

Fuzzy Clustering: More than just Fuzzification

Frank Klawonn

Department of Computer Science
Ostfalia University of Applied Sciences
Salzdahlumer Str. 46/48
D-38302 Wolfenbuettel, Germany

Biostatistics

Helmholtz Centre for Infection Research
Inhoffenstr. 7
D-38124 Braunschweig, Germany

Rudolf Kruse

Department of Computer Science
Otto-von-Guericke-University
Universitaetsplatz 2
D-39106 Magdeburg

Roland Winkler

Leibniz Institute of Astrophysics Potsdam (AIP)
An der Sternwarte 16
14482 Potsdam
Germany

Abstract

The initial idea of extending the classical k-means clustering technique to an algorithm that uses membership degrees instead of crisp assignments of data objects to clusters led to the invention of a large variety of new fuzzy clustering algorithms. However, most of these algorithms are concerned with cluster shapes or outliers and could have been defined without any problems in the context of crisp assignments of data objects to clusters. In this paper, we demonstrate that the use of membership degrees for these algorithms – although it is not necessary from the theoretical point of view – is essential for these algorithms to function in practice. With crisp assignments of data objects to clusters these algorithms would get stuck most of the time in a local minimum of their underlying objective function, leading to undesired clustering results. In other contributions it was shown that the use of membership degrees can avoid this problem of local minima but it also introduces new problems, especially for clus-

ters with varying density and for high-dimensional data, at least if fuzzy clustering is carried out with the simple standard fuzzifier .

1 Introduction

Cluster analysis is an unsupervised data analysis task, trying to group data objects into clusters, such that similar data objects are assigned to the same cluster whereas dissimilar data objects should belong to different clusters. This informal definition of cluster analysis already implies that a notion of similarity is needed to cluster data. It is out of the scope of this paper, to provide an overview on the large variety of existing approaches to clustering data. We confine our discussion to clustering multivariate real-valued data and to extensions of the classical k-means clustering algorithm (see for instance [1]), since until today the large majority of fuzzy clustering approaches is based on the initial generalisation of k-means clustering to fuzzy c-means clustering as it was proposed by Dunn [2] and Bezdek [3, 4].

An obvious question to be asked is why there is or was a need to extend an algorithm like k-means clustering to allow for membership degrees instead of crisp assignments of data objects to clusters. First of all, the idea of “soft” assignments of data objects to clusters in terms of probabilities has been around long before in the form of mixture models, especially the well-known Gaussian mixture models (for an overview see for instance [5]). Gaussian mixture models assume that the clusters originate from multivariate normal distributions, so that the whole data set can be described as a sample from a mixture of normal distributions. Such mixture models estimate the parameters of the underlying multivariate normal distributions as well as the a priori probabilities for each cluster or multivariate normal distribution to generate a data point. On this basis, mixture models compute (posterior) probabilities that a specific data point was generated by a specific multivariate normal distribution, i.e. the probability whether it belongs to the corresponding cluster. In a Gaussian mixture model, data points that can be assigned very well to a specific cluster will have a probability close to one for this cluster and a probability close to zero for the other clusters, whereas ambiguous data points between clusters will have moderate probabilities to two or more clusters. One could argue that the concept of fuzziness is different from that of probability and therefore a membership degree to a cluster is different from the probability to belong to a cluster. But at least the original fuzzy extension of the k-means clustering algorithm puts a probabilistic constraint on the membership degrees. So this cannot be the main reason for the use of fuzzy clustering.

Connecting fuzzy rule-based systems to data is another motivation for fuzzy clustering. Sugeno and Yasukawa [6] used fuzzy clustering to cluster the output or target attribute of a fuzzy system to generate fuzzy rules. Since in this case only one-dimensional data are considered for clustering, this is essentially a fuzzified discretisation technique. But could this not also be done by standard discretisation techniques like equi-width, equi-frequency or V-optimal discreti-

sation (for an overview see for example [7]) and then using the resulting intervals to define corresponding fuzzy sets? In [8, 9] multidimensional fuzzy clusters are used to construct fuzzy rules. But there are specifically tailored methods to derive fuzzy rules from data, for instance neuro-fuzzy methods [10].

