Proceedings of the 2007 IEEE Symposium on
Foundations of Computational Intelligence (FOCI 2007)

Likelihood Based Fuzzy Clustering for Data Sets of Mixed Features

Mahnhoon Lee
Computational Intelligence Group
Thompson Rivers University, Canada
mlee@tru.ca

Roelof K. Brouwer, *Senior Member*, *IEEE*
Visiting Professor
University of Stellenbosch, South Africa
rkbrouwer@ieee.org

Abstract – A noble clustering algorithm is presented for data sets of mixed features: numerical, ordinal and nominal. The algorithm uses the concept of fuzzy clustering to reduce negative effect from noises, and uses the iterative partitional algorithm founded on an optimization function to reduce the time complexity. The optimization function uses the likelihood for each individual feature as the optimization criterion of the similarity or likeliness between patterns and clusters, not like the fuzzy c-means clustering algorithm based on distance or the EM clustering algorithm. Hence the algorithm can quickly find fuzzy clusters having different distributions in the each feature level. The simulations show the algorithm to be quite efficient.

I. Introduction

Clustering partitions patterns into groups called clusters that have similar properties in terms of their features. Features can be numerical, ordinal and nominal. Many different types of clustering algorithms [1, 2] have been presented. The iterative partitional type clustering algorithms have been known to find quickly good quality clusters in a data set of numerical features as well as ordinal and nominal features with some modifications. The algorithms have an iteration loop founded on certain optimization functions regarding to the similarity or likeliness between patterns and clusters. At each iteration, the algorithms partition patterns into clusters, and the clusters are updated to better ones. The examples of the iterative partitional clustering algorithms are the k-means clustering algorithm [3, 4], the fuzzy c-mean clustering algorithm [5, 6], the modified k-means and fuzzy c-means algorithms [7-10], and the EM clustering algorithms [11, 12].

The iterative partitional clustering algorithms can be classified into hard clustering and soft clustering according to the methods of assignment of patterns into clusters. In hard clustering, such as k-means and EM [11], patterns are exclusively partitioned into clusters. Hence hard clustering algorithms tend to quickly get stuck in a local solution. One way of alleviating this problem is to use soft clustering, such as fuzzy c-means and EM with soft assignment [12]. The time complexity of the EM algorithm with soft assignment is high due to the additional use of the simulated annealing method after the EM algorithm. In soft clustering, patterns are partitioned into clusters with certain degrees repeatedly in the iteration loop.

The iterative partitional clustering algorithms can be classified in another way, regarding to the optimization criteria to measure the similarity or likeliness between patterns and clusters. The k-means, fuzz c-means and their variances use various types of distances, i.e., general type distances like Euclidean, Mahalanobis and so on, between patterns and the prototypes of clusters. Therefore, those algorithms do not efficiently separate non-convex type clusters as well as clusters having different distribution models. On the contrary the EM algorithm uses the likelihood as the optimization criterion. Gaussian distributions are assumed in clusters, and multivariate Gaussian probability density functions are used to measure the likelihoods of clusters given patterns. Therefore, the algorithm can find clusters having Gaussian distributions. However, the patterns of mixed features cannot be partitioned using the algorithm.

All the above iterative partitional clustering algorithms have some advantages and disadvantages. None of them, within our knowledge, supports all the concepts in good clustering algorithms, which were discussed in the above paragraphs: iterative partitioning for the less time complexity; soft clustering for reducing negative effect from noises; the likelihood as the optimization criterion for clusters having different distribution models; clustering of patterns of mixed features. In this paper, we present a likelihood based fuzzy clustering algorithm that supports all of those concepts.

In the next section, we discuss the meaning of the likelihood and how the likelihood can be used as the optimization criterion in iterative paritional clustering algorithms for patterns of mixed features. In section III, our likelihood based fuzzy clustering algorithm is presented, and the experimental results and concluding remarks follow.

II. Likelihood, Probability, and Iterative Partitional Clustering

There are several types of clustering algorithms – hierarchical, constructive, iterative partitional, and so on [1, 2]. The iterative partitional clustering algorithms have been effectively used in many application areas. Those algorithms find at each iteration in an iteration loop the best clusters to which each pattern might belong, and the clusters are updated to better ones as the algorithms repeat the iteration. Then the decision step in those algorithms to find the best clusters fitting to the pattern finds the cluster y maximizing the likeliness that the implication $x_1 \wedge x_2 \wedge x_3 \wedge x_4 \Rightarrow y$ (shortly $x_1x_2x_3x_4 \Rightarrow y$) is true. (For the purpose of easy explanation in this section, we assume that all patterns in a given data set

have four features, and we let x_1, x_2, x_3 and x_4 be the feature values of a pattern in the given data set.)

