Artificial Metaplasticity for Deep Learning: Application to WBCD Breast Cancer Database Classification

Juan Fombellida[1(✉)], Santiago Torres-Alegre[1],
Juan Antonio Piñuela-Izquierdo[2], and Diego Andina[1]

[1] Group for Automation in Signals and Communications,
Technical University of Madrid, Madrid, 28040, Spain
jfv@alumnos.upm.es, d.andina@upm.es

[2] Universidad Europea de Madrid, Villaviciosa de Odón, Madrid, Spain

Abstract. Deep Learning is a new area of Machine Learning research that deals with learning different levels of representation and abstraction in order to move Machine Learning closer to Artificial Intelligence. Artificial Metaplasticity are Artificial Learning Algorithms based on modelling higher level properties of biological plasticity: the plasticity of plasticity itself, so called Biological Metaplasticity. Artificial Metaplasticity aims to obtain general improvements in Machine Learning based on the experts generally accepted hypothesis that the Metaplasticity of neurons in Biological Brains is of high relevance in Biological Learning. This paper presents and discuss the results of applying different Artificial Metaplasticity implementations in Multilayer Perceptrons at artificial neuron learning level. To illustrate their potential, a relevant application that is the objective of state-of-the-art research has been chosen: the diagnosis of breast cancer data from the Wisconsin Breast Cancer Database. It then concludes that Artificial Metaplasticity also may play a high relevant role in Deep Learning.

Keywords: Metaplasticity · Deep learning · Plasticity · MLP · MMLP · AMP · WBCD · Feature extraction · Machine learning · Artificial neural network

1 Introduction

In this research we progress on previous works [2], [3], [7], [8]. In the experiments that have been performed in the frame of this investigation, several neural networks belonging to the multiplayer perceptron type have been used to classify the patterns available in the Wisconsin Breast Cancer Database (WBCD)[10].

We will compare the results obtained for the classification of breast cancer data using several different implementations of the Artificial Metaplasticity Multilayer Perceptron (AMMLP) theory [7]. The experiments will use in different forms the AMMLP inherent distribution estimations of patterns in the learning sets.

J.M. Ferrández Vicente et al. (Eds.): IWINAC 2015, Part II, LNCS 9108, pp. 399–408, 2015.
DOI: 10.1007/978-3-319-18833-1_42

Inside each set of experiments several training methods for AMMLP have been used. The first step is to optimize the parameters used in the nominal Backpropagation (BPA) algorithm for well known Multilayer Perceptrons (MLPs) [4] to be sure that the results obtained with the modifications of the method are compared with the best performance possible using BPA. The following experiments use *a posteriori* probability estimation of the input distributions inside the MLP and AMMLP theory to implement the Metaplasticity Learning algorithm. Finally the main objective of the article is covered with the successful results presented, in comparison with state-of-the-art methods.

For assessing this algorithm's accuracy of classification, we used the most common performance measures: specificity, sensitivity and accuracy. The results obtained were validated using the 10-fold cross-validation method.

The remainder of this paper is organized as follows. Section 2 presents a detailed description of the database and the algorithms. In Section 3 the experimental results obtained are present. A brief discussion of these results is showed in Section 4. Finally section 5 summarizes the main conclusions.

2 Materials and Methods

2.1 WBCD Dataset

Breast cancer is a malignant tumor that develops from breast cells. Although research has identified some of the risk factors that increase a woman's chance of developing breast cancer, the inherent cause of most breast cancers remains unknown.

The correct pattern classification of breast cancer is an important worldwide medical problem. Cancer is one of the major causes of mortality around the world and research into cancer diagnosis and treatment has become an important issue for the scientific community. If the cancerous cells are detected before they spread to other organs, the survival rate is greater than 97%. For this reason, the use of classifier systems in medical diagnosis is increasing. Artificial intelligence classification techniques can enhance current research.

This study analyzed the Wisconsin Breast Cancer Database (WBCD). This data base has been used several times in the literature so is very useful in order to compare the results with the state of the art. This data base contains 699 patterns, each of this pattern is composed by 9 numerical attributes that corresponds to different physical characteristics that can be considered as markers of the possible presence of cancer in the sample.

2.2 Data Preparation

Numerically the attributes have been evaluated manually by an expert with values between 1 and 10, being value 1 the closest to an indicator of a benign nature of the sample and value 10 the closest to an indicator of a malicious nature of the sample. The database contains a field that indicates the final diagnosis of the nature of the sample. This value will be used as the ideal output of the networks during the supervised training.

