Fuzzy Clustering in Parallel Universes

Bernd Wiswedel and Michael R. Berthold
Department of Computer and Information Science, University of Konstanz
78457 Konstanz, Germany
{wiswedel,berthold}@inf.uni-konstanz.de

***Abstract*— We propose a modified fuzzy c-Means algorithm that operates on different feature spaces, so-called parallel universes, simultaneously. The method assigns membership values of patterns to different universes, which are then adopted throughout the training. This leads to better clustering results since patterns not contributing to clustering in a universe are (completely or partially) ignored. The outcome of the algorithm are clusters distributed over different parallel universes, each modeling a particular, potentially overlapping, subset of the data. One potential target application of the proposed method is biological data analysis where different descriptors for molecules are available but none of them by itself shows global satisfactory prediction results. In this paper we show how the fuzzy c-Means algorithm can be extended to operate in parallel universes and illustrate the usefulness of this method using results on artificial data sets.**

I. Introduction

In recent years, researchers have worked extensively in the field of cluster analysis, which has resulted in a wide range of (fuzzy) clustering algorithms [1], [2]. Most of the methods assume the data to be given in a single (mostly high-dimensional numeric) feature space. In some applications, however, it is common to have multiple representations of the data available. Such applications include biological data analysis, in which, e. g. molecular similarity can be defined in various ways. Fingerprints are the most commonly used similarity measure. A fingerprint in a molecular sense is a binary vector, whereby each bit indicates the presence or absence of a molecular feature. The similarity of two compounds can be expressed based on their bit vectors using the Tanimoto coefficient for example. Other descriptors encode numerical features derived from 3D maps, incorporating the molecular size and shape, hydrophilic and hydrophobic regions quantification, surface charge distribution, etc. [3]. Further similarities involve the comparison of chemical graphs, inter-atomic distances, and molecular field descriptors. However, it has been shown that often a single descriptor fails to show satisfactory prediction results [4].

Other application domains include web mining where a document can be described based on its content and on anchor texts of hyperlinks pointing to it [5]. Parts in CAD-catalogues can be represented by 3D models, polygon meshes or textual descriptions. Image descriptors can rely on textual keywords, color information, or other properties [6].

In the following we denote these multiple representations, i. e. different descriptor spaces, as *Parallel Universes* [7], each of which having representations of all objects of the data set. The challenge that we are facing here is to take advantage of the information encoded in the different universes to find clusters that reside in one or more universes each modeling one particular subset of the data. In this paper, we develop an extended fuzzy c-Means (FCM) algorithm [8] that is applicable to parallel universes, by assigning membership values from objects to universes. The optimization of the objective function is similar to the original FCM but also includes the learning of the membership values to compute the impact of objects to universes.

In the next section, we will discuss in more detail the concept of parallel universes; section III presents related work. We formulate our new clustering scheme in section IV and illustrate its usefulness with some numeric examples in section V.

II. Parallel Universes

We consider parallel universes to be a set of feature spaces for a given set of objects. Each object is assigned a representation in each single universe. Typically, parallel universes encode different properties of the data and thus lead to different measures of similarity. (For instance, similarity of molecular compounds can be based on surface charge distribution or fingerprint representation.) Note, due to these individual measurements they can also show different structural information and therefore exhibit distinctive clustering. This property differs from the problem setting in the so-called *Multi-View Clustering* [9] where a single universe, i. e. view, suffices for learning but the aim is on binding different views to improve the classification accuracy and/or accelerating the learning process. The objective for our problem definition is on identifying clusters located in different universes whereby each cluster models a subset of the data based on some underlying property.

Since standard clustering techniques are not able to cope with parallel universes, one could either restrict the analysis to a single universe at a time or define a descriptor space comprising all universes. However, using only one particular universe omits information encoded in the other representations and the construction of a joint feature space and the derivation of an appropriate distance measure are cumbersome and require great care as it can introduce artifacts.

