

Thresholded Learning Matrix for Efficient Pattern Recalling

Mario Aldape-Pérez, Israel Román-Godínez,
and Oscar Camacho-Nieto

Center for Computing Research, CIC
National Polytechnic Institute, IPN
Mexico City, Mexico

mario@aldape.org.mx, iromanb05@sagitario.cic.ipn.mx, oscarcc@cic.ipn.mx
<http://www.aldape.org.mx>

Abstract. The *Lernmatrix*, which is the first known model of associative memory, is a heteroassociative memory that can easily work as a binary pattern classifier if output patterns are appropriately chosen. However, this mathematical model undergoes fundamental patterns misclassification whenever crossbars saturation occurs. In this paper, a novel algorithm that overcomes *Lernmatrix* weaknesses is proposed. The crossbars saturation occurrence is solved by means of a dynamic threshold value which is computed for each recalled pattern. The algorithm applies the dynamic threshold value over the ambiguously recalled class vector in order to obtain a sentinel vector which is used for uncertainty elimination purposes. The efficiency and effectiveness of our approach is demonstrated through comparisons with other methods using real-world data.

Keywords: Associative Memories, Dynamic Threshold, Lernmatrix, Pattern Classification, Supervised Learning.

1 Introduction

Karl Steinbuch, a pioneer of artificial neural networks, was one of the first researchers that developed functional structures (square arrays known as crossbars) which use conditional connections for adaptation purposes in categorization tasks [1]. *Lernmatrix* transcendence is evidenced by Kohonen's statement, where he points out that Correlation Matrices substitute Steinbuch's *Lernmatrix* [2]. The *Lernmatrix*, which is the first known model of associative memory, is a heteroassociative memory that can easily work as a binary pattern classifier if output patterns are appropriately chosen [3]. Nonetheless, complete recall is not guaranteed whenever crossbars saturation occurs [4]. In the following section, a brief description of associative memories fundamentals is presented. In Section 3, *Thresholded Learning Matrix* foundations are presented. An illustrative example is presented in Section 4 while in Section 5 some experimental results are shown using real-world data. The *Thresholded Learning Matrix* advantages, as well as a short conclusion will be discussed in Section 6.

2 Associative Memories

An associative memory M is a system that relates input patterns and output patterns as follows: $x \longrightarrow \boxed{\mathbf{M}} \longrightarrow y$ with x and y , respectively, the input and output pattern vectors. Each input vector forms an association with its corresponding output vector. For each k integer and positive, the corresponding association will be denoted as: (x^k, y^k) . Associative memory M is represented by a matrix whose ij -th component is m_{ij} [5]. Memory M is generated from an *a priori* finite set of known associations, called the fundamental set of associations. If μ is an index, the fundamental set is represented as: $\{(x^\mu, y^\mu) \mid \mu = 1, 2, \dots, p\}$ with p as the cardinality of the set. The patterns that form the fundamental set are called fundamental patterns. If it holds that $x^\mu = y^\mu \forall \mu \in \{1, 2, \dots, p\}$ M is auto-associative, otherwise it is heteroassociative; in this case, it is possible to establish that $\exists \mu \in \{1, 2, \dots, p\}$ for which $x^\mu \neq y^\mu$. If we consider the fundamental set of patterns $\{(x^\mu, y^\mu) \mid \mu = 1, 2, \dots, p\}$ where n and m are the dimensions of the input patterns and output patterns, respectively, it is said that $x^\mu \in A^n$, $A = \{0, 1\}$ and $y^\mu \in A^m$. Then the j -th component of an input pattern is $x_j^\mu \in A$. Analogously, the j -th component of an output pattern is represented as $y_j^\mu \in A$. A distorted version of a pattern x^k to be recuperated will be denoted as \tilde{x}^k . If when feeding an unknown input pattern x^ω with $\omega \in \{1, 2, \dots, k, \dots, p\}$ to an associative memory M , it happens that the output corresponds exactly to the associated pattern y^ω , it is said that recuperation is correct [6].

