Fuzzy Clustering of Quantitative and Qualitative Data

Christian Döring, Christian Borgelt, and Rudolf Kruse
Dept. of Knowledge Processing and Language Engineering
Otto-von-Guericke-University of Magdeburg
Universitätsplatz 2, D-39106 Magdeburg, Germany,
phone: +49.391.67.11358
fax: +49.391.67.12018
{doering,borgelt,kruse}@iws.cs.uni-magdeburg.de

***Abstract*— In many applications the objects to cluster are described by quantitative as well as qualitative features. A variety of algorithms has been proposed for unsupervised classification if fuzzy partitions and descriptive cluster prototypes are desired. However, most of these methods are designed for data sets with variables measured in the same scale type (only categorical, or only metric). We propose a new fuzzy clustering approach based on a probabilistic distance measure. Thus a major drawback of present methods can be avoided which lies in the vulnerability to favor one type of attributes.**

I. Introduction

Clustering mixed feature-type data sets is a task frequently encountered in data analysis. It may occur, for instance, in the field of user modeling when mining descriptive user segments is aimed at grouping users according to their particular interests and behavior. Little work has been done in defining and comparing algorithms that form expressive descriptions (prototypes) of fuzzy clusters from data described by a mix of quantitative and qualitative features. Ismail and El-Sonbaty were the first who applied the concept of fuzziness when partitioning datasets of symbolic objects [1]. Their *symbolic fuzzy c-means* approach can handle a wide variety of different attribute types including ordinal variables and intervals while forming expressive cluster prototypes. For instance, these prototypes can contain the weighted frequencies of modalities in the cluster. They applied Diday's dissimilarity measure for symbolic objects [2]. Although this measure allows for the calculation of dissimilarity also for continuous attributes, the cluster centers they construct are not appropriate for this quantitative type. Thus the symbolic fuzzy c-means algorithm is not suitable for mixture-type data sets. Yang, Hwang, and Chen overcome this limitation by modifying and extending the dissimilarity measure as well as the construction of cluster prototypes [3]. Their approach is able to cluster symbolic and fuzzy feature components. Values of continuous attributes can be embodied in their respective parameterizations of trapezoidal fuzzy numbers. That way, all attributes of mixed-type objects are utilized for finding clusters. Results of the method on toy data sets are provided.

Other related methods place more restrictions on the set of allowed attribute types. The *fuzzy k-modes* algorithm is restricted to categorical variables and finds the fuzzy cluster modes when using the simple matching dissimilarity measure for categories [4]. An extension called *k-prototypes* algorithm for dealing with both metric and categorical features has been proposed for the *hard k-modes* algorithm. Dissimilarity is then computed separately for the respective types of feature components. In the aggregated dissimilarity a weight parameter controls the influence of the nominal features on the total value [5]. Another approach that limits itself to categorical variables is a fuzzy-statistical algorithm [6]. As explained in [7], however, the ("pure") prototypes are assumed only for theoretical substantiation but they are not constructed. Similarly, all relational clustering algorithms are suitable to perform the classification task for mixed-type objects. These algorithms require a distance matrix as input, do not reference the actual input data, and as result they yield the indices of the most typical objects in the clusters. In this way, cluster prototypes are not constructed which were useful for characterizing the clusters. With our approach we want to overcome the following limitations of the present methods:

Problem 1: Formation of representative cluster prototypes. In the literature this issue is often motivated only, because the weighted mean equations for calculating new cluster centers do not work for symbols. Certainly this problem arises when alternately improving a clustering solution. Our motivation for prototypes which are able to represent the clusters' characteristics lies in the requirement that they should also be informative for a user, since prototypes are the result of data analysis. This is a major research objective in the presence of heterogenous feature and data types [8].

Problem 2: Calculation of dissimilarity. In our problem setting the Euclidean distance is inappropriate for determining the membership degrees of objects to clusters. For the calculation of dissimilarity between cluster centers and objects a variety of dissimilarity measures is available [2], [9]. However, special care has to be taken when the distances regarding the qualitative and quantitative features are computed separately first and aggregated to a total dissimilarity later. Then it has to be ensured that the qualitative and quantitative components are commensurable in order to avoid favoring one type of attributes. This can be achieved by standardizing numeric

data and/or introducing weights into the distance measure. However, weight parameters are always problematic, since the objective choice of values is often difficult [5]. Further, the classification results are easily distorted or can be manipulated when weights are chosen by the user [9].

