Abstract—For a semi-supervised classification system, with the increase of the training samples number, the system needs to be continually updated. As the size of samples set is increasing, many unreliable samples will also be increased. In this paper, we use fuzzy c-means (FCM) clustering to take out some samples that are useless, and extract the intersection between the original training set and the cluster after using FCM clustering. The intersection between every class and cluster is reliable samples which we are looking for. The experiment result demonstrates that the superiority of the proposed algorithm is remarkable.

Index Terms—Fuzzy c-means clustering, fuzzy k-nearest neighbor classifier, instance selection.

1. Introduction

In recent years, semi-supervised classification has been receiving more and more attention in machine learning and pattern recognition. The motivation behind it is to employ a large number of unlabeled data to help to build a better classifier from the labeled data\(^1\). That is, through “learning” these unlabeled data and labeled data, a training set will be continuous renewed and the optimal classification performance can be obtained at last. Generally, the training set stems from real world, especially in noisy environments, and is stored and searched in a semi-supervised classification system. Several unreliable data often arise because of some erroneously labeled samples.

To deal with this problem, some excellent methods were proposed, e.g., the feature component\(^2\), leaders algorithm\(^3\)\(^4\), data normalization\(^5\), data standardization\(^6\), hierarchical clustering\(^7\), and multivariate regression trees\(^8\). In these papers, the goal is to select sample sets which contain unreliable samples, i.e., reducing the size of sample sets and maintaining the prediction accuracy\(^9\)\(^-\)\(^13\).

As we have already seen, sample sets selection attempts to reduce the number of rows in the training set with no loss of classification accuracy and obtain an improvement in the classification system\(^13\)\(^-\)\(^16\). In this paper, we use fuzzy c-means (FCM) to take out some unreliable samples, which is different from the former existed methods. We compare the classes in the initial training set with the clusters after using FCM, and then extract their intersection. The intersection is the updated training set which we are looking for. These samples in the intersection are the representative samples and will be remained “important samples” in each class.

The rest of the paper is organized as follows. Section 2 reviews the fuzzy k-nearest neighbor classifier (FKNN) and FCM clustering algorithm. The novel method using the FCM clustering method is proposed in Section 3. Section 4 gives experiment results using two different synthetic data. The conclusion is presented in Section 5.

2. Preliminaries

In this section, we briefly review the FKNN classifier and FCM clustering algorithm.

2.1 FKNN Classifier

FKNN was first proposed by Keller\(^17\) to solve a problem that each of the labeled samples is given equal importance in the process of deciding the class membership of the patterns to be classified. It is an extension of the k-nearest neighbor classifier (KNN), whose form of its results is different from the crisp version. Its nature is to assign a number to every element in the universe, which indicates that the degree which the element belongs to is a fuzzy set. The advantage in FKNN is that the degree of membership in a set can be specified, rather than just the binary\(^18\)\(^-\)\(^20\). This algorithm firstly finds the k-nearest neighbors to each testing sample according to the dissimilarity measure, gives initial membership of labeled samples, and then makes a decision according to the labeled neighbors, usually by assigning the label of the class that gains the most support voted by these k neighbors.

In FKNN, the initial degree of each labeled sample, which belongs to each class is binary (the degree may be 0
or 1). Actually, these degrees should not crisply be 0 or 1. In this case, a measure function that weighs the importance of samples is needed. The initial membership function is necessary for the initial partition matrix \( U = \{ u_{ij}(y_i) \}_{i=1}^n \) in FKNN, where \( t \) is the number of the labeled samples, \( c \) is the number of classes. \( u_{ij}(y_i) = u_{ij} \) denotes the degree of the labeled sample \( y_i = [y_{i1}, y_{i2}, \cdots, y_{in}]^T \) in the \( j \)th class. Let \( k \) be the number of the nearest neighbors, \( T = \{y_1, y_2, \cdots, y_t\} \) the labeled training set, and \( x = [x_1, x_2, \cdots, x_n]^T \) a testing sample, we calculate the Euclidean distances between the testing sample \( x \) and the samples \( y_t \). The training set \( T \) is composed by \( c \) classes \( T = \{P_1, P_2, \cdots, P_c\} \), where \( T = \{y_1, y_2, \cdots, y_t\} = \{P_1, P_2, \cdots, P_c\} \). In this algorithm, we try to find the \( k \)-nearest neighbors. The algorithm is summarized as follows.

