

A New Tool for Engineering Education: Hepatitis Diagnosis using Associative Memories*

MARIO ALDAPE-PÉREZ, CORNELIO YÁÑEZ-MÁRQUEZ, OSCAR CAMACHO-NIETO and AMADEO J. ARGÜELLES-CRUZ

Center for Computing Research, CIC-IPN Building, Nueva Industrial Vallejo, G.A. Madero, Mexico City 07738, Mexico.
E-mail: maldape@ieec.org, cyanez@cic.ipn.mx, ocamacho@ipn.mx, jamadeo@cic.ipn.mx

Classification is one of the key issues in medical diagnosis. In this paper, a new tool for engineering education is presented: it is an automatic hepatitis diagnosis system based on associative memories. The characteristic of this approach is twofold: first, learning the fundamental set of associations in order to get an associative memory; second, computing a differential associative memory in order to get a threshold value for each unknown input pattern to be classified. Hepatitis disease dataset, taken from UCI machine learning repository, was used as medical dataset. Classification accuracy of the proposed approach is 82.67% and it was assessed using stratified 10 fold cross-validation. The correct diagnosis performance of the proposed approach is validated not only using classification accuracy, but also performing sensitivity and specificity analysis. The results presented in this paper demonstrate associative memories potential for automatic medical diagnosis systems.

Keywords: associative memories; engineering education; decision support systems; machine learning; pattern classification

1. Introduction

Hepatitis is an abnormal condition of the liver characterized by the presence of inflammatory cells in the tissue of the organ. Depending on malady duration, hepatitis is considered acute when it lasts less than six months and chronic when it persists longer [1]. There are a number of different viruses, known as the hepatitis viruses, which cause most cases of liver damage worldwide. The type of hepatitis is named for the virus that causes it; for example, hepatitis A is caused by the hepatitis A virus (HAV). The hepatitis virus family consists of several different viruses, namely, hepatitis A, hepatitis B, hepatitis C, hepatitis D and hepatitis E [2–4]. All of these viruses cause acute or short-term viral hepatitis; however, hepatitis B, C, and D viruses can also cause chronic hepatitis, which can lead to cirrhosis of the liver, liver failure, and liver cancer [5].

Unlike most daily decisions, many health-care decisions have important implications for the quality of life of the patient, and involve significant uncertainties and trade-offs. The uncertainties may be about the diagnosis, the accuracy of available diagnostic tests, the prevalence of the disease and its attendant risk factors. For such kind of complex decisions, which are inherently affected by so many uncertainties, it is indispensable to have computational tools that help to identify which variables or features of the problem should have a major impact on our decision. It is also needed to apply effective mathematical models, as

well as efficient algorithms that allow to decrease the level of uncertainty in the diagnosis of the disease.

Early models of learning matrices appeared more than four decades ago [6–8], and since then associative memories have attracted the attention of major research groups worldwide. From a connectionist model perspective an associative memory can be considered a special case of the neural computing approach for pattern recognition [9–11]. Furthermore, associative memories have a number of properties, including a rapid, compute efficient best-match and intrinsic noise tolerance that make them ideal for many applications [12–14]. As a consequence, associative memories have emerged as a computational paradigm to efficiently solve pattern recognition tasks such as: automatic color matching [15], efficient retrieval of grayscale images [16], text translation [17], feature selection [18], image compression [19] and classification of cancer recurrence [20] among many others.

In this paper, a novel algorithm is presented. The proposed algorithm is applied to diagnose diseases; particularly, it is applied to help diagnose hepatitis effectively. The experimental outcomes suggest that to perform reliable diagnosis of diseases it is not sufficient to take into account only the classification accuracy parameter. Notwithstanding, it is necessary to conduct a sensitivity and specificity analysis.

In the following section, the main characteristics of the dataset that was used along the experimental phase are presented. In section 3, a brief description of associative memories fundamentals is presented. In section 4, Linear Associator technical details are

presented. In section 5, Delta Associative Memory mathematical foundations are presented. In section 6, a brief introduction to performance evaluation methods in a binary classification problem is presented, while in section 7 some experimental results are shown using real-world data. Delta Associative Memory advantages, as well as a short conclusion will be discussed in section 8.