A clear argument in favour of fuzzy clustering is the possibility of new cluster validity measures like Bezdek’s partition coefficient or Bezdek’s partition entropy [4] that only make sense in the context of fuzzy clustering. Such validity measures are used to judge whether a clustering result is meaningful and also to determine the number of clusters. In [11], visualisation techniques based on the membership degrees were used to judge the quality of clusters.

But there is one very important reason speaking in favour of fuzzy clustering that is discussed in more detail in this paper. Fuzzy clustering can help to avoid algorithmic problems from which methods like the k-means clustering algorithm suffer. The result of k-means clustering highly depends on the initialisation of the algorithm, leading to undesired clustering results. This is not true for fuzzy clustering. In this sense, fuzzy clustering is not only an improvement of k-means clustering. It also opens the possibility to introduce more flexible and sophisticated clustering models than the simple k-means algorithm, still avoiding the problem of undesired clustering results.

After a brief introduction of the fuzzy extension of the k-means algorithm in Section 2, Section 3 briefly reviews a number of extensions to the fuzzy c-means algorithm that would theoretically but not practically function if the same extensions were applied to hard k-means clustering. Section 4 demonstrates and explains why these algorithms – although essentially not being fuzzy – require the concept of fuzzy clustering. In Section 5 we discuss alternatives to the original fuzzification of the k-means clustering algorithm before we briefly conclude the paper in Section 6.

2 From crisp to fuzzy clustering

Although first ideas on fuzzy clustering were already proposed at the end of the sixties of the last century by Ruspini [12], the breakthrough for fuzzy clustering came with the fuzzification of the k-means algorithm. In the k-means algorithm, each cluster is represented by a so-called prototype which is supposed to be the centre of the corresponding cluster. Given a data set $\{x_1, \dots, x_n\} \subset \mathbb{R}^q$, the algorithm is based on the objective function

$$f = \sum_{i=1}^c \sum_{j=1}^n u_{ij} d_{ij} \tag{1}$$

that should be minimised under the constraints

$$\sum_{i=1}^c u_{ij} = 1 \quad \text{for all } j \in \{1, \dots, n\} \tag{2}$$

where $u_{ij} \in \{0, 1\}$ indicates whether data vector x_j is assigned to cluster i ($u_{ij} = 1$) or not ($u_{ij} = 0$). $d_{ij} = \|x_j - v_i\|^2$ is the squared Euclidean distance between data vector x_j and cluster prototype v_i .

The number of clusters¹ c must be fixed in advance. Usually, the number of clusters is not known. In this case, the clustering can be carried out with different numbers of clusters and based on validity measures (see for instance [13, 14]) or resampling techniques [15], the number of clusters is chosen in such a way that it yields the best value for the validity measure or is most stable for the resampling. Determining the number of clusters is a separate issue that will not be further addressed in this paper.

Unfortunately, there is no closed form solution for the constrained optimisation problem posed by Equations (1) and (2). Alternating optimisation is the common heuristic approach to solve this optimisation problem. Alternating optimisation requires that the parameters to be optimised are split into two sets. In the case of k-means clustering, one set of parameters consists of the cluster prototypes v_i , while the assignments u_{ij} of the data vectors to the clusters form the other set of parameters. If one set of parameters is considered to be fixed, there is a closed form solution for the other set of parameters. When the cluster prototypes are fixed then in order to minimise (1), each data vector should be assigned to the closest cluster prototype. Assuming the assignments to the clusters to be fixed, (1) is minimised when the prototypes are the centres of gravity of the data vectors assigned to the corresponding cluster.

Alternating optimisation exploits these partial solutions of the optimisation problem by initialising usually the cluster centres randomly, for instance by randomly picking data vectors as cluster centres, and then alternatingly optimise the parameter sets u_{ij} and v_i . Although this strategy cannot give a guarantee to find the global minimum of the objective function (1), it does at least reduce the value of the objective function in each step and will therefore also converge because the objective function is non-negative [16]. The problem of convergence in a clustering result that does not correspond to the global minimum is illustrated in Section 4.