III. Related Work

Clustering in parallel universes is a relatively new field of research. In [6], the DBSCAN algorithm is extended and

0-7803-9187-X/05/$20.00

applied to parallel universes. DBSCAN uses the notion of dense regions by means of core objects, i. e. objects that have a minimum number k of objects in their (ϵ-) neighborhood. A cluster is then defined as a set of (connected) dense regions. The authors extend this concept in two different ways: They define an object as a neighbor of a core object if it is in the ϵ-neighborhood of this core object either (1) in any of the representations or (2) in all of them. The cluster size is finally determined through appropriate values of ϵ and k. Case (1) seems rather weak, having objects in one cluster even though they might not be similar in any of the representational feature spaces. Case (2), in comparison, is very conservative since it does not reveal local clusters, i. e. subsets of the data that only group in a single universe. However, the results in [6] are promising.

Another clustering scheme called "Collaborative fuzzy clustering" is based on the FCM algorithm and was introduced in [10]. The author proposes an architecture in which objects described in parallel universes can be processed together with the objective of finding structures that are common to all universes. Clustering is carried out by applying the c-Means algorithm to all universes individually and then by exchanging information from the local clustering results based on the partitioning matrices. Note, the objective function, as introduced in [10], assumes the same number of clusters in each universe and, moreover, a global order on the clusters which—in our opinion—is very restrictive due to the random initialization of FCM.

A supervised clustering technique for parallel universes was given in [7]. It focuses on a model for a particular (minor) class of interest by constructing local neighborhood histograms, so-called Neighborgrams for each object of interest in each universe. The algorithm assigns a quality value to each Neighborgram and greedily includes the best Neighborgram, no matter from which universe it stems, in the global prediction model. Objects that are covered by this Neighborgram are finally removed from consideration in a sequential covering manner. This process is repeated until the global model has sufficient predictive power.

Blum and Mitchell [5] introduced co-training as a semi-supervised procedure whereby two different hypotheses are trained on two distinct representations and then bootstrap each other. In particular they consider the problem of classifying web pages based on the document itself and on anchor texts of inbound hyperlinks. They require a conditional independence of both universes and state that each representation should suffice for learning if enough labeled data were available. The benefit of their strategy is that (inexpensive) unlabeled data augment the (expensive) labeled data by using the prediction in one universe to support the decision making in the other.

Other related work includes reinforcement clustering [11] and extensions of partitioning methods—such as k-Means, k-Medoids, and EM—and hierarchical, agglomerative methods, all in [9].

IV. Clustering Algorithm

In this section, we introduce all necessary notation, review the FCM algorithm and formulate a new objective function that is suitable to be used for parallel universes. The technical details, i. e. the derivation of the objective function, can be found in the appendix section.

In the following, we consider $|U|$, $1 \leq u \leq |U|$, parallel universe, each having representational feature vectors for all objects $\vec{x}_{i,u} = (x_{i,u,1}, \ldots, x_{i,u,a}, \ldots x_{i,u,A_u})$ with A_u the dimensionality of the u-th universe. We depict the overall number of objects as $|T|$, $1 \leq i \leq |T|$. We are interested in identifying c_u clusters in universe u. We further assume appropriate definitions of distance functions for each universe $d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2$ where $\vec{w}_{k,u} = (\vec{w}_{k,u,1}, \ldots, \vec{w}_{k,u,a}, \ldots \vec{w}_{k,u,A_u})$ denotes the k-th prototype in the u-th universe.

We confine ourselves to the Euclidean distance in the following. In general, there are no restrictions to the distance metrics other than the differentiability. In particular, they do not need to be of the same type in all universes. This is important to note, since we can use the proposed algorithm in the same feature space, i. e. $\vec{x}_{i,u_1} = \vec{x}_{i,u_2}$ for any u_1 and u_2, but different distance measure across the universes.

A. Formulation of new objective function

The original FCM algorithm relies on one feature space only and minimizes the objective function as follows. Note that we omit the subscript u here as we consider only one universe:

$$J_m = \sum_{i=1}^{|T|} \sum_{k=1}^{c} v_{i,k}^m \, d\left(\vec{w}_k, \vec{x}_i\right)^2 \, .$$

$m \in (1, \infty)$ is a fuzzyfication parameter, and $v_{i,k}$ the respective value from the partition matrix, i. e. the degree to which pattern $\vec{x}_i$ belongs to cluster k. This function is subject to minimization under the constraint

$$\forall\, i : \sum_{k=1}^{c} v_{i,k} = 1 \, ,$$

requiring that the coverage of any pattern i needs to accumulate to 1.