2. Hepatitis disease dataset

Hepatitis disease dataset has been widely used as test dataset not only in machine learning but also in knowledge discovery. This dataset was donated by the Jozef Stefan Institute, former Yugoslavia, now Slovenia. It was taken from the University of California at Irvine machine learning repository [21]. The purpose of the dataset is to predict the presence or absence of hepatitis disease on a patient. Hepatitis disease dataset consists of 155 instances belonging to two different classes (32 ‘die’ cases, 123 ‘live’ cases). Each instance consists of 20 attributes, 13 binary, 6 attributes with discrete values and a class label.

3. Associative memories

An associative memory M is a system that relates input patterns and output patterns as follows:

$$x \rightarrow \boxed{M} \rightarrow y$$

with x and y the input and output pattern vectors, respectively. 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) . An associative memory M is represented by a matrix whose ij -th component is m_{ij} . Associative memory M is generated from a set of known associations, called the fundamental set of associations. If μ is an index, the fundamental set is represented as: $\{(x^\mu, y^\mu) | \mu = 1, 2, \dots, p\}$ 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 autoassociative, 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 $\{(x^\mu, y^\mu) | \mu = 1, 2, \dots, p\}$ where n and m are the dimensions of the input patterns and output patterns, respectively, it is said that $x^\mu \in A^n$ and $y^\mu \in A^m$ where $A = \{0, 1\}$, then the j -th component of an input pattern $x^\mu \in A^n$ is $x_j^\mu \in A$. Analogously, the i -th component of an output pattern $y^\mu \in A^m$ is represented as $y_i^\mu \in A$. Therefore the

fundamental input and output patterns are represented as follows:

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

A distorted version of a pattern x^k to be recalled will be denoted as \tilde{x}^k . An unknown input pattern to be recalled will be denoted as x^ω . If when an unknown input pattern x^ω with $\omega \in \{1, 2, \dots, k, \dots, p\}$ is fed to an associative memory M , it happens that the output corresponds exactly to the associated pattern y^ω , it is said that recalling is correct.

4. Linear associator

The Linear Associator [11, 14], which is one of the classical models of associative memories, is a heteroassociative memory that can easily work as a binary pattern classifier if output patterns are appropriately chosen. In this section, Linear Associator mathematical foundations are presented.

4.1 Learning phase

Let $\{(x^\mu, y^\mu) | \mu = 1, 2, \dots, p\}$ be the fundamental set. In order to obtain an associative memory M , the learning phase is done in two stages:

1. Consider each one of the p associations (x^μ, y^μ) , so an m by n matrix is obtained according to

$$y^\mu \cdot (x^\mu)^t = \begin{pmatrix} y_1^\mu x_1^\mu & \cdots & y_1^\mu x_j^\mu & \cdots & y_1^\mu x_n^\mu \\ \vdots & & \vdots & & \vdots \\ y_i^\mu x_1^\mu & \cdots & y_i^\mu x_j^\mu & \cdots & y_i^\mu x_n^\mu \\ \vdots & & \vdots & & \vdots \\ y_m^\mu x_1^\mu & \cdots & y_m^\mu x_j^\mu & \cdots & y_m^\mu x_n^\mu \end{pmatrix} \quad (1)$$

2. An associative memory M is obtained by adding all the p matrices

$$M = \sum_{\mu=1}^p y^\mu \cdot (x^\mu)^t = [m_{ij}]_{m \times n} \quad (2)$$

In this way the ij -th component of an associative memory M is expressed as follows:

$$m_{ij} = \sum_{\mu=1}^p y_i^\mu x_j^\mu \quad (3)$$

4.2 Recalling phase

Linear Associator recalling phase is done by operating an associative memory M with an unknown input pattern x^ω , where $\omega \in \{1, 2, \dots, k, \dots, p\}$. Operate $M \cdot x^\omega$ as follows:

$$M \cdot x^\omega = \left[\sum_{\mu=1}^p y^\mu \cdot (x^\mu)^t \right] \cdot x^\omega \quad (4)$$

Let's expand expression (4), which is:

$$M \cdot x^\omega = y^\omega \cdot [(x^\omega)^t \cdot x^\omega] + \sum_{\mu \neq \omega} y^\mu \cdot [(x^\mu)^t \cdot x^\omega] \quad (5)$$

Expression (5) let us know about which restrictions have to be observed thus correct recalling is achieved.