In the original data base there are 16 samples whose attributes are not completely filled. In order to work with a homogeneous set of patterns with all the numerical attributes filled, incomplete elements have been eliminated from the experiment. Finally we will use 683 patterns that are divided in 444 benign samples (65%) and 239 malicious samples (35%).

To obtain results statistically independent of the distribution of the patterns a 10 fold cross validation evaluation method has been considered. Using this method the possible dependence of the results with the distribution of the samples in the training or performance evaluation sets is eliminated: all the samples are used to train the networks and all the samples are used to evaluate the performance of the results in different executions of the experiment for the same initial neural networks, mean values are calculated to establish the final performance results.

It it has empirically been proved that the classifiers based on neural networks produce better results if the training sets are equilibrated presenting the same number of patterns belonging to each one of the possible classes. In order to achieve this situation in the creation of the sets used to train and to evaluate the system some malicious patterns will be repeated instead of eliminating some benign patterns to get these equilibrated sets. It has been considered better to duplicate a small number of malicious elements as inputs for the networks instead of losing the potential information present in some of the benign elements.

For these experiment we have used ten data sets with the following distribution of patterns:

- G1: 90 total patterns: 45 benign and 45 malicious
- G2: 90 total patterns: 45 benign and 45 malicious
- G3: 90 total patterns: 45 benign and 45 malicious
- G4: 88 total patterns: 44 benign and 44 malicious
- G5: 88 total patterns: 44 benign and 44 malicious
- G6: 88 total patterns: 44 benign and 44 malicious
- G7: 88 total patterns: 44 benign and 44 malicious
- G8: 88 total patterns: 44 benign and 44 malicious
- G9: 88 total patterns: 44 benign and 44 malicious
- G10: 90 total patterns: 45 benign and 45 malicious

Using these elements as initial sets we will create 10 different final folders. In each one of the training sets that will be used as inputs to the networks for training the system and evaluating the evolution of the error will consists in 9 of the previous 10 groups. The final evaluation of the performance of the network will use the other element. The 10 folders will be created with the variation of the initial set that is used for evaluation and not for training.

The networks are trained from the same initial conditions presenting the data corresponding to each of the 10 folders. Finally the mean values of the results will be calculated to eliminate the possible statistical influence in the results due to the concrete fixed selection of some patterns to train the system and the fixed selection of other patterns to evaluate the results.

2.3 Artificial Metaplasticity Neural Network Model

ANNs, widely used in pattern classification within medical fields, are biologically inspired distributed parallel processing networks based on the neuron organization and decision-making process of the human brain [8]. In this paper we continue with our previous work [7] applying metaplasticity to the MLP for classifying breast cancer patterns.

The concept of biological metaplasticity was defined in 1996 by Abraham [1] and now is widely applied in the fields of biology, neuroscience, physiology, neurology and others [1], [9]. Recently, Ropero-Peláez [8], Andina [2] and Marcano-Cerdeño [5] have introduced and modeled the biological property metaplasticity in the field of ANNs, obtaining excellent results.

For these experiments Multiplayer Perceptron (MLP) neural networks have been used with a input composed by 9 attributes contained in each single pattern, a hidden layer composed by 8 neurons (previous experiments proved that 8 neurons is enough to get the flexibility needed), and an output layer with just 1 neuron (result of the classification).

The activation function used in all the neurons of the system is sigmoidal, input patterns set normalized and the initialization of the weights of the neurons is random but included in an interval $[-0.5, +0.5]$, parameter value σ in the sigmoidal activation function is constant and equal to 1. Doing this so the range of inputs to the activation function $\sigma \sum \omega_i x_i$ will be limited to the interval $[-4.5, +4.5]$. Then the initial part of the training is compliant with the premise of not saturating the output of the neurons.

To introduce AMP in an arbitrary MLP training [7], all that has to be done is to introduce a weighting function $\frac{1}{f_X^*}$ in the MLP learning equation that has the properties of a probability density function [3]. Then, it is up to the designer to find a function that improves MLP learning. Several have been already proposed [2], [3], [7], [8], and we introduce a new one in this paper.

$$w_{ij}^{(s)}(t+1) = w_{ij}^{(s)}(t) - \eta \frac{\partial E^* [W(t)]}{\partial w_{ij}^{(s)}} = w_{ij}^{(s)}(t) - \eta \frac{1}{f_X^*} \frac{\partial E[W(t)]}{\partial w_{ij}^{(s)}} \tag{1}$$

where $s, j, i \in N$ are the MLP layer, node and input counters, respectively, for each $W(t)$ component, $w_{ij}^{(s)}(t) \in R$ and being $\eta \in R^+$ the learning rate.

