ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

Parameter-Free Regularization in Extreme Learning Machines with Affinity Matrices

Leonardo J. Silvestre[1,2], André P. Lemos[3], João P. Braga[4], Antônio P. Braga[3] *

1- Graduate Program in Electrical Engineering -
Federal University of Minas Gerais
Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brazil

2- Dept. of Computing and Electronics - Federal University of Espírito Santo
Email: leonardo.silvestre@ufes.br

3 - Dept. of Electronics Engineering - Federal University of Minas Gerais

4 - Dept. of Chemistry - Federal University of Minas Gerais

Abstract. This paper proposes a novel regularization approach for Extreme Learning Machines. Regularization is performed using *a priori* spacial information expressed by an affinity matrix. We show that the use of this type of *a priori* information is similar to perform Tikhonov regularization. Furthermore, if a parameter free affinity matrix is used, like the cosine similarity matrix, regularization is performed without any need for parameter tunning. Experiments are performed using classification problems to validate the proposed approach.

1 Introduction

Regularization techniques have been shown to be useful for improving the generalization capability of neural networks [1]. These techniques impose smoothness constraints to neural network learning, transforming it from an ill-posed problem to a well-posed one. In order to perform regularization, *a priori* information about the problem can be used.

An affinity matrix, or similarity matrix, is a matrix containing measures of similarities between every pair of observations of a given dataset. These matrices have already been used in several data analysis problems, including clustering, dimensionality reduction, image segmentation and link analysis [2]. The structural information represented by an affinity matrix seems to be a good candidate of *a priori* information which may be used to perform regularization.

Extreme Learning Machine (ELM) [3] is a learning algorithm for single layer feedforward networks which can be easily modified to include a regularization term. Several modifications have already been proposed to include a regularization term during the estimation of the output layer weights [4, 5, 6]. However, up to our knowledge, all existing methods have regularization parameters that usually are adjusted using some heuristic, like cross-validation or a statistical information criterion.

This paper proposes a modification on ELM learning algorithm adding a regularization term computed using *a priori* information represented by an affinity matrix. Given the ELM two-step learning process, an affinity matrix is used to

*This work has been supported by the Brazilian agency CAPES.

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

transform each element of the hidden layer output matrix, inserting information regarding the spacial organization of the input samples. Next, the transformed matrix is used to estimate output layer weights. We prove that, if a parameter-free affinity matrix is used, like the cosine similarity matrix, a regularization effect is achieved based on *a priori* structural information without any regularization parameter.

The remainder of this paper is organized as follows. Section 2 presents the proposed learning algorithm and an analytical demonstration of its regularization effect. Next, Section 3 presents the numerical experiment results for supervised and semi-supervised classification problems. Finally, discussions and conclusions are presented in Section 4 .

2 Regularization in ELMs with Affinity Matrices

Given a set of N distinct observations $(\mathbf{x_i}, \mathbf{y_i})$, in which $\mathbf{x_i} = [x_{i1}, x_{i2}, \cdots, x_{im}]^T \in \mathbf{R^m}$ and $y_i \in \mathbf{R}$ for $i = 1, \cdots, N$, a single-layer feedforward network (SLFN) can be used to model these observations as follows:

$$\hat{y_i} = \sum_{j=1}^{l} w_j g(\mathbf{x_i}) = \sum_{j=1}^{l} w_j g(\mathbf{v_j x_i} + b_j), \quad i = 1, \cdots, N \tag{1}$$

in which l is the number of hidden layer neurons, $g()$ is an activation function, $\mathbf{v_j} = [v_{j1}, v_{j2}, \cdots, v_{jm}]^T$ is the weight vector connecting the inputs to the j neuron, w_j is the weight connecting the j neuron to the output and b_j is the j neuron bias term, for $j = 1 \ \cdots \ l$.

