Report Number: WUCS-93-44

1993

Supervised Competitive Learning Part II: SCL with Fuzzy Logic

Ce Wang, Takayuki D. Kimura, and Thomas H. Fuller Jr.

Supervised Competitive Learning (SCL) is described in an accompanying paper [1]; SCL assembles a set of learning modules into a supervised learning system to address the stability-plasticity dilemma. That paper reported results using backpropagation networks as the learning modules (SCL/BP). Here (Part II) we report SCL results using learning modules based on fuzzy logic (SCL/FZ). Although its learning algorithm is very different from that of backpropagation networks, fuzzy logic also suffers the stability-plasticity dilemma. A simulator on handwritten digit and gesture recognition was constructed to demonstrate the utility of SCL/FZ; it is recognized 98% of test digits, and 91%... Read complete abstract on page 2.

Supervised Competitive Learning Part II: SCL with Fuzzy Logic

Ce Wang, Takayuki D. Kimura, and Thomas H. Fuller Jr.

Complete Abstract:

Supervised Competitive Learning (SCL) is described in an accompanying paper [1]; SCL assembles a set of learning modules into a supervised learning system to address the stability-plasticity dilemma. That paper reported results using backpropagation networks as the learning modules (SCL/BP). Here (Part II) we report SCL results using learning modules based on fuzzy logic (SCL/FZ). Although its learning algorithm is very different from that of backpropagation networks, fuzzy logic also suffers the stability-plasticity dilemma. A simulator on handwritten digit and gesture recognition was constructed to demonstrate the utility of SCL/FZ; it is recognized 98% of test digits, and 91% of test gestures. In this paper, we also compare SCL/BP with SCL/FZ for recognizing handwritten digits.

Supervised Competitive Learning Part II: SCL with Fuzzy Logic

Ce Wang, Takayuki D. Kimura and Thomas H. Fuller, Jr.

WUCS-93-44

September 1993

Department of Computer Science
Washington University
Campus Box 1045
One Brookings Drive
St. Louis, Missouri 63130-4899

Reprinted from Intelligent Engineering through Artificial Neural Networks, Volume 2. New York: ASME Press, 1992, C. Dagli et al ed., pages 351-356. This work supported by the Kumon Machine Project.

Supervised Competitive Learning
Part II: SCL with Fuzzy Logic

Ce Wang, Takayuki D. Kimura,
and Thomas H. Fuller, Jr.

WUCS-93-44

Reprinted from *Intelligent Engineering through Artificial Neural Networks, Volume 2. New York: ASME Press, 1992.*
Editor: C. Dagli et al.
(pages 351-356)

November 1992

Department of Computer Science
Washington University
Campus Box 1045
One Brookings Drive
St. Louis, MO 63130-4899

This work was supported by the Kumon Machine Project.

Supervised Competitive Learning
Part II: SCL with Fuzzy Logic

Ce Wang, Takayuki D. Kimura, and Thomas H. Fuller, Jr.[1]

Department of Computer Science, Washington University in St. Louis

ABSTRACT

Supervised Competitive Learning (SCL) is described in an accompanying paper [1]; SCL assembles a set of learning modules into a supervised learning system to address the stability-plasticity dilemma. That paper reported results using backpropagation networks as the learning modules (SCL/BP). Here (Part II) we report SCL results using learning modules based on fuzzy logic (SCL/FZ). Although its learning algorithm is very different from that of backpropagation networks, fuzzy logic also suffers the stability-plasticity dilemma. A simulator on handwritten digit and gesture recognition was constructed to demonstrate the utility of SCL/FZ; it recognized 98% of test digits, and 91% of test gestures. In this paper, we also compare SCL/BP with SCL/FZ for recognizing handwritten digits.

1. Introduction

A major problem in developing pen-based computers is handwriting recognition. Pen-based recognition technology is generally based on pattern matching, and its performance is not yet satisfactory for practical uses. Therefore, systems based on other technologies, such as fuzzy logic and neural networks, are currently studied.

A vector-based handwritten character recognition system takes a sequence of pen-location points as input. The great variations of handwritten characters makes it difficult to create a set of standard reference characters. Some automatic feature abstraction algorithms provide the way to find a set of features necessary to classify each input patterns. But in general, these features are *fuzzy* in nature.

