Available online www.jocpr.com

Journal of Chemical and Pharmaceutical Research, 2014, 6(3):102-112

Research Article

ISSN : 0975-7384
CODEN(USA) : JCPRC5

Incremental learning fuzzy measures with Choquet integrals in fusion system

Xuefei Li[1*], Huimin Feng[2] and Hongbo Shao[3]

[1]College of Science, Agricultural University of Hebei, Baoding, China
[2]College of Mathematics and Computer Science, Hebei University, Baoding, China
[3]College of Science, Agricultural University of Hebei, Baoding, China

---

ABSTRACT

A new neural network architecture is introduced for incremental learning fuzzy measure for Choquet integrals based fusion system. The proposed approach differs from other determining fuzzy measure methods by its suitability to incremental learning fuzzy measures. The fuzzy measure is treated as the weights of the neural networks. An additional neural network is introduced to accommodate new data when new samples arrive, including examples that correspond to previously unseen classes. Furthermore, the algorithm does not require access to previously used data during subsequent incremental learning sessions, but preserve the center of a batch of samples. At the same time, it does not forget previously acquired knowledge. The outputs of the resulting classifier fusion systems are combined using a weighted majority procedure where the weights are dependent on the performance of the fusion system and the distance of testing samples and the centers of each batch. We present simulation results on several benchmark datasets as well as a real-world classification task. Initial results indicate that the proposed algorithm works rather well in practice.

Key words: Choqute integral, fuzzy measure, non-negative measure, neural network, fusion system

---

INTRODUCATION

Many types of classifiers, e.g. decision trees [1]–[2], neural networks [3], support vector machines [4], have been proposed in the recent decades, but it is hard to say which type is the best [5]. In many practical tasks, it is very difficult to design one classifier with satisfying performance. Hasen and Salamon [6] showed that the generalization ability could be significantly improved through fusing a number of neural networks, i.e. training many neural networks and then fusing their classification results. Since this technology is easy and behaves remarkably well, it has been successfully applied to diversified areas [7]–[9], and the types of classifiers can not only be neural networks but also be decision trees, k-Nearest Neighbors, etc.

In multiple classifier fusion, a number of classifiers are designed for a given classification task. The classification results from all classifiers can be aggregated by an operator such as average, weighted average, ordered weighted average or fuzzy integral [10]–[12]. Through aggregating all classifiers' predictions by operators, the final classification can be obtained. Weighted average and ordered weighted average operators are good choices to deal with the different importance of individual classifiers, but the method of weighted average is under an assumption that the interaction does not exist among the classifiers. However, this assumption may not be true in many real problems. The fuzzy integral [10-14] will be our better choice. The fuzzy integral as the fusion tool, in which the non-additive measure can clearly express the interaction among classifiers and the importance of classifiers, has its particular advantages. In fuzzy integral based fusion models, determining suitable fuzzy measures is one of the keys and one of main difficulties. There have been some methods to determine fuzzy measures such linear programming and quadratic programming [15], genetic algorithm [16], neural network [17], and pseudo-gradient [18]. So far as is known to the authors, all these methods determine suitable fuzzy measures in a single training session, learning all the data concurrently. Once that training is finished, the fuzzy measures will not change. The performance of the fusion system relies heavily on the availability of a representative set of training examples. In many practical applications, acquisition of a representative training data is expensive and time consuming. Consequently, it is not uncommon for such data to become available in small batches over a period of time. In such settings, it is necessary to update existing fuzzy measures in an incremental fashion to accommodate new data without compromising classification performance on old data. We proposed neural network architecture to incrementally learn fuzzy measures for the aim: learning new information without forgetting previously acquired knowledge. Our approach has the following characters:

1) It is able to learn additional information from new data;
2) It needs little information about the original data, doesn't require access to all the original data, used to determine the existing fuzzy measures;
3) It could preserve previously acquired knowledge (that is, it should not suffer from catastrophic forgetting).

The rest of this paper is organized as follows. In Section 2, we provide an overview of Choquet integrals based multiple classifier fusion system. In Section 3, we show the neural network architecture for incremental learning fuzzy measures algorithm in detail. In Section 4, we give the simulation results obtained on some the benchmark real-world databases. Finally, in Section 5, we summarize our conclusions and point at future research directions.