The Backpropagation (BP) algorithm presents some limitations and problems during the MLP training [6]. The Artificial Metaplasticity on MLP algorithm (AMMLP) aims to improve BP algorithm by including a variable learning rate $\eta(x) = \frac{\eta}{f_X^*}$ in the training phase affecting the weights in each iteration step based on an estimation of the inherent distribution of training patterns.

Considering this basic methods some alternative weighting functions or alternative AMMLP implementations are considered and studied in this paper. For each modification the same initial networks have been used.

Artificial Metaplasticity by Gaussian Weighting Function or by Inputs *a Posteriori* Distributions. Two cases are considered:

- A given probability distribution known or assumed: One suboptimal solution [7] is:

$$f_X^*(x) = \frac{A}{\sqrt{(2\pi)^N} \cdot e^{B \sum_{i=1}^{N} x_i^2}} = \frac{1}{w_X^*(x)} \tag{2}$$

  where $w_X^*(x)$ is defined as $1/f_X^*(x)$, N is the number of neurons in the MLP input, and parameters A and $B \in R^+$ are algorithm optimization values which depend on the specific application of the AMLP algorithm.

- Considering the estimation of *a posteriori* probability density function. In this case:

$$\widehat{y}_L \cong P(H_l/x) = f_X^*(x) \tag{3}$$

  where $\widehat{y}_L$ is the output of the neuron that estimate the *a posteriori* probability. Equation 3 takes advantage of the inherent *a posteriori* probability estimation for each input class of MLP outputs. Note that if this is not the case, as it happens in first steps of BPA training algorithm, the training may not converge. In this first steps, the outputs of the MLP does not provide yet any valid estimation of the probabilities, but rather random values corresponding to initial guess of the MLP weights. In these first steps of training is better either to apply ordinary BPA training or to use another valid weighting function till BPA starts to minimize the error objective.
  For two classes in the training set we only need one output (for N classes classification problems only N-1 output neurons are needed). If the desired output activation correspond to input vectors $x \in H_1$, then the AMP is implemented by:

$$f_X^*(x) = \widehat{y}_L \tag{4}$$

  and for the complementary class patterns $x \in H_0$:

$$f_X^*(x) = (1 - \widehat{y}_L) \tag{5}$$

  to swap the roles of "1" and "0" is a choice of the designer.

3 Results

3.1 Network Characteristics

In this section we present the results obtained in this research. All the models used in this study were trained and tested with the same data and validated using 10-fold cross-validation. The MLP and AMMLP proposed as classifiers implemented in MATLAB (software MATLAB version R2012a).

Structure of the network

- Number of inputs: equal to the number of attributes of the pattern (9).
- Number of hidden layers: 1
- Number of neurons included in the hidden layer: Based on previous experience 8 neurons are considered ideal for a tradeoff between the flexibility in the definition of the decision regions and the complexity of the system.
- Number of neurons in the output layer: 1 to classify in two classes.
- Activation function: Sigmoidal with output included in the interval $(0, 1)$.

Conditions considered to finalize the network training:

- Reach a defined number of inputs presented to the network to have enough iterations to reach a stable output without overspecializing the network.

3.2 Evaluation Method

In each one of the experiments 50 networks have been considered and trained. Using the 10 fold cross validation method the results are not dependant of the concrete patterns used for training and for performance evaluation. Using 50 different initial networks and calculating mean values we assure that the results are independent of the initial random value in the creation of the networks.

The following hypothesis are defined:

- $H\,(1/1)$: The pattern is malicious and has been classified as malicious.
- $H\,(1/0)$: The pattern is benign and has been classified as malicious.
- $H\,(0/1)$: The pattern is malicious and has been classified as benign.
- $H\,(0/0)$: The pattern is benign and has been classified as benign.

Using these hypothesis it is possible to build the confusion matrix:

Table 1. Confussion matrix model

True Positive $H\,(1/1)$	False Positive $H\,(1/0)$
False Negative $H\,(0/1)$	True Negative $H\,(0/0)$

To evaluate the performance of the classifiers two measures are used and defined as follows:

$$Sensitivity(SE) = \frac{TP}{TP + FN}(\%) \tag{6}$$

$$Accuracy(AC) = \frac{TP + TN}{TP + TN + FP + FN}(\%) \tag{7}$$

Where TP, TN, FP, and FN stand for true positive, true negative, false positive and false negative, respectively.