II. Fuzzy Clustering

Most fuzzy clustering algorithms are objective function based: they determine an optimal (fuzzy) partition of a given data set $\mathbf{X} = \{\vec{x}_j \mid j = 1, \ldots, n\}$ into clusters by minimizing an objective function

$$J(\mathbf{X}, \mathbf{U}, \mathbf{C}) = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^m d_{ij} \tag{1}$$

subject to the constraints

$$\sum_{j=1}^{n} u_{ij} > 0, \qquad \text{for all } i \in \{1, \ldots, c\}, \qquad \text{and} \tag{2}$$

$$\sum_{i=1}^{c} u_{ij} = 1, \qquad \text{for all } j \in \{1, \ldots, n\}, \tag{3}$$

where $u_{ij} \in [0, 1]$ is the membership degree of datum $\vec{x}_j$ to cluster i and d_{ij} is the distance between datum x_j and cluster i. The $c \times n$ matrix $\mathbf{U} = (u_{ij})$ is called the fuzzy partition matrix and $\mathbf{C}$ describes the set of clusters by stating location parameters (i.e. the cluster center) and maybe size and shape parameters for each cluster. The parameter m, $m > 1$, is called the *fuzzifier* or *weighting exponent*. It determines the "fuzziness" of the classification: with higher values for m the boundaries between the clusters become softer, with lower values they get harder. Usually $m = 2$ is chosen.

Constraint (2) guarantees that no cluster is empty and constraint (3) ensures that the membership degrees of a datum to the clusters sum up to 1 and thus that each datum has the same total influence. Because of the second constraint this approach is usually called *probabilistic fuzzy clustering*, since with it the membership degrees for a datum formally resemble the probabilities of its being a member of the corresponding clusters. The partitioning property of a probabilistic clustering algorithm, which "distributes" the weight of a datum to the different clusters, is due to this constraint.

Unfortunately, the objective function J cannot be minimized directly. Therefore an iterative algorithm is used, which alternately optimizes the membership degrees and the cluster parameters. That is, first the membership degrees are optimized for fixed cluster parameters, then the cluster parameters are optimized for fixed membership degrees. The main advantage of this scheme is that in each of the two steps the optimum can be computed directly. By iterating the two steps the joint optimum is approached (although it cannot be guaranteed that the global optimum will be reached—the algorithm may get stuck in a local minimum of the objective function J).

The update formulae are derived by simply setting the derivative of the objective function J w.r.t. the parameters to optimize equal to zero (necessary condition for a minimum). Independent of the chosen distance measure we thus obtain the following update formula for the membership degrees [10]:

$$u_{ij} = \frac{d_{ij}^{-\frac{1}{m-1}}}{\sum_{t=1}^{c} d_{tj}^{-\frac{1}{m-1}}}, \tag{4}$$

The update formulae for the cluster parameters depend, of course, on what parameters are used to describe a cluster (location, shape, size) and on the chosen distance measure. Therefore a general update formula cannot be given.

The approach we develop in this paper is based on a mixture model for the process that generated the data and from this model assumption we derive a distance measure that is inversely proportional to the probability that a datum was generated by a cluster, similar to the idea underlying fuzzy maximum likelihood estimation (FMLE) [11]. Details are derived in the next two sections.

III. Mixture Models

In a mixture model it is assumed that a given data set $\mathbf{X} = \{\vec{x}_j \mid j = 1, \ldots, n\}$ has been drawn from a population of c clusters. Each cluster is characterized by a k-variate probability distribution, which is described by a prior probability of the cluster and a conditional probability density function (cpdf). The data generation process may then be imagined as follows: first a cluster i, $i \in \{1, \ldots, c\}$, is chosen for an example, indicating the cpdf to be used, and then the example is sampled from this cpdf. Consequently the probability of occurrence of a data point $\vec{x}$ can be computed as

$$p_{\vec{X}}(\vec{x}; \Theta) = \sum_{i=1}^{c} p_C(i; \Theta_i) p_{\vec{X}|C}(\vec{x}|i; \Theta_i),$$

where C is a random variable describing the cluster i chosen in the first step, $\vec{X}$ is a random vector describing the attribute values of the data point, and $\Theta = \{\Theta_1, \ldots, \Theta_c\}$ with each Θ_i, $i = 1, \ldots, c$, containing the parameters for one cluster (that is, its prior probability and the parameters of the cpdf) [12].