In general, the appealing model of cognition [13-15] is to generalize Aristotelian implication $\alpha\beta\gamma\delta \Rightarrow \varepsilon$ by finding that symbol ε which maximizes a posterior probability $P(\varepsilon \mid \alpha\beta\gamma\delta)$ (for concreteness, four assumed fact symbols α, β, γ and δ, and a conclusion symbol ε, each drawn from its own separate lexicon, with juxtaposition indicating Boolean AND.) The posterior probability is also called the *likelihood*.

Therefore, assuming that the features are mutually independent, the iterative paritional clustering problem becomes now the problem to find the cluster y maximizing the likelihood $P(y \mid x_1 x_2 x_3 x_4)$ for each pattern having feature values x_1, x_2, x_3 and x_4. By the Bayes' rule, the likelihood can be rewritten in (2.1).

$$P(y \mid x_1 x_2 x_3 x_4) = \frac{P(x_1 x_2 x_3 x_4 \mid y) P(y)}{P(x_1 x_2 x_3 x_4)} \tag{2.1}$$

One of the iterative partitional clustering algorithm, the EM algorithm, finds clusters in a data set of patterns of numerical features only, using the likelihood in (2.1). In the computation of the probability $P(x_1 x_2 x_3 x_4 \mid y)$ in the right side part in Formula (2.1), the algorithm uses a multivariate probability density function for each cluster, assuming each cluster has a multivariate Gaussian distribution. Hence the algorithm cannot be directly used in clustering of patterns of mixed features, such as numerical, ordinal and nominal.

In the iterative partitional clustering problem, it is assumed that the number of clusters is given and only one instance of each cluster is handled obviously in the interation loop. This implies that $P(y)$'s, i.e., the probabilities of clusters, in the right side part of Formula (2.1) are all equal. Therefore, finding the cluster y maximizing the likelihood $P(y \mid x_1 x_2 x_3 x_4)$, given each pattern having feature values x_1, x_2, x_3 and x_4, is equal to finding the cluster y maximizing the probability $P(x_1 x_2 x_3 x_4 \mid y)$. Furthermore, the assumption that the features of given patterns are mutually independent gives us:

$$P(x_1 x_2 x_3 x_4 \mid y) = P(x_1 \mid y) P(x_2 \mid y) P(x_3 \mid y) P(x_4 \mid y) \tag{2.2}$$

Then the problem of finding the cluster y maximizing the likeliness that the implication $x_1 x_2 x_3 x_4 \Rightarrow y$ is true becomes now the problem of finding the cluster maximizing the right side part in Formula (2.2). This reasoning conclusion is also advocated by the claim in [16] that a correct model of vertebrate cognition for $\alpha\beta\gamma\delta \Rightarrow \varepsilon$ is the maximization of cogency $P(\alpha\beta\gamma\delta \mid \varepsilon)$ and the maximization of the cogency is equal to the maximization of $P(\alpha \mid \varepsilon) P(\beta \mid \varepsilon) P(\gamma \mid \varepsilon) P(\delta \mid \varepsilon)$. We summarize all the above explanations as the next proposition. The proposition is the basis of our likelihood based fuzzy clustering algorithm for patterns of mixed features.

Proposition 1: The iterative partitional clustering problem is equal to the problem finding the cluster maximizing the product of the individual probabilities of feature values of a pattern, given the cluster.

Now we can handle any type of feature in the iterative partitional clustering problem, as long as we know how to find the probability of a feature value given a cluster, e.g., $P(x_i \mid y)$ in the right side part in (2.2). The meaning of the probability $P(x_i \mid y)$ is the probability of the feature value x_i, given all the same type feature values of the cluster y, not as the set of multivariate patterns belonging to the cluster. Hence $P(x_i \mid y)$ is the probability expressing how often x_i occurs in the set of all the same type feature values of y. That is, $P(x_i \mid y)$ is a sort of frequency of x in the set of all the same type feature values of y.