Fuzzy logic is a mathematical approach to deal with fuzziness, i.e. fuzzy features and fuzzy decision rules. Its statistical methods also deal with inexact information, but they rely on probability density functions some of whose parameters must be estimated from large experimental data sets. Offhand assumptions within these functions would not provide a proper empirical basis for our application. The fuzzy logic approach serves as a numerical model-free estimator. They estimate a function without requiring a mathematical description of how the output functionally depends on the input, that is, they "learn from samples." The freedom from models is the key advantage over the traditional statistical approaches. It is felt that fuzzy logic provides a useful approach to pattern classification, particularly, in problems having imprecisely defined input patterns and a small number of samples, and where statistical independence can not be assumed.

Like some other adaptive learning systems, such as a backpropagation neural networks, a fuzzy learning system also suffers the stability-plasticity dilemma [Grossberg, 1986]: How can the learning system prevent new learning from washing away prior learned knowledge? A compounding system of Supervised Competitive Learning that resolves the stability-plasticity dilemma is explored in

1 Assistant Professor, Department of Mathematics and Computer Science, Principia College

the accompanying paper (Part I). Based on the same learning scheme, instead of using the backpropagation networks, we propose an SCL algorithm using fuzzy sets in the learning modules (SCL/FZ). We adaptively update prototypes according to sample data. For each unknown input pattern, the system applies a membership function to measure a set of feature similarities to the existing prototypes. Then the system maps this array of membership function values to a character similarity measure and so classifies the input pattern. If the input pattern is not found similar to any existing prototypes, the system creates a new subclass with the label specified by the user.

Due to the variations in writing styles, two characters belonging to the same character class may have very different features. For example, some people write the numeral "5" with a single stroke while some others write it with two strokes. So we need to divide each character class into several subclasses. Even for each subclass, a single pattern may not be sufficient to describe that subclass. Multiple patterns are sometimes necessary. But how many writing styles are there for each character class? We don't know in advance. Although we could sample a large set of handwriting data, it is impossible to exhaust all the possibilities. In general, prototypes are generated based on feature clusters. Deterministic clustering algorithms generate class partitions such that each pattern is assigned to exactly one cluster. However, a pattern is often "between" clusters. It could belong to one cluster nearly as well as to another. Only fuzzy clustering methods can yield more accurate representations of real data objects. A fuzzy clustering algorithm (from Ruspini and Bezdek) is discussed in [Pao, 1989]. This algorithm requires us to input the number of current cluster centers. But we are dealing with an evolving environment. No number of cluster centers can be preconceived. SCL/FZ provides an approach to determine feature clusters by similarity measures and we use the ensemble average of the many values obtained for a particular feature of a particular cluster.

Another challenge in fuzzy systems is the design of membership functions. Fuzzy logic serves as a similarity detector. Basically it maps an input pattern to the degree of membership of a class by means of the membership function. The membership function is defined as $\mu_A: X \rightarrow [0, 1]$, where X is a collection of input patterns. But what is its exact formula? Some applications such as the fuzzy system for handwritten Chinese character recognition [Cheng et al., 1989] use exponential functions. Another typical formula of the membership function is

$$\mu = (1 + | (p_i - p) / E |^{F})^{\alpha} \qquad (1\text{-}1)$$

where p represents an input pattern, p_i represents the corresponding value of the ith prototype in a fuzzy recognition system, and E, F, and α are three parameters to be determined. Notice that no matter what formula is used, there are some parameters to be determined. Is there a principle for the parameter determination? A method based on the class separability measure is presented in [Cheng et al., 1989]. This method again needs to employ some probability density functions; for example, the class conditional probability density functions. We select the membership functions in the form of equation (1-1) and determine the parameters E, F, and α empirically.

This work is organized into three major parts. In the next section we describe the SCL/FZ recognition system. In section 3, we report the results of experiment on handwritten character recognition. In the last section, we compare the performances of two learning systems: SCL/FZ and SCL/BP in terms of recognition rate and computational complexity.

