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.

An Extreme Learning Approach to Active Learning

Euler Guimarães Horta[1,2] and Antônio Pádua Braga[2] *

1- Instituto de Ciência e Tecnologia
Universidade Federal dos Vales do Jequitinhonha e Mucuri
Diamantina, MG, Brazil

2- Programa de Pós-Graduação em Engenharia Elétrica
Universidade Federal de Minas Gerais
Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brasil

Abstract. We propose in this paper a new active learning method that makes no considerations about the data distribution and does not need to adjust any free parameter. The proposed algorithm is based on extreme learning machines (ELM) and a perceptron with analytical calculation of weights. We show that the proposed model have good results using a reduced set of patterns.

1 Introduction

The main idea of active learning is that the algorithm can interact with the expert and ask for labels only for the most informative patterns. This is important in a scenario where many unlabeled data are available and labels have a high cost. Many authors consider that the most informative patterns are those closer to a separating hyperplane [1, 2, 3, 4]. In order to perform active learning many studies make considerations that can often not be realistic as, for example, considering that data is linearly separable and that the data distribution is uniform [2, 3, 4]. Furthermore, in many methods it is necessary to adjust free parameters what requires an additional labeled dataset in order to perform cross-validation. Many of these methods can use kernel functions to linearize the data in the feature space, but the most common kernel functions require kernel parameters to be adjusted.

All these characteristics of the most common active learning methods motivated the development of a new algorithm which can work with arbitrary distributions of the dataset, without linear separability assumptions, that has only the number of hidden neurons of a Extreme Learning Machine as a parameter to set and that is capable to perform on-line learning. In this study we propose a new active learning algorithm that is based on extreme learning machines [5] using a large number of hidden neurons that must be greater than the dimension of the pattern. The linear separator in the output layer is based on the perceptron algorithm proposed by Fernandez-Delgado et al. [6] that, according to the authors, minimizes the training error and maximizes the margin, behaving as a linear SVM, but being free of tuning parameters and permitting on-line learning.

*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.

This paper is organized as follows: section 2 describes the proposed method; section 3 presents the results of applying the proposed method in different benchmarks; section 4 presents the conclusions.

2 Proposed Method

The neural network training based on Extreme Learning Machines has become very popular in recent years. Huang et al. [5] proposed an algorithm consisting of a neural network with a hidden layer of N neurons with weight vectors i ($\mathbf{w}_i$) randomly chosen, resulting in the weight matrix $W_{ih} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_N]$. The weights of the output layer ($\mathbf{w}_{ho}$) are calculated using a linear classifier which separates the data in the feature space defined by propagation of data through the hidden layer. This idea is related to Cover's theorem [7], which shows that when the data is projected onto a high-dimensional space there is a higher probability to becoming linearly separable. The number of hidden neurons should be larger than the dimension of the input data [8]. Huang et al. [5] showed that the algorithm works correctly only if the number of patterns is equal or greater than the number of hidden neurons. To perform active learning using ELMs we must choose another linear classifier in the output layer because the main objective is to build classifiers using fewer labels. In this case the number of labels could be lesser than the number of hidden neurons.

An algorithm to carry on such a linear separation is the model proposed recently by Fernandez-Delgado et al [6]. They proposed a new analytical approach to train perceptrons that maximizes margin and minimizes error.

According to the authors, the algorithm works as a SVM in which all training patterns are support vectors with Lagrange multipliers equal to 1. The weights are adjusted as follows:

$$\mathbf{w}_0 = \frac{\sum_{k=1}^{N} d_k \mathbf{x}_k}{\| \sum_{k=1}^{N} d_k \mathbf{x}_k \|} \tag{1}$$

For deductions and more details see [6]. In this equation $\mathbf{x}_k$ is a pattern and d_k is the corresponding label. To obtain the classification to a desired pattern just calculate $y(\mathbf{x}) = sign(\mathbf{w}_0^T \mathbf{x})$.

This approach does not need to adjust free parameters and allows on-line learning, since it simply add the new patterns, multiplied by its desired output, to the weight vector and normalizes the resulting vector again to obtain $||\mathbf{w}_0|| = 1$. Thus it is expected that active learning could improve the generalization capability of this model, by learning only the most relevant patterns.

The question we must answer is: how to find the optimal number of labels needed to obtain maximum generalization capability for this model? We believe that a good way is to extend the well known perceptron convergence theorem [9] for the model of Fernandez-Delgado et al [6] and use the result as a criterion to decide which patterns should be labeled.