A straight forward “fuzzification” of the k-means clustering approach would be to relax the constraints $u_{ij} \in \{0, 1\}$ to $u_{ij} \in [0, 1]$. It is, however, easy to show that for the global minimum of the objective function (1) $u_{ij} \in \{0, 1\}$ will still hold even with the relaxed constraint. Therefore, Dunn [2] and Bezdek [3, 4] proposed to introduce a so-called fuzzifier $m > 1$ into the objective function (1), leading to

$$f = \sum_{i=1}^c \sum_{j=1}^n u_{ij}^m d_{ij}. \quad (3)$$

Note that the fuzzifier m does not have any effects when the constraints $u_{ij} \in \{0, 1\}$ are imposed. Dunn [2] considered only the case $m = 2$, whereas Bezdek [3, 4] analysed the objective function (3) for arbitrary $m > 1$. The fuzzifier

¹For k-means clustering, the number of clusters is usually denoted by the letter k – hence the name k-means clustering – whereas c is mostly used in the context of fuzzy clustering. In this paper, we always denote the number of clusters by the letter c .

m controls how “fuzzy” the clusters tend to be. For $m \rightarrow 1$, the membership degrees u_{ij} tend to the values 0 and 1, i.e. FCM becomes k-means clustering. For $m \rightarrow \infty$, the clustering is not meaningful, since this implies $u_{ij} \rightarrow \frac{1}{c}$ for all $i \in \{1, \dots, c\}$ and all $j \in \{1, \dots, n\}$.

Based on the modified objective function (3) a corresponding alternating optimisation scheme can be derived for the fuzzy version of the k-means algorithm. This algorithm is called fuzzy c-means (FCM). Instead of assigning each data vector to the closest cluster centre, the membership degree of a data vector to a specific cluster is calculated based on the relative distances of the data vector to the other clusters.

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{d_{ij}}{d_{kj}}\right)^{\frac{1}{m-1}}}, \quad (4)$$

The prototypes or cluster centres are now computed as weighted centres depending on the membership degrees.

$$v_i = \frac{\sum_{j=1}^n u_{ij}^m x_j}{\sum_{j=1}^n u_{ij}^m}, \quad (5)$$

As in the k-means algorithms, Equations (4) and (5) are closed form solutions for the objective function (3), given the corresponding other set of parameters is considered to be fixed, so that the FCM algorithm can also be based on an alternating optimisation scheme.

3 Fuzzy clustering algorithms which are not fuzzy

FCM gave rise to a number of new clustering algorithms that are able to handle more complex cluster structures than the ones based on the simple cluster centre of k-means and FCM. These new approaches are usually addressed as fuzzy clustering algorithms although they could be formulated in a corresponding way as extensions of k-means clustering as well. They actually do not touch the computation of the membership degrees but replace the Euclidean distance by another function. We briefly review a small selection of such algorithms in this section.

Because FCM is based on the Euclidean distance between a data vector and a cluster centre, it essentially assumes that clusters are (hyper-)spheres of roughly the same size. The Gustafson-Kessel (GK) algorithm [17] introduces for each cluster a symmetric, positive-definite matrix² – similar to the covariance matrix of a multivariate normal distribution – as an additional set of parameters, so that it can adapt to ellipsoidal clusters. This means that in the alternating optimisation scheme the step for updating the cluster centres must also include the update of these symmetric, positive-definite matrices. The closed form solutions for these matrices are the covariance matrices of the corresponding clusters with

²In order to avoid degenerated solutions, the determinant of each matrix is constrained to a fixed value, usually 1.

a normalisation factor to have a fixed determinant. As for the cluster centres in FCM, these covariance matrices are computed as weighted covariance matrices based on the membership degrees of the data vectors to the corresponding cluster. The update equation for the membership degree remains the same as in FCM but with the distances induced by the corresponding symmetric, positive-definite matrices. One could also define a crisp version of GK as an extension of the k-means algorithm by simply replacing the weighted covariance matrices by the covariance matrices defined by the data vectors assigned to the corresponding cluster. This can also be considered as a weighted sum because those data vectors assigned to the specific cluster obtain the weight 1, whereas other data vectors obtain the weight 0.