We can separate features into two categories – discrete ones and continuous ones. Nominal features and ordinal features fall in the discrete feature category. Discrete numerical features, e.g., integers, are also discrete. The numerical features of real numbers are the continuous feature type. When the feature is discrete, the normalized frequency distribution over the feature space might be used to find $P(x_i \mid y)$. When the feature is continuous, a parameterized probability density function representing the set of all the same type feature values of y, or the normalized frequency distribution over the discretized feature of the continuous feature might be used to find $P(x_i \mid y)$. Further discussion is given in subsection III.*B*.

III. Likelihood Based Iterative Partitional Fuzzy Clustering

Now we present an iterative partitional fuzzy clustering algorithm, based on Proposition 1 explained in the previous section. The algorithm has the very similar algorithmic structure as the fuzzy c-means clustering algorithm [5, 6]. Hence we first give a brief explanation of the fuzzy c-means clustering algorithm in the next subsection for the better understanding our algorithm explained in subsection *B*.

A. Fuzzy c-means clustering algorithm

The fuzzy c-means clustering algorithm for patterns consisting of all numerical features is defined by: Minimizing the objective function in (3.1).

$$Q = \sum_{i=1}^{c} \sum_{j=1}^{N} u_{i,j}^{m} \left\| X_j - V_i \right\|^2 \tag{3.1}$$

subject to

$$\begin{aligned} & u_{i,j} \geq 0, \ \forall i = 1, ..., c; \forall j = 1, ..., N \\ & \sum_{i=1}^{c} u_{i,j} = 1, \ \forall j = 1, ..., N \\ & \sum_{j=1}^{N} u_{i,j} > 0, \ \forall i = 1, ..., c \\ & m > 1 \end{aligned} \tag{3.2}$$

where $X_1,...,X_N$ are N patterns; c is the number of clusters; $u_{i,j}, \forall i = 1,...,c; \forall j = 1,...,N,$ is the membership of the pattern X_j to the i^{th} cluster; $V_i, \forall i = 1,...,c,$ is the prototype of the i^{th} cluster; and m is the indicator of fuzziness of clusters.

The minimal solution of the above objective function is:

$$u_{s,t} = \frac{1}{\sum_{i=1}^{c} \left(\frac{\|X_t - V_s\|}{\|X_t - V_i\|} \right)^{\frac{2}{m-1}}} \qquad \forall s = 1,...,c; \forall t = 1,...,N \tag{3.3}$$

$$V_s = \frac{\sum_{j=1}^{N} u_{s,j}^m X_j}{\sum_{j=1}^{N} u_{s,j}^m} \qquad \forall s = 1,...,c \tag{3.4}$$

From the above Formulas (3.3) and (3.4), we can see that the computation of the memberships in (3.3) uses the distances between patterns and the prototypes of clusters, and the prototypes are updated from the memberships. Formulas (3.3) and (3.4) are computed repeatedly in Algorithm 1 until a termination criterion is met.

Algorithm 1. (Fuzzy c-Means Clustering)

Initialization:

Initialize $V_i, \forall i = 1,...,c,$ with random numbers;

Repeat:

(Step 1) Compute $u_{i,j}, \forall i = 1,...,c; \forall j = 1,...,N,$ using (3.3);

(Step 2) Compute $V_i, \forall i = 1,...,c,$ using (3.4);

while $\varepsilon \le \max_{i,j} \{ |old\_u_{i,j} - new\_u_{i,j}| \}$;

B. Proposed likelihood based fuzzy clustering algorithm

The idea of our likelihood based fuzzy clustering algorithm for patterns of mixed features is to use the likelihoods of clusters, given patterns, as explained in the previous section with Proposition 1, instead of the distances used in the fuzzy c-means clustering algorithm (3.1). We explain our algorithm in the top-down manner. We start the explanation of the algorithm with the next objective functions (3.5) and (3.6). The objective functions are very similar to the objective function (3.1) in the fuzzy c-means clustering algorithm, except the optimization criteria: Maximizing

$$Q = \sum_{i=1}^{c} \sum_{j=1}^{N} w_{i,j}^m L_{i,j} \tag{3.5}$$

or equivalently, minimizing

$$Q = \sum_{i=1}^{c} \sum_{j=1}^{N} w_{i,j}^m U_{i,j} \tag{3.6}$$

subject to

$$\begin{aligned} & w_{i,j} \ge 0, \ \forall i = 1,...,c; \forall j = 1,...,N \\ & \sum_{i=1}^{c} w_{i,j} = 1, \ \forall j = 1,...,N \\ & \sum_{j=1}^{N} w_{i,j} > 0, \ \forall i = 1,...,c \end{aligned} \tag{3.7}$$