The above objective function assumes all cluster candidates to be located in the same feature space and is therefore not directly applicable to parallel universes. To overcome this, we introduce a matrix $z_{i,u}$, $1 \leq i \leq |T|$, $1 \leq u \leq |U|$, encoding the membership of patterns to universes. A value $z_{i,u}$ close to 1 denotes a strong contribution of pattern $\vec{x}_i$ to the clustering in universe u, and a smaller value, a respectively lesser degree. $z_{i,u}$ has to satisfy standard requirements for membership degrees: it must accumulate to 1 considering all universes and must be in the unit interval.

The new objective function is given with

$$J_{m,n} = \sum_{i=1}^{|T|} \sum_{u=1}^{|U|} z_{i,u}^n \sum_{k=1}^{c_u} v_{i,k,u}^m d_u \left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2 \, . \qquad (1)$$

Parameter $n \in (1, \infty)$ allows (analogous to m) to have impact on the fuzzyfication of $z_{i,u}$: The larger n the more equal the distribution of $z_{i,u}$, giving each pattern an equal impact to all universes. A value close to 1 will strengthen the composition of $z_{i,u}$ and assign high values to universes where a pattern shows good clustering behavior and small values to those where it does not. Note, we now have $|U|$ different partition matrices (v) to assign membership degrees of objects to cluster prototypes.

As in the standard FCM algorithm, the objective function has to fulfill side constraints. The coverage of a pattern among the partitions in each universe must accumulate to 1:

$$\forall\, i,\, u : \sum_{k=1}^{c_u} v_{i,k,u} = 1\,. \tag{2}$$

Additionally, as mentioned above, the membership of a pattern to different universes has to be in total 1, i. e.

$$\forall\, i : \sum_{u=1}^{|U|} z_{i,u} = 1\,. \tag{3}$$

The minimization is done with respect to the parameters $v_{i,k,u}$, $z_{i,u}$, and $\vec{w}_{k,u}$. Since the derivation of the objective function is more of technical interest, please refer to the appendix for details.

The optimization splits into three parts. The optimization of the partition values $v_{i,k,u}$ for each universe; determining the membership degrees of patterns to universes $z_{i,u}$ and finally the adaption of the center vectors of the cluster representatives $\vec{w}_{k,u}$.

The update equations of these parameters are given in (4), (5), and (6). For the partition values $v_{i,k,u}$, it follows

$$v_{i,k,u} \;=\; \frac{1}{\sum\limits_{\bar{k}=1}^{c_u} \left(\frac{d_u(\vec{w}_{k,u}, \vec{x}_{i,u})^2}{d_u\left(\vec{w}_{\bar{k},u}, \vec{x}_{i,u}\right)^2} \right)^{\frac{1}{m-1}}}\,. \tag{4}$$

Note, this equation is independent of the values $z_{i,u}$ and is therefore identical to the update expression in the single universe FCM. The optimization with respect to $z_{i,u}$ yields

$$z_{i,u} \;=\; \frac{1}{\sum\limits_{\bar{u}=1}^{|U|} \left(\frac{\sum_{k=1}^{c_u} v_{i,k,u}^m d_u(\vec{w}_{k,u}, \vec{x}_{i,u})^2}{\sum_{k=1}^{c_{\bar{u}}} v_{i,k,\bar{u}}^m d_{\bar{u}}(\vec{w}_{k,\bar{u}}, \vec{x}_{i,\bar{u}})^2} \right)^{\frac{1}{n-1}}}\,, \tag{5}$$

and update equation for the adaption of the prototype vectors $\vec{w}_{k,u}$ is of the form

$$w_{k,u,a} \;=\; \frac{\sum_{i=1}^{|T|} z_{i,u}^n\, v_{i,k,u}^m\, x_{i,u,a}}{\sum_{i=1}^{|T|} z_{i,u}^n\, v_{i,k,u}^m}\,. \tag{6}$$

Thus, the update of the prototypes depends not only on the partitioning value $v_{i,k,u}$, i. e. the degree to which pattern i belongs to cluster k in universe u, but also to $z_{i,u}$ representing the membership degrees of patterns to the current universe of interest. Patterns with larger values $z_{i,u}$ will contribute more to the adaption of the prototype vectors, while patterns with a smaller degree accordingly to a lesser extent.

Equipped with these update equations, we can introduce the overall clustering scheme in the next section.