Here we consider a model in which the cpdf for each cluster consists of two parts, one for the numeric and one for the nominal attributes. We use the following notation: let I and K be the sets of indices of the numeric and the nominal attributes, respectively. The projection of the random vector $\vec{X}$ to the numeric attributes is denoted by $\vec{X}[I]$, the projection of a data point $\vec{x}$ by $\vec{x}[I]$. For the nominal part, $\vec{X}[K]$ denotes the projection of the random vector to the nominal attributes, X_l, $l \in K$, denotes the random variable describing the l-th nominal attribute, the domain of which is a set of s_l categories, coded by their indices, i.e. $\mathrm{dom}(X_l) = \{1, \ldots, s_l\}$. Finally, x_l denotes the value of the attribute X_l in a data point $\vec{x}$.

We assume that the numeric attributes are conditionally independent of the nominal ones given the cluster, so that their joint cpdf can be computed as a product of two terms, one for the numeric and one for the nominal attributes, i.e.

$$p_{\vec{X}|C}(\vec{x}|i; \Theta_i) = p_{\vec{X}[I]|C}(\vec{x}[I]|i; \Theta_i) \cdot p_{\vec{X}[K]|C}(\vec{x}[K]|i; \Theta_i).$$

Furthermore we assume that the joint cpdf of the numeric attributes is a multivariate normal distribution, i.e.

$$p_{\vec{X}[I]|C}(\vec{x}[I]|i;\Theta_i) = N(\vec{x}[I];\vec{\mu}_i,\mathbf{\Sigma}_i)$$
$$= \frac{1}{(2\pi|\mathbf{\Sigma}_i|)^{\frac{|I|}{2}}} \exp\left(-\frac{1}{2}(\vec{x}[I]-\vec{\mu})^\top \mathbf{\Sigma}_i^{-1}(\vec{x}[I]-\vec{\mu})\right),$$

where $\vec{\mu}_i$ is the mean vector and $\mathbf{\Sigma}_i$ the covariance matrix of the normal distribution, $i = 1,\ldots,c$.

Finally we assume that the nominal attributes are conditionally independent given the class, so that the joint probability of a combination of nominal attribute values (i.e. the probability of a vector $\vec{x}[K]$) can be computed as a product of conditional probabilities, one for each attribute, i.e.

$$p_{\vec{X}[K]|C}(\vec{x}[K]|i;\Theta_i) = \prod_{l\in K} p_{X_l|C}\left(x_l|i;\vec{\theta}_{l|i}\right),$$

where the $\vec{\theta}_{l|i}$, $l \in K$, are vectors

$$\vec{\theta}_{l|i} = \left(\vec{\theta}_{l|i}[1],\ldots,\vec{\theta}_{l|i}[s_l]\right)$$

stating the conditional probabilities of the different categories of the attribute X_l in the i-th cluster.

Consequently the parameters Θ_i of the i-th cluster are

$$\Theta_i = \{\theta_i, \vec{\mu}_i, \mathbf{\Sigma}_i, \vec{\theta}_{l_1|i},\ldots,\vec{\theta}_{l_{|K|}|i}\},$$

where θ_i is the prior probability of the i-th cluster and it is $\forall r,s; 0 \le r,s \le |K| : l_r, l_s \in K \wedge (r < s \rightarrow l_r < l_s)$, i.e. the vectors holding the cluster-specific conditional probability distributions are sorted w.r.t. the attribute index.

Assuming that the examples in a data set are independent and are drawn from the same distribution (i.e., that the distributions of their underlying random vectors $\vec{X}_j$ are identical), we can compute the probability of occurrence of the data set $\mathbf{X}$ as

$$P(\mathbf{X};\Theta) = \prod_{j=1}^{n}\sum_{i=1}^{c} \theta_i N(\vec{x}_j[I];\vec{\mu}_i,\mathbf{\Sigma}_i) \prod_{l\in K} \vec{\theta}_{l|i}[x_{jl}].$$

Note that we do not know the value the random variable C_j, which indicates the cluster, has for each example case $\vec{x}_j$. However, given the data point, we can compute the posterior probability that a data point has been sampled from the cpdf of the i-th cluster using Bayes' rule as

$$p_{C|\vec{X}}(i|\vec{x};\Theta) = \frac{p_C(i;\theta_i)\, p_{\vec{X}|C}(\vec{x}|i;\theta_i)}{p_{\vec{X}}(\vec{x};\Theta)}$$
$$= \frac{p_C(i;\theta_i)\, p_{\vec{X}|C}(\vec{x}|i;\theta_i)}{\sum_{t=1}^{c} p_C(t;\theta_t)\, p_{\vec{X}|C}(\vec{x}|t;\theta_t)}. \quad (5)$$

This posterior probability may be used to complete the data set w.r.t. the cluster, namely by splitting each example $\vec{x}_j$ into c examples, one for each cluster, which are weighted with the posterior probability $p_{C|\vec{X}_j}(i|\vec{x}_j;\Theta)$. This idea is used in the well known expectation maximization (EM) algorithm [13].