Step 1) Initialize the value \( k \), training set \( T \), and membership matrix \( U \). For the given testing sample \( x \), calculate the distance between the \( n \)-dimensional testing sample \( x \) and all labeled samples \( y_t \), i.e., \( d(x, y_t) = \|x - y_t\| \).

Step 2) Sort the distances \( \{d(x, y_t)|i = 1, 2, \cdots, t\} \), determine and list the \( k \)-nearest neighbors \( T_k = \{y_{k1}^t, y_{k2}^t, \cdots, y_{kn}^t\} \) that are closest to \( x \) according to the sort.

Step 3) Gather the labels of the \( k \)-nearest neighbors, which respectively belong to \( c \) different subset \( P_j \), \( j = 1, 2, \cdots, c \).

Step 4) The testing sample \( x \) will be classified based on the majority classes of its nearest neighbors. The degree of the membership of \( x \) to the \( j \)th class is

\[
u_j(x) = \frac{\sum_{i=1}^{n} w_i u_{ij}}{\sum_{i=1}^{n} w_i}
\]

where \( w_i \) represents the importance of \( y_{ik} \), if \( j_{\text{max}} \) satisfies \( u_{j_{\text{max}}}(x) = \max \{ u_{ij}(x) | j = 1, 2, \cdots, c \} \), i.e., \( j_{\text{max}} = \text{arg max} \{ u_{ij}(x) \} \), then \( x \) is assigned to \( P_{j_{\text{max}}} \).

2.2 FCM Clustering

Since Zadeh[18] first articulated the fuzzy set theory which gave rise to the concept of partial membership, based on a membership function, fuzziness has received increasing attention. In the classical \( k \)-means procedure, each data point is assumed to be in exactly one cluster, i.e., the membership is either 1 or 0. In FCM, this condition can be assumed that each sample has some graded or “fuzzy” membership in a cluster.

Suppose that we are given a set of samples \( T = \{y_1, y_2, \cdots, y_t\} = \{P_1, P_2, \cdots, P_c\} \) and an initial number of classes \( c \), FCM clustering is based on the minimization of the objective function:

\[
J_m = \sum_{i=1}^{n} \sum_{j=1}^{c} u_{ij}^m \| y_i - v_j \|
\]

where \( m \) is any real number greater than 1, \( v_j \) is the center of the \( j \)th cluster, and \( \| y_i - v_j \| \) is any norm expression of the similarity between any measured data and the center. Fuzzy partitioning is carried out through an iterative optimization of the objective function shown in (2) with the update of membership \( u_{ij} \) and the cluster centers \( v_j \):

\[
u_{ij} = \frac{1}{\sum_{j=1}^{c} \left( \frac{\| y_i - v_j \|}{\| y_i - v_{j_{\text{max}}} \|} \right)^{\frac{2}{m-1}}} \sum_{j=1}^{c} u_{ij}^m\]

\[
v_j = \frac{\sum_{i=1}^{n} u_{ij}^m y_i}{\sum_{i=1}^{n} u_{ij}^m}.
\]

Then the algorithm is as follows.

Step 1) Input the value of \( c \), \( \varepsilon \), which is a threshold value, and initialize \( U \) matrix.

Step 2) Calculate the centers vectors \( v_j \) by (4).

Step 3) Update \( U = (u_{ij})_{n \times c} \) by (3).

Step 4) Repeat Step 2) and Step 3) until \( \| U^{t+1} - U^t \| \leq \varepsilon \).