B. Clustering algorithm

Similar to the standard FCM algorithm, clustering is carried out in an iterative manner, involving three steps:

1) Update of the partition matrices (v)
2) Update of the membership degrees (z)
3) Update of the prototypes ($\vec{w}$)

More precisely, the clustering procedure is given as:

(1)	*Given:* Input pattern set described in $\|U\|$ parallel universes: $\vec{x}_{i,u}$, $1 \le i \le \|T\|$, $1 \le u \le \|U\|$
(2)	*Select:* A set of distance metrics $d_u(\cdot,\cdot)^2$, and the number of clusters for each universe c_u, $1 \le u \le \|U\|$, define parameter m and n
(3)	*Initiate:* Partition matrices $v_{i,k,u}$ with random values and the cluster prototypes by drawing samples from the data. Assign equal weight to all membership degrees $z_{i,u} = \frac{1}{\|U\|}$.
(4)	*Train:*
(5)	*Repeat*
(6)	Update partitioning values $v_{i,k,u}$ according to (4)
(7)	Update membership degrees $z_{i,u}$ according to (5)
(8)	Compute prototypes $\vec{w}_{i,u}$ using (6)
(9)	*until* a termination criterion has been satisfied

The algorithm starts with a given set of universe definitions and the specification of the distance metrics to use. Also, the number of clusters in each universe needs to be defined in advance. The membership degrees $z_{i,u}$ are initialized with equal weight (line (3)), thus having the same impact on all universes. The optimization phase in line (5) to (9) is—in comparison to the standard FCM algorithm—extended by the optimization of the membership degrees, line (7). The possibilities for the termination criterion in line (9) are manifold. One can stop after a certain number of iterations or use the change of the value of the objective function (1) between two successive iterations as stopping criteria. There are also more sophisticated approaches, for instance the change to the partition matrices during the optimization.

Just like the FCM algorithm, this method suffers from the fact that the user has to specify the number of prototypes to be found. Furthermore, our approach even requires the definition of cluster counts *per* universe. There are numerous approaches to suggest the number of clusters in the case of the standard FCM, [12], [13] to name but a few. Although we have not yet studied their applicability to our problem definition we do believe that some of them can be adapted to be used in our context as well.

V. Experimental Results

In order to demonstrate this approach, we generated synthetic data sets with different numbers of parallel universes. For simplicity we restricted the size of a universe to 2 dimensions and generated 2 Gaussian distributed clusters