I. THE FUSION MODEL BASED ON CHOQUET INTEGRALS

Suppose that $X=\{x_1,x_2,\cdots,x_n\}$ is a set of classifiers. The output of classifier x_j is a c-dimensional nonnegative vector $[d_{j,1}, d_{j,2},\ldots,d_{j,c}]$ where c is the number of classes. Without loss of generality, let $d_{i,j}\in[0, 1]$ denoting the support from classifier x_j to the hypothesis that sample submitted for classification comes from the i-th class C_i for j=1,2,…,n, i=1,2,…, c. The larger the support, the more likely the class label C_i. All outputs of classifiers for a particular sample can be organized in a matrix:

$$DP=\begin{bmatrix} d_{1,1}\cdots d_{1,i}\cdots d_{1,c} \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ d_{j,1}\cdots d_{j,i}\cdots d_{j,c} \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ d_{n,1}\cdots d_{n,i}\cdots d_{n,c} \end{bmatrix}$$

Each column of DP matrix can be regarded as a function defined on the classifier set X, $f_j : X\to[0,1]$, $f_j(x_i)=d_{j,i}$, $i=1,2,...,c$, $j=1,2,...,n$. For each class C_i, we need to determine a nonnegative set function μ_i on the power set $P(X)$ of X. μ_i can represent not only the importance of individual classifiers but also the interaction among classifiers towards samples of C_i class. Set functions have some special cases.

Definition 1 [19]**:** Let X be a nonempty and finite set and $P(X)$ be the power set of X. Then $(X; P(X))$ is a measurable space. A set function $\mu : P(X)\to(-\infty,+\infty)$ is called a fuzzy measure or a monotone measure, if

(FM1) $\mu(\varnothing)=0$, (vanishing at the empty set).
(FM2) $\mu(A)\geq 0$, for any $A\subset X$ (non-negativity).
(FM3) $\mu(A)\leq\mu(B)$ if $A\subset B$, $A\subset X$, $B\subset X$ (monotonicity).

Set function μ is called an efficiency measure if it satisfies (FM1) and (FM2); μ is called a signed efficiency measure if it satisfies (FM1) only. Any fuzzy measure is a special case of the efficiency measure and fuzzy measures and efficiency measures are nonnegative set functions. Fuzzy measures have a more monotone constraint than efficiency measures, so fuzzy measures are sometimes called nonnegative monotone set functions. In multiple classifier fusion, nonnegative set functions are used to describe the importance of classifiers and the interaction among classifiers. The value of set function at a single-point-set $\mu(\{x_i\})$ presents the contribution of the single classifier x_i towards classification, and the value of set function at other sets, such as $\mu(\{x_i,x_j,x_k\})$, presents the joint contribution of classifiers towards classification. Mainly the methods to determine the nonnegative set functions have two types. One is to specify by experts and the other is to learn from the history data such linear programming, quadratic programming [15], genetic algorithm [16], neural network [17], pseudo-gradient [18]. So far as authors know, existing methods are determining the fuzzy measures in a single training session and can not update the fuzzy measures in an incremental fashion to accommodate new data without compromising classification performance on old data. In section 3, an incremental learning fuzzy measures method is introduced in details.

Once the set functions are available, we can use the fuzzy integral to aggregate the outputs from all classifiers. The i-th column of DP matrix, $d_{j,I}$, j=1, 2, …, n, can be regarded as a function f_i defined on classifier set X, $f_i(x_j) = d_{j,i}$. The integral of function f_i with respect to nonnegative set μ_i is the degree of fusion system classifying an example to class C_i. If necessary, we can obtain the crisp class label $C_t = \arg(\max_{1\le i\le c}\{\int f_i d\mu_i\})$.

Usually the type of fuzzy integral is chosen in advance. Choquet fuzzy integral is often selected in fusion process because the addition and the multiplication operators are used in Choquet integral and most researchers prefer now to use the Chquet integral in classifier fusion models [14, 20].

Following is the definition of the Choquet integral.
Consider a nonempty and finite set $X = \{x_1, x_2, \cdots, x_n\}$. $P(X)$ is the power set of X; $\mu : P(X) \to [0,+\infty)$ is a set function denoting the fuzzy measure, and $f : X \to [0,+\infty)$ is a function. First, $\{f(x_1), f(x_2), \cdots, f(x_n)\}$ are rearranged into a non-decreasing order, that is,

$$f(x_1^*) \le f(x_2^*) \le \cdots \le f(x_n^*)$$

where $\{x_1^*, x_2^*, \cdots, x_n^*\}$ is a permutation of $\{x_1, x_2, \cdots, x_n\}$. Then the Choquet integral of function f with respect to the measure μ are evaluated as

$$(C)\int f d\mu = \sum_{i=1}^{n}[f(x_i^*) - f(x_{i-1}^*)]\mu\{x_i^*, x_{i+1}^*, \cdots, x_n^*\} \tag{1}$$

where $f(x_0^*) = 0$.