The perceptron convergence theorem states that the Rosenblatt's algorithm [9] will converge with a number of iterations less than or equal to a maximum

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.

value if the problem is linearly separable [10]. In this algorithm a pattern is learned by perceptron only if classification has been incorrect. The proof of this theorem, proposed by [11], can be extended to Fernandez-Delgado's perceptron. For this perceptron all the available patterns are used for training, regardless if its classification was correct or incorrect. Thus, it is easy to extend the deduction of this theorem for this model to obtain the maximum number of labels needed to ensure the algorithm's convergence:

$$n_{max} = \frac{\beta + 2\theta}{\alpha^2} \tag{2}$$

Where

$$\alpha = \min_{\mathbf{x}(n)\in\zeta} |\mathbf{w}^T\mathbf{x}(n)|, \qquad \beta = \max_{\mathbf{x}(k)\in\zeta} ||\mathbf{x}(k)||^2, \qquad \theta = \max_{\mathbf{x}(k)\in\zeta} |\mathbf{w}^T(k)\mathbf{x}(k)| \tag{3}$$

and ζ is the training set.

Equation 2 shows that Fernandez-Delagado's perceptron converges using at most $\frac{\beta+2\theta}{\alpha^2}$ labels.

2.1 Extreme Active Learning Machines

We will build an active learning model that consists of an ELM hidden layer and an output layer based on Fernandez-Delgado's perceptron. The first layer will project the data onto ELM feature space and the second one will perform a linear separation of the data in this new space. Algorithm 1 presents the proposed method. In this algorithm a pattern of a set of candidates patterns C is randomly chosen and projected onto ELM feature space and its label will be queried only if the calculated n_{max} is greater than the number of already learned patterns, otherwise the pattern is disposed. The process continues until evaluates all candidate patterns or until the maximum number of labels is reached. This maximum number depends of the available resources to obtain labels from the expert.

3 Experiments and Results

The experiments in this study aim to compare the proposed method, here called EALM, with perceptrons of Dasgupta et al [3], here called *PDKCM*, and Cesa-Bianchi et al [2], here called *PCBGZ*, with Tong et al [1] SVM, here called *SVMTK* and with an SVM trained using all training patterns, here called *SVMALL*. Our goal is to use these methods as linear outputs for an ELM hidden layer applied in non-linearly separable problems. In all cases, the ELM hidden layer is the same, consisting of 1000 neurons and with weights randomly selected in the range [-3, 3], as proposed by [8]. Patterns with missing values are removed. All datasets used in this work are from UCI Machine Learning repository [12].

In all cases, 30% of the datasets have been separated to adjust free parameters for the models *PDKCM*, *PCBGZ* and the *SVMs*. The adjustment was made using the *10-fold cross-validation*, similarly to the work of Monteleoni et al [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.

Input : Candidates set C, initial size of the training set m, maximum number of labels L
Output : Weight vector $\mathbf{w}$
Method:

Propagate all patterns from C through the ELM layer to create the set C_{ELM};
Take and remove at random m pattens from C_{ELM} and query its labels;
Calculate $\mathbf{w}_{ho} = \frac{\sum_{k=1}^{m} d_k \phi(\mathbf{x}_k)}{||\sum_{k=1}^{m} d_k \phi(\mathbf{x}_k)||}$;
repeat
  Take and remove at random a pattern $\mathbf{x}$ from C_{ELM};
  Calculate α, β and θ using equation 3;
  Calculate $n = \frac{\beta+2\theta}{\alpha^2}$;
  if $n > m$ **then**
    Query the label d to pattern $\mathbf{x}$;
    Update $\mathbf{w} = \frac{\mathbf{w}+d\phi(\mathbf{x})}{||\mathbf{w}+d\phi(\mathbf{x})||}$;
    $m = m + 1$;
  end
until *($m = L$) or ($C_{ELM} = \emptyset$)* ;

Algorithm 1: Extreme Active Learning Machine

This adjustment was carried out to obtain the best possible AUC (Area Under the ROC Curve) with a reduced number of labels. For the model PCBGZ, we also tested the optimal parameter $b = (max_{x \in C}||x||^2)/2$ [2], and we called the resulting model as *PCBGZ-OPT*. For SVMs the regularization parameter C has been set using the values of range $\{2^{-5}, 2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 1, 2, 2^2, ..., 2^{14}\}$, as proposed by [6].

With the remainder data was performed *10-fold cross-validation* obtaining the mean accuracy, mean AUC and the average number of used labels. The cross-validation was performed 10 times. All data was normalized in order that the input patterns have zero mean and unity standard deviation.