This is expressed as:

$$(x^\mu)^t \cdot x^\omega = \begin{cases} 1 & \text{if } \mu = \omega \\ 0 & \text{if } \mu \neq \omega \end{cases} \quad (6)$$

If condition (6) is met, then a correct recalling is expected. Therefore, expression (5) is expressed as:

$$M \cdot x^\omega = y^\omega \quad (7)$$

5. Delta associative memory

In this section, a novel algorithm that overcomes Linear Associator weaknesses is proposed. Due to the fact that an order relation between patterns implies an order relation between their characteristic set and vice versa [22, 23], *cross-talk* influence can be annulled by means of a dynamic threshold value which is computed for each unknown input pattern to be classified. Delta Associative Memory algorithm applies the same learning phase as the Linear Associator, while a completely different recalling phase is proposed.

In what follows, let M be an associative memory whose ij -th component is denoted by m_{ij} and let ω be an index such that $\omega \in \{1, 2, \dots, k, \dots, p\}$. Let $x^\omega \in \mathbb{R}^n$ be an unknown input pattern to be classified and let $m, n \in \mathbb{Z}^+$ be the dimension of the output patterns and input patterns, respectively.

Definition 1. Differential Associative Memory. A Differential Associative Memory is denoted by Ψ^ω . The ij -th component of Ψ^ω is obtained according to the following rule:

$$\psi_{ij}^\omega = |m_{ij} - x_j^\omega| \quad (8)$$

$$\forall i \in \{1, 2, \dots, m\}, \forall j \in \{1, 2, \dots, n\}.$$

Definition 2. Maximum threshold value. The maximum threshold value, denoted by ζ^ω , is obtained according to the following rule:

$$\zeta^\omega = \bigvee_{i=1}^m \left(\bigvee_{j=1}^n (\psi_{ij}^\omega) \right) \quad (9)$$

Where \vee is the maximum operator.

Definition 3. Minimum threshold value. The minimum threshold value, denoted by α^ω , is obtained according to the following rule:

$$\alpha^\omega = \bigwedge_{i=1}^m \left(\bigwedge_{j=1}^n (\psi_{ij}^\omega) \right) \quad (10)$$

Where \wedge is the minimum operator.

Definition 4. Delta Associative Memory (DAM). Let θ^ω be the dynamic threshold value, such that $\alpha^\omega \leq \theta^\omega \leq \zeta^\omega$. A Delta Associative Memory is denoted by Δ^ω . The ij -th component of Δ^ω , denoted by δ_{ij}^ω , is obtained according to the following rule:

$$\delta_{ij}^\omega = \begin{cases} 1 & \text{if } |m_{ij} - x_j^\omega| \leq \theta^\omega \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

Definition 5. Positive contributions vector. The Positive contributions vector is denoted by σ^ω . The i -th component of σ^ω , denoted by σ_i^ω , is obtained according to the following rule:

$$\sigma_i^\omega = \sum_{j=1}^n \delta_{ij}^\omega \quad (12)$$

Definition 6. Transition vector. The Transition vector is denoted by τ^ω . The i -th component of τ^ω , denoted by τ_i^ω , is obtained according to the following rule:

$$\tau_i^\omega = \begin{cases} 1 & \text{if } \sigma_i^\omega = \left(\bigvee_{h=1}^m (\sigma_h^\omega) \right) \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

Where \vee is the maximum operator.