II. THE INCREMENTAL LEARNING METHOD BASED ON NEURAL NETWORK ARCHITECTURE
In this section we will introduce neural network architectures to accomplish incremental learning fuzzy measures. First, another formula to evaluate the Choquet integral is introduced as following.

$$(c)\int f\,\mathrm{d}\mu = \sum_{A\subset X} \max(0, \min_{x\in A}(f(x)) - \max_{x\in \bar{A}}(f(x))) \times \mu(A) \tag{2}$$

Let $g(f, A) = \max(0, \min_{x\in A}(f(x)) - \max_{x\in \bar{A}}(f(x)))$, then we have

$$\begin{aligned}(c)\int f\,\mathrm{d}\mu &= \sum_{A\subset X} g(f,A)\times\mu(A)\\ &= [g(f,\{x_1\})\quad g(f,\{x_2\})\quad g(f,\{x_1,x_2\})\quad\cdots\quad g(f,X)]\cdot\begin{bmatrix}\mu(\{x_1\})\\ \mu(\{x_2\})\\ \mu(\{x_1,x_2\})\\ \vdots\\ \mu(X)\end{bmatrix}\end{aligned} \tag{3}$$

It inspires us to build a neural network to evaluate the Choquet integral as Fig. 1. The function f= $\{f(x_1)$, $f(x_2)$, $\cdots, f(x_n)\}$ is the inputs of neural network. Then f is transformed into 2^n-1 dimension vector $[g(f,\{x_1\})\quad g(f,\{x_2\})\quad g(f,\{x_1,x_2\})\quad\cdots\quad g(f,X)]$ corresponding all subset of X. That is, the second layer is to calculate to $g(f,A)$ for each nonempty subset A of X. The output of the neural network is the value of the Choquet integral of f. The weights from the second layer to the output layer are the value of the fuzzy measure for each subset A of X. The weights can be adjusted through the training of the neural network, such back propagation algorithm.
In the fusion system, the function f is the supports of base classifiers for a testing example belonging to a class, and the fuzzy measure, i.e. the weights, represents not only the importance of individual base classifiers but also the interaction between them towards samples of this class. The output, i.e. the value of the Choquet integral, is the support of the fusion system.

In industrial life, data usually become available gradually; this fact requires data analysis systems to have the capability to learn information incrementally. Learning from new data without forgetting prior

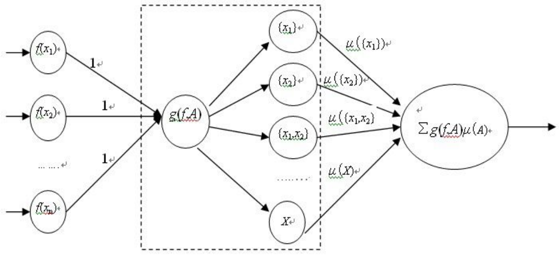

Figure 1. The neural network for the Choquet integral

knowledge is known as incremental learning. Various algorithms have been suggested for incremental learning the in the literature. One of typical algorithms is the (fuzzy) ARTMAP[21] , which is based on generating new decision clusters in response to new patterns that are sufficiently different from previously seen instances. Since previously generated clusters are always retained, ARTMAP does not suffer from catastrophic forgetting. Furthermore, ARTMAP does not require access to previously seen data, and it can accommodate new classes. ARTMAP is a very powerful and versatile algorithm; however, it has its own drawbacks. In many applications, researchers have noticed that ARTMAP is very sensitive to selection of the vigilance parameter, to the noise levels in the training data and to the order in which the training data is presented to the algorithm. Furthermore, the algorithm generates a large number of clusters causing overfitting, resulting in poor generalization performance, if the vigilance parameter is not chosen correctly. Therefore, this parameter is typically chosen in an *ad hoc* manner by trial and error. Various algorithms have been suggested to overcome such difficulties [22-23]. Robi Polikar [24] etc proposed a Learn++ algorithm: incremental learning algorithm for supervised neural networks. Learn++ utilizes ensemble of classifiers by generating multiple hypotheses using training data sampled according to carefully tailored distributions. The outputs of the resulting classifiers are combined using a weighted majority voting procedure. Voting weights are determined based on performances of the hypotheses on their own training data subset. This is suboptimal, since the performance of a hypothesis on a specific subset of the input space does not guarantee the performance of that hypothesis on an unknown instance, which may come from a different subset of the space. So we adopted the Learn++ algorithm to implement incremental learning fuzzy measures, but adopted the distance-based and performance-based weights in stead of original performance-based weights. The distance indicates that which subset is the closest from testing sample and which should have more importance in classifying testing sample. Different subsets will different importance on classifying testing sample.