2.1 The Steinbuch's Lernmatrix

The *Lernmatrix* is a heteroassociative memory that can easily work as a binary pattern classifier if output patterns are appropriately chosen [3]. Typically it accepts binary patterns suchlike $\mathbf{x}^\mu \in A^n$, $A = \{0, 1\}$ as input and returns binary patterns suchlike $\mathbf{y}^\mu \in A^m$ as output; it is worth pointing out that there are m different classes, each one coded by a simple rule: class $k \in \{1, 2, \dots, m\}$ will be represented by a column vector whose components will be assigned by $y_k^\mu = 1$, so $y_j^\mu = 0$ for $j = 1, 2, \dots, k-1, k+1, \dots, m$; hence, the class statements are given in a 1-out-of- m -code [1], also known as *one-hot* codification [7].

Learning Phase. Find the adequate operators and a way to generate a matrix M that will store the p associations of the fundamental set $\{(x^1, y^1), \dots, (x^p, y^p)\}$, where $x^\mu \in A^n$ and $y^\mu \in A^m \forall \mu \in \{1, 2, \dots, p\}$. The following matrix, will keep the pattern association values after the Learning Phase is completed:

$$\begin{array}{cccc}
 & x_1^\mu & \cdots & x_j^\mu & \cdots & x_n^\mu \\
 y_1^\mu & m_{11} & \cdots & m_{1j} & \cdots & m_{1n} \\
 \vdots & \vdots & & \vdots & & \vdots \\
 y_i^\mu & m_{i1} & \cdots & m_{ij} & \cdots & m_{in} \\
 \vdots & \vdots & & \vdots & & \vdots \\
 y_m^\mu & m_{m1} & \cdots & m_{mj} & \cdots & m_{mn}
 \end{array} \tag{1}$$

Step 1. Each one of the m_{ij} components of M in (1) is initialized with zero.

Step 2. For each fundamental pattern association $\{(x^\mu, y^\mu) | \mu = 1, 2, \dots, p\}$, modify each one of the m_{ij} components of M in (1) according to the following rule:

$$m_{ij} = m_{ij} + \Delta m_{ij} \quad (2)$$

where Δm_{ij} is obtained as follows:

$$\Delta m_{ij} = \begin{cases} +\varepsilon & \text{if } y_i^\mu = 1 = x_j^\mu \\ -\varepsilon & \text{if } y_i^\mu = 1 \text{ and } x_j^\mu = 0 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

and ε is any positive constant, previously chosen.

Recalling Phase. The Recalling Phase for the Steinbuch's *Lernmatrix* consists of finding the class which an input pattern $\mathbf{x}^\omega \in A^n$ belongs to. Finding the class means getting $\mathbf{y}^\omega \in A^m$ that corresponds to \mathbf{x}^ω ; according to the construction method of all \mathbf{y}^μ , the class should be obtained without ambiguity. The i -th component of y_i^ω is obtained according to the following rule, where \vee is the *maximum* operator:

$$y_i^\omega = \begin{cases} 1 & \text{if } \sum_{j=1}^n m_{ij} \cdot x_j^\omega = \bigvee_{h=1}^m \left[\sum_{j=1}^n m_{hj} \cdot x_j^\omega \right] \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

It has to be mentioned that Steinbuch's *Lernmatrix* undergoes fundamental patterns misclassification every time crossbars saturation occurs. In order to get a better understanding of the crossbars saturation phenomenon, it is important to include the concept of characteristic set introduced by Sánchez-Garfias in [8].

Definition 1. Let $A = \{0, 1\}$ and $x^\omega \in A^n$ be a pattern. We call *characteristic set* of x^ω to the index set $T^\omega = \{j \mid (x_j^\omega > 0)\}$ and its cardinality is denoted by $|T^\omega|$.

Definition 2. Let $A = \{0, 1\}$ and $x^\alpha, x^\beta \in A^n$ be two patterns, then $x^\alpha \leq x^\beta$ if $\forall j \in \{1, 2, \dots, n\}$ it holds that $x_j^\alpha = 1 \rightarrow x_j^\beta = 1$.