In order to a SLFN, composed by l neurons, be able to approximate N observations with a null error there must exist $\mathbf{v_j}$, w_j and b_j, for $j = 1, \cdots, l$ such that:

$$HW = Y \tag{2}$$

with:

$$H = \begin{bmatrix} g(\mathbf{v_1 x_1} + b_1) & \cdots & g(\mathbf{v_p x_1} + b_l) \\ \vdots & \cdots & \vdots \\ g(\mathbf{v_1 x_N} + b_1) & \cdots & g(\mathbf{v_p x_N} + b_l) \end{bmatrix}_{N \times l} \tag{3}$$

$W = [w_1 \ \cdots \ w_l]^T$ and $Y = [y_1 \ \cdots \ y_N]^T$

ELM is a SLFN learning algorithm in which $\mathbf{v_j}$ and b_j for $j = 1 \ \cdots \ l$ are randomly assigned and the output weights are computed using the Moore-Penrose pseudo-inverse:

$$\hat{W} = (H^T H)^{-1} H^T Y = H^+ Y \tag{4}$$

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

The original ELM learning algorithm does not impose any control on the network response smoothness. Regularization techniques can be used to control smoothness and consequently improve ELM generalization performance.

In this paper, regularization is performed by modifying the original ELM algorithm inserting *a priori* information expressed by an affinity matrix. One illustrative example of an affinity matrix is the cosine similarity matrix, in which each element of the resulting $N \times N$ matrix is computed as follows:

$$p_{ij} = \frac{\mathbf{x_i x_j}}{\|\mathbf{x_i}\| \, \|\mathbf{x_j}\|} \tag{5}$$

The procedure proposed in this paper transforms the hidden layer matrix H using an affinity matrix P. Each original value of H is transformed in order to include structural information of the training dataset represented by P. The resulting H' is defined using the following expression:

$$H' = PH \tag{6}$$

The output layer weights are then computed as follows:

$$W' = (H'^T H')^{-1} H'^T Y. \tag{7}$$

Each element h'_{ij} of H' is defined as $h'_{ij} = \sum_{k=1}^{N} p_{ik} h_{kj}$. This expression can be rewritten as $h'_{ij} = p_{ii} h_{ij} + \sum_{k=1, k\neq i}^{N} p_{ik} h_{kj}$. Then, H' can be decomposed into:

$$H' = AH + B, \tag{8}$$

in which

$$A = \begin{bmatrix} p_{11} & 0 & \dots & 0 \\ 0 & p_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & p_{NN} \end{bmatrix}_{N \times N} \tag{9}$$

and

$$B = \begin{bmatrix} \sum\limits_{\substack{1 \leq k \leq N, \\ k \neq 1}} p_{1k} h_{k1} & \sum\limits_{\substack{1 \leq k \leq N, \\ k \neq 1}} p_{1k} h_{k2} & \dots & \sum\limits_{\substack{1 \leq k \leq N, \\ k \neq 1}} p_{1k} h_{kl} \\ \vdots & \vdots & \ddots & \vdots \\ \sum\limits_{\substack{1 \leq k \leq N, \\ k \neq N}} p_{Nk} h_{k1} & \sum\limits_{\substack{1 \leq k \leq N, \\ k \neq N}} p_{Nk} h_{k2} & \dots & \sum\limits_{\substack{1 \leq k \leq N, \\ k \neq N}} p_{Nk} h_{kl} \end{bmatrix}_{N \times l} \tag{10}$$

If matrix P is symmetric and normalized, i.e., $p_{ij} = 1$ if $i = j$, $p_{ij} = p_{ji}$ and $-1 \leq p_{ij} \leq 1$ for $i \neq j$, $A = I_{N \times N}$ and $H' = H + B$. Then, (7) can be rewritten as follows:

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

$$\begin{aligned} W' &= ((H+B)^T(H+B))^{-1}H'^TY \\ &= (H^TH + H^TB + B^TH + B^TB)^{-1}(PH)^TY \\ &= (H^TH + \Lambda)^{-1}H^TP^TY \\ &= (H^TH + \Lambda)^{-1}H^TY', \end{aligned} \tag{11}$$

in which $Y' = P^TY = PY$ and $\Lambda = H^TB + B^TH + B^TB$. Since $B = H' - H$ it is easy to show that $\Lambda = H'^TH' + H^TH$.