2. SCL/FZ: SCL with Fuzzy Logic

The SCL model is defined in the accompanying paper (Part I). Here we briefly review it to clarify later discussion. The basic idea of the compounding system of

SCL is to use each learning module as a similarity detector for a prototype and adopt the prototype resetting mechanism [Carpenter 1988] to create new prototypes. In the SCL/FZ system, we use the learning scheme shown in Figure 2.1. Now, instead of using the backpropagation networks, we use fuzzy sets as learning modules. Due to the difference between the fuzzy logic training and the neural network training, the algorithm for the selector is slightly different. We rewrite the algorithm to relate it to the fuzzy system below.

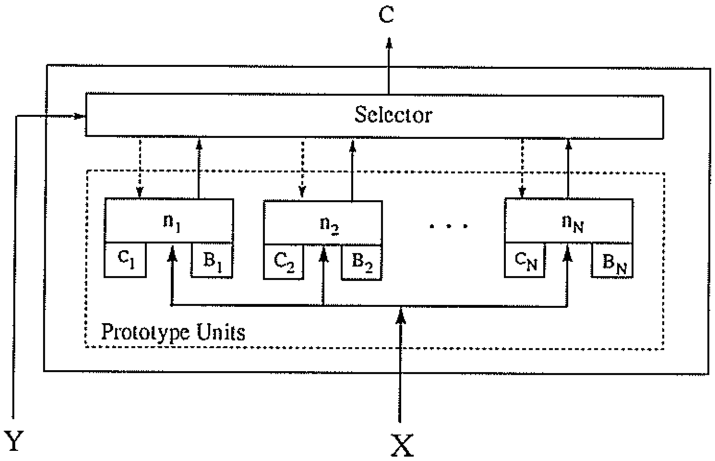

Figure 2.1 SCL Scheme

Note that B_i here is defined a little differently from that in the accompanying paper (Part I). In addition to a set of typical patterns, B_i carries an array of cluster centers of these patterns. We consider B_i the prototype maintained by prototype unit n_i to represent subcategory C_i. The algorithm is:

Parameters: $0 < \rho < 1$
Initialization: $C_i = \Lambda$, $f_i = 0$, $B_i := \phi$ (the empty set), for $1 \leq i \leq N$.

1. Get the input pattern X.
2. $K := \{ i \mid C_i \neq \Lambda$ and $n_i(X) > \rho \}$.
3. If $K = \phi$ then Produce Λ; Goto 7.
4. Find j such that $n_j(X) = \max\{ n_i(X) \mid i \in K\}$.
5. Produce C_j.
6. If accepted then Goto 1.
7. Get the correct category name Y.
8. $K := \{ i \mid C_i = Y$ and $n_i(X) > \rho\}$.
9. If $K = \phi$ then Goto 13.
10. Find j such that $n_j(X) = \max\{ n_i(X) \mid i \in K\}$.
11. Modify B_j by adding X to it; $f_j := f_j + 1$.
12. For all i such that $i \notin K$ and $n_i(X) \geq \rho$, modify B_i such that $n_i(X) < \rho$; Goto 2.
13. $K := \{ i \mid C_i = \Lambda\}$.
14. If $K \neq \phi$ then Select $j \in K$; Goto 11.
15. Find j such that $f_j = \min\{ f_i \mid 1 \leq i \leq N\}$.
16. $f_j = 0$; $B_j := \phi$; Goto 11.

Because it is not clear how to modify B_i to let $n_i(X) < \rho$, the algorithm currently in use skips step 12 (add "Goto 2" to the end of step 11). The negative training (step 12) remains an interesting unsolved problem. Also, in current experiments, we are not using the variable f_i to record the occurrence frequency of the pattern belonging to subcategory C_i. When the system exhausts the capacity to adopt new patterns, it simply exits the learning process.

Fuzzy Logic Learning Module: The digitizing tablet collects the number of strokes followed by a sequence of x,y coordinates when a character is drawn on its writing surface. We apply the feature abstraction algorithm to the pen data to get the feature vector of the input character and use it as the input pattern X to the recognition system. When X is compared with the existing prototypes, we first calculate the feature similarities of X to each prototype and then map them to a character similarity measure.