IV. The Proposed Algorithm

The mixture model provides the means to define the similarity measure for mixed-feature types data sets and it also allows for the formation of expressive cluster prototypes. We define the distance d_{ij} between the datum $\vec{x}_j$ and cluster i as the reciprocal of the probability that the datum $\vec{x}_j$ occurred *and* that it was generated by the component distribution underlying the cluster i. Then a high probability results in a small distance value, whereas a low probability that the datum was created by the distribution of cluster i indicates a large distance. Constructed in this intuitive way the distance measure is the reciprocal of the numerator in eqn. (5) of the posterior probabilities. We obtain

$$d_{ij} = \frac{1}{p_{C_j}(i;\Theta_i)\, p_{\vec{X}_j|C_j}(\vec{x}_j|i;\Theta_i)}$$
$$= \frac{1}{\theta_i\, N(\vec{x}_j[I];\vec{\mu}_i,\mathbf{\Sigma}_i)} \prod_{l\in K} \vec{\theta}_{l|i}[x_{jl}]^{-1}.$$

This definition has another interesting property: Inserted into the update equation for the membership degrees (4) it yields membership degrees that are equal to the posterior probabilities of the data points *provided* the fuzzifier $m = 2$ (see eqn. (5)). Only for this value of the weighting exponent the partial assignments u_{ij} of the data points to the clusters are their posterior probabilities [14]. Defining the distance measure in analogy to the posterior probabilities has been done first by Gath and Geva in [11].

Given the update equations for the membership degrees and the similarity measure as above, we still need to find expressions for re-calculating the parameters. Unfortunately, an updating scheme for the cluster parameters Θ_i cannot be derived by minimizing the objective function. Inserting the distance measure into eqn. (1) and setting its derivative w.r.t. the parameters equal to zero does not lead to analytically solvable expressions for the minimizing parameter values.

Using the mixture model, however, we can calculate the probability of occurrence of the data set for fixed assignments of data to clusters. After the membership degrees have been determined, we take into account that the partition of the data set is fuzzy. That is, each generated instance has a certain case weight in each cluster, which is discussed in more detail later. For now, let w_{ij} be the weight of example $\vec{x}_j$ with which it is generated by the cpdf underlying cluster i. The probability of the data to occur is then given by

$$P(\mathbf{X};\Theta) = \prod_{j=1}^{n}\prod_{i=1}^{c}\left(\theta_i\, N(\vec{x}_j[I];\vec{\mu}_i,\mathbf{\Sigma}_i) \prod_{l\in K} \vec{\theta}_{l|i}[x_{lj}]\right)^{w_{ij}}. \quad (6)$$

Since the w_{ij} are fixed during the re-estimation of the prototype parameters, we can determine those parameter values that maximize the above probability. By doing so we obtain the maximum likelihood estimates of the cpdf parameters in Θ_i. Instead of maximizing eqn. (6) directly, we take the logarithm of this formula (which leaves the maximum unchanged, because the logarithm is monotonous) and get the

following function of the parameters Θ_i:

$$\begin{aligned} F(\Theta) &= \log P(\mathbf{X};\Theta) \\ &= \sum_{j=1}^{n}\sum_{i=1}^{c} w_{ij}\log\left(\theta_i\right) && (7) \\ &+ \sum_{j=1}^{n}\sum_{i=1}^{c} w_{ij}\log\left(N(\vec{x}_j[I];\vec{\mu}_i,\mathbf{\Sigma}_i)\right) && (8) \\ &+ \sum_{j=1}^{n}\sum_{i=1}^{c} w_{ij}\sum_{l\in K}\log\left(\vec{\theta}_{l|i}[x_{lj}]\right) && (9) \end{aligned}$$

Since the terms containing the different parameters in Θ_i are not related, they can be maximized independently. The generalized estimators for the prior probabilities θ_i as well as the estimators for $\vec{\mu}_i$ and $\mathbf{\Sigma}_i$ in the normally distributed continuous features are obtained by maximizing terms (7) and (8), respectively. The well-known maximum likelihood estimators are [15]:

$$\widehat{\theta_i} = \frac{1}{n}\sum_{j=1}^{n} w_{ij}, \quad (10)$$

$$\widehat{\vec{\mu}_i} = \frac{\sum_{j=1}^{n} w_{ij}\vec{x}_j[I]}{\sum_{j=1}^{n} w_{ij}}, \quad (11)$$

$$\widehat{\mathbf{\Sigma}}_i = \frac{\sum_{j=1}^{n} w_{ij}(\vec{x}_j[I]-\vec{\mu}_i)(\vec{x}_j[I]-\vec{\mu}_i)^\top}{\sum_{j=1}^{n} w_{ij}}. \quad (12)$$