The fuzzy *c*-varieties (FCV) algorithm [18, 4] searches for clusters in the form of linear manifolds. Again, the simple cluster centres of FCM must be extended in this case by the vectors spanning the corresponding linear manifold. Instead of covariance matrices as in GK, these vectors need to be calculated in FCV. The update equation is again a weighted formula based on the membership degrees. As for GK, one could introduce a crisp version of FCV in the same way.

Shell clustering algorithms [19] are mainly designed for image analysis purposes to find lines, shells of circles or ellipses or more generally quadratic forms. As for GK and FCV one could directly define crisp versions of shell clustering algorithms.

Noise clustering [20] is an approach that can deal with noise in the data, i.e. with data vectors not fitting properly to any cluster. For this purpose an additional noise cluster is introduced. The noise cluster does not have a prototype or cluster centre but only a fixed large distance to all data vectors. Therefore, data vectors far away from all clusters obtain a high membership degree to the noise cluster. It is again obvious that one could extend the k-means algorithm by adding a noise cluster and there is no specific need for membership degrees.

All these and many other fuzzy clustering algorithms have been introduced in the context of fuzzy clustering and as extensions of FCM although their crisp versions could have been defined long before as extensions of k-means clustering. The next section tries to explain why the fuzzy clustering context was necessary for these algorithms.

4 Avoidance and introduction of local minima

The left hand side of Figure 1 shows a simple two-dimensional data set with four clusters and a result of k-means clustering. The cluster centres are shown as full circles. The assignment of data vectors to different clusters is indicated by different symbols. The clustering result is not what we would expect. It corresponds to a local minimum of the objective function of k-means clustering. The clustering result considers the two data clusters on the left hand side as one cluster and splits the upper right data cluster into two clusters. We have run the k-means algorithm 10,000 times on this data set with random initialisations,

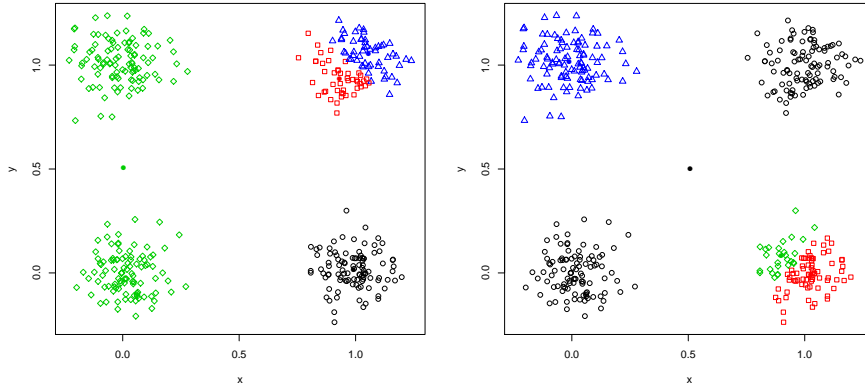


Figure 1: Undesired partitions resulting from crisp clustering algorithms like k -means (left) or the crisp version of the Gustafson-Kessel algorithm (right).

i.e. four randomly chosen data vectors as initial cluster centres. In 6,243 out of these 10,000 runs the k -means algorithm found the correct clusters, but in the remaining runs the clustering result looked similar³ to the one in the left hand side of Figure 1. This means that k -means clustering has roughly a 38% chance of failing on this simple data set. The fact that k -means clustering is very sensitive to the initialisation is not new at all. One can, of course, repeat the k -means algorithm more than once and choose the clustering result with the smallest value for the objective function in order to reduce the chance of finding only a local minimum. Or one could try to apply more sophisticated initialisation techniques as in the k -means++ algorithm [21]. However, one could also simply apply FCM. FCM found the right clusters and cluster centres in all of the 10,000 runs – 0% failure rate for this data set. It should be noted that the problem of local minima for k -means clustering increases with the number of clusters because the chance of having a data clusters with more than one prototype and other data clusters with no prototype at the initialisation increases quickly with the number of clusters. Such initialisations are more prone to lead to a local minimum.