where

N	number of patterns
M	number of features
$c > 1$	number of clusters
$m > 1$	fuzziness factor
$X \triangleq \{X_1,..., X_N\}$	N patterns of M mixed features
$X_j \triangleq (x_{j,1},..., x_{j,M})$	j^{th} pattern
$L_{i,j}$	likelihood of the i^{th} cluster, given X_j
$U_{i,j}$	log-unlikelihood of the i^{th} cluster, given X_j
$W \triangleq [w_{i,j}]_{c \times N}$	membership matrix of X_j to the i^{th} cluster

(3.8)

Relying on Proposition 1 in section 2, we use the next definition (3.9) of the likelihood $L_{i,j}$ as the optimization criterion in the objective function (3.5).

$$L_{i,j} \triangleq \prod_{k=1}^{M} p_{i,j,k} \tag{3.9}$$

where $p_{i,j,k}$ is the probability of $x_{j,k}$, i.e, k^{th} feature value of j^{th} pattern, given the i^{th} cluster C_i, and is defined in (3.10) as explained in Proposition 1:

$$p_{i,j,k} \triangleq P(x_{j,k} \mid C_i) \tag{3.10}$$

The log-unlikelihood $U_{i,j}$ in (3.6) is defined as:

$$U_{i,j} \triangleq -\ln L_{i,j} \tag{3.11}$$

Then the objective function in (3.6) is rewritten as follow.

$$\begin{aligned} Q &= \sum_{i=1}^{c} \sum_{j=1}^{N} w_{i,j}^m U_{i,j} \\ &= \sum_{i=1}^{c} \sum_{j=1}^{N} w_{i,j}^m \left(-\ln \prod_{k=1}^{M} p_{i,j,k} \right) \\ &= -\sum_{i=1}^{c} \sum_{j=1}^{N} w_{i,j}^m \sum_{k=1}^{M} \ln p_{i,j,k} \end{aligned} \tag{3.12}$$

The objective in our likelihood based fuzzy clustering algorithm is to find the memberships $w_{i,j}, \forall i = 1,...,c; \forall j = 1,...,N,$ minimizing the objective function (3.6). First, assuming the log-unlikelihoods $U_{i,j}, \forall i = 1,...,c; \forall j = 1,...,N,$ (or equivalently the probabilities $p_{i,j,k}, \forall i = 1,...,c; \forall j = 1,...,N; \forall k = 1,...,M,$ in (3.10),) are given, the minimal solutions $w_{s,t}, \forall s = 1,...,c; \forall t = 1,...,N,$ are computed. Later, assuming the memberships are given, the probabilities $p_{i,j,k}, \forall i = 1,...,c; \forall j = 1,...,N; \forall k = 1,...,M,$ are computed. These two steps are repeated until a termination criterion is met, in the similar way of Algorithm 1 in the previous subsection.

In order to compute $w_{s,t}, \forall s = 1,...,c; \forall t = 1,...,N,$ we include the Lagrange multiplier and have the following function in (3.13).

$$R_t = \sum_{i=1}^{c} w_{i,t}^m U_{i,t} - \lambda \left(\sum_{i=1}^{c} w_{i,t} - 1 \right) \tag{3.13}$$

The partial derivative of R_t with respect to $w_{s,t}, \forall s = 1,...,c,$ is

$$\frac{\partial R_t}{\partial w_{s,t}} = m U_{s,t} w_{s,t}^{m-1} - \lambda \tag{3.14}$$

From $\frac{\partial R_t}{\partial w_{s,t}} = 0,$ we can find the minimal solution $w_{s,t}$ regarding to the Lagrange multiplier and the log-unlikelihoods, as follows.

$$mU_{s,t} w_{s,t}^{m-1} - \lambda = 0 \Rightarrow w_{s,t} = \left(\frac{\lambda}{mU_{s,t}}\right)^{\frac{1}{m-1}} \tag{3.15}$$

Now we can find the Lagrange multiplier λ by applying the unity constraint in (3.7).