Definition 7. Unambiguously recalled class vector. The Unambiguously recalled class vector, denoted by y^ω , is obtained according to the following rule:

$$y^\omega = \begin{cases} \tau^\omega & \text{if } \sum_{i=1}^m \tau_i^\omega \leq 1 \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

5.1 Learning phase

Generate a matrix M that will store the p associa-

tions of the fundamental set $\{(x^1, y^1), \dots, (x^p, y^p)\}$, where $x^\mu \in \mathbb{R}$ and $y^\mu \in A^m$ $\forall \mu \in \{1, 2, \dots, p\}$. It is worth pointing out that there are m different classes. Therefore, each one of the input patterns belongs to a k class, $k \in \{1, 2, \dots, m\}$, represented by a column vector y^μ , 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, also known as *one-hot* codification [17].

Given:

The fundamental set of associations $\{(x^\mu, y^\mu) | \mu = 1, 2, \dots, p\}$ with p as the cardinality of the set.

Algorithm:

Obtain p matrices according to expression (1).

```

for  $\mu = 1$  to  $p$  do
{
  for  $i = 1$  to  $m$  do
  {
    for  $j = 1$  to  $n$  do
    {
      Compute  $m_{ij}$  using expression (2).
    }
  }
}
}

```

5.2 Classification phase

Finding the class which an unknown input pattern $x^\omega \in \mathbb{R}^n$ belongs to. Finding the class means getting $y^\omega \in A^m$ that corresponds to x^ω .

If when an unknown input pattern x^ω is fed to an associative memory M , it happens that the output corresponds exactly to the associated pattern y^ω , it is said that classification is correct.

Given:

An unknown input pattern $x^\omega \in \mathbb{R}^n$

Algorithm:

Obtain a Differential Associative Memory Ψ^ω , using expression (8).

Compute the maximum threshold value ζ^ω , using expression (9).

Compute the minimum threshold value α^ω , using expression (10).

Initialize the dynamic threshold value θ^ω , ie, $\theta^\omega = \alpha^\omega$

While $\sum_{i=1}^m \tau_i^\omega > 1$ and $\theta^\omega < \zeta^\omega$ do

{

Obtain a Delta Associative Memory Δ^ω , using expression (11).

Compute the positive contributions σ^ω , using expression (12).

Compute the transition vector τ^ω , using expression (13).

Assign the recalled class vector y^ω , using expression (14).

Increment the dynamic threshold value θ^ω , i.e., $\theta^\omega = \theta^\omega + 1$

}

Assign the unambiguously recalled class vector y^ω , using expression (14).

6. Performance evaluation methods

There are three main performance indicators of a binary classification test: sensitivity, specificity and classification accuracy. These indicators are computed from the confusion matrix.

6.1 Confusion matrix

A confusion matrix is typically used in supervised learning for classifier performance estimation. It contains information, arranged in rows and columns, about the actual condition and the classification outcome. In general, a confusion matrix is of size $L \times L$, where L is the number of different class labels [24]. The diagonal elements represent correctly classified instances while the cross-diagonal elements represent misclassified instances. Table 1 shows the confusion matrix for a two class classifier (binary classification test).

The entries of the confusion matrix are as follows:

- True Positive (TP) refers to those instances whose actual condition is positive and the test outcome is positive.
- True Negative (TN) refers to those instances whose actual condition is negative and the test outcome is negative.
- False Positive (FP) refers to those instances whose actual condition is negative and the test outcome is positive.
- False Negative (FN) refers to those instances whose actual condition is positive and the test outcome is negative.

Table 1. Confusion matrix for a two class classifier (binary classification test)

		Actual condition	
		Positive	Negative
Test outcome	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

6.2 Sensitivity and specificity

Sensitivity and specificity are statistical measures of the performance of a binary classification test.

From a medical diagnostic perspective, sensitivity and specificity are used for assessing the results of diagnostic and screening tests [25].