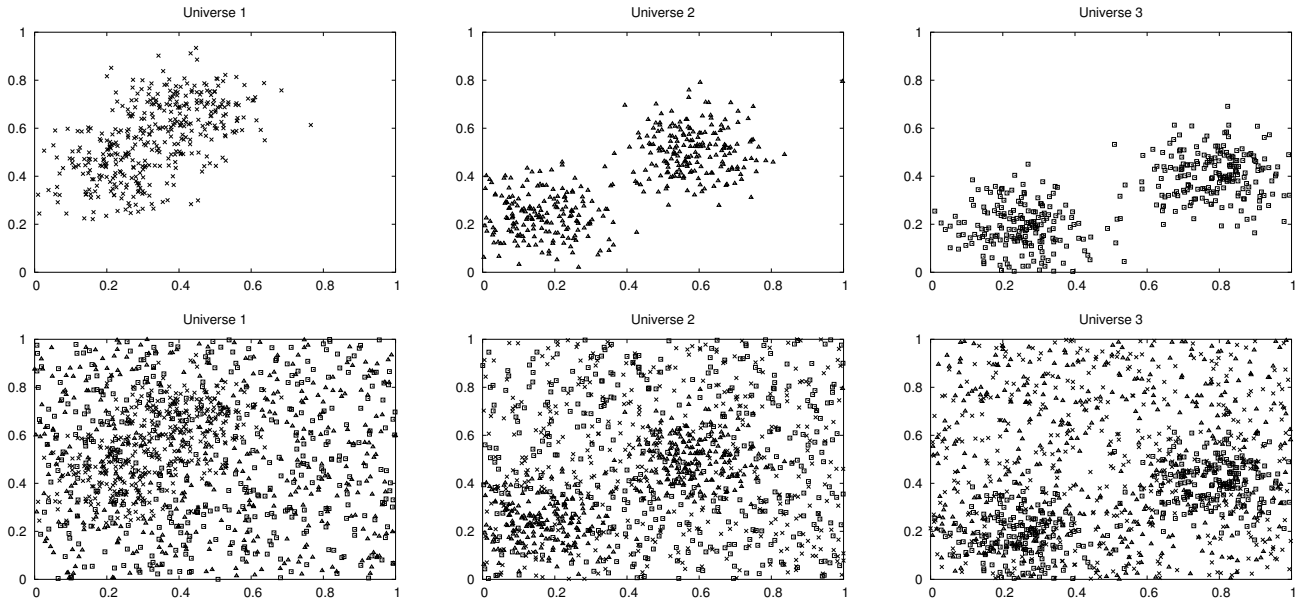

Fig. 1. Three universes of a synthetic data set. The top figures show only objects that were generated within the respective universe (using two clusters per universe). The bottom figures show all patterns; note that most of them (i. e. the ones from the other two universes), are noise in this particular universe. For clarification we use different shapes for objects that origin from different universes.

(per universe). We then assigned each object to one of the universes and drew its features in that universe according to the distribution of the cluster (randomly picking one of the two). The features of this object in the other universes were drawn from a uniform distribution, i. e. they represent noise in these universes. Figure 1 shows an example data set with three universes. The top figures show only the objects that were generated to cluster in the respective universe, i. e. they define the reference clustering. The bottom figures include all objects and show the universes as they are presented to the clustering algorithm. Approximately 2/3 of the data do not contribute to clustering in a universe and therefore are noise.

To compare the results we applied the standard FCM algorithm to the joint feature space of all universes and set the number of desired clusters to the overall number of generated clusters. Thus, the numbers of dimensions and clusters were two times the number of universes.We forced a crisp cluster membership decision based on the highest value of the partition values, i. e. the cluster to a pattern i is determined by $\bar{k} = \arg\max_{1 \le k \le c}\{v_{i,k}\}$. When the universe information was taken into account, a cluster decision is based on the highest value of $z_{i,u} \cdot v_{i,k,u}$. Thus, universe and cluster index (u, k) for pattern i are computed as $(\bar{u}, \bar{k}) = \arg\max_{\substack{1 \le u \le |U| \\ 1 \le k \le c_u}} \{z_{i,u} \cdot v_{i,k,u}\}$.

We used the following quality measure to compare different clustering results [6]:

$$Q_K(C) = \sum_{C_i \in C} \frac{|C_i|}{|T|} \cdot (1 - \mathrm{entropy}_K(C_i)) \ ,$$

where K is the reference clustering, i. e. the clusters as generated, C the clustering to evaluate, and $\mathrm{entropy}_K(C_i)$ the entropy of cluster C_i with respect to K. This function is 1 if C equals K and 0 if all clusters are completely puzzled such that they all contain an equal fraction of the clusters in K. Thus, the higher the value, the better the clustering.

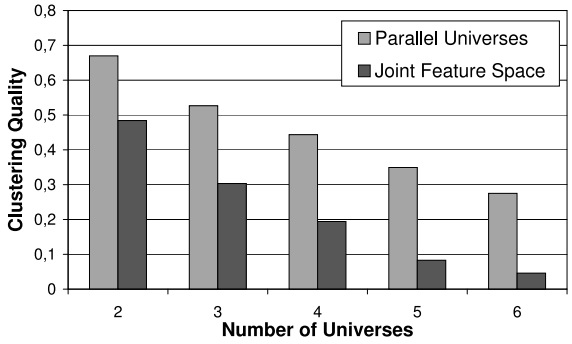

Fig. 2. Clustering quality for 5 different data sets. The number of universes ranges from 2 to 6 universes. Note how the cluster quality of the joint feature space drops sharply whereas the parallel universe approach seems less affected. An overall decline of cluster quality is to be expected since the number of clusters to be detected increases.

Figure 2 summarizes the quality values for 5 experiments compared to the standard FCM. The number of clusters ranges from 2 to 6. Clearly, for this data set, our algorithm takes advantage of the information encoded in different universes and identifies the major parts of the original clusters. Obviously this is by no means proof that the method will always detect clusters spread out over parallel universes but these early results are quite promising.