In this section, FCM clustering groups a set of labeled samples \( T = \{y_1, y_2, \cdots, y_t\} = \{P_1, P_2, \cdots, P_c\} \) into \( c \) clusters \( T' = \{y_1^{\text{cluster}}, y_2^{\text{cluster}}, \cdots, y_t^{\text{cluster}}\} \).

3. Using FCM to Select Samples

3.1 Finding the Intersection to Select Samples

In this section, we describe the process of finding out the unreliable samples and explain the FKNN algorithm using FCM clustering. FCM clustering is an unsupervised learning algorithm, aiming to group \( t \) samples into \( c \) clusters in which each sample belongs to the cluster with the nearest class center.

As the training samples are continuously updating, some erroneously labeled samples should be taken out of the samples set (the training set). In this paper, the main idea is that FCM clustering groups the labeled training samples into \( c \) clusters. Then some reliable data are extracted by comparing classes \( T = \{P_1, P_2, \cdots, P_c\} \) with clusters \( T' = \{P_1^{\text{cluster}}, P_2^{\text{cluster}}, \cdots, P_c^{\text{cluster}}\} \). Once we find out the most representative data, then we say the remaining data is useless. Formally, the training sample set \( T = \bigcup_{j=1}^{c} P_j \), \( |T| = t \), where \( |\cdot| \) is the number of elements in \( T \), FCM
clustering is used to group $T$ into $c$ clusters, i.e., $T' = \{P_1^{\text{cluster}}, P_2^{\text{cluster}}, \ldots, P_c^{\text{cluster}}\}$.

Based on $P_j$ and $P_j^{\text{cluster}}$, we have $P_j^{\text{new}} = P_j \cap P_j^{\text{cluster}} = \{y | y \in P_j, y \in P_j^{\text{cluster}}\}$. The relationship between the training set and the modified training set is $\tilde{T} = \bigcup_{j=1}^{c} P_j^{\text{new}} \subseteq T'$. Then the intersection $P_j^{\text{new}}$ of the group and the corresponding cluster is the optimal training set, which eliminates the redundant samples. We have $P_j^{\text{new}} \subseteq P_j$, $P_j^{\text{new}} \subseteq P_j^{\text{cluster}}$, and we use $P_j - P_j^{\text{new}}$ to denote the unreliable data. Then the modified training set is $\tilde{T} = \bigcup_{j=1}^{c} P_j^{\text{new}}$, where $\tilde{T} = \sum_{j=1}^{c} P_j^{\text{new}}$ is the size of the modified training set. Based on the above discussion, we can briefly describe the algorithm as follows:

Step 1) Initialize the sample set $T = \{P_1, P_2, \ldots, P_c\}$.

Step 2) By FCM, we group the sample set $T$ into $c$ clusters $T' = \{P_1^{\text{cluster}}, P_2^{\text{cluster}}, \ldots, P_c^{\text{cluster}}\}$.

Step 3) Comparing each group $P_j$ with each cluster $P_j^{\text{cluster}}$, we have the intersection of a group $P_j$ and a cluster $P_j^{\text{cluster}}$, $P_j^{\text{new}} = P_j \cap P_j^{\text{cluster}}$, where $j = 1, 2, \ldots, c$.

Step 4) Obtain the new training set $\tilde{T} = \bigcup_{j=1}^{c} P_j^{\text{new}}$.

In fact, the main idea in the above discussion is to seek out the subset $P_j^{\text{new}}$ which can represent the initial training set. Clearly, $\tilde{T} = \{P_1^{\text{new}}, P_2^{\text{new}}, \ldots, P_c^{\text{new}}\}$ is the proposed training set.

### 3.2 FKNN Classifier Based on the Training Set after FCM Clustering

A problem that each of the training samples is given equal importance in deciding the class membership of the samples to be classified arises, regardless of their typicalness. FKM is modified by [17] assigning class membership to a sample, and weighting $k$ different neighbors.