Lemma 1. Let $A = \{0, 1\}$ and $x^\alpha, x^\beta \in A^n$ be two patterns, then $x^\alpha \leq x^\beta \longleftrightarrow T^\alpha \subseteq T^\beta$.

The proof of this lemma appears in [9]. The lemma means that an order relation between patterns implies an order relation between their characteristic set and vice versa. Misclassification usually takes place either when \mathbf{y}^ω has more than one bit as class identifier, which is clearly an erroneously coded class vector, or whenever the characteristic set of one or more learned patterns is a subset of the characteristic set of another fundamental pattern. In any of these cases correct classification is not guaranteed [10].

3 Thresholded Learning Matrix

In this section, a novel algorithm that overcomes *Lernmatrix* weaknesses is proposed. Due to the fact that an order relation between patterns implies an order relation between their characteristic set and vice versa, crossbars saturation occurrence can be solved along the recalling phase by means of a dynamic threshold value which is computed for each recalled pattern. *Thresholded Learning Matrix* algorithm applies the same learning phase as the *Lernmatrix*, while a completely different recalling phase is proposed.

Definition 3. Let $A = \{0, 1\}$ and $x^\omega \in A^n$ be a pattern, so the ij -th component of an associative memory M is denoted by m_{ij} , then an ambiguously recalled class vector C^ω is obtained according to the following rule:

$$C_i^\omega = \sum_{j=1}^n m_{ij} \cdot x_j^\omega \quad (5)$$

Definition 4. Let \vee be the maximum operator and ε be any positive constant, then the dynamic threshold value θ^ω is obtained according to the following rule:

$$\theta^\omega = \frac{1}{\varepsilon} \bigvee_{h=1}^p [C_h^\omega] \quad (6)$$

If we substitute (5) in (6), the dynamic threshold value θ^ω can be expressed as follows:

$$\theta^\omega = \frac{1}{\varepsilon} \bigvee_{h=1}^p \left[\sum_{j=1}^n m_{hj} \cdot x_j^\omega \right] \quad (7)$$

Definition 5. Let ε be any positive constant and m_{ij} be the ij -th component of an associative memory M , then the positive contribution value of each one of the fundamental patterns is obtained according to the following rule:

$$U_i = \frac{1}{\varepsilon} \sum_{j=1}^n \{m_{ij} \mid (m_{ij} > 0)\} \quad (8)$$

Definition 6. Let θ^ω be the dynamic threshold value, ε be any positive constant and m_{ij} be the ij -th component of an associative memory M , then a thresholded class vector Z^ω is obtained according to the following rule:

$$Z_i^\omega = \begin{cases} 1 & \text{if } \sum_{j=1}^n \frac{1}{\varepsilon} (m_{ij} \cdot x_j^\omega) = \theta^\omega \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

Definition 7. Let U be the positive contribution value of each one of the fundamental patterns and θ^ω be the dynamic threshold value, then the sentinel vector S^ω is obtained according to the following rule:

$$S_i^\omega = \frac{1}{\theta^\omega} U_i \quad (10)$$

Definition 8. Let Z^ω be a thresholded class vector and S^ω be the sentinel vector, then the unambiguously recalled class vector R^ω is obtained according to the following rule:

$$R_i^\omega = Z_i^\omega . S_i^\omega \quad (11)$$

Using the previous definitions it is possible to enunciate The *Thresholded Learning Matrix* Recalling Rule. The i -th component of y^ω is obtained according to the following rule:

$$y_i^\omega = \begin{cases} 1 & \text{if } R_i^\omega = 1 \\ 0 & \text{otherwise} \end{cases} ; \quad \forall i \in \{1, 2, \dots, p\} \quad (12)$$

where \vee is the *maximum* operator and R^ω is the unambiguously recalled class vector.