In order to obtain estimators for the parameters of the multinomial distributions we optimize term (9). We introduce Lagrange multipliers $\lambda_{l|i}$ for the constraints that $\sum_{r=1}^{s_l}\vec{\theta}_{l|i}[r]=1$ and set the derivative of this term w.r.t. the parameters $\vec{\theta}_{l|i}[r]$, $r=1,\ldots,s_l$, to zero (necessary condition for a maximum):

$$\begin{aligned} &\frac{\partial}{\partial\theta_{a|t}[b]}\left(\sum_{j=1}^{n}\sum_{i=1}^{c} w_{ij}\sum_{l\in K}\log\vec{\theta}_{l|i}[x_{jl}] + \lambda_{l|i}\left(1-\sum_{r=1}^{s_l}\vec{\theta}_{l|i}[r]\right)\right) \\ &= \sum_{j=1}^{n} w_{tj}\delta_{x_{ja},b}\frac{1}{\vec{\theta}_{a|t}[b]} - \lambda_{a|t} \stackrel{!}{=} 0, \quad (13) \end{aligned}$$

where $\delta_{\alpha,\beta}$ is the Kronecker symbol,

$$\delta_{\alpha,\beta} = \begin{cases} 1, & \text{if } \alpha=\beta, \\ 0, & \text{otherwise.} \end{cases}$$

From expression (13) we get s_l equations (for $b=1,\ldots,s_l$) for each value of t, $t=1,\ldots,c$:

$$\vec{\theta}_{a|t}[b] = \frac{1}{\lambda_{a|t}}\sum_{j=1}^{n} w_{tj}\delta_{x_{ja},b}. \quad (14)$$

Summing these equations over b yields

$$\begin{aligned} \sum_{b=1}^{s_l}\vec{\theta}_{a|t}[b] &= \sum_{b=1}^{s_l}\frac{1}{\lambda_{a|t}}\sum_{j=1}^{n} w_{tj}\delta_{x_{ja},b} \\ \Leftrightarrow \quad 1 &= \frac{1}{\lambda_{a|t}}\sum_{j=1}^{n} w_{tj}\sum_{b=1}^{s_l}\delta_{x_{ja},b} \\ \Leftrightarrow \quad \lambda_{a|t} &= \sum_{j=1}^{n} w_{tj}. \end{aligned}$$

Inserting this into eqn. (14) we obtain the maximum likelihood estimator for the probabilities of category r of the nominal attribute X_l given cluster i:

$$\widehat{\vec{\theta}_{l|i}[r]} = \frac{\sum_{j=1}^{n} w_{ij}\delta_{x_{jl},r}}{\sum_{j=1}^{n} w_{ij}}. \quad (15)$$

Summarizing, the weighted relative frequencies of the attribute categories in the clusters are used as estimates for their conditional probabilities. They are stored in the corresponding prototypes.

In order to completely describe the proposed algorithm the case weights w_{ij} have to be specified. We do so in analogy to the well-known objective function based algorithm of Gustafson and Kessel [16]. In this algorithm the formulae for updating the cluster means and the implicitly calculated covariance matrices for detecting hyper-ellipsoidal clusters are as follows [10]:

$$\vec{\mu}_i = \frac{\sum_{j=1}^{n} u_{ij}^m\vec{x}_j[I]}{\sum_{j=1}^{n} u_{ij}^m}, \quad (16)$$

$$\mathbf{\Sigma}_i = \frac{\sum_{j=1}^{n} u_{ij}^m(\vec{x}_j[I]-\vec{\mu}_i)(\vec{x}_j[I]-\vec{\mu}_i)^\top}{\sum_{j=1}^{n} u_{ij}^m}. \quad (17)$$

Apparently, they look very similar to the estimators with unspecified case weights as derived above (eqn. 11 and 12). The results of Gustafson and Kessel suggest to choose $w_{ij}=u_{ij}^m$ to arrive at analogous equations. This has been done first in the Fuzzy Maximum Likelihood Estimation (FMLE) algorithm as described in detail in [10]. We refer to our approach as extended FMLE, since this method is developed to cluster data sets which can also contain nominal attributes and since it employs the same weighting of instances.