Sensitivity or True Positive Rate (TPR) represents the proportion of truly diseased persons in a screened population who are identified as being diseased by the test. Sensitivity is a measure of the probability of correctly diagnosing a condition. Sensitivity is computed using the following expression:

$$\text{sensitivity} = \frac{\#TP}{\#TP + \#FN} \quad (15)$$

Specificity or True Negative Rate (TNR) is the proportion of truly healthy persons who are identified as so by the screening test. Specificity is a measure of the probability of correctly identifying a healthy person. Specificity is computed using the following expression:

$$\text{specificity} = \frac{\#TN}{\#FP + \#TN} \quad (16)$$

6.3 Classification accuracy

Classification accuracy of any algorithm can be estimated taking into account the overall number of test patterns that are correctly classified. In the present paper, classification accuracy results were estimated using the following expression:

$$\text{accuracy}(T) = \frac{\sum_{\omega=1}^{|T|} \text{assess}(x^\omega)}{|T|}; \quad x^\omega \in T \quad (17)$$

Where T is the set of unknown input patterns to be classified (test set) [26].

Each time the classification result of a test pattern $x^\omega \in T$ is equal to the actual condition of that pattern, an integer value equal to 1 will be assigned to the assessment function, as shown in the following expression:

$$\text{assess}(x^\omega) = \begin{cases} 1 & \text{if } \text{classify}(x^\omega) = y^\xi \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

Where y^ξ is the actual condition of a test pattern x^ω and $\text{classify}(x^\omega)$ returns the classification result of a test pattern x^ω by Delta Associative Memory algorithm, as indicated in section 5.2.

7. Experimental results

Throughout the experimental phase, hepatitis disease dataset, taken from UCI machine learning

repository [21], was used as test set to estimate the performance of the proposed method in the diagnosis of liver disease. The main characteristics of this data set have been expounded in section 2. Delta Associative Memory performance was compared against the performance achieved by the fifteen best-performing algorithms which are included in WEKA 3: Data Mining Software in Java [27]. Further information on each of the algorithms used during the experimental phase, can be found in the book Data Mining: Practical Machine Learning Tools and Techniques [28].

The experimental phase has been carried out as follows: firstly, in order to obtain reasonably unbiased performance estimates, the dataset was broken into K partitions (in our case $K = 10$). Afterwards, in order to obtain an associative memory, the same number of input vectors for each class was randomly taken, which ensures a balanced classifier. Classification accuracy of each one of the compared algorithms was calculated using 10-fold cross-validation technique. Sensitivity, specificity and classification accuracy for each one of the compared algorithms have been summarized in Table 2. As it is shown in Table 2, the best-performing method in terms of specificity and classification accuracy is our proposal, called Delta Associative Memory (DAM).

8. Conclusions

This paper presents a new algorithm to perform pattern classification tasks efficiently. The proposed algorithm is applied to diagnose a disease; particularly, it is applied to help diagnose hepatitis effectively. Since many health-care decisions have important implications for the quality of life of the

Table 2. Classification accuracy, sensitivity and specificity values for hepatitis disease dataset using 10 fold cross-validation. The fifteen best-performing algorithms shown in this table are included in WEKA 3: Data Mining Software in Java [27]

	Method	Sensitivity	Specificity	Accuracy
1	HyperPipes	0.871	0.143	54.194
2	DecisionStump	0.941	0.214	61.290
3	FLR	0.400	0.671	52.258
4	ConjunctiveRule	0.835	0.371	62.581
5	RandomTree	0.541	0.557	54.839
6	LADTree	0.647	0.471	56.774
7	REPTree	0.812	0.400	62.581
8	BFTree	0.800	0.429	63.226
9	SimpleCart	0.753	0.457	61.936
10	RBFNetwork	0.788	0.586	69.677
11	JRip	0.741	0.614	68.387
12	FT	0.765	0.600	69.032
13	PART	0.706	0.643	67.742
14	NBTree	0.765	0.614	69.677
15	DecisionTable	0.812	0.614	72.258
*	DAM (our proposal)	0.812	0.829	82.667

patient, it is not sufficient to issue a medical diagnosis taking into account only the classification accuracy parameter. The experimental outcomes suggest that in order to perform reliable diagnosis of diseases it is necessary to conduct a sensitivity and specificity analysis. The performance of the proposed model in the diagnosis of liver disease is validated not only using classification accuracy, but also performing sensitivity and specificity analysis. Experimental results have shown that the best performing method in terms of specificity and classification accuracy is Delta Associative Memory.