Weight assignment has often been used to improve the performance of classification and clustering analysis [21]. Some important samples are assigned a higher weight, while less important samples are given a lower weight. In this section, different weights $w_1, w_2, \ldots, w_c$ are assigned to the $k$ nearest neighbors. Generally speaking, a number in $[0, 1]$ can be assigned to the samples to indicate its importance. In FKN, weight assignment $w_j = 1/d(x, y_j)^{2/(1-\alpha)}$ can completely represent the samples’ importance. The FKN classifier algorithm based on the modified training set is as follows:

Step 1) Input the sample set $T$, and an unlabeled testing sample $x$, we group the sample set $T$ into $c$ clusters $T' = \{P_1^{\text{cluster}}, P_2^{\text{cluster}}, \ldots, P_c^{\text{cluster}}\}$ using FCM.

Step 2) Compare $T = \{P_1, P_2, \ldots, P_c\}$ and $T' = \{P_1^{\text{cluster}}, P_2^{\text{cluster}}, \ldots, P_c^{\text{cluster}}\}$, and seek out $\tilde{T} = \bigcup_{j=1}^{c} P_j^{\text{new}} = \{y_1^{\text{new}}, y_2^{\text{new}}, \ldots, y_c^{\text{new}}\}$.

Step 3) Initialize $k (1 \leq k < n)$, $m = 2$, and generate the samples’ initial membership matrix $U = (u_{ij})_{n \times c}$, where the membership value $u_{ij}$ is in the interval $[0, 1]$.

Step 4) For the given $x$, calculate the distance $d(x, y_j^{\text{new}}) = \|x - y_j^{\text{new}}\|$ between $x = (x_1, x_2, \ldots, x_d)^T$ and all the samples $y_j^{\text{new}} = [y_1^{\text{new}}, y_2^{\text{new}}, \ldots, y_c^{\text{new}}]^T$ corresponding to the attribute set for the $j$th sample in $\tilde{T} = \{y_1^{\text{new}}, y_2^{\text{new}}, \ldots, y_c^{\text{new}}\}$.

Step 5. Sort the distance $\{d(x, y_j^{\text{new}})|j = 1, 2, \ldots, \tilde{T}\}$, and choose the $k$ nearest neighbors $\tilde{T}_k = \{(y_1^{\text{new}})^k, (y_2^{\text{new}})^k, \ldots, (y_c^{\text{new}})^k\}$.

Step 6. Compute $u_j(x)$ (the membership of testing sample $x$ in the $j$th class) using (1), if $f_{\text{max}} = \arg \max_j u_j(x)$, then $x$ belongs to the $j_{\text{max}}$ class $P_j^{\text{new}}$.

From this algorithm, the core samples in each class are extracted. FCM clustering is used to group training samples to different clusters according to a distance metric. Its goal is to find training samples with the attributes that are relatively similar to the attributes of the testing sample.

### 4. Experiments

In this section, we will evaluate the performance of the FKNN classifier based on FCM clustering using two different synthetic data. We respectively used the training samples and the modified training samples to classify the same testing samples so as to compare their recognition rate. A two-dimensional (three-dimensional) synthetic data set $D_1$ ($D_2$) for a three-class (four-class) problem was generated as follows. Each class has 2000 (4000) patterns which were independent and identically distributed (i.i.d) and drawn from a normal distribution with the mean as $(0, 0), (3, 0), (1.5, 3), (0, 0, 0), (3, 0, 0), (0, 3, 0), (0, 0, 3))$ and the same covariance identity matrix as $I_{2 \times 2}$ ($I_{3 \times 3}$) (see details in Table 1).