This seems to indicate that the use of membership degrees instead of crisp assignments can make undesired local minima of the objective function vanish. And indeed, this is the case as was demonstrated in [22].

The comparison of the results for k -means clustering and FCM is based on a final defuzzification of the FCM result, i.e. data objects are assigned to the cluster with the highest membership degree in order to make the two algorithms comparable. This also applies to the comparison between the crisp and fuzzy

³Two of the data clusters are covered by one cluster and one data cluster is split into two clusters.

version of GK that we discuss below. Of course, in this way the additional information and benefit of membership degrees gets lost. The fuzzification is only carried out for algorithmic purposes. This idea of relaxation in optimisation, i.e. permitting at least in an intermediate step continuous instead of discrete values – here continuous membership degrees between 0 and 1 instead of only the discrete values 0 and 1 – is well known, for instance in integer programming where typically so-called cutting planes are introduced to derive an integer solution from a continuous solution to a constraint optimisation problem [23]. Approaches to minimise k-means clustering based on such ideas are described in [24].

It should also be noted that for the sake of the comparison between the crisp and the fuzzy version of clustering algorithm, we use only data sets with well-separated clusters where fuzzy membership degrees would or should tend to be close to 0 and 1 anyway. The important and more difficult problem of overlapping clusters is not a topic of this paper.

k-means clustering and FCM are very simple algorithms in terms of their cluster parameters in the form of just cluster centres. The clustering algorithms mentioned in the previous section can handle more complex cluster structures. But the price for this flexibility to adapt to more complex cluster structures is that there are more local minima in the objective function.

When we apply the crisp version of GK to the data set in Figure 1 – the resulting symmetric, positive definite matrices should be roughly the identity matrix since the clusters are spherical – in addition to the local minima for k-means clustering as shown on the left hand side of Figure 1 new local minima are introduced into the objective function. An example for such an additional local minimum is shown on the right hand side of Figure 1. The data clusters in the lower left and the upper right are covered by one elongated cluster (ellipsoidal cluster with no data points close to its centre) and the lower right data cluster is split into two clusters. Because of the longer computation time, we have run the crisp version of GK only 1,000 times with random initialisations of the cluster centres and identity matrices as the initial symmetric positive definite matrices. The crisp version of GK found the right clusters only in 479 cases. So the more complex crisp GK increased the 38% failure rate of the simple k-means algorithm to more than 52% for this very simple data set. The original fuzzy version of GK found the right clusters in 967 out of 1,000 runs, a failure rate of about 3% compared to 52% of the crisp version.

Similar considerations apply to the other algorithms mentioned in the previous section.

So it seems essential to avoid local minima for more complex cluster structures and more complex data sets by making use of membership degrees.

Unfortunately, the use of the fuzzifier can also introduce new undesired local minima or – even worse – undesired global minima. Figure 2 shows a data set and the clustering result for FCM. There are two data clusters, a very dense one in the lower left and a sparser cluster in the upper right. Data vectors drawn as black circles have a membership degree greater than 0.5 to the first cluster, data vectors drawn as red squares have a membership degree greater than 0.5

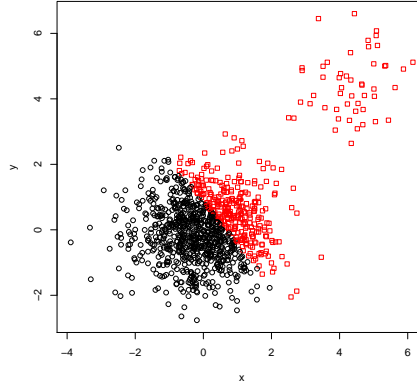


Figure 2: An undesired partition resulting from FCM.