From Fig. 2, it is clear that the “triangle” sample is further from the “circle” samples than “plus signs” samples and “asterisk” samples. The fuzzy measures learned from the “circle” samples are less important than those learned from “plus signs” samples and “asterisk” samples. When we classify the “triangle” sample, in weighted majority the weight for the results from the fuzzy measures obtained from “circle” samples should accommodate the distance of the “triangle” sample and “circle” samples. We determine the weight based on the performance and the distance.

Like general fusion system, for each subset D_k in succession of a problem, a number of base classifiers are trained. The outputs of base classifiers are fused by the Choquet integral with the fuzzy measures trained according neural network (shown in Fig. 1) for each class. That is, there is a fusion system on each subset. If we have received totally *K* subsets in succession, there are *K* fusion systems. The incremental learning fuzzy measures algorithm is given as follows.

Algorithm for incremental learning fuzzy measures

Input: for each subset of database drawn from $\mathcal{D}_k$, k=1, 2,…, K

- Sequence of m_k training examples S=[(x_1, y_1),(x_2, y_2),…,(x_{m_k}, y_{m_k})].
- Classifier learning algorithm CLearn.
- Integer T_k, specifying the number of base classifier for database $\mathcal{D}_k$.

Do for k=1, 2,…,K

1. Call CLearn, providing it with $\mathcal{D}_k$ to train T_k base classifiers.
2. Determine the fuzzy measures by provide the neural network shown in figure 1 the outputs of T_k base classifiers.
3. Calculate the classification correct accuracy of the fusion of $\mathcal{D}_k$ base classifiers based on the fuzzy measures on database $\mathcal{D}_k$ Denoting the accuracy as p_k.
4. Calculated the center of $\mathcal{D}_k$ as C_k.
5. Calculated the weight $w_k = \frac{p_k}{\sum_{k=1}^{K} p_k}$

Outputs: the base classifiers, the fuzzy measures, the weights, the centers for each $\mathcal{D}_k$, k=1, 2,…, K.

When a testing sample x^* arrives, it will be provided to each fusion system. The weighted majority will be used to fuse the results where the result of k^{th} fusion system is denoted by O_k. Noting, the weights in weighted majority should be modified according to the distance:

$$w_k^{'} = \frac{w_k \times dis(x^*, C_k)}{\sum_{k=1}^{K} w_k \times dis(x^*, C_k)}$$

where $dis(x^*, C_k)$ is the distance between x^* and the center C_k. The final classification result is the weighted majority of K fusion systems:

$$O_{final} = \arg\max_{y \in Y} \sum_{O_k = y} w_k^{'}$$

The weights are dependent on the performance of fuzzy measures and the distance between the testing sample and the center of subset of samples. Better performance

the fusion system has, bigger the weight is. Closer the distance is, bigger the weight is. Our model can be summarized in Fig.3. In our model, the additional information can be learned from new data and previously acquired knowledge can be preserved. Note that, we only need to preserve the centers of original subset of database and do not require access to the original data used to train the fuzzy measures. The knowledge in continually new samples could be learned, but the space for storage of previous data will not increase much over time.

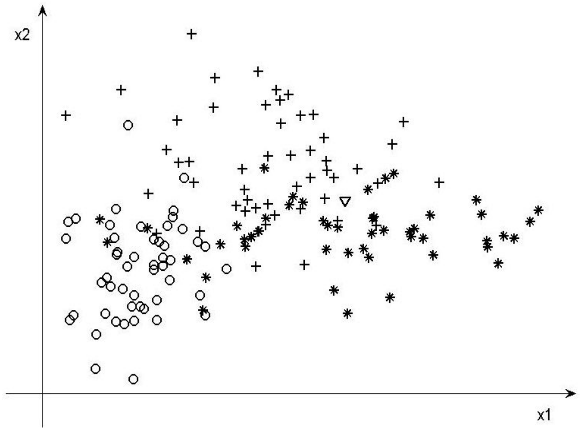

Figure 2. An illustration of training samples with different importance upon testing sample (triangle is the testing samples, plus signs, circle and asterisk are training samples)