VI. Conclusion

We considered the problem of unsupervised clustering in parallel universes, i. e. problems where multiple representations are available for each object. We developed an extension of the fuzzy c-Means algorithm that uses membership degrees to model the impact of objects to the clustering in a particular universe. By incorporating these membership values into the objective function, we were able to derive update equations which minimize the objective with respect to these values, the partition matrices, and the prototype center vectors. The clustering algorithm works in an iterative manner using these equations to compute a (local) minimum. The result are clusters located in different parallel universes, each modeling only a subset of the overall data and ignoring data that do not contribute to clustering in a universe.

We demonstrated that the algorithm performs well on a synthetic data set and exploits the information of having different universes nicely. Further studies will concentrate on the applicability of the proposed method to real world data, heuristics that adjust the number of clusters per universe, and the influence of noisy data.

Acknowledgment

This work was partially supported by the Research Training Group 1024 funded by the Deutsche Forschungsgemeinschaft (DFG).

Appendix

In order to compute a minimum of the objective function (1) with respect to (2) and (3), we exploit a Lagrange technique to merge the constrained part of the optimization problem with the unconstrained one. This leads to a new objective function F_i that we minimize for each pattern $\vec{x}_i$ individually,

$$\begin{aligned} F_i &= \sum_{u=1}^{|U|} z_{i,u}^n \sum_{k=1}^{c_u} v_{i,k,u}^m d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2 \\ &\quad + \sum_{u=1}^{|U|} \mu_u \left(1 - \sum_{k=1}^{c_u} v_{i,k,u}\right) + \lambda \left(1 - \sum_{u=1}^{|U|} z_{i,u}\right). \end{aligned} \quad (7)$$

The parameters λ and μ_u, $1 \leq u \leq |U|$, denote the Lagrange multiplier to take (2) and (3) into account. The necessary conditions leading to local minima of F_i read as

$$\frac{\partial F_i}{\partial z_{i,u}} = 0, \quad \frac{\partial F_i}{\partial v_{i,k,u}} = 0, \quad \frac{\partial F_i}{\partial \lambda} = 0, \quad \frac{\partial F_i}{\partial \mu_u} = 0\,, \quad 1 \leq u \leq |U|, \quad 1 \leq k \leq c_u. \quad (8)$$

In the following we will derive update equations for the z and v parameters. Evaluating the first derivative of the equations in (8) yields the expression

$$\frac{\partial F_i}{\partial z_{i,u}} = n\, z_{i,u}^{n-1} \sum_{k=1}^{c_u} v_{i,k,u}^m d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2 - \lambda = 0,$$

and hence

$$z_{i,u} = \left(\frac{\lambda}{n}\right)^{\frac{1}{n-1}} \left(\frac{1}{\sum_{k=1}^{c_u} v_{i,k,u}^m d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2}\right)^{\frac{1}{n-1}}. \quad (9)$$

We can rewrite the above equation

$$\left(\frac{\lambda}{n}\right)^{\frac{1}{n-1}} = z_{i,u} \left(\sum_{k=1}^{c_u} v_{i,k,u}^m d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2\right)^{\frac{1}{n-1}}. \quad (10)$$

From the derivative of F_i w. r. t. λ in (8), it follows

$$\begin{aligned} \frac{\partial F_i}{\partial \lambda} &= 1 - \sum_{u=1}^{|U|} z_{i,u} = 0 \\ \sum_{u=1}^{|U|} z_{i,u} &= 1\,, \end{aligned} \quad (11)$$

which returns the normalization condition as in (3). Using the formula for $z_{i,u}$ in (9) and integrating it into expression (11) we compute

$$\begin{aligned} &\sum_{u=1}^{|U|} \left(\frac{\lambda}{n}\right)^{\frac{1}{n-1}} \left(\frac{1}{\sum_{k=1}^{c_u} v_{i,k,u}^m d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2}\right)^{\frac{1}{n-1}} = 1 \\ &\left(\frac{\lambda}{n}\right)^{\frac{1}{n-1}} \sum_{u=1}^{|U|} \left(\frac{1}{\sum_{k=1}^{c_u} v_{i,k,u}^m d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2}\right)^{\frac{1}{n-1}} = 1. \end{aligned} \quad (12)$$