Analysing (11), one can note that the use of an affinity matrix during ELM training leads to a Tikhonov regularization effect [7]. The Λ matrix represents the regularization term and it is computed using only H and P. If a parameter-free affinity matrix is used, such as the cosine similarity matrix (5), no regularization parameter is necessary.

Furthermore, please note that (11) uses $Y' = PY$ instead of Y. That is, in order to compute the output layer weights, each output observation y_i is transformed from its original value into a weighted sum of all other output observations, $y'_i = \sum_{k=1}^{N} p_{ik}y_k$, in which the weight associated with a given y_k is defined as the similarity between $\mathbf{x_i}$ and $\mathbf{x_k}$. This modification may improve ELM generalization for semi-supervised learning problems, since by using Y' instead of Y, *a priori* information expressed by P is used to define output labels for unlabeled observations. That is, the output for each unlabeled observation is defined as the weighted sum of all labeled observations outputs.

3 Numerical Experiments

In this section, the proposed modified ELM learning algorithm is evaluated using supervised and semi-supervised classification problems.

Firstly, experiments are performed to evaluate the regularization effect, and consequently the generalization improvement, achieved by the inclusion of the affinity matrix during ELM learning on supervised classification problems.

Next, the proposed algorithm is evaluated using semi-supervised classification problems. It is expected that the use of a modified output vector in (11) and the regularization term would improve the network generalization for semi-supervised problems.

Four binary classification datasets were considered in all experiments: *Australian Credit* (`acr`), *Statlog (Heart)* (`hea`), *Pima Indians Diabetes* (`pid`) and *Wisconsin Breast Cancer* (`wbc`). The datasets were obtained from UCI [8].

All the observations with missing values were removed and the outputs were normalized to be in $\{-1,\ 1\}$. The inputs were normalized to zero mean and unit variance and 10-fold cross validation was used to evaluate accuracy. The sigmoid activation function was used in all experiments and the hidden layer weights were sampled from a uniform distribution within the interval $[-0.5\ \ 0.5]$. The affinity matrix used in all experiments was the parameter-free cosine similarity matrix (5).

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

3.1 Supervised Classification

Firstly, the performance of the proposed method is compared with the original ELM algorithm using supervised learning problems. The comparison is performed to evaluate the regularization effect observed by the inclusion of *a priori* information during learning.

Tables 1-4 details test set accuracy (mean ± stdev) for distinct number of hidden layer neurons (l).

l	Met.	ELM
10	85.51±4.7	80.87±4.5
30	86.81±4.5	86.67±3.6
100	86.52±4.4	84.93±4.6
1000	86.38±4.1	52.46±5.0

Table 1: Classification results for the `acr` dataset

l	Met.	ELM
10	85.56±7.3	81.11±10.5
30	85.93±6.0	84.44±4.9
100	83.70±6.6	75.93±10.8
1000	83.33±7.3	73.70±5.6

Table 2: Classification results for the `hea` dataset

l	Met.	ELM
10	73.84±5.0	75.39±2.7
30	74.48±4.9	75.92±7.1
100	72.91±3.8	73.57±3.9
1000	74.35±5.5	58.98±6.8

Table 3: Classification results for the `pid` dataset

l	Met.	ELM
10	96.79±2.1	96.78±2.0
30	97.51±2.2	96.05±2.2
100	97.36±1.5	95.90±1.5
1000	97.07±2.8	85.35±4.2

Table 4: Classification results for the `wbc` dataset

3.2 Semi-supervised Classification

Finally, the performance of the proposed method is evaluated using semi-supervised problems. All datasets considered in this work are originally supervised classification problems. Thus, in order to evaluate the performance on semi-supervised problems the datasets were artificially modified by removing the labels of a randomly selected subset of the training observations. For different proportions of labeled observations $nl \in \{0.01, 0.05, 0.1, 0.5\}$, $(1-nl)\%$ of the observations were randomly selected and had their outputs set to 0.