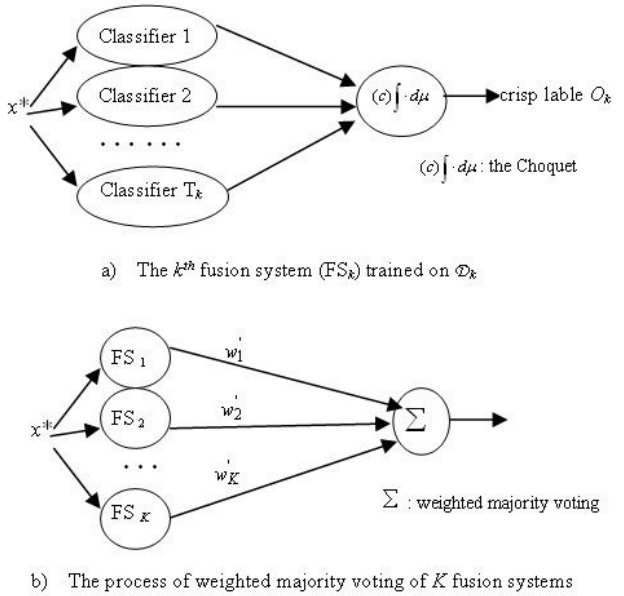

Figure 3. The classification of a testing sample x^* in our model

EXPERIMENTAL SECTION

In this section we present several experiments to test the incremental learning fuzzy measures algorithm on a synthetic database and benchmark databases. The synthetic database is concentric circles in two- dimensional space shown in Fig. 4. The points in one class are uniformly distributed into a circle of radius 0.3 centered on (0.5, 0.5). The points in another class are uniformly distributed into a ring centered on (0.5, 0.5) with internal and external radius equal to 0.3 and 0.5, respectively. The true boundary of two classes is the circle centered on (0.5, 0.5) with radius equal to 0.3. Six real-world databases are from UCI machine learning repository [25].

1) Optical digits database consisted of 5620 samples of digitized handwritten characters. The characters were numerals 0–9, and they were digitized on an 8×8 grid, creating 64 attributes. Fig. 5 shows sample images of this database.
2) Pima Indians diabetes database consisted of 768 samples from National Institute of Diabetes and Digestive and Kidney Diseases. The diagnostic, binary-valued variable investigated is whether the patient shows signs of diabetes according to World Health Organization criteria (i.e., if the 2 hour post-load plasma glucose was at least 200 mg/dl at any survey examination or if found during routine medical care). Each sample is described by 8 numeric-valued attributes. All samples come from two different classes.
3) Wine database consists of 13 input attributes, 178 samples, and 3 classes. All attributes are continuous. These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different

cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines.
4) Letter recognition database consists of 16 input attributes, 20000samples, and 26 classes. The objective is to identify each of a large number of black-and-white rectangular pixel displays as one of the 26 capital letters in the English alphabet. The character images were based on 20 different fonts and each letter within these 20 fonts was randomly distorted to produce a file of 20,000 unique stimuli. Each stimulus was converted into 16 primitive numerical attributes (statistical moments and edge counts) which were then scaled to fit into a range of integer values from 0 through 15.
5) Ionosphere database is to classify the radar returned from the ionosphere. This radar data was collected by a system in Goose Bay, Labrador. This system consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kilowatts. See the paper for more details. The targets were free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not; their signals pass through the ionosphere. This is a binary classification task. Received signals were processed using an autocorrelation function whose arguments are the time of a pulse and the pulse number. There were 17 pulse numbers for the Goose Bay system. Instances in this database are described by 2 attributes per pulse number, corresponding to the complex values returned by the function resulting from the complex electromagnetic signal. There are 351 samples consisting of 34 attributes.
6) CoverType database consists of 54 attributes, 581012 samples, and 7classes. Predicting forest cover type from cartographic variables only (no remotely sensed data). The actual forest cover type for a given observation (30 x 30 meter cell) was determined from US Forest Service (USFS) Region 2 Resource Information System (RIS) data. Independent variables were derived from data originally obtained from US Geological Survey (USGS) and USFS data. Data is in raw form (not scaled) and contains binary (0 or 1) columns of data for qualitative independent variables (wilderness areas and soil types). This study area includes four wilderness areas located in the Roosevelt National Forest of northern Colorado. These areas represent forests with minimal human-caused disturbances, so that existing forest cover types are more a result of ecological processes rather than forest management practices. Some background information for these four wilderness areas: Neota (area 2) probably has the highest mean elevational value of the 4 wilderness areas. Rawah (area 1) and Comanche Peak (area 3) would have a lower mean elevational value, while Cache la Poudre (area 4) would have the lowest mean elevational value. As for primary major tree species in these areas, Neota would have spruce/fir (type 1), while Rawah and Comanche Peak would probably have lodgepole pine (type 2) as their primary species, followed by spruce/fir and aspen (type 5). Cache la Poudre would tend to have Ponderosa pine (type 3), Douglas-fir (type 6), and cottonwood/willow (type 4). The Rawah and Comanche Peak areas would tend to be more typical of the overall dataset than either the Neota or Cache la Poudre, due to their assortment of tree species and range of predictive variable values (elevation, etc.) Cache la Poudre would probably be more unique than the others, due to its relatively low elevation range and species composition.