Discussion: Fuzzy Maximum Likelihood Estimation is very similar to the well known Expectation Maximization (EM) algorithm applied to mixture decomposition under the assumption of a hidden variable indicating the class membership [13]. The relatedness of the methods is intimated by the almost identical derivation of the estimators. The differences, however, are made explicit in the different choice of the instance weights w_{ij} in both algorithms.

The EM algorithm maximizes the probability of the data set to occur by first calculating the posterior probabilities that a data point was created by the cpdfs of the clusters. This is done for fixed model parameters and yields the partial weights with which a data point belongs to the clusters (see eqn. (5)). Then these weighted assignments to clusters are used as instance weights in the estimators in order to optimize the likelihood. Thus, in the EM algorithm the partial assignments of a datum to the clusters are identical to the instance weight that is used when estimating the parameters, such that $w_{ij}=p_{C|\vec{X}}(i|\vec{x};\Theta)$.

In Fuzzy Maximum Likelihood Estimation, on the other hand, the partial assignments, namely the membership degrees u_{ij}, are different from the instance weights that are used when re-estimating the parameters, which are u_{ij}^m. Even in the case $m=2$, where the calculated membership degrees

TABLE I

THE PROBABILISTIC MODEL USED TO GENERATE THE DATA.

	A_1		A_2		A_3			A_3		
cluster	μ	σ^2	μ	σ^2	1	2	3	1	2	3
1	4	1	5	1	0.1	0.2	0.7	0.7	0.1	0.2
2	3	1	3	1	0.2	0.7	0.1	0.1	0.2	0.7
3	6	1	3	1	0.7	0.1	0.2	0.2	0.7	0.1

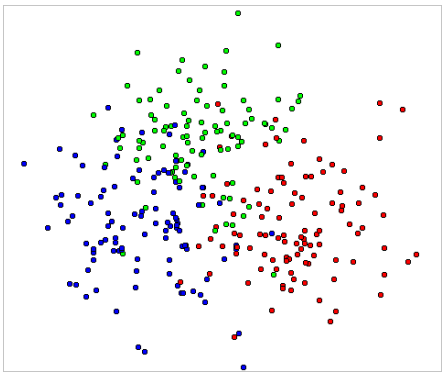

Fig. 1. The data points with classes.

u_{ij} correspond to the posterior probabilities (due to the special definition of the distance measure that is used in this approach), the case weights $w_{ij} = u_{ij}^m \leq u_{ij}$.

There is no choice of m such that the FMLE algorithm becomes identical to the EM algorithm, because m also appears in formula (4) for computing the membership degrees, which rules out the choice $m = 1$. It is the coupling of the exponents m and $\frac{1}{m-1}$ that distinguishes FMLE from EM.

V. EXPERIMENTS

Since our clustering approach is based on an explicit probabilistic model, real world data sets are not well suited for experiments, because for such data sets the underlying model is not known and thus an assessment whether it was recovered well is difficult, if possible at all. Therefore we relied on artificial data sets, which we generated with Monte Carlo simulation from a naive Bayes classifier. We used three classes and four attributes, two of them numeric and two nominal.

We ran several experiments using different models (i.e. naive Bayes classifiers) and different seed values for the data generator. From the generated data sets we selected one that shows the effects of using symbolic attributes in clustering as well as the differences to standard expectation maximization. The underlying model is shown in Table I. Attributes A_1 and A_2 are numeric and assumed to be conditionally independent given the cluster and normally distributed. Their columns state their mean values μ and variances σ^2. Attributes A_3 and A_4 are nominal with three values each and their columns state the conditional probabilities of these values given the cluster.

From this model we generated 300 data points randomly. The numeric part of this data set, i.e. the values of A_1 and A_2, with the color of the data points indicating the cluster that generated them, is shown in Figure 1. Obviously there is no clear cluster structure, so that a clustering approach that uses only the numeric attributes is not likely to recover the underlying model. However, the relative frequencies of the nominal attributes' values, which are close to the conditional probabilities in Table I, fairly clearly indicate the generating cluster and thus it can be expected that using them in the clustering process helps recovering the underlying model.

TABLE II

THE RESULT OF (EXTENDED) FMLE ON ALL ATTRIBUTES.