The results are shown in Table 1. The number of labels used effectively by a model consists of the labels used to adjust free parameters and the labels obtained from the active learning process, so Table 1 shows the number of labels achieved by active learning (AL Labels) and the number of labels effectively used (Effective Labels). As can be seen, the best results were obtained for *EALM* model and for *SVMTK* model.

4 Conclusion

Active learning has attracted attention of many researchers in recent years because large amounts of data has been generated and labeling can have a high

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.

cost. This motivates developing good classifiers using fewer labels. Moreover most of the existing models needs to adjust free parameters what requires an additional labeled dataset for this purpose.

The *EALM* model makes no constraints on the data distribution and does not require tuning free parameters. If we consider that the total number of labels required for active learning is the sum of the labels used to adjust the free parameters with the number of labels used for training, we can conclude that the *EALM* model uses fewer patterns than the other models and it is more practical. The closeness of *EALM* results with those obtained by the SVM based models is certainly due to the fact that the Fernandez-Delgado's perceptron [6] works as a SVM where all training patterns are considered support vectors. Furthermore, *EALM* only stores the weights vector which allows on-line learning, with low computational cost if compared with SVMs.

References

[1] Simon Tong and Daphne Koller. Support Vector Machine Active Learning with Applications to Text Classification. *Journal of Machine Learning Research*, pages 45–66, 2001.

[2] Nicolò Cesa-Bianchi, Claudio Gentile, and Luca Zaniboni. Worst-Case Analysis of Selective Sampling for Linear Classification. *Journal of Machine Learning Research*, 7:1205–1230, 2006.

[3] Sanjoy Dasgupta, Adam Tauman Kalai, and Claire Monteleoni. Analysis of Perceptron-Based Active Learning. *Journal of Machine Learning Research*, 10:281–299, 2009.

[4] Claire Monteleoni and Matti Kääriäinen. Practical Online Active Learning for Classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 328, 2007.

[5] G Huang, Q Zhu, and C Siew. Extreme learning machine: Theory and applications. *Neurocomputing*, 70(1-3):489–501, December 2006.

[6] Manuel Fernandez-Delgado, Jorge Ribeiro, Eva Cernadas, and Senén Barro Ameneiro. Direct parallel perceptrons (DPPs): fast analytical calculation of the parallel perceptrons weights with margin control for classification tasks. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, 22(11):1837–48, November 2011.

[7] T M Cover. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *Ieee Transactions On Electronic Computers*, EC-14(3):326–334, 1965.

[8] Benoît Frénay and Michel Verleysen. Using SVMs with randomised feature spaces: an extreme learning approach. In *European Symposium on Artificial Neural Networks - Computational Intelligence and Machine Learning*, number April, pages 315–320, 2010.

[9] F. Rosenblatt. *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Spartan, Washington, DC, 1962.

[10] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Macmillan, New York, 1994.

[11] N. Nilsson. *Learning Machines*. McGraw-Hill, New York, 1965.

[12] A. Frank and A. Asuncion. Uci machine learning repository. 2010.

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.