$$w_{s,t} = \left(\frac{\lambda}{mU_{s,t}}\right)^{\frac{1}{m-1}}$$

$$\Rightarrow \quad \sum_{i=1}^{c} w_{i,t} = \left(\frac{\lambda}{m}\right)^{\frac{1}{m-1}} \sum_{i=1}^{c} \left(\frac{1}{U_{i,t}}\right)^{\frac{1}{m-1}} = 1 \tag{3.16}$$

$$\Rightarrow \quad \left(\frac{\lambda}{m}\right)^{\frac{1}{m-1}} = \frac{1}{\sum_{i=1}^{c} \left(\frac{1}{U_{i,t}}\right)^{\frac{1}{m-1}}}$$

The Lagrange multiplier λ is applied back into the formula (3.15) to obtain the final $w_{s,t}$.

$$w_{s,t} = \left(\frac{\lambda}{mU_{s,t}}\right)^{\frac{1}{m-1}} = \frac{1}{\sum_{i=1}^{c} \left(\frac{U_{s,t}}{U_{i,t}}\right)^{\frac{1}{m-1}}} \tag{3.17}$$

Next, we start the discussion how to compute the conditional probabilities $p_{i,j,k}$'s in (3.10), assuming the fuzzy memberships $w_{s,t}, \forall s = 1,...,c; \forall t = 1,...,N,$ are given. (Note that the log-likelihoods $U_{i,j}$'s in (3.11) are computed from $p_{i,j,k}$'s.) We consider the different types of features. We can separate features into two categories – discrete and continuous. Nominal and ordinal features as well as discrete numerical features fall in the discrete feature category. The numerical features of real numbers are the continuous feature type.

First, we assume that the k^{th} feature is discrete. A finite family of discrete values is not easily expressed with a differentiable distribution model. Hence we assume a simple sphere-shape model, and then the probability $p_{i,j,k}$ can be expressed in the term of the distance from the center or mode of the family, depending on the type of feature – numerical, ordinal or nominal. A mode of a discrete family is a most-frequent element in the family. The concept of mode is used in a variant of the fuzzy c-means clustering algorithm [8].

$$p_{i,j,k} \triangleq 1 - d\left(x_{l^*,k}, x_{j,k}\right), \tag{3.18}$$

where $x_{l^*,k}$ is a weighted center of all k^{th} feature members of the i^{th} cluster if the k^{th} feature is numerical, otherwise a mode. The distance function $d(\cdot,\cdot)$ is the Euclidean distance in the numerical feature, or $d(\cdot,\cdot)$ for an ordinal or nominal feature is defined as

$$d(x,y) \triangleq \begin{cases} 0 & x = y \\ 1-\varepsilon & x \neq y, \end{cases} \quad \text{where } 0<\varepsilon<<1 \tag{3.19}$$

Minimizing the objective function in (3.12) is equivalent to maximizing the inner sum. This is because the inner sum is independent of features, and $w_{s,j} \geq 0;\ \ln p_{s,j,k} \leq 0$, and hence the inner sum is negative. We rewrite the inner sum:

$$\begin{aligned} \sum_{j=1}^{N} w_{s,j}^m \ln\left(1 - d\left(x_{t,k}, x_{j,k}\right)\right) &= \sum_{j=1;x_{t,k}=x_{j,k}}^{N} w_{s,j}^m \ln(1) + \sum_{j=1;x_{t,k}\neq x_{j,k}}^{N} w_{s,j}^m \ln\varepsilon \\ &= \sum_{j=1;x_{t,k}\neq x_{j,k}}^{N} w_{s,j}^m \ln\varepsilon \\ &= \ln\varepsilon \sum_{j=1;x_{t,k}\neq x_{j,k}}^{N} w_{s,j}^m \\ &= \ln\varepsilon \left(\sum_{j=1}^{N} w_{s,j}^m - \sum_{j=1;x_{t,k}=x_{j,k}}^{N} w_{s,j}^m\right) \end{aligned} \tag{3.20}$$

Since $\ln\varepsilon \leq 0;\ \sum_{j=1}^{N} w_{s,j}^m - \sum_{j=1;x_{t,k}=x_{j,k}}^{N} w_{s,j}^m \geq 0$, maximizing the above inner sum is equal to minimizing $\sum_{j=1}^{N} w_{s,j}^m - \sum_{j=1;x_{t,k}=x_{j,k}}^{N} w_{s,j}^m$. Since $\sum_{j=1}^{N} w_{s,j}^m$ is a constant and $0 \leq \sum_{j=1;x_{t,k}=x_{j,k}}^{N} w_{s,j}^m \leq \sum_{j=1}^{N} w_{s,j}^m$, minimizing $\sum_{j=1}^{N} w_{s,j}^m - \sum_{j=1;x_{t,k}=x_{j,k}}^{N} w_{s,j}^m$ is equal to maximizing $\sum_{j=1;x_{t,k}=x_{j,k}}^{N} w_{s,j}^m$. This means that $x_{t,k}$ is a most frequent feature value w.r.t W, i.e., $x_{t,k}$ is a mode.