50 initial networks have been trained and evaluated with the 10 folder cross validation algorithm. From the results obtained for the same network with each one of the folders the mean confusion matrix is obtained for each network. Once these 50 mean values are calculated an additional calculation is made and the final mean value is obtained as the final result of the experiment.

The most important figure in these experiments in the sensitivity (considered as true positive percentage), these is due to the intrinsic nature of the experiment (it is much more important to detect all the malicious patterns than classifying as malicious a benign input).

Nominal Backpropagation Algorithm. Results for this case have been presented in [7] but it is now relevant to add the graphical evolution of learning, shown in Figure 1.

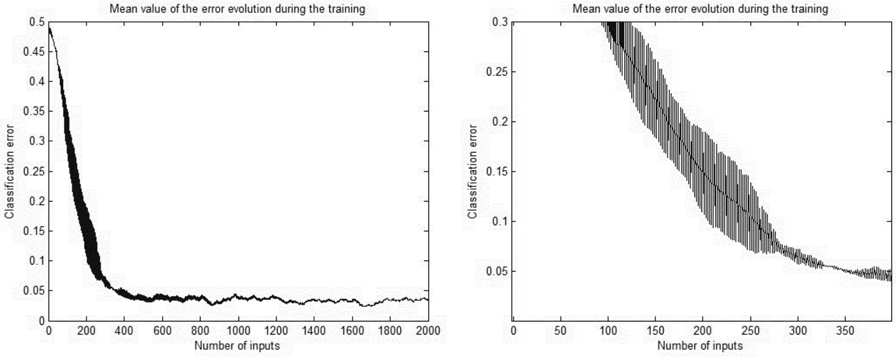

Fig. 1. Evolution of the evolution of the classification error (detail in right figure) $\eta = 1$ - Nominal Backpropagation - 2000 input patterns

Gaussian AMP Function. Results for this case have also been presented in [7] but, as in the case of nominal BPA, it is now relevant to add the graphical evolution of learning, shown in Figure 2

AMP Based on the Output of the Network. In order to check the theoretical approach of applying the *a posteriori* estimation of the probability distribution as AMP function it is necessary to reach a point when the network has started to learn. This experiment will be divided in two parts:

- The initial part of the training will use the backpropagation classic algorithm using learning rate $\eta = 25$ (empirically determined, as in usual BPA) until classification error reach 0.3.

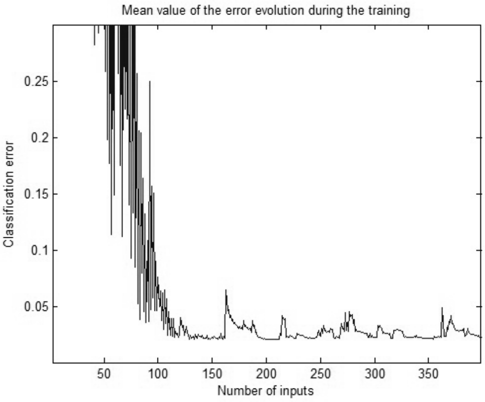

Fig. 2. Detail of the evolution of the classification error $A = 10$ $B = 0.45$ - Gaussian AMP - 2000 input patterns

- The second and principal part of the training will use equations 1, 3, 4 and 5 from error 0.3 to the finalization of the training.

- % Accuracy = 97.9009
- % Sensitivity = 98.8739

Confusion matrix for these parameters rate is shown in table 2.

Table 2. Confusion matrix $\eta = 25$ - Output as probability estimation - 2000 input patterns

98.87 %	3.07 %
1.13 %	96.93 %

The evolution of the classification error during the training phase is shown in Figure 3.

The ROC of this experiment can be found on Figure 4, the area under the curve associated is 0.995.

4 Discussion

- The best sensitivity result is obtained by the Gaussian AMP and the best accuracy result is obtained by AMP based on the output of the network.
- If we compare the results obtained by the nominal BPA, with both AMP training methods we can observe a considerable improvement in the quality of the performance (for both accuracy and sensitivity). We can also observe a considerable quicker learning for AMP, measured in iterations to achieve the final error.