We make use of (10) and substitute $\left(\frac{\lambda}{n}\right)^{\frac{1}{n-1}}$ in (12). Note, we use $\bar{u}$ as parameter index of the sum to address the fact that it covers all universes, whereas u denotes the current universe of interest. It follows

$$\begin{aligned} 1 &= z_{i,u} \left(\sum_{k=1}^{c_u} v_{i,k,u}^m d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2\right)^{\frac{1}{n-1}} \\ &\quad \times \sum_{\bar{u}=1}^{|U|} \left(\frac{1}{\sum_{k=1}^{c_{\bar{u}}} v_{i,k,\bar{u}}^m d_{\bar{u}}\left(\vec{w}_{k,\bar{u}}, \vec{x}_{i,\bar{u}}\right)^2}\right)^{\frac{1}{n-1}}, \end{aligned}$$

which can be simplified to

$$1 = z_{i,u} \sum_{\bar{u}=1}^{|U|} \left(\frac{\sum_{k=1}^{c_u} v_{i,k,u}^m d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2}{\sum_{k=1}^{c_{\bar{u}}} v_{i,k,\bar{u}}^m d_{\bar{u}}\left(\vec{w}_{k,\bar{u}}, \vec{x}_{i,\bar{u}}\right)^2}\right)^{\frac{1}{n-1}},$$

and returns an immediate update expression for the membership $z_{i,u}$ of pattern i to universe u (see also (5)):

$$z_{i,u} = \frac{1}{\sum_{\bar{u}=1}^{|U|} \left(\frac{\sum_{k=1}^{c_u} v_{i,k,u}^m d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2}{\sum_{k=1}^{c_{\bar{u}}} v_{i,k,\bar{u}}^m d_{\bar{u}}\left(\vec{w}_{k,\bar{u}}, \vec{x}_{i,\bar{u}}\right)^2}\right)^{\frac{1}{n-1}}}.$$

Analogous to the calculations above we can derive the update equation for value $v_{i,k,u}$which represents the partitioning value of pattern i to cluster k in universe u. From (8) it follows

$$\frac{\partial F_i}{\partial v_{i,k,u}} = z_{i,u}^n\, m\, v_{i,k,u}^{m-1} d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2 - \mu_u = 0,$$

and thus

$$v_{i,k,u} = \left(\frac{\mu_u}{m\, z_{i,u}^n\, d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2} \right)^{\frac{1}{m-1}}, \quad (13)$$

$$\left(\frac{\mu_u}{m\, z_{i,u}^n} \right)^{\frac{1}{m-1}} = v_{i,k,u} \left(d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2 \right)^{\frac{1}{m-1}}. \quad (14)$$

Zeroing the derivative of F_i w. r. t. μ_u will result in condition (2), ensuring that the partition values sum to 1, i. e.

$$\frac{\partial F_i}{\partial \mu_u} = 1 - \sum_{k=1}^{c_u} v_{i,k,u} = 0\,. \quad (15)$$

We use (13) and (15) to come up with

$$1 = \sum_{k=1}^{c_u} \left(\frac{\mu_u}{m\, z_{i,u}^n\, d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2} \right)^{\frac{1}{m-1}},$$

$$1 = \left(\frac{\mu_u}{m\, z_{i,u}^n} \right)^{\frac{1}{m-1}} \sum_{k=1}^{c_u} \left(\frac{1}{d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2} \right)^{\frac{1}{m-1}}. \quad (16)$$

Equation (14) allows us to replace the first multiplier in (16). We will use the $\bar{k}$ notation to point out that the sum in (16) considers all partitions in a universe and k to denote one particular cluster coming from (13),

$$1 = v_{i,k,u} \left(d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2 \right)^{\frac{1}{m-1}} \times \sum_{\bar{k}=1}^{c_u} \left(\frac{1}{d_u\left(\vec{w}_{\bar{k},u}, \vec{x}_{i,u}\right)^2} \right)^{\frac{1}{m-1}}$$

$$1 = v_{i,k,u} \sum_{\bar{k}=1}^{c_u} \left(\frac{d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2}{d_u\left(\vec{w}_{\bar{k},u}, \vec{x}_{i,u}\right)^2} \right)^{\frac{1}{m-1}}$$