Second, let's assume that the k^{th} feature is continuous. One idea is to assume a simple sphere model for a family of continuous values, with the use of the probability in (3.18). (Note that the simple sphere model is used in the k-means clustering algorithm, fuzzy c-means clustering algorithm and their variants.) Another approach is to assume that the continuous feature has a parameterized probability density function, e.g., Gaussian distribution. (Note that the EM clustering algorithm assumes Gaussian distributions in clusters.)

Let $f_{i,k}$ be the Gaussian probability density function of the set of all the k^{th} feature values of the i^{th} cluster. $f_{i,k}$ with the standard deviation $\sigma_{i,k}$ and the mean $\mu_{i,k}$ is defined as follows.

$$f_{i,k}(x) = \frac{1}{\sigma_{i,k}\sqrt{2\pi}} e^{-\frac{(x-\mu_{i,k})^2}{2\sigma_{i,k}^2}} \tag{3.21}$$

Then $f_{i,k}$ can be used as $p_{i,j,k}$:

$$p_{i,j,k} \triangleq f_{i,j,k} \triangleq f_{i,k}(x_{j,k}) = \frac{1}{\sigma_{i,k}\sqrt{2\pi}} e^{-\frac{(x_{j,k}-\mu_{i,k})^2}{2\sigma_{i,k}^2}} \tag{3.22}$$

For that purpose, we need to compute $\sigma_{i,k}$ and $\mu_{i,k}$ from the set of all the k^{th} feature values of the fuzzy i^{th} cluster, not crisp. The computations are done by solving the following equations with the partial derivatives of the objective function in (3.12).

$$\frac{\partial Q}{\partial \mu_{s,t}} = \frac{\partial\left(\sum_{i=1}^{c}\sum_{j=1}^{N} w_{i,j}^m U_{i,j}\right)}{\partial \mu_{s,t}} = 0 \tag{3.23}$$

$$\frac{\partial Q}{\partial \sigma_{s,t}} = \frac{\partial \left(\sum_{i=1}^{c} \sum_{j=1}^{N} w_{i,j}^{m} U_{i,j} \right)}{\partial \sigma_{s,t}} = 0 \tag{3.24}$$

The equation in (3.23) is rewritten as follow, and $\mu_{i,k}$ is obtained:

$$\begin{aligned} & \frac{\partial \left(-\sum_{i=1}^{c} \sum_{j=1}^{N} w_{i,j}^{m} \sum_{k=1}^{M} \ln f_{i,j,k} \right)}{\partial \mu_{s,t}} \\ &= -\sum_{j=1}^{N} w_{s,j}^{m} \frac{\partial \left(\ln f_{s,j,t} \right)}{\partial \mu_{s,t}} \\ &= -\sum_{j=1}^{N} w_{s,j}^{m} \frac{\partial \left(\ln \left(\frac{1}{\sigma_{s,t}\sqrt{2\pi}} e^{-\frac{(x_{j,t}-\mu_{s,t})^2}{2\sigma_{s,t}^2}} \right) \right)}{\partial \mu_{s,t}} \\ &= -\frac{1}{\sigma_{s,t}^2} \sum_{j=1}^{N} w_{s,j}^{m} (x_{j,t} - \mu_{s,t}) = 0 \end{aligned} \tag{3.25}$$

$$-\frac{1}{\sigma_{s,t}^2} \sum_{j=1}^{N} w_{s,j}^{m} (x_{j,t} - \mu_{s,t}) = 0$$

$$\Rightarrow \quad \sum_{j=1}^{N} w_{s,j}^{m} (x_{j,t} - \mu_{s,t}) = 0 \tag{3.26}$$

$$\Rightarrow \quad \mu_{s,t} = \frac{\sum_{j=1}^{N} w_{s,j}^{m} x_{j,t}}{\sum_{j=1}^{N} w_{s,j}^{m}}$$

In the similar way, we can obtain $\sigma_{i,k}$ from (3.24), as follow.

$$\sigma_{s,t}^2 = \frac{\sum_{j=1}^{N} w_{s,j}^{m} (x_{j,t} - \mu_{s,t})^2}{\sum_{j=1}^{N} w_{s,j}^{m}} \tag{3.27}$$

Now, assuming that the fuzzy memberships $w_{s,t}, \forall s = 1,...,c; \forall t = 1,...,N,$ are given, we can compute the probability $p_{i,j,k}$ of the k^{th} feature value $x_{j,k}$ in the set of all the k^{th} feature values of the i^{th} cluster, from (3.18) when the feature is discrete, or from (3.22) when the feature is continuous.