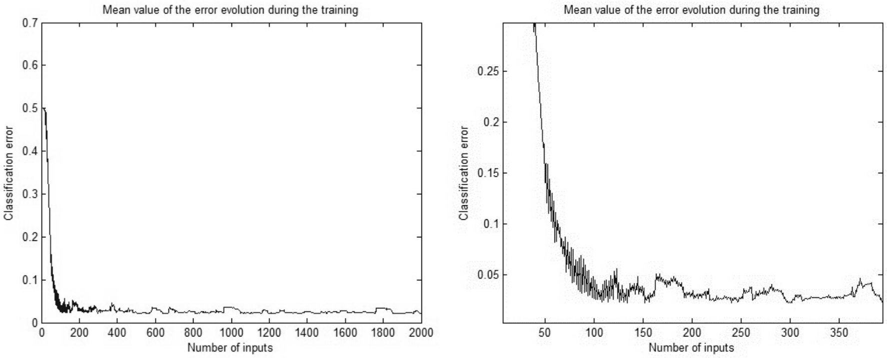

Fig. 3. Evolution of the classification error $\eta = 25$ (detail in right figure) - AMP based on the output of the network - 2000 input patterns

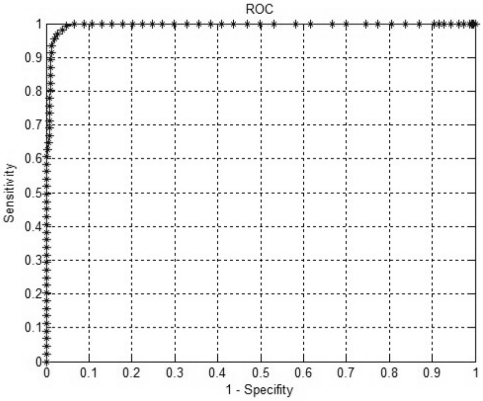

Fig. 4. ROC $\eta = 25$ - AMP based on the output of the network - 2000 input patterns

- Observing the evolution of the classification error, in the first two experiments (BPA and Gaussian AMP) there are a lot of peaks in the learning evolution. In the last AMP where the output of the network is used, the evolution of the error is more natural.

5 Conclusions

In this paper, several alternative Artificial Metaplasticity Learning implementations on MLPs show improved learning over the classical MLPs that corresponds to a uniform plasticity model. Not only in final performance, but also in the evolution of learning. Applied to a challenge and impact relevant classification problem, cancer detection, the learning has been more natural (regular), taken less training steps and provided better performance. Due to the general nature of the Metaplasticity concept, that is as wide as plasticity itself, this results are coherent with the hypothesis that modelling Metaplasticity in Machine Learning must be a relevant issue for Deep Learning Algorithms, as it is plasticity.

References

1. Abraham, W.C.: Activity-dependent regulation of synaptic plasticity (metaplasticity) in the hippocampus. In: The Hippocampus: Functions and Clinical Relevance, pp. 15–26. Elsevier, Amsterdam (1996)
2. Andina, D., Ropero-Pelaez, J.: On the biological plausibility of artificial metaplasticity learning algorithm. Neurocomputing (2012), http://dx.doi.org/10.1016/j.neucom.2012.09.028
3. Andina, D., Alvarez-Vellisco, A., Jevtic, A., Fombellida, J.: Artificial metaplasticity can improve artificial neural network learning. Intelligent Automation and Soft Computing; Special Issue in Signal Processing and Soft Computing 15(4), 681–694 (2009)
4. Andina, D., Pham, D.: Computational Intelligence for Engineering and Manufacturing. Springer, The Nederlands (2007)
5. Benchaib, Y., Marcano-Cedeño, A., Torres-Alegre, S., Andina, D.: Application of Artificial Metaplasticity Neural Networks to Cardiac Arrhythmias Classification. In: Ferrández Vicente, J.M., Álvarez Sánchez, J.R., de la Paz López, F., Toledo Moreo, F. J. (eds.) IWINAC 2013, Part I. LNCS, vol. 7930, pp. 181–190. Springer, Heidelberg (2013)
6. Leung, H., Haykin, S.: The complex backpropagation algorithm. IEEE Transactions on Signal Processing 39(9), 2101–2104 (1991)
7. Marcano-Cedeño, A., Quintanilla-Dominguez, J., Andina, D.: Breast cancer classification applying artificial metaplasticity algorithm. Neurocomputing 74(8), 1243–1250 (2011)
8. Ropero-Pelaez, J., Andina, D.: Do biological synapses perform probabilistic computations? Neurocomputing (2012), http://dx.doi.org/10.1016/j.neucom.2012.08.042
9. Kinto, E.A., Del Moral Hernandez, E., Marcano, A., Ropero Peláez, J.: A preliminary neural model for movement direction recognition based on biologically plausible plasticity rules. In: Mira, J., Álvarez, J.R. (eds.) IWINAC 2007. LNCS, vol. 4528, pp. 628–636. Springer, Heidelberg (2007)
10. http://archive.ics.uci.edu/ml/datasets.html