In all experiments, we set K=3 and the number of base classifiers is 4. That is, we will obtain 3 fusion systems and $4\times3=12$ classifiers when the incremental learning algorithm terminates. We adopt the fuzzy decision trees as base classifiers where in fuzzifying data, we use the triangular membership function and select its slopes in the way that adjacent membership functions cross at the membership value 0.5, the significant level α=0.4 and the truth level threshold β=0.8.

On concentric circles database, we can see that single decision trees create decision boundaries with portions perpendicular to the feature axes from Fig. 6. The learned boundary is zigzag. The difference of the learned boundary and the true boundary is big. The performance of classification is unsatisfied. The fuzzy measure is determined through the neural network in Fig. 1. Then the results of 4 base decision trees are aggregated through the Choquet integral, the learned boundary becomes a smooth curve in Fig. 7. The learned boundary is a rough circle, the performance of classification is improved. But the difference between the true boundary and the learned boundary is distinct. After the process of incremental learning fuzzy measures are finished, the learned boundary is smoother and closer to the true boundary in Fig. 8. The figures show that aggregating multiple classifiers could improve the performance of classification. The incremental learning fuzzy measures algorithm does integrate the previous knowledge and the new knowledge contained in new samples.

For six databases from UCI, we run 10-fold cross validation 50 times on each database. We set K=3 in incremental learning fuzzy measures algorithm. That is, the training samples are randomly parted into three groups. In all experiments, previously seen data were not used in subsequent stages of learning. The incremental algorithm is compared with the quadratic programming [15], the genetic algorithm [16] and the neural networks [17]. The diversity of classifiers is very important. If all base classifiers are the same, the fusion will be meaningless. We use different attributes subset to design diverse base classifiers. For each base classifier 80% attributes is randomly chosen from original attributes set.

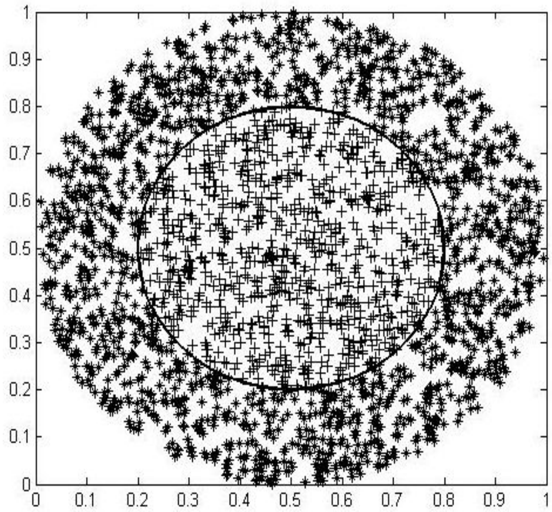

Figure 4. Sample images from the optical digits database

Figure 5. The concentric circles database (with the boundary between two classes)

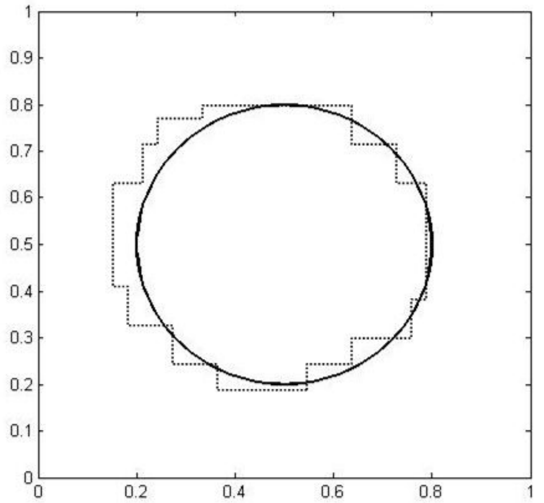

Figure 6. The learned boundary of a fuzzy decision tree on concentric circles database (solid line--ture boundary, dotted line—learned boundary)