	A_1		A_2		A_3			A_3		
cluster	μ	σ^2	μ	σ^2	1	2	3	1	2	3
1	3.8	1.2	5.0	0.8	.06	.21	.74	.80	.01	.20
2	2.9	1.0	2.8	1.0	.23	.70	.07	.05	.22	.72
3	6.1	1.0	3.1	1.1	.76	.06	.18	.25	.62	.14

The results of processing this data set with the different algorithms are shown in Figures 2 to 7, which depict the numeric subspace of the data. Figures 2 and 3 show the generating model and a naive Bayes classifier induced from the data set. These are the reference models with which the results of the clustering algorithms are to be compared, because they show the "true" cluster structure. Figures 4 and 5 show the results of the EM algorithm using only the numeric or all attributes, respectively. Using only the numeric attributes there is a considerable difference in cluster shape to the reference, which is clearly reduced if the nominal attributes are taken into account. The same holds, in an even more pronounced way, for the (extended) FMLE algorithm, the results of which on only the numeric or on all attributes are shown in Figure 6 and 7, respectively. With the help of the nominal attributes the cluster structure is much better recovered. The result of the (extended) FMLE algorithm on all attributes, represented in the same way as the generating model, is shown in Table II.

From these and other experiments we can also report the following: EM proved to be the much more stable algorithm. FMLE tends to reduce the prior probability of one or more clusters to (almost) zero—as can already be guessed from the fact that in Figure 6 the blue (left) cluster is driven far to left, thus covering fewer data points than the other clusters. Sometimes this becomes extreme, with one cluster being driven completely out of the bulk of the data points. This tendency is even more pronounced if FMLE is not initialized with the result of the fuzzy c-means algorithm, while EM can be run directly without problems (initialized with Latin hypercube sampling). Using all attributes, (extended) FMLE becomes more stable, but only if the nominal attributes provide clear information about the cluster structure. However, even in the above example there are two possible results for the FMLE on all attributes, which occur about equally often and of which the better is shown in Figure 7. On the other hand, if there is a fairly clear structure in the numeric part of the data, using uninformative nominal attributes does not deteriorate the results, neither for EM nor for FMLE, so that we can conjecture that taking nominal attributes into account does not do any harm (this has to be confirmed by more tests, though).

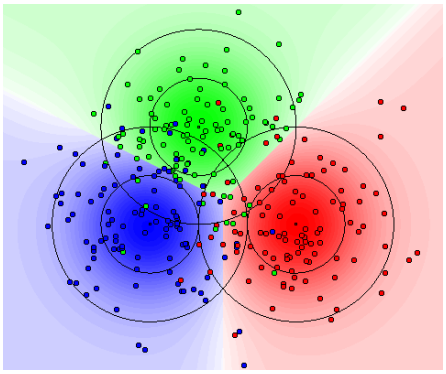

Fig. 2. The generating model.

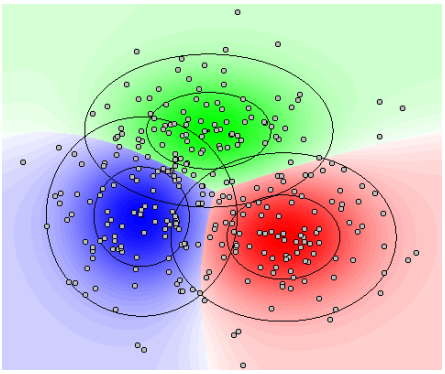

Fig. 4. EM on numeric attributes.

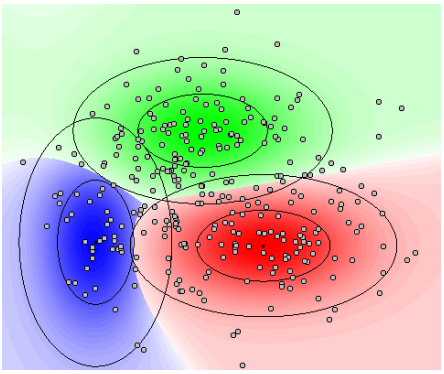

Fig. 6. FMLE on numeric attributes.

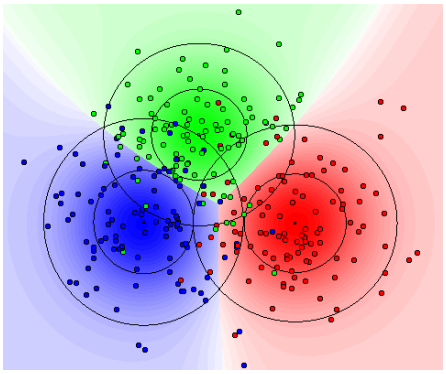

Fig. 3. Induced naive Bayes classifier.