As the last part in this subsection, Algorithm 2, our likelihood based fuzzy clustering algorithm for patterns of mixed features, is following. Algorithm 2 starts with random initial probabilities $p_{i,j,k}, \forall i = 1,...,c; \forall j = 1,...,N; \forall k = 1,...,M.$ The algorithm iterates two steps to find the cluster memberships from $p_{i,j,k}$'s, and next $p_{i,j,k}$'s from the memberships.

Algorithm 2. (Likelihood based fuzzy clustering)

Initialize $p_{i,j,k}, \forall i = 1,...,c; \forall j = 1,...,N; \forall k = 1,...,M,$ with random numbers;

Repeat:

(Step 1) Compute $w_{i,j}, \forall i = 1,...,c; \forall j = 1,...,N,$ using (3.17);

(Step 2) Compute $p_{i,j,k}$ for $\forall i = 1,...,c;\ \forall j = 1,...,N;$ $\forall k = 1,...,M$, using (3.18) or (3.22) according to the type of the k^{th} feature;

while $\varepsilon \le \max_{i,j} \left\{ \left| old\_w_{i,j} - new\_w_{i,j} \right| \right\}$;

Algorithm 3. (Likelihood for a continuous feature having a Gaussian distribution)

The fuzzy cluster memberships $w_{i,j}, \forall i = 1,...,c; \forall j = 1,...,N,$ are given;

(Step 1) Compute $\mu_{i,k}, \forall i = 1,...,c; \forall k = 1,...,M$, using (3.26);

(Step 2) Compute $\sigma_{i,k}, \forall i = 1,...,c; \forall k = 1,...,M$, using (3.27);

(Step 3) Compute $p_{i,j,k}, \forall i = 1,...,c; \forall j = 1,...,N; \forall k = 1,...,M,$ using (3.22);

IV. EXPERIMENTAL RESULTS

We compared two algorithms, the fuzzy c-means clustering algorithm (denoted FCM) and our likelihood based fuzzy clustering algorithm (denoted LFCM,) with synthetic data sets of patterns of two numerical features only. This is because data sets of patterns of two numerical features can be drawn easily and hence helpful for better understanding of the difference of the two algorithms. The comparisons were done with respect to the performance measures – partition coefficient (denoted BPC) and partition entropy (denoted BPE) in [17-19], and another measure partition index (denoted XBPI) in [19]. (Note that the more BPC, the better; the less BPE, the better; and the less XBPI, the better.)

We used two different synthetic data sets. One data set has two clusters that have normal distribution of different variances. The other data set has three clusters that have normal distribution of different variances also. The simulations were performed 10 times and averages were obtained. The next Table 1 shows the better performance of our algorithm LFCM for the both synthetic data sets in terms of the above three performance measurements. Figures 1 and 2 show the clusters to which patterns belong to mainly. We can easily see which one of the two algorithms separates clusters more clearly, especially around the borders between clusters.

	FCM 2 clusters	LFCM 2 clusters	FCM 3 clusters	LFCM 3 clusters	FCM 4 clusters	LFCM 4 clusters
BPC	0.868	**0.960**	0.783	**0.940**	0.655	**0.896**
BPE	0.232	**0.067**	0.363	**0.105**	0.675	**0.188**
XBPI	0.094	**0.069**	0.181	**0.069**	0.156	**0.098**

Table 1. Comparison results of the two clustering algorithms, with the data sets having 2, 3 and 4 clusters

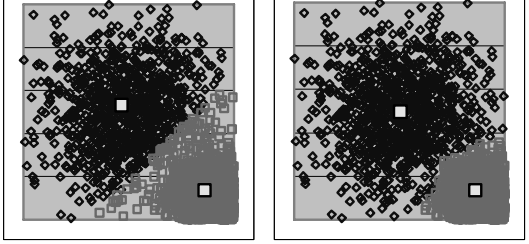

Fig. 1. Crisp clusters found by FCM and LFCM from a numerical data set having two clusters