Table 1: Average results of 10 runs of 10-fold cross-validation

Key	AL Labels	Effective Labels	Ac	AUC
		EALM		
HRT	88.58 ± 5.97	88.58	0.85 ± 0.01	0.84 ± 0.01
WBCO	39.10 ± 2.79	39.10	0.98 ± 0.00	0.98 ± 0.00
WBCD	80.07 ± 3.96	80.07	0.97 ± 0.01	0.97 ± 0.01
PIMA	188.16 ± 6.18	188.16	0.76 ± 0.01	0.72 ± 0.01
SNR	103.89 ± 2.88	103.89	0.71 ± 0.03	0.72 ± 0.03
ION	105.30 ± 5.86	105.30	0.89 ± 0.01	0.87 ± 0.02
AUST	137.92 ± 8.34	137.92	0.86 ± 0.01	0.86 ± 0.01
LIV	163.54 ± 5.43	163.54	0.60 ± 0.02	0.60 ± 0.02
GER	295.71 ± 8.26	295.71	0.75 ± 0.01	0.68 ± 0.01
SPAM	515.48 ± 18.56	515.48	0.92 ± 0.00	0.91 ± 0.00
		PDKCM		
HRT	65.18 ± 1.43	146.18	0.77 ± 0.02	0.77 ± 0.02
WBCO	63.46 ± 2.50	268.46	0.97 ± 0.00	0.97 ± 0.01
WBCD7	36.01 ± 0.87	207.01	0.93 ± 0.02	0.92 ± 0.02
PIMA	131.02 ± 3.86	361.02	0.72 ± 0.02	0.70 ± 0.02
SNR	77.01 ± 1.85	139.01	0.71 ± 0.03	0.72 ± 0.03
ION	51.01 ± 1.77	156.01	0.83 ± 0.03	0.80 ± 0.03
AUST	145.56 ± 3.18	352.56	0.83 ± 0.01	0.83 ± 0.01
LIV	142.22 ± 3.56	246.22	0.61 ± 0.02	0.61 ± 0.02
GER	247.85 ± 5.07	547.85	0.71 ± 0.01	0.64 ± 0.01
SPAM	314.13 ± 6.07	1694.13	0.87 ± 0.01	0.87 ± 0.01
		PCBGZ		
HRT	50.83 ± 2.10	131.83	0.78 ± 0.02	0.78 ± 0.02
WBCO	182.32 ± 2.98	387.32	0.97 ± 0.01	0.97 ± 0.01
WBCD	203.67 ± 2.79	374.67	0.96 ± 0.01	0.96 ± 0.01
PIMA	193.01 ± 4.35	423.01	0.71 ± 0.01	0.69 ± 0.02
SNR	93.28 ± 1.77	155.28	0.71 ± 0.03	0.70 ± 0.02
ION	116.28 ± 3.33	221.28	0.86 ± 0.01	0.83 ± 0.02
AUST	170.40 ± 2.75	377.40	0.82 ± 0.02	0.81 ± 0.02
LIV	156.05 ± 2.65	260.05	0.59 ± 0.03	0.59 ± 0.03
GER	180.43 ± 3.37	480.43	0.70 ± 0.01	0.63 ± 0.01
SPAM	1168.10 ± 12.43	2548.10	0.89 ± 0.01	0.89 ± 0.01
		PCBGZ-OPT		
HRT	168.47 ± 0.36	249.47	0.77 ± 0.03	0.76 ± 0.03
WBCO	424.06 ± 0.74	629.06	0.97 ± 0.01	0.96 ± 0.01
WBCD	353.04 ± 0.54	524.04	0.96 ± 0.01	0.96 ± 0.01
PIMA	480.08 ± 0.51	710.08	0.69 ± 0.02	0.67 ± 0.02
SNR	130.46 ± 0.30	192.46	0.71 ± 0.03	0.70 ± 0.02
ION	219.55 ± 0.33	324.55	0.86 ± 0.03	0.83 ± 0.03
AUST	430.49 ± 0.61	637.49	0.79 ± 0.01	0.79 ± 0.01
LIV	215.38 ± 0.29	319.38	0.59 ± 0.03	0.59 ± 0.03
GER	624.64 ± 0.63	924.64	0.70 ± 0.01	0.64 ± 0.02
SPAM	2860.56 ± 1.69	4240.56	0.89 ± 0.01	0.89 ± 0.01
		SVMTK		
HRT	45.70 ± 4.24	126.70	0.82 ± 0.01	0.82 ± 0.01
WBCO	18.70 ± 1.83	223.70	0.97 ± 0.00	0.97 ± 0.00
WBCD	40.00 ± 2.79	211.00	0.98 ± 0.00	0.97 ± 0.00
PIMA	138.60 ± 26.06	368.60	0.75 ± 0.01	0.72 ± 0.01
SNR	59.60 ± 9.71	121.60	0.75 ± 0.02	0.75 ± 0.02
ION	55.80 ± 5.77	160.80	0.90 ± 0.01	0.87 ± 0.01
AUST	68.90 ± 13.90	275.90	0.85 ± 0.01	0.85 ± 0.01
LIV	96.40 ± 26.01	200.40	0.64 ± 0.02	0.64 ± 0.02
GER	207.40 ± 17.33	507.40	0.74 ± 0.01	0.66 ± 0.01
SPAM	340.00 ± 25.26	1720.00	0.92 ± 0.00	0.92 ± 0.00
		SVMALL		
HRT	170.00	251.00	0.81 ± 0.01	0.83 ± 0.08
WBCO	430.00	635.00	0.97 ± 0.00	0.97 ± 0.02
WBCD	358.00	529.00	0.98 ± 0.00	0.99 ± 0.02
PIMA	485.00	715.00	0.74 ± 0.01	0.71 ± 0.08
SNR	131.00	193.00	0.81 ± 0.01	0.82 ± 0.10
ION	221.00	326.00	0.91 ± 0.01	0.90 ± 0.07
AUST	434.00	641.00	0.84 ± 0.01	0.84 ± 0.04
LIV	217.00	321.00	0.67 ± 0.02	0.62 ± 0.11
GER	630.00	930.00	0.75 ± 0.01	0.64 ± 0.04
SPAM	2898.00	4278.00	0.93 ± 0.00	0.93 ± 0.02