Tables 5-8 details test set accuracy (mean ± stdev) for each scenario considered. The number of hidden neurons l was set to 100 in all experiments.

4 Discussions and Conclusions

The use of affinity matrices during ELM learning allowed not only to improve the accuracy on supervised learning problems (in most of the evaluated scenarios) but also prevented overfitting as the number of hidden layer neurons (l) were increased. This is an interesting result since it suggests that the regularization approach proposed may achieve good generalization results without the need to select regularization parameters and/or an optimum number of hidden neurons.

The results obtained for semi-supervised problems suggest that the proposed method is able to achieve better accuracy using only a small fraction of labeled

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

%Labeled	Met.	ELM
0.01	58.55±7.1	67.97±4.4
0.05	77.54±6.2	75.51±7.8
0.1	83.33±4.5	69.28±5.8
0.5	86.96±3.4	84.20±4.5

Table 5: Classification results - semi-supervised case (dataset: `acr`)

%Labeled	Met.	ELM
0.01	57.04±10.6	42.59±10.4
0.05	66.30±5.1	79.63±9.3
0.1	74.81±7.4	70.00±8.5
0.5	80.37±8.2	70.74±9.0

Table 6: Classification results - Semi-supervised case (dataset: `hea`)

%Labeled	Met.	ELM
0.01	56.64±4.5	55.47±4.7
0.05	71.23±4.3	64.21±8.2
0.1	71.88±3.1	59.26±8.2
0.5	73.44±2.8	71.09±3.4

Table 7: Classification results - Semi-supervised case (dataset: `pid`)

%Labeled	Met.	ELM
0.01	93.27±3.3	96.64±1.5
0.05	91.35±3.8	89.75±4.2
0.1	95.91±2.8	85.81±4.9
0.5	97.21±1.5	94.43±2.3

Table 8: Classification results - Semi-supervised case (dataset: `wbc`)

observations. This is also a suggestive result since for several problems it is easy to gather huge amounts of data however costly to label them.

This paper proposes and validates the use of affinity matrices during ELM learning as a viable and promising alternative to perform ELM regularization. An affinity matrix is used to compute a regularization term and expresses *a priori* information about the spacial organization of the input observations used during learning.

Future work shall address the evaluation of different affinity matrices and comparison with alternative ELM regularization algorithms for supervised and semi-supervised problems.

References

[1] Federico Girosi, Michael Jones, and Tomaso Poggio. Regularization theory and neural networks architectures. *Neural Computation*, 7:219–269, 1995.

[2] Rómer Rosales and Brendan Frey. Learning generative models of similarity matrices. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pages 485–492. Morgan Kaufmann Publishers Inc., 2002.

[3] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: a new learning scheme of feedforward neural networks. In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, volume 2, pages 985–990 vol.2, july 2004.

[4] Wanyu Deng, Qinghua Zheng, and Lin Chen. Regularized extreme learning machine. In *Computational Intelligence and Data Mining, 2009. CIDM'09. IEEE Symposium on*, pages 389–395. IEEE, 2009.

[5] Yoan Miche, Mark Van Heeswijk, Patrick Bas, Olli Simula, and Amaury Lendasse. Tropelm: A double-regularized elm using lars and tikhonov regularization. *Neurocomputing*, 74(16):2413–2421, 2011.

[6] Qi Yu, Yoan Miche, Emil Eirola, Mark Van Heeswijk, Eric Séverin, and Amaury Lendasse. Regularized extreme learning machine for regression with missing data. *Neurocomputing*, 102:45–51, 2013.

[7] A. N. Tikhonov. Solution of incorrectly formulated problems and the regularization method. *Soviet Math. Dokl*, 1963.

[8] A. Asuncion and D.J. Newman. UCI machine learning repository, 2007.