The results presented in this paper demonstrate associative memories potential for automatic medical diagnosis systems.

Acknowledgments—The authors of the present paper would like to thank the following institutions for their support to develop this work: Science and Technology National Council of Mexico (CONACyT), National System of Researchers of Mexico (SNI), National Polytechnic Institute of Mexico (IPN, Project No. SIP-IPN 20120679) and Institute of Science and Technology of the Federal District (ICyTDF, Project No. PICSA12-083).

References

- National Institute of Diabetes and Digestive and Kidney Diseases, U.S. Department of Health and Human Services: Viral Hepatitis: A through E and Beyond, <http://digestive.niddk.nih.gov/>, Accessed 21 September 2009.
- MedlinePlus, U.S. National Library of Medicine, National Institutes of Health: Hepatitis A, <http://medlineplus.gov/>, Accessed 21 September 2009.
- MedlinePlus, U.S. National Library of Medicine, National Institutes of Health: Hepatitis B, <http://medlineplus.gov/>, Accessed 21 September 2009.
- MedlinePlus, U.S. National Library of Medicine, National Institutes of Health: Hepatitis C, <http://medlineplus.gov/>, Accessed 21 September 2009.
- National Institute of Allergy and Infectious Diseases, National Institutes of Health, U.S. Department of Health and Human Services: Viral Hepatitis, <http://www3.niaid.nih.gov/>, Accessed 21 September 2009.
- K. Steinbuch, Die lernmatrix, *Kybernetik*, **1**, 1961, pp. 36–45.
- K. Steinbuch and H. Frank, Nichtdigitale lernmatrizen als perzeptoren, *Kybernetik*, **1**, 1961, pp. 117–124.
- K. Steinbuch, Adaptive networks using learning matrices, *Kybernetik*, **2**, 1964, pp. 148–152.
- H. Kazmierczak and K. Steinbuch, Adaptive systems in pattern recognition, *IEEE Transactions on Electronic Computers*, EC-12, 1963, pp. 822–835.
- K. Steinbuch and B. Widrow, A critical comparison of two kinds of adaptive classification networks, *IEEE Transactions on Electronic Computers*, EC-14, 1965, pp. 737–740.
- T. Kohonen, Correlation matrix memories, *IEEE Transactions on Computers*, C-21, 1972, pp. 353–359.
- K. Steinbuch and U. A. W. Piske, Learning matrices and their applications, *IEEE Transactions on Electronic Computers*, EC-12, 1963, pp. 846–862.
- J. A. Anderson, A memory storage model utilizing spatial correlation functions, *Kybernetik*, **5**, 1968, pp. 113–119.
- J. A. Anderson, A simple neural network generating an interactive memory, *Mathematical Biosciences*, **14**, 1972, pp. 197–220.
- C. Yáñez, E. Felipe-Riveron, I. López-Yáñez and R. Flores-Carapia, A novel approach to automatic color matching, *Lecture Notes in Computer Science*, **4225**, 2006, pp. 529–538.
- C. Yáñez-Márquez, L.P. Sánchez-Fernández and I. López-Yáñez, Alpha-beta associative memories for gray level patterns, *Lecture Notes in Computer Science*, **3971**, 2006, pp. 818–823.
- M. E. Acevedo-Mosqueda, C. Yáñez-Márquez and I. López-Yáñez, Alpha-beta bidirectional associative memories: theory and applications, *Neural Processing Letters*, **26**, 2007, pp. 1–40.
- M. Aldape-Pérez, C. Yáñez-Márquez and A. J. Argüelles-Cruz, Optimized associative memories for feature selection, *Lecture Notes in Computer Science*, **4477**, 2007, pp. 435–442.
- E. Guzmán, O. Pogrebnyak, C. Yáñez and L. P. S. Fernandez, Morphological transform for image compression, *EURASIP Journal on Advances in Signal Processing*, **2008**, 2008, 15 pp.
- M. E. Acevedo, M. A. Acevedo and F. Felipe, Classification of cancer recurrence with alpha-beta bam, *Mathematical Problems in Engineering*, **2009**, 2009, 14 pp.
- UCI machine learning repository, <http://archive.ics.uci.edu/ml/>, Accessed 21 September 2009.
- C. Yáñez-Márquez, *Associative memories based on order relations and binary operators* (In Spanish), Ph.D. thesis, Artificial Intelligence Laboratory, Center for Computing Research, Mexico City, Mexico, 2002.
- M. Aldape-Pérez, I. Román-Godínez and O. Camacho-Nieto, Thresholded learning matrix for efficient pattern recalling, *Lecture Notes in Computer Science*, **5197**, 2008, pp. 445–452.
- R. Kohavi and F. Provost, Glossary of terms special issue on applications of machine learning and the knowledge discovery process, *Machine Learning*, **30**, 1998, pp. 271–274.
- M. G. M. Hunink, P. P. Glasziou, J. E. Siegel, J. C. Weeks, J. S. Pliskin, A. S. Elstein and M. C. Weinstein, *Decision Making in Health and Medicine: Integrating Evidence and Values*, Cambridge University Press, Cambridge, UK, 2001.
- A. Watkins, *Exploiting Immunological Metaphors in the Development of Serial, Parallel and Distributed Learning Algorithms*, Ph.D. thesis, Computing Laboratory, University of Kent, Canterbury, UK, 2005.
- M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, The weka data mining software: An update, *SIGKDD Explorations*, **11**, 2009, pp. 10–18.
- I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, 2005.