Learning Phase. Generate a matrix M that will store the p associations of the fundamental set $\{(x^1, y^1), \dots, (x^p, y^p)\}$, where $x^\mu \in A^n$ and $y^\mu \in A^m \forall \mu \in \{1, 2, \dots, p\}$ according to Step 1 and Step 2 as stated in Section 2.1. Once the learning phase is completed, an associative memory M will be obtained.

Recalling Phase. Find the class which an input pattern $\mathbf{x}^\omega \in A^n$ belongs to. Finding the class means getting $\mathbf{y}^\omega \in A^m$ that corresponds to \mathbf{x}^ω . Using the previous definitions it is possible to enunciate The *TLM* Recalling Phase.

Step 1. Obtain the dynamic threshold value θ^ω using (7).

Step 2. Obtain the positive contribution value of each one of the fundamental patterns using (8).

Step 3. Obtain the thresholded class vector Z^ω using (9).

Step 4. Obtain the sentinel vector S^ω using (10).

Step 5. Obtain the unambiguously recalled class vector R^ω using (11).

Step 6. Obtain the class y^ω which an input pattern \mathbf{x}^ω belongs to.using (12).

After the Recalling Phase is completed and according to the construction method of all \mathbf{y}^μ , the class y^ω is obtained without ambiguity for each one of the fundamental patterns $x^\mu \in A^n \forall \mu \in \{1, 2, \dots, p\}$.

4 Numerical Results

Example 4.1. Let $p = 3$, $n = 5$, $m = 3$. Given the fundamental patterns $\{(x^\mu, y^\mu) \mid \mu = 1, 2, \dots, p\}$, obtain a *Thresholded Learning Matrix*. The fundamental associations will be denoted as: $\{(x^1, y^1), (x^2, y^2), \dots, (x^3, y^3)\}$.

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

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

Learning phase. Generate a matrix M that will store the p associations of the fundamental set, according to Step 1 and Step 2 as stated in Section 2.1. Obtain the positive contribution value U of each one of the fundamental patterns using (8).

$$M = \begin{pmatrix} \varepsilon & -\varepsilon & -\varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & -\varepsilon & \varepsilon \\ \varepsilon & -\varepsilon & \varepsilon & -\varepsilon & -\varepsilon \end{pmatrix}; \quad U = \begin{pmatrix} 2 \\ 3 \\ 2 \end{pmatrix}$$

Once the learning phase is completed, an associative memory M will be obtained, as well as the positive contribution value U of each one of the fundamental patterns.

Recalling Phase. Find the class which an input pattern $\mathbf{x}^\omega \in A^n$ belongs to. Finding the class means getting $\mathbf{y}^\omega \in A^m$ that corresponds to \mathbf{x}^ω . The dynamic threshold value θ^ω is obtained using (7).

$$C^1 = M \cdot x^1 = \begin{pmatrix} 2\varepsilon \\ 2\varepsilon \\ 0 \end{pmatrix}; \quad \theta^1 = 2$$

Using (9) a thresholded class vector Z^ω is obtained, as well as the sentinel vector S^ω using (10).

$$Z^1 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}; \quad S^1 = \begin{pmatrix} 1 \\ 1.5 \\ 1 \end{pmatrix}$$

Using (11) the unambiguously recalled class vector R^ω is obtained.

$$R^1 = Z^1 \cdot S^1 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 1.5 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1.5 \\ 0 \end{pmatrix} \rightarrow y^1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Finally, the class y^ω which an input pattern \mathbf{x}^ω belongs to is obtained using (12). Due to paper space limitations, only the first association (x^1, y^1) recalling results are shown. The reader can easily verify that the whole fundamental set of patterns is completely recalled.