to the other cluster. The clustering result in Figure 2 does not correspond to an undesired local minimum of the objective function. It is the global minimum of the objective function. Even if FCM is initialised with correct cluster centres of the data clusters, it will converge to the result in Figure 2. The computation of the membership degrees (4) and the update equation (5) for the cluster centres explain why this happens. According to Equation (4) a data vector will always have a non-zero membership degree to all clusters, except in the rare case that it coincides with the cluster centre. Even if the cluster centres are initially chosen as the centres of the data clusters, each data vector of the dense lower left cluster will have a small, but positive membership degree to the upper right cluster. Due to Equation (5) each of these data vectors will pull the cluster centre of the upper right cluster slightly the lower left. The problem here is that because of the high density of the lower left data cluster, there are so many data vectors pulling slightly at the other cluster centre that the few data vectors in the sparse upper right data cluster cannot keep the cluster prototype where it should be.

High-dimensional data impose another problem for algorithms like FCM. It can be shown for such a data that placing all cluster centres close to centre of the whole data set is a very large and pronounced undesired local minimum of the objective function [25].

5 Beyond the fuzzifier

There are, of course, other concepts than the original simple fuzzifier that can avoid or at least reduce the problems mentioned at the end of the previous section. A very good overview on different approaches to fuzzy clustering can be found in [26].

Possibilistic clustering [27] was one of the first concepts after the introduction of the fuzzifier. In possibilistic clustering the probabilistic constraint that the membership degrees of a data vector to all clusters must sum up to 1 is dropped and replaced by a penalty term for low membership degrees. This, however, leads to an objective function whose global minimum is achieved when all cluster centres cover the same best data cluster. To avoid this, the algorithms is run in such a way that it converges in a local minimum. Therefore, possibilistic clustering does not fall properly into the category of objective function-based algorithms. There is an improved objective function-based version of possibilistic clustering [28]. However, this approach does not admit a closed form solution for the computation of the membership degrees, so that it requires an iteration procedure within the alternating optimisation scheme and leads to long computation times.

In [29] a generalisation of the fuzzifier is proposed where the simple power function $g(u_{ij}) = u_{ij}^m$ can be replaced by a differentiable, increasing convex function $g : [0, 1] \rightarrow [0, 1]$ with $g(0) = 0$ and $g(1) = 1$. However, in order to guarantee a closed form solution for the update equation for the membership degrees, g must be chosen very carefully. A very convenient example for g is the so-called polynomial fuzzifier which can also be interpreted as a convex combination of the objective functions for FCM and k-means clustering. In [25] it was demonstrated that the polynomial fuzzifier has less problems with high-dimensional data than FCM.

A completely different approach does not use a fuzzifier function at all but uses instead a regularisation function to rule out the crisp solution for the membership degrees [30, 31]. In contrast to FCM, these regularisation approaches as well as the polynomial fuzzifier admit zero membership degrees and can therefore avoid or at least reduce the problems of FCM mentioned at the end of the previous section. They are slightly more complex than FCM in the computation scheme for the membership degrees but still provide a closed form solution.

6 Conclusions

We have demonstrated in this paper that one main advantage of fuzzy clustering is the avoidance of local minima that correspond to undesired clustering results. This avoidance of local minima makes it possible to define algorithms for more complex clustering structures that would be possible in principle in the context of crisp clustering but would very often fail in practice because of the large number of local minima that a complex clustering scheme introduces into the objective function. Therefore, even if one is interested in a crisp clustering partition, these more sophisticated algorithm should always be first run with membership degrees and a defuzzification should only be carried out at the very end.

Although FCM can help to avoid local minima, it also leads to new problems as demonstrated at the end of Section 4. For low-dimensional data with clusters of roughly the same density, the standard fuzzifier for fuzzy clustering is

often sufficient. But for higher-dimensional data and very heterogeneous clusters there are better alternatives than the simple fuzzifier as mentioned in the previous section. However, it seems that their use is still not common. Reasons for this might be the limited availability of software tools for them and that they are not well known enough.

References

- [1] R. Duda, P. Hart, *Pattern Classification and Scene Analysis*, Wiley, New York, 1973.
- [2] J. Dunn, A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters, *Cybernetics and Systems* 3 (3) (1973) 32–57.
- [3] J. C. Bezdek, *Fuzzy mathematics in pattern classification*, dissertation, Cornