Forecasting Algorithm Developed from the Neuro - Fuzzy System

Phichit Napook
Center of Excellence in Digital Socio-Economy,
Social Research Institute,
Chiang Mai University,
Thailand
phichit_n@cmu.ac.th

Narissara Eiamkanitcaht
[1]Department of Computer Engineering, Faculty of Engineering,
[2]Social Research Institute,
Chiang Mai University,
Thailand
narisara@eng.cmu.ac.th

Ubicomp/ISWC'15 Adjunct, September 07 - 11, 2015, Osaka, Japan

ISBN 978-1-4503-3575-1/15/09 $15.00
DOI: http://dx.doi.org/10.1145/2800835.2800983

Abstract

Forecasting is the process to estimate the future by analysis the past and present information. In data analysis the accurate forecast results are usually based on the reliable classification algorithm. This paper proposes the new Neuro-Fuzzy algorithm combined with the quadratic function. The classification performance of the propose algorithm is shown by the accuracy of 7 standard datasets from the UCI machine learning repository. The satisfactory classification accuracy is shown in the experiment. The proposed structure can classify by both direct calculation and using rule based classification. The classification structure after training process is easy to apply in any application.

Author Keywords

Data classification; forecasting algorithm; Neuro-fuzzy system.

ACM Classification Keywords

I.2.1: Applications and Expert Systems

Introduction

Forecasting or prediction is a greater role in planning the strategy in many organizations. Furthermore the forecast algorithm is applied in many fields of science work. In [1], the research proposed the usage of neural network to forecast the need of electrical power consumption. The neural network algorithm also used to forecast the lowest temperature of the Chandigarh in [2]. The neural network can be used to find the relationship of the input data and output data for calculating the final result. The structure of neural networks can apply to modify for use with various datasets for prediction purpose. Many researchers use this benefit structure to use in the learning process. Furthermore, the robustness with the unseen data is appropriate to use as the prediction tool. In [3-5] use neural network to combine with the fuzzy logic system, which outstanding of data interpretable. The new structure called the neural-fuzzy systems, those have proven the better performance in many experimental [6-10].

In [7-8] present the structure of high performance Neuro-fuzzy. The structure includes the 3 fuzzy membership function creation embedded in the neural network. The original data is transformed by the fuzzy values and use in the learning process of neural network. The parameters in neural network use the back propagation algorithm to update. The parameters of fuzzy membership also update simultaneously. After finishing the trained process, the final parameters can be used as a rule base classification with high accuracy. Moreover, the sorted of weights results can be used as the feature selection which have been proved in the experiment as the high informative features by results the high accuracy rate after used in the classification task. In [9] has extended the concept from [8] to develop the structure and find the appropriate number of the membership functions. The success of [9] is continued developing in [10] by using Golden section search to search the proper number of features for creating the rule based classification. The Golden section search technique is used to find the maximum or minimum value of the function. The results show the development of the system.

This research proposed the development of adaptive dynamic clustering of Neuro-fuzzy system for classification [10] by applying the Quadratic function to search the appropriate number of features for used in the rule based classification and reduced the complexity of using the Golden section search.

The proposed method

The algorithm in this paper improves the searching part from algorithm in [10], which present higher performance of classification than other algorithms. The structure of the system shown in figure 1. The details of each section is described as follows.

a. Dynamic clustering

The first step will perform the dynamic clustering. Each feature is sorted and applied the incremental clustering. The first sample is assigned to cluster 1. The following sample will assigned to cluster 1, if the value of distance function is less than threshold, otherwise assigned to new cluster. The process is iterative until all samples have been assigned cluster. The average value of membership in clusters is used as the criteria to agglomerate the cluster. The cluster member that has

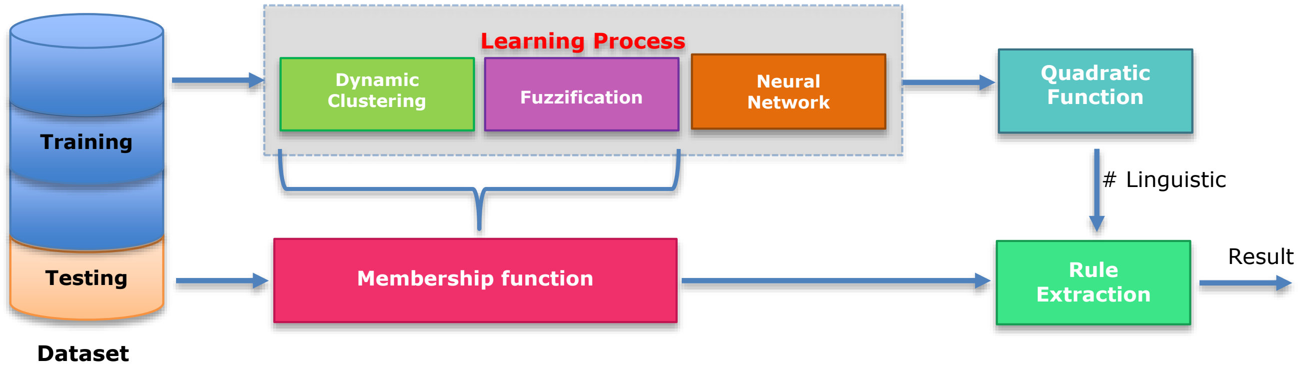

Figure 1: The structure of the proposed model.

less than the threshold are agglomerated, until all clusters have member greater than threshold. Finally, combined the cluster that has the same meaning together.

b. Fuzzification

This part the clusters from previous step to create the membership function. The Gaussian membership functions are used to calculate membership value as

$$G_{nk} = \begin{cases} 0, \text{ if } \sigma_{nk} = 0 \text{ and } x_n \neq \mu_{nk} \\ e^{\left(-\frac{(x_n - \mu_{nk})^2}{2\sigma_{nk}^2}\right)}, \quad \text{if } \sigma \neq 0 \\ 1, \text{ if } \sigma_{nk} = 0 \text{ and } x_n = \mu_{nk} \end{cases} \quad (1).$$

The value G_{nk} is the membership value, n is the considered feature, the letter k represents the cluster of the dataset, x_n is the original input, μ_{nk} is the mean value of the cluster and finally, σ_{nk} is the standard deviation value of the cluster. The result from a fuzzy process use as the input in neural network process. The value of fuzzy membership value is transformed to the binary value by

$$b_{nk} = \begin{cases} 1, \text{ if } G_{nk} \text{ is maximum} \\ 0, \text{ othewise} \end{cases} \quad (2).$$

The b_{nk} is the binary value of cluster k in the feature n .

c. Neural network

The structure of the neural network in this work is the single layer perceptron. The input layer has a number of nodes equal to the number of the cluster from the dynamic cluster process. The output layer has a number of nodes equal to the number of classes of dataset. The log Sigmoid is used as the activation function for all nodes. The output from each node defined by

$$y_o = \frac{1}{(1+e^{-V_o})} \quad (3),$$

where

$$V_o = \sum_{i=1}^{n}\sum_{j=1}^{k} b_{ij} w_{ijo} \quad (4).$$

The b_{ij} is the binary value of the feature i at a node j, the w_{ijo} is the weight of the output node. At output layer the error of each node can calculated from

$$e_o(l) = d_o(l) - y_o(l) \quad (5).$$

The $e_o(l)$ is the error of the output node O at the iteration l, $d_o(l)$ is the desired output and $y_o(l)$ is the output value. The gradient at output layer is defined by

$$\delta_o(l) = e_o(l)\varphi'(V_o(l)) \quad (6).$$

The weights of the network can be modified by

$$w_{ijo}(l+1) = w_{ijo}(l) - \eta\delta_o(l) b_{ij}(l) \quad (7).$$

d. Quadratic function

After training process, the appropriate number of weights is needed to create the rule based classification. The Quadratic function is used in this part to select the set of informative weights. The quadratic function is a method to find the proper value. First step initializes the value of parameters x_0, x_1, x_2 and x_3 to find the proper number of weights to create the rule. The value x_0 is set to zero, the x_1 is the random value between *0* to maximum numbers of weights. The value x_2 is the maximum numbers of weights. All values are used in the following equation

$$f(x) = \sum_{k=1}^{x} A_k,\; A = \begin{cases} 1, \text{ if } w_k \text{ is } b_{w_k} \textit{ equal to } 1 \\ 0, \quad \text{otherwise} \end{cases} \quad (8).$$

The algorithm to find the value of $f(x_0), f(x_1)$ and $f(x_2)$ described in Figure 2.

Algorithm Calculate Function Value (x)

```
Sort weights
Sum=0
For i=1 to all training data
    j=0
    While A~=1 and j<x
        If b_nk of w_nko is 1
            A=1
        Else  A=0
        End
        j=j+1
    End
    Sum=Sum+A
End
Return Sum
```

Figure 2: Algorithm to Calculate Function.

The next step calculates the value of x_3 from

$$x_3 = \frac{f(x_2)(x_0^2 - x_1^2) + f(x_0)(x_1^2 - x_2^2) + f(x_1)(x_2^2 - x_0^2)}{2f(x_2)(x_0 - x_1) + 2f(x_0)(x_1 - x_2) + 2f(x_1)(x_2 - x_0)} \quad (9).$$

The value of x_3 can calculate the value of $f(x_3)$ following algorithm in Figure 2. If the value of the $f(x_3)$ is equal to the value of $f(x_2)$ then let $x_2 = x_3$, $f(x_2) = f(x_3)$. Iterate the random process of x_1 in range x_0 to x_2, then calculate the value of $f(x_1)$. If the value of $f(x_3)$ is less than the value of $f(x_2)$ then let $x_0 = x_3$, $f(x_2) = f(x_3)$. Then lets $x_1 = x_3$, $f(x_1) = f(x_3)$ and calculate the value of x_3. Perform the iteration process until the value of $f(x_0) = f(x_1)$ then uses the value of x_1 to create the rule based classification rules.

Experimental method and results

The dataset from the UCI machine learning [11] is used to verify the accuracy performance of the proposed method.

Dataset	Size	Feature	Cluster	Type
Wine	178	13	3	Integer, Real
Breast Canner	699	9	2	Integer
Iris	150	4	3	Real
Vote	435	16	2	Categorical
Zoo	101	16	7	Integer, Categorical
Seeds	210	7	3	Real
Liver Disorder	345	6	2	Integer, Real, Categorical

Table 1: Details of the dataset use in the experimental.

The classification results from the proposed method compared to other methods are displayed in table1.The method of the 10 fold cross validation is used in all experiments. The results from direct calculate shown in table 2 and results from rule extraction shown in table 3 as follows.

Average accuracy of 10 fold cross validation		
Dataset	**GSS[10]**	**Quadratic**
Wine	99.44%	99.44%
Breast Canner	97.14%	97.14%
Iris	96.67%	96.67%
Vote	96.78%	96.78%
Zoo	97.03%	97.03%
Seeds	93.81%	93.81%
Liver Disorder	69.86%	69.86%

Table 2: The average accuracy from direct calculate of the proposed method compared with method proposed in [10].

Average accuracy of 10 fold cross validation		
Dataset	**GSS[10]**	**Quadratic**
Wine	99.44%	99.44%
Breast Canner	95.71%	95.71%
Iris	96.67%	96.67%

Average accuracy of 10 fold cross validation		
Dataset	**GSS[10]**	**Quadratic**
Vote	96.09%	96.09%
Zoo	93.07%	93.07%
Seeds	91.90%	91.90%
Liver Disorder	69.86%	69.86%

Table 3: The average accuracy from Rule extraction of the proposed method compared with method proposed in [10].

The classification results in table 2 and table 3 shown the same average accuracy in all datasets.

Time of 10 fold cross validation (sec.)		
	GSS[10]	**Quadratic**
Wine	892.92	754.31
Breast Canner	1,377.45	1302.13
Iris	95.44	93.09
Vote	707.30	706.74
Zoo	152.39	147.96
Seeds	616.67	581.13
Liver Disorder	726.05	707.53

Table 4: The execution time comparison.

However to compare the execution time as shown in table 4, it is clearly seen that the proposed algorithm in this paper use less time than [10] in every dataset. Comparing to the used of the Golden section method, results in table 2 to table 3 show the same performance of using Quadratic function in all datasets. The reason is the Golden section method using constant values in the search algorithm, where the Quadratic method is using parameters that can converge to the solution faster.

The next experiment is using the algorithm propose in this paper with the real world dataset. The following dataset is the carbon dioxide equivalent calculate from the activity of students in Chiang Mai University dormitory. The details of a dataset is the carbon dioxide equivalent of 27 activities from 18 dormitories. Figure 3 shows the scatter plot of the features in this dataset. The carbon dioxide from electrical consumption is represented in x axis of all graphs. The values of y axis from graph 1 to graph 6 in figure 3 are the carbon dioxide equivalent of electrical consumption, water consumption, fuel from motorbike that produce Co2, CH4, N2O and Greenhouse gases respectively. It is quite clear from the graph that all dormitories can divided into 2 clusters. The K-mean is applied and the dormitories are divided into 2 classes that is the high carbon dioxide emission and low carbon dioxide emission.

The average accuracy of 10 fold cross validation classification results from the student dormitories from both direct calculate and rule is 88.23%, show the good performance of using algorithm with the real word dataset.

Conclusion

The experimental results shown that the accuracy of classification of the proposed algorithm is comparable to the structure proposed in [10], but use less execution time. Meaning that the overall performance of the proposed method is better than the method proposed in [10]. Various types of dataset are used in the experimental to show the generalization of the proposed Neuro-fuzzy classification structure. The experimental result shows that the propose structure can be applied in many applications to forecasting the usage of resources, for managing the resource planning in the future.