5 Experimental Results

In order to estimate how the *Thresholded Learning Matrix* algorithm performs with high dimensional data, 20 grayscale images obtained from the *Third International Fingerprint Verification Competition* (FVC2004) were used as fundamental patterns (Figure 1). The experimental phase was implemented using Matlab v.7.0 (R14) over a PC with an Intel Core 2 Duo microprocessor @ 2.13 GHz with 2048 MB of RAM and 120 GB of hard disk space. Each one of the

**Fig. 1.** Fundamental Input Patterns**Table 1.** Averaged recalling results

	<i>Algorithm1 [1]</i>	<i>Algorithm2 [10]</i>	<i>TLM</i>
Number of vectors	20	20	20
Vector size, n	153600 bits	153600 bits	153600 bits
Learning time	11.69 (sec)	11.69 (sec)	11.69 (sec)
Recalling time	10.52 (sec)	11.82 (sec)	10.75 (sec)
Error rate (%)	11.2	9.6	0

fundamental patterns produces a $160 \times 120 \times 8$ bits vector. Accordingly, each pattern association results in a 153600 bits vector, consequently, a 20 by 153600 bits matrix is obtained after the learning phase is completed.

The experimental phase was carried out as follows: after a 20 by 153600 bits matrix initialization process was concluded, the first association was learned and recalled. Subsequently, the first and second associations were learned and recalled; after that, the same procedure continued in a consecutive manner until the fundamental set of patterns was completely learned and recalled. The above mentioned procedure was executed 10 times, each time changing the fundamental patterns order randomly. The averaged recalling results are shown in Table 1. A relevant thing to mention about the recalling criterion that was used along the experimental phase is that, in this case, perfect recall means that all of the 153600 bits were exactly recovered. Particularly, outstanding results were achieved by the *Thresholded Learning Matrix* algorithm (the whole fundamental set of patterns was perfectly recalled).

6 Conclusions and Ongoing Research

In this paper a new algorithm which is based on the first known model of associative memory has been introduced. *Lernmatrix* weaknesses are solved by means of a dynamic threshold value which is computed for each recalled pattern. The algorithm applies the dynamic threshold value over the ambiguously recalled

class vector in order to obtain a sentinel vector which is used for uncertainty elimination purposes. Experimental results have shown that this algorithm is an efficient way to improve classifier accuracy; furthermore, our recalling rule proposal not only is simpler but swifter than the ones presented in [1] and [10].

Acknowledgments. The authors of the present paper would like to thank the following institutions for their economical support to develop this work: National Polytechnic Institute, Mexico (CIC, SIP, PIFI, COFAA) , CONACyT and SNI.

References

1. Steinbuch, K.: Die Lernmatrix. *Kybernetik* 1(1), 36–45 (1961)
2. Kohonen, T.: Correlation Matrix Memories. *IEEE Transactions on Computers* C-21(4), 353–359 (1972)
3. Steinbuch, K., Frank, H.: Nichtdigitale Lernmatrizen als Perzeptoren. *Kybernetik* 1(3), 117–124 (1961)
4. Yáñez-Márquez, C., Díaz-de-León Santiago, J.L.: Lernmatrix de Steinbuch. IT-48 Serie Verde, CIC-IPN, México (2001)
5. Hassoun, M.H.: *Associative Neural Memories*. Oxford University Press, New York (1993)
6. Yáñez-Márquez, C.: *Associative Memories Based on Order Relations and Binary Operators* (in Spanish). PhD Thesis. Center for Computing Research, México (2002)
7. Chren, W.A.: One-hot residue coding for high-speed non-uniform pseudo-random test pattern generation. In: *IEEE International Symposium on Circuits and Systems ISCAS 1995*, pp. 401–404. IEEE Press, New York (1995)
8. Sánchez Garfías, F.A., Díaz-de-León Santiago, J.L., Yáñez Márquez, C.: Lernmatrix de Steinbuch: condiciones necesarias y suficientes para recuperación perfecta de patrones. *Research on Computing Science* 1, 437–448 (2002)
9. Sánchez Garfías, F.A., Díaz-de-León Santiago, J.L., Yáñez Márquez, C.: New Results on the Lernmatrix Properties. *Research on Computing Science* 10, 91–102 (2004)
10. Román-Godínez, I., López-Yáñez, I., Yáñez-Márquez, C.: A New Classifier Based on Associative Memories. In: *Proc. 15th International Conference on Computing CIC 2006*, pp. 55–59. IEEE Computer Society, Los Alamitos (2006)