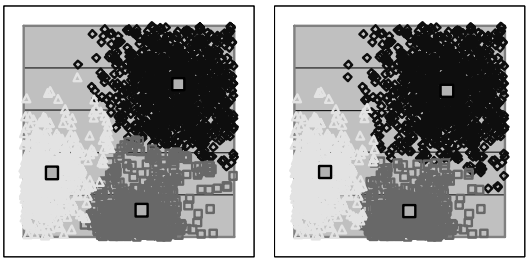

Fig. 2. Crisp clusters found by FCM and LFCM from a numerical data set having three clusters

V. CONCLUDING REMARKS

We presented a novel fuzzy clustering algorithm, not hard clustering in order to reduce negative effect from noises. The algorithm has the same time complexity as the fuzzy c-means clustering algorithm, which means the algorithm runs fast. The algorithm is based on the likelihood as the EM algorithm so that the algorithm can find clusters of different distributions. Furthermore, the likelihood is expressed as the product of the probabilities of individual features, and hence even patterns of mixed features can be clusters, not like the EM algorithm. The simulation showed the effectiveness of our algorithm.

It is expected that the proposed algorithm can be easily updated to the generalized clustering problem for the patterns of non-singleton feature values, such as fuzzy sets and intervals.

REFERENCES

[1] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data Clustering: A Review," *ACM Computing Survey*, vol. 31, 1999.

[2] P. Berkhin, "Survey of Clustering Data Mining Techniques," Accrue Software, San Jose, CA 2002.

[3] Z. B. MacQueen, "Some Methods of Classification and Analysis of Multivariate Observations," presented at Berkely Symposium on Mathematical Statistics and Probability, 1967.

[4] S. P. Lloyd, "Least Squares Quantization in PCM," *IEEE Transactions on Information Theory*, vol. IT-28, pp. 129-137, 1982.

[5] J. C. Dunn, "A Fuzzy Relative of the ISODATA Process and Its Use in detecting Compact Well-Separated Clusters," *Journal of Cybernetics*, vol. 3, pp. 32-57, 1973.

[6] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press, 1981.

[7] Y. El-Sonbaty and M. A. Ismail, "Fuzzy clustering for symbolic data," *Fuzzy Systems, IEEE Transactions on*, vol. 6, pp. 195-204, 1998.

[8] Z. Huang and M. K. Ng, "A fuzzy k-modes algorithm for clustering categorical data," *Fuzzy Systems, IEEE Transactions on*, vol. 7, pp. 446-452, 1999.

[9] O. M. San, V.-N. Huynh, and Y. Nakamori, "An Alternative Extension of the k-Means Algorithm for Clustering Categorical Data," *International Journal of Applied Mathematics and Computer Science*, vol. 14, pp. 241-247, 2004.

[10] R. Krishnapuram, A. Joshi, O. Nasraoui, and L. Yi, "Low-Complexity Fuzzy Relational Clustering Algorithms for Web Mining," *IEEE Trans. Fuzzy Systems*, vol. 9, pp. 595-607, 2001.

[11] A. P. Dempster, N. M. Laird, and R. D. B., "Maximum-Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society*, vol. B, pp. 1-38, 1977.

[12] S. Zhong and J. Ghosh, "A Unified Framework for Model-based Clustering," *Journal of Machine Learning Research*, vol. 4, pp. 1001-1037, 2003.

[13] E. A. Bender, *Mathematical Methods in Artificial Intelligence*. Los Alamitos, CA: IEEE Computer Society Press, 1996.

[14] N. J. Nilsson, *Artificial Intelligence: A New Synthesis*. San Francisco: Morgan Freeman Publishers, 1998.

[15] J. Pearl, *Causality*. Cambridge, UK: Cambridge University Press, 2000.

[16] R. Hecht-Nielsen, "Cogent Confabulation," *Neural Networks*, vol. 18, pp. 111-115, 2005.

[17] J. C. Bezdek, "Cluster Validity with Fuzzy Sets," *Journal of Cybern*, vol. 3, pp. 58-73, 1974.

[18] J. C. Bezdek, "Mathematical Models for Systematic and Taxonomy," presented at International Conference on Numerical Taxonomy, San Francisco, 1975.

[19] X. L. Xie and G. A. Beni, "Validity Measures for Fuzzy Clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 3, pp. 841-846, 1991.