mathematical Morphology based on Linear Combined Metric Spaces on  $\mathbb{Z}^2$  (Part I): Constant Time Morphological Operations, *Journal of Mathematical Imaging and Vision*, 12, 155-168.

- [78] Sossa-Azucla, J.H., Yáñez-Márquez, C. & Díaz-de-León S., J.L. (2001). Computing geometric moments using morphological erosions. *Pattern Recognition*, 34, 2, 271-276.
- [79] Steinbuch, K. (1961). Die Lernmatrix, *Kybernetik*, 1, 1, 36-45.
- [80] Steinbuch, K. & Frank, H. (1961). Nichtdigitale Lernmatrizen als Perzeptoren, *Kybernetik*, 1, 3, 117-124.
- [81] Storkey, A. J. (1997). Increasing the capacity of a Hopfield network without sacrificing functionality. *International Conference on Artificial Neural Networks*, 451-456.
- [82] Stright, J. R., Coffield, P. C. & Brooks, G. W. (1998). Analog VLSI implementation of a morphological associative memory. *Proceedings of the SPIE*, 3452-03, 14-22.
- [83] Turner, M. & Austin, J. (1997). Matching performance of binary correlation matrix memories, *Neural Networks*, 10, 9, 1637-1648.
- [84] Villanueva, N. & Figueroa, J. (1998). Superposición de información en memorias aleatorias: análisis cualitativo, *Memorias del Simposium Internacional de Computación CIC'98*, 559-564.
- [85] Willshaw, D., Buneman, O. & Longuet-Higgins, H. (1969). Non-holographic associative memory. *Nature*, 222, 960-962.
- [86] Yáñez, C. & Díaz-de-León, J. L. (1999). Nuevas Memorias Asociativas Basadas en Operadores Binarios. *Memoria del TAINA '99, México, D.F.*, 245-257.
- [87] Yáñez-Márquez, C. & Díaz-de-León Santiago, J.L. (2001). *Lernmatrix de Steinbuch*, IT-48. Serie Verde, ISBN 970-18-6688-6, CIC-IPN, México.
- [88] Yáñez-Márquez, C. & Díaz-de-León Santiago, J.L. (2001). *Linear Associator de Anderson-Kohonen*, IT-50, Serie Verde, ISBN 970-18-6690-8. CIC-IPN, México.
- [89] Yáñez-Márquez, C. & Díaz-de-León Santiago, J.L. (2001). *Memoria Asociativa Hopfield*, IT-52, Serie Verde, ISBN 970-18-6692-4, CIC-IPN, México.
- [90] Yáñez-Márquez, C. & Díaz-de-León Santiago, J.L. (2001). *Modelo ADAM*, IT-54, Serie Verde, ISBN 970-18-6694-0, CIC-IPN, México.

- [91] Yáñez-Márquez, C. & Díaz-de-Lcón Santiago, J.L. (2001). *Modelo SDM*, IT-56, Serie Verde, ISBN 970-18-6696-7, CIC-IPN, México.
- [92] Yáñez-Márquez, C. & Díaz-de-Lcón Santiago, J.L. (2001). *Memorias Morfológicas Autoasociativas*, IT-58, Serie Verde, ISBN 970-18-6698-3, CIC-IPN, México.
- [93] Yáñez-Márquez, C. (2002). *Memorias Asociativas basadas en Relaciones de Orden y Operadores Binarios*. Tesis doctoral. CIC-IPN, México.
- [94] Zboril, F. (1997). An application of the sparse distributed memory, *Proceedings of the ASIS 97*, 127-132.

**Impreso en los Talleres Gráficos  
de la Dirección de Publicaciones del  
Instituto Politécnico Nacional  
Tresguerras 27, Centro Histórico, México, DF  
Noviembre 2003. Edición: 500 ejemplares.**

**Esta edición ha sido coordinada por:  
Luis Hernández Lara**





# Introducción a las Memorias Asociativas

El propósito de los autores al presentar esta obra, es poner en manos del amable lector un trabajo conciso, serio y profesional que cubre los principales hechos históricos y los modelos más importantes de memorias asociativas que han surgido, a nivel mundial, desde la década de los sesenta. Este libro será de interés para investigadores y estudiantes de licenciatura, maestría y doctorado cuyo trabajo involucre alguno de los tópicos de memorias asociativas.



**Dr. Cornelio Yáñez Márquez.** Nacionalidad Mexicana. Licenciado en Física y Matemáticas (1989) por la ESFM-IPN. Grados de maestro en ciencias (1995) en ingeniería de cómputo y de doctor en ciencias de la computación (2002), obtenidos en el CIC-IPN. Actualmente, es profesor investigador titular C del CIC-IPN y miembro fundador del Grupo de Robótica y Análisis de Imágenes. Mención honorífica en el examen de grado de doctorado y Presea Lázaro Cárdenas 2002, recibida de manos del C. Presidente de la República. Miembro del Sistema Nacional de Investigadores. Nombramiento de *asociado honorario* de la Asociación Boliviana para el Avance de la Ciencia (ABAC),

perteneciente a la Academia Nacional de Ciencias de Bolivia. Otorgamiento oficial de *testimonio de bienvenida* y entrega de las llaves de la Ciudad de la Provincia de Cercado Cochabamba, Bolivia por su trayectoria profesional, en mayo de 2003.

Áreas de Interés: Memorias Asociativas, Redes Neuronales, Morfología Matemática, Análisis de Imágenes y Robótica Móvil.

cyanez@cic.ipn.mx



**Dr. Juan Luis Díaz de León Santiago.** Nacionalidad Mexicana. Grados de maestro en ciencias (1993) en control automático y de doctor en ciencias (1996) en morfología matemática, obtenidos en el CINVESTAV-IPN. Investigador invitado (1996) y profesor visitante (1997) de la Universidad de Florida. Director de Tecnología (2000-2001) del Sistema Nacional de Seguridad Pública. Actualmente, es Director e investigador titular C del CIC-IPN y Director Fundador del Grupo de Robótica y Análisis de Imágenes (GRAI). Miembro del Sistema Nacional de Investigadores y Premio Nacional de Investigación y Desarrollo Tecnológico de Excelencia *Luis Enrique Erro*.

Nombramiento de *asociado honorario* de la Asociación Boliviana para el Avance de la Ciencia (ABAC), perteneciente a la Academia Nacional de Ciencias de Bolivia. Otorgamiento oficial de *testimonio de bienvenida* y entrega de las llaves de la Ciudad de la Provincia de Cercado Cochabamba, Bolivia por su trayectoria profesional. Distinguido por el Departamento de Cochabamba, Bolivia, con el Reconocimiento "Honor al Mérito" por sus aportes a la investigación, en mayo de 2003.

Áreas de Interés: Morfología Matemática, Análisis de Imágenes, Redes Neuronales Morfológicas, Teoría de Control y Robótica Móvil.

jdiaz@cic.ipn.mx

ISBN: 970-36-0116-2