Mario Aldape-Pérez received the B.E. degree in electrical engineering from the Autonomous Metropolitan University, Azcapotzalco, Mexico, in 2004, and the M.Sc. degree in computer engineering and the Ph.D. degree in computer science from the National Polytechnic Institute of Mexico, Mexico, in 2007 and 2011, respectively. In 2010, he joined the Department of Computer Systems Engineering from the School of Computing, National Polytechnic Institute of Mexico, as an Associate Professor, and in 2012 became Professor. His current research interests include Unconventional Computing, Associative Memories, Soft Computing and FPGA Implementation of High Performance Pattern Classification algorithms. Dr. Aldape-Pérez is a member of the IEEE Computer Society. He is currently the President of the Academy of Digital Systems at the School of Computing of the National Polytechnic Institute of Mexico.

Cornelio Yáñez-Márquez received the M.Sc. degree in computer engineering and the Ph.D. degree in computer science from the National Polytechnic Institute of Mexico, Mexico, in 1995 and 2002, respectively. Since then, he has been with the Center for Computing Research (CIC-IPN), Mexico, where he is currently the Head of the Laboratory of Unconventional

Computing and Neural Networks. His research interests cover Neural Networks and Unconventional Computing, Associative Memory, Pattern Recognition, Mathematical Morphology and Soft Computing, with over 100 technical publications. Dr. Yáñez-Márquez is the leader and founder of Alpha-Beta Research.

Oscar Camacho-Nieto received the M.Sc. degree in computer engineering and the Ph.D. degree in computer science from the National Polytechnic Institute of Mexico, Mexico, in 1995 and 2003, respectively. He was a Senior Lecturer at the Center for Computing Research (CIC-IPN), Mexico, for several years. He is currently the Director of the Center for Innovation and Technology Development in Computing (CIDETEC-IPN), Mexico.

Amadeo J. Argüelles-Cruz received the M.Sc. degree in computer engineering and the Ph.D. degree in computer science from the National Polytechnic Institute of Mexico, Mexico, in 1997 and 2007, respectively. Since then, he has been with the Center for Computing Research (CIC-IPN), Mexico, where he is currently the Head of the Department of Research in Computer Science.