Finally, the update rule for $v_{i,k,u}$ arises as (see also 4):

$$v_{i,k,u} = \frac{1}{\sum_{\bar{k}=1}^{c_u} \left(\frac{d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2}{d_u\left(\vec{w}_{\bar{k},u}, \vec{x}_{i,u}\right)^2} \right)^{\frac{1}{m-1}}}\,.$$

For the sake of completeness we also derive the update rules for the cluster prototypes $\vec{w}_{k,u}$. We confine ourselves to the Euclidean distance here, assuming the data is normalized[1]:

$$d_u\left(\vec{w}_{k,u}, \vec{x}_{i,u}\right)^2 = \sum_{a=1}^{A_u} \left(w_{k,u,a} - x_{i,u,a}\right)^2\,, \quad (17)$$

[1]The derivation of the updates using other than the Euclidean distance works in a similar manner.

with A_u the number of dimensions in universe u and $w_{k,u,a}$ the value of the prototype in dimension a. $x_{i,u,a}$ is the value of the a-th attribute of pattern i in universe u, respectively. The necessary condition for a minimum of the objective function (1) is of the form $\nabla_{\vec{w}_{k,u}} J = 0$. Using the Euclidean distance as given in (17) we obtain

$$\frac{\partial J_{m,n}}{\partial w_{k,u,a}} = 0 = 2 \sum_{i=1}^{|T|} z_{i,u}^n\, v_{i,k,u}^m\, \left(w_{k,u,a} - x_{i,u,a}\right)$$

$$w_{k,u,a} \sum_{i=1}^{|T|} z_{i,u}^n\, v_{i,k,u}^m = \sum_{i=1}^{|T|} z_{i,u}^n\, v_{i,k,u}^m\, x_{i,u,a}$$

$$w_{k,u,a} = \frac{\sum_{i=1}^{|T|} z_{i,u}^n\, v_{i,k,u}^m\, x_{i,u,a}}{\sum_{i=1}^{|T|} z_{i,u}^n\, v_{i,k,u}^m}\,,$$

which is also given with (6).

References

[1] D. J. Hand, H. Mannila, and P. Smyth, *Principles of Data Mining*. MIT Press, 2001.

[2] F. Höppner, F. Klawoon, R. Kruse, and T. Runkler, *Fuzzy Cluster Analysis*. Chichester, England: John Wiley, 1999.

[3] G. Cruciani, P. Crivori, P.-A. Carrupt, and B. Testa, "Molecular fields in quantitative structure-permeation relationships: the VolSurf approach," *Journal of Molecular Structure*, vol. 503, pp. 17–30, 2000.

[4] A. Schuffenhauer, V. J. Gillet, and P. Willett, "Similarity searching in files of three-dimensional chemical structures: Analysis of the bioster database using two-dimensional fingerprints and molecular field descriptors." *Journal of Chemical Information and Computer Sciences*, vol. 40, no. 2, pp. 295–307, 2000.

[5] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of the eleventh annual Conference on Computational Learning Theory (COLT'98)*. ACM Press, 1998, pp. 92–100.

[6] K. Kailing, H.-P. Kriegel, A. Pryakhin, and M. Schubert, "Clustering multi-represented objects with noise." in *PAKDD*, 2004, pp. 394–403.

[7] D. E. Patterson and M. R. Berthold, "Clustering in parallel universes," in *Proceedings of the 2001 IEEE Conference in Systems, Man and Cybernetics*. IEEE Press, 2001.

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

[9] S. Bickel and T. Scheffer, "Multi-view clustering," in *Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM'04)*, 2004, pp. 19–26.

[10] W. Pedrycz, "Collaborative fuzzy clustering," *Pattern Recognition Letters*, vol. 23, no. 14, pp. 1675–1686, 2002.

[11] J. Wang, H.-J. Zeng, Z. Chen, H. Lu, L. Tao, and W.-Y. Ma, "ReCoM: Reinforcement clustering of multi-type interrelated data objects," in *In Proceedings of the 26th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR'03)*, 2003, pp. 274–281.

[12] R. R. Yager and D. P. Filev, "Approximate clustering via the mountain method," *IEEE Trans. Systems Man Cybernet.*, vol. 24, no. 8, pp. 1279–1284, August 1994.

[13] N. B. Venkateswarlu and P. S. V. S. K. Raju, "Fast ISODATA clustering algorithms," *Pattern Recognition*, vol. 25, no. 3, pp. 335–342, 1992.