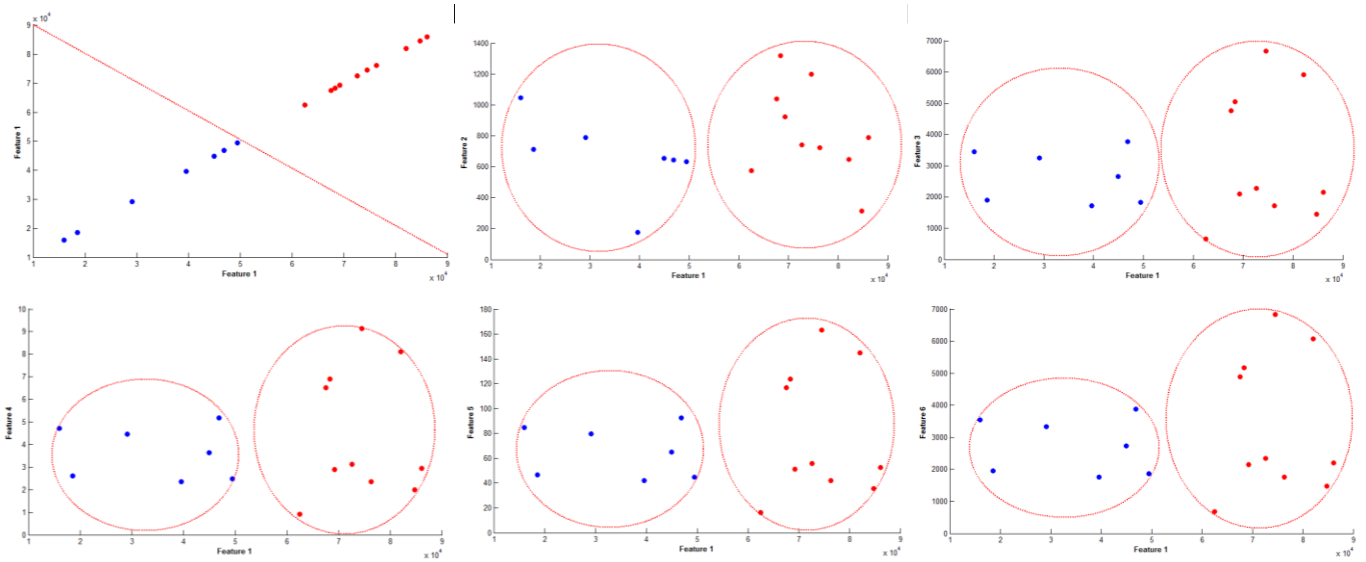

Figure 3: Scatter plot of the carbon dioxide equivalent.

Acknowledgements

This paper was supported in part by the Coordinating Center for Thai Government Science and Technology Scholarship Students (CSTS) National Science and Technology Development Agency (NSTDA).

References

1. Zhang H.T., Xu, F.Y. and Zhou, L.: Artificial neural network for load forecasting in smart gird. *In 2010 International Conference on Machine Learning and Cybernetics (ICMLC)*, pp. 3200-3205.
2. Kaur A., Singh H. 2011. Artificial Neural Network in forecasting Minimum Temperature. *In International Journal of Electronics & Communication Technology*, pp. 101-105.
3. Haykin S. 2008. Neural Networks and Learning Machines. Prentice Hall, New York.

4. Badiru A.D., Cheung, J.Y. 2002. Fuzzy Engineering Expert System with Neural Network Applications. John Wiley, New York.
5. Kriesel, D. 2007. A Brief Introduction to Neural Networks. Retrieved August, 15, 2011, pp. 37-124.
6. Chakraborty D., Pal N.R. 2004. A Neuro-Fuzzy Scheme for Simultaneous Feature Selection and Fuzzy Rule-Based Classification. *In Proceedings of IEEE Transactions on Neural Network*, 110-123.
7. Eiamkanitchat N., Theera-Umpon N., Auephanwiriyakul S. 2010. A Novel Neuro-Fuzzy Method for Linguistic Feature Selection and Rule-Based Classification. *In Proceedings of the 2nd International Conference on Computer and Automation Engineering* (ICCAE), 247-252.
8. Eiamkanitchat N., Theera-Umpon N., Auephanwiriyakul S. 2010. Colon Tumor Microarray Classification Using Neural Network with Feature Selection and Rule-Based Classification. *In Proceedings of Advances in Neural Network Research and Applications*, 363-372.
9. Wongchomphu P., Eiamkanitchat N. 2014. Enhance Neuro-Fuzzy System for Classification Using Dynamic Clustering. *In Proceedings of the 4th Joint International Conference on Information and Communication Technology, Electronic and Electrical Engineering (JICTEE)*, 1-6.
10. Napook P., Eiamkanitchat N. 2015. The Adaptive Dynamic Clustering of Neuro-Fuzzy System for Classification. *In Proceedings of 6th International Conference on Information Science and Applications (ICISA)*, 721-728.
11. Bache, K, Lichman, M, (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.