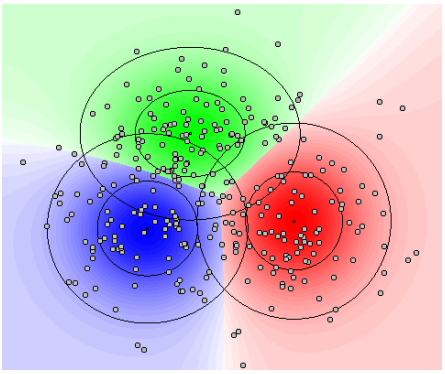

Fig. 5. EM on all attributes.

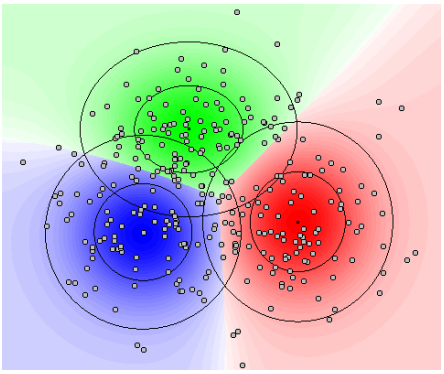

Fig. 7. FMLE on all attributes.

VI. Conclusions

In this paper we gave a brief overview of present approaches for fuzzy clustering mixed-type data. Our new approach is based on a probabilistic model and thus circumvents the problems of weighting dissimilarity components that can result from separately computing distances regarding the different attribute types. The formed cluster prototypes contain the weighted means and covariance matrices of the numeric attributes and weighted frequencies of the categories of the nominal attributes in the cluster. Being the results of the mining process, they are more informative than the prototypes of present approaches [4]. Our experiments show that nominal attributes help the clustering algorithm to find a good partition. However, the extended FMLE seems to be inferior to the classical EM algorithm, because it is less stable and fairly sensitive to the initialization of the cluster prototypes.

References

[1] Y. El-Sonbaty and M. Ismail, "Fuzzy clustering for symbolic data," *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 2, pp. 195–204, May 1998.

[2] K. Chidananda Gowda and E. Diday, "Symbolic clustering using a new dissimilarity measure," *Pattern Recognition*, vol. 24, no. 6, pp. 567–578, 1991.

[3] M.-S. Yang, P.-Y. Hwang, and D.-H. Chen, "Fuzzy clustering algorithms for mixed feature variables," *Fuzzy Sets and Systems*, vol. 141, no. 2, pp. 301–317, January 2004.

[4] Z. Huang and M. K. Ng, "A fuzzy k-modes algorithm for clustering categorical data," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 4, August 1999.

[5] Z. Huang, "Clustering large data sets with mixed numeric and categorical values," in *Proceedings of the First Pacific-Asia Conference on Knowledge Discovery and Data Mining*, ser. Lecture Notes in Artifical Intelligence, 1997, pp. 21–34.

[6] M. A. Woodbury and J. A. Clive, "Clinical pure types as a fuzzy partition," *J. Cybern.*, vol. 4-3, pp. 111–121, 1974.

[7] J. Bezdek, *Pattern Recognition With Fuzzy Objective Function Algorithms*. New York: Plenum Press, 1981.

[8] R. Kruse and C. Borgelt, "Information mining: Editorial," *Int. Journal of Approximate Reasoning (IJAR)*, vol. 32, pp. 63–65, 2003.

[9] H. H. Bock, *Automatische Klassifikation*. Göttingen, Zürich: Vadenhoeck & Ruprecht, 1974.

[10] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler, *Fuzzy Clustering*. Chichester, United Kingdom: Wiley, 1999.

[11] I. Gath and A. B. Geva, "Unsupervised optimal fuzzy clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 773—781, 1989.

[12] B. S. Everitt and D. J. Hand, *Finite Mixture Distributions*. London: Chapman & Hall, 1981.

[13] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood estimation from incomplete data via the em algorithm (with discussion)," *Journal of the Royal Statistical Society Series B*, vol. 39, pp. 1–38, 1977.

[14] J. Bezdek, J. Keller, R. Krishnapuram, and N. Pal, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*. Boston, London: Kluwer, 1999.

[15] J. Bilmes, "A gentle tutorial on the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models," University of Berkeley, Tech. Rep. ICSI-TR-97-021, 1997. [Online]. Available: citeseer.ist.psu.edu/bilmes98gentle.html

[16] E. E. Gustafson and W. C. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," in *Proc. of the IEEE Conference on Decision and Control, San Diego, Californien*. Piscataway, NJ: IEEE Press, 1979, pp. 761—766.