The framework of a speech identification system [Pal and Majunder, 1977,1978] underlies the design of our fuzzy learning modules. The issue is how to weight the total effect of an array of membership function values to make a fuzzy decision on pattern classification. Their system demonstrated that the pattern classification can be done by a similarity measure defined somewhat arbitrarily in terms of a ratio of the membership functions of a pattern and of cluster prototypes.

Now the pattern in question is X in the form of an array of features. Assume that X is an L-dimensional vector, the size of which should be much smaller than that of the raw data. First, we transfer X to an array of the membership function values, that is,

$$P_j(X) = (p_{1j}, p_{2j} ,..., p_{Lj}) \tag{2-1}$$

where the values of p_{ij} lie in the interval [0,1]. $P_j(X)$ can be considered as the feature similarity measure of X to the prototype B_j. And each element of $P_j(X)$ is obtained by the membership function formulated as

$$p_{ij} = (1 + | (x_i - a_{ij}) / E |^F)^{\alpha} \tag{2-2}$$

where,

p_{ij} denotes the degree to which the feature i is possessed by the prototype B_j,
x_i is the value of feature i in pattern X,
a_{ij} is the cluster center value of feature i of the prototype B_j,
E and F are positive constants to be determined, and,
α is a constant, usually with the value -1.

Assume that there are N subcategories C_1, C_2, ..., C_j, ..., C_N in the recognition system, and B_j carries H_j patterns for each subcategory C_j. Let $B_j^{(h)} = (b_{1j}^{(h)}, ..., b_{Lj}^{(h)})$ be the hth pattern in B_j, where h = 1, ..., H_j. We transfer each of these patterns into the fuzzy references represented in terms of membership function values, i.e.

$$R_j^{(h)} = (r_{1j}^{(h)}, r_{2j}^{(h)}, ... ,r_{Lj}^{(h)}) \tag{2-3}$$

where $r_{ij}^{(h)}$ denotes the degree to which the property i is possessed by the hth pattern of B_j, and

$$r_{ij}^{(h)} = (1 + | (b_{ij}^{(h)} - a_{ij}) / E |^F)^{\alpha} \tag{2-4}$$

In both (2-2) and (2-4), we need to know the value a_{ij}. As we have mentioned, the system prototypes are determined by the feature clusters. But it may not be possible for us to obtain these feature clusters by applying the fuzzy clustering algorithm mentioned in section 1 because no number of cluster centers can be preconceived. So in the design of our system, we may consider the above a_{ij} as the ensemble average of the many values obtained for the ith feature of the prototype B_j, i.e.

$$a_{ij} = (1 / H_j) \sum_{h=1}^{H_j} b_{ij}^{(h)} \tag{2-5}$$

The data for the computation of $P_j(X)$ and $R_j^{(h)}$ are tabulated in Table 2.1.

Table 2.1 The data for the computation of $P_j(X)$ and $R_j^{(h)}$.

B_j:	$B_j^{(1)}$:	$b_{1j}^{(1)}$	...	$b_{ij}^{(1)}$	...	$b_{Lj}^{(1)}$
	$B_j^{(2)}$:	$b_{1j}^{(2)}$	...	$b_{ij}^{(2)}$	...	$b_{Lj}^{(2)}$
	...	...	...	...	...	...
	$B_j^{(Hj)}$:	$b_{1j}^{(Hj)}$	...	$b_{ij}^{(Hj)}$	...	$b_{Lj}^{(Hj)}$
	X:	x_1	...	x_i	...	x_L

The similarity vector $S_j(X)$ for the pattern X with respect to the subcategory C_j is defined to be

$$S_j(X) = (s_{1j}, ..., s_{Lj}) \tag{2-6}$$

where

$$s_{ij} = (1/H_j) \sum_{h=1}^{H_j} s_{ij}^{(h)} \tag{2-7}$$

The values $s_{ij}^{(h)}$ are obtained from two membership function values p_{ij} and $r_{ij}^{(h)}$ through the relationship

$$s_{ij}^{(h)} = (1 + w_i \mid 1 - p_{ij} / r_{ij}^{(h)} \mid) ^{-2} \tag{2-8}$$

The numerical value of s_{ij} denotes the degree of similarity of the ith feature of X with that of C_j. In (2-8) the w_i are positive constants that can be individually tailored to indicate the relative sensitivity of the classification to deviations from the prototype values.