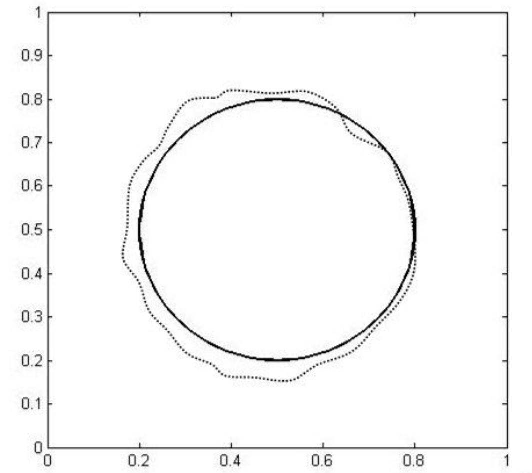

Figure 7. The learned boundary of a fusion system on concentric circles database (solid line--ture boundary, dotted line—learned boundary)

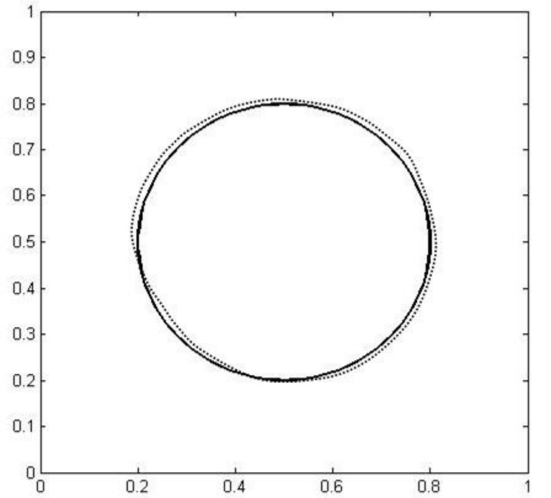

Figure 8. The learned boundary of incremental learning model on concentric circles database (solid line--ture boundary, dotted line—learned boundary)

TABLE I The comparison between incremental learning fuzzy measures algorithm and three learning fuzzy measures algorithm on entire training samples (accuracy)

Data set	Incremental	Quadratic	Genetic	Neural network
Optical digits	0.9496	0.9481	0.9484	0.9436
Pima	0.7746	0.7701	0.7677	0.7648
Wine	0.9727	0.9736	0.9648	0.9725
Letter recogntion	0.8652	0.8589	0.8532	0.8607
Ionosphere	0.9195	0.9016	0.9052	0.9062
CoverType	0.6601	0.6647	0.6642	0.6588

The comparison results are shown in Table 1. From the results of experiment, it can be seen that the performance of the incremental learning fuzzy measures algorithm is similar to or better than that of three learning fuzzy measures algorithm on entire training samples. The genetic algorithm has better performance in finding the solution of optimization problems. It indicates that the learned through genetic algorithm is easy to be overfitting especially when there are noise data or outliers in database and is very time consuming. Similarly the over-fitting is the shortage of quadratic programming and neural networks. In incremental model, there are multiple neural networks for a fuzzy measure corresponding to different training sets of samples. It works like an aggregating system to avoid overfitting in some sense.

CONCLUSION

The appropriate fuzzy measure is one of the keys in a successful multiple classifiers fusion system. In many real-world applications where data arrives over time, it is mainly concerned with learning models in an ever-changing environment. An incremental learning fuzzy measures algorithm is proposed through neural network architecture. The calculation of Choquet integral is transformed into a neural network which the weights from the hidden layer to the output layer are the fuzzy measure and can be learned through the training process. When a new set of samples arrive, new neural networks are trained to learn the fuzzy measures corresponding the new samples. The results of classification corresponding individual fuzzy measure learned on different subset of samples will be fused by weighted majority where the weights are dependent on the performance of fuzzy measures and the distance between the testing sample and the center of subset of samples. Better the performance is, bigger the weight is. Closer the distance is, bigger the weight is. In our model, the additional information can be learned from new data and previously acquired knowledge can be preserved. Note that, we only need to preserve the centers of original subset of database and do not require access to the original data used to train the fuzzy measures. The process of classify unseen samples is like to the aggregation system, so it can avoid overfitting in some sense.

Our future work can focus on the following two problems. 1) Finding a better way than the distance between the testing sample and the center of each subset of database, because the center can represent well the information of a subset of database when the samples distribute like sphere or cloud. It is not the case when the samples distribute in a strip region. 2) Attempting to describe the type of distribution of samples which our incremental learning model is appropriate for. It will give much help in applying and improving our model. 3) Finding the appropriate size of each patch for our model.