In $D_1$ ($D_2$), a total of 6000 (16000) patterns, of which 3000 (8000) patterns were set as the testing set and 3000 (8000) patterns were set as the training set, were the original experiment data. The number of training set scaled down from 3000 (8000) to 2717 (7147) after the FCM clustering, and the number of testing set was invariable. The training set’s remaining rate was 0.9057 (0.8934) (see details in Table 1). We respectively used training samples and the modified training samples to classify the same testing samples so as to compare their recognition rate. Fig. 1 shows two algorithms’ recognition rates for the same data on adjusted values of neighbors, where $k$ is from 1 to 10.
### Table 1: Fundamental state of the three different data (D1)

<table>
<thead>
<tr>
<th>Items</th>
<th>D1</th>
<th>D2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of samples</td>
<td>6000</td>
<td>16000</td>
</tr>
<tr>
<td>Number of class</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Number of attribute</td>
<td>C1</td>
<td>C1</td>
</tr>
<tr>
<td>class</td>
<td>C2</td>
<td>C2</td>
</tr>
<tr>
<td></td>
<td>C3</td>
<td>C3</td>
</tr>
<tr>
<td></td>
<td>C4</td>
<td>C4</td>
</tr>
<tr>
<td>Testing number</td>
<td>3000</td>
<td>8000</td>
</tr>
<tr>
<td>Training number</td>
<td>1000</td>
<td>2000</td>
</tr>
<tr>
<td>Remained number (new</td>
<td>1000</td>
<td>2000</td>
</tr>
<tr>
<td>training set)</td>
<td>2000</td>
<td>2000</td>
</tr>
<tr>
<td>Remaining rate</td>
<td>0.8800</td>
<td>0.9050</td>
</tr>
</tbody>
</table>

### Table 2: Data of experiment results

<table>
<thead>
<tr>
<th>Items</th>
<th>Recognition number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>k=1</td>
</tr>
<tr>
<td>Original (D1-3000)</td>
<td>2568</td>
</tr>
<tr>
<td>Modified (D1-3000)</td>
<td>2725</td>
</tr>
<tr>
<td>Original (D2-8000)</td>
<td>6756</td>
</tr>
<tr>
<td>Modified (D2-8000)</td>
<td>7116</td>
</tr>
</tbody>
</table>

Fig.1. Comparison results: (a) experiment result for D1 and (b) experiment result for D2.

As shown in Fig. 1 and Table 2, the proposed technique is better than the original method. For the same testing samples, the modified training set can greatly improve the recognition rate. It is clear that the improved method does eliminate a lot of unreliable samples. In semi-supervised classification, as the data size is becoming larger and larger, the FCM clustering algorithm plays a increasingly great role in eliminating unreliable samples.

### 5. Conclusions

In this paper, FCM clustering is used to extract a representative subset from the original training data, i.e., taking out some unreliable samples. The experiment result shows that FCM clustering is suitable for finding the reliable samples. It takes out some unreliable samples which have an impact on the classification results. From another point of view, we realize the purpose of samples selecting. This algorithm use FCM to cluster samples, which is an unsupervised method. Some more supervised information, such as statistical information, will be added to the algorithm in the future research.

### Acknowledgment

The authors would like to thank the support of research funds of Sichuan Key Laboratory of Intelligent Network Information Processing under Grant No. SGXZID1002-10 and Sichuan Key...
Technology Research and Development Program under Grant No. 2012GZ0019 and No. 2011FZ0051.

References


Chao Zhang was born in Hubei Province, China, in 1986. He received the B.S. degree from the Hubei University of Education, Wuhan in 2008 and the M. S. degree from Xihua University, Chengdu in 2011, both in mathematics. He is currently an assistant with the Department of Road Traffic Management, Sichuan Police College. His research interests include pattern recognition, intelligent transportation system, and information theory.

Jian-Mei Cheng was born in Xinjiang Province, China, in 1984. She received the B.S. degree from the Yili Normal University, Xinjiang in 2008 and the M.S. degree from Xihua University, Chengdu in 2011, both in mathematics. She is currently an assistant with the Department of Road Traffic Management, Sichuan Police College. Her research interests include applied mathematics and intelligent transportation system.

Liang-Zhong Yi was born in Sichuan Province, China in 1963. He received the Ph.D. degree from Southwest Jiaotong University, Chengdu in 2007. He is currently the Vice President of Sichuan Police College. His research interests include applied mathematics, intelligent information processing, and logical reasoning.