The similarity value of the input pattern X to the prototype B_j representing the subcategory C_j is defined to be the length of $S_j(X)$. We compute the normalized similarity values, which reside in [0,1], from each similarity vectors by

$$\mid S_j(X) \mid =((1/L) \sum_{i=1}^{L} s_{ij}^2)^{1/2} \tag{2-9}$$

The closed form of $\mid S_j(X) \mid$ is

$$\mid S_j(X) \mid = ((1/L) \sum_{i=1}^{L} ((1/H_j) \sum_{h=1}^{H_j} (1+w_i \mid 1- p_{ij}/r_{ij}^{(h)} \mid) ^{-2z})^2)^{1/2} \tag{2-10}$$

where j = 1, ..., N. We use $\mid S_j(X) \mid$ as the $n_j(x)$ for the SCL/FZ and select the winning subcategory C_j for X by the decision rule,

$$\mid S_j(X) \mid = \max(\mid S_i (X) \mid \mid i = 1, ..., N) \tag{2-11}$$

3. Experiments

The experimental results on SCL/FZ for handwritten digit recognition are tabulated in Figure 3.1.

Parameters				Confidence	Number of	Training	Recognition Rate	
α	E	F	z	Value ρ	Prototypes	Time (Min.)	Train(800)	Test(600)
-1	1	1	1	0.8	88	4	96.62%	95.67%
-1	1	1	1	0.85	88	4	96.88%	96.83%
-1	1	1	1	0.9	157	11	99.50%	97.17%

Figure. 3.1 Handwritten Digit Recognition Results: SCL/FZ

4. Comparison and Conclusion

The comparison of performances of the SCL/FZ and SCL/BP is illustrated in Figure 4.1. There is no significant difference in computational complexity between the two systems. The SCL/BP system achieved higher recognition rates because of negative training. The experimental results from both systems demonstrate that the SCL is a powerful technique for resolving the stability-plasticity dilemma. Our next research is to introduce negative training into SCL/FZ. We believe that it will significantly improve the performance of the fuzzy recognition system.

System	Number of Prototypes	Training Time (Min.)	Recognition Rate (Testing Data)
SCL/BP	37	4	98.1%
SCL/BP	35	19	99.7%
SCL/FZ	88	4	96.8%
SCL/FZ	157	11	97.2%

Figure 4.1 Comparison on SCL/FZ and SCL/BP

5. References

[1] Thomas H. Fuller, Jr. and Takayuki Dan Kimura, "Supervised Competitive Learning Part I: SCL with Backpropagation Networks," Submitted to ANNIE'92, St. Louis, MO, 1992.

[2] Grossberg, S. "Competitive Learning: From Interactive Activation to Adaptive Resonance. Cognitive Science 11, 1986, 23-63.

[3] Yoh-Han Pao, "Adaptive Pattern Recognition and Neural Networks," Addison-Wesley Publishing Company, Inc., 1989.

[4] Bart Kosko, "Neural Networks and Fuzzy Systems," Prentice-Hall, Inc., 1992.

[5] Fang-Hsuan Cheng, Wen-Hsing Hsu, and Chien-An Chen, "Fuzzy Approach to Solve the Recognition Problem of Handwritten Chinese Characters," Pattern Recognition, Vol. 22, No. 2, 1989, pp. 133-141.

[6] L. A. Zadeh, "Fuzzy Sets," Information and Control, Vol. 8, 1965, pp. 338-353.

[7] L. A. Zadeh, "Fuzzy Logic," Computer, Vol. 21, No. 4, April, 1988, pp. 83-93.

[8] S. K. Pal and D. D. Majunder, "Fuzzy Sets and Decisionmaking Approaches in Vowel and Speaker Recognition," IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-7, August, 1977, pp. 625 ~ 629.

[9] S. K. Pal and D. D. Majunder, "On Automatic Plosive Identification Using Fuzziness in Property Sets," IEEE Transactions on Systems, Man, and Cybernetics, Vol.8, 1978, 302-7.