Acknowledgment
This research is supported by the National Natural Science Foundation of China (61170040), by the Natural Science Foundation of Hebei Province (F2011201063, F2012201023, F2013201110, F2013201060 and F2013201220), and by the Key Scientific Research Foundation of Education Department of Hebei Province (ZD2010139).
Corresponding author: Huimin Feng

REFERENCES

[1] Quinlan J.R., *Machine Learning*, **1986,** vol. 1, pp. 81-106.
[2] Qing hua Hu, Xunjian Che, Lei Zhang, David Zhang, Maozu Guo, Daren Yu, *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 11, **2011,** pp. 2052-2064.
[3] Kabir, H., Ying Wang, Ming Yu, Qi-Jun Zhang. *IEEE Transactions on Microwave Theory and Techniques*, Vol. 58, no. 1, **2010**, pp. 145 - 156.
[4] Harmsen M., Fischer B., Schramm H., Seidl T., Deserno T. M., *IEEE Journal of Biomedical and Health Informatics*, **2013,**Vol. 17 , no. 1, pp. 190 - 197.
[5] Fabio Roli, Josef Kittler, *Information Fusion*, Guest eitorial, **2002,** vol. 3, pp. 243-243.
[6] Hansen L.K., Salamon P., *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **1990,** vol. 12, no. 10, pp. 993-1001.
[7] Abdallah A.C.B., Frigui H., Gader P., *IEEE Transactions on Fuzzy Systems*, **2012,** vol. 20, no. 5, pp. 849 - 864.
[8] Fernando Enríquez, Fermín L. Cruz, F. Javier Ortega, Carlos G. Vallejo, José A. Troyano, *Information Fusion*, **2013**, Vol. 14, no. 3, pp. 255–267.
[9] Caiyun Zhang, Zhixiao Xie, *Geocarto International*, **2013**, pp. 1-16, doi: 10.1080/10106049.2012.756940.
[10] Antanas Verikas, Arunas Lipnichas, Kerstin Malmqvist. *Pattern Recognition Letters*, **1999,**vol. 20, no. 4, pp. 429-444.
[11] Andrey Temko, Dusan Macho, Climent Nadeu, *Pattern Recognition*, **2008,**Vol. 41, no. 5, pp. 1814-1823.
[12] Hui-min Feng, Xue-fei Li, Wen-jing Lv, *Journal of Hebei University* (*Natural Science Edition*), **2013,**vol. 33, no. 3, pp. 231-235.
[13] Xue-fei Li, Zhi-hua Gu, Hui-min Feng, *Journal of Hebei University* (*Natural Science Edition*), **2012,** vol.32, no. 4, pp. 342-348.
[14] Anderson D.T., Havens T.C., Wagner C., Keller J.M., Anderson M.F., Wescott D.J., “Sugeno fuzzy integral generalizations for Sub-normal Fuzzy set-valued inputs,” 2012 *IEEE International Conference on Fuzzy Systems*, **2012,** pp. 1- 8.
[15] Michel Grabisch, Jean-Marie Nicolas. *Fuzzy Sets and Systems*, **1994,**vol. 65, pp. 255-271.
[16] Zhenyuan Wang, Kwong-sak Leung, Jia Wang, *Fuzzy Sets and Systems*, **1999,** vol. 102, pp. 463-469.
[17] Xufei Li, Huimin Feng, Ruihong Wang, “Dynamic fuzzy measures based on neural networks,” *International Conference on Electronic Information and Electrical Engineering*, **2012,** pp. 583-586.
[18] Zhenyuan Wang, Kwong-Sak Leung, George J.Klir, *International Journal of Intelligent Systems*, **2006,**vol. 21, pp. 1073-1092.
[19] Zhenyuan Wang, George J. Klir. *Fuzzy measure theory,* Plenum Press, New York, A division of Plenum Publishing Corporation, **1992**.

[20] Hui-min feng, Xue-fei Li, jun-fen Chen, "A comparative study of four fuzzy integrals for classifier fusion," *Proceedings of 2010 International Conference on Machine Learning and Cybernetics*, **2010,** pp. 332-338.
[21] G. A. Carpenter, S. Grossberg, N. Markuzon, J. H. Reynolds, and D. B. Rosen, *IEEE Transaction on Neural Network*, **1992,**vol. 3, pp. 698–713.
[22] J. R. Williamson, *Neural Network*, **1996,** vol. 9, no. 5, pp. 881–897.
[23] Kulkarni P., *Reinforcement and Systemic Machine Learning for Decision Making*, New York: Wiley-IEEE Press, **2012.**
[24] Robi Polikar, Lalita Udpa, Satish S. Udpa, Vasant Honavar, *IEEE Transactions on Systems, Man and Cybernetics-Part C: Applications and Reviews*, **2001,**vol. 31, no. 4, pp. 497-508.
[25] UCI Repository of Machine Learning Databases and Domain Theories [Online]. Available: *ftp://ftp.ics.uci.edu/pub/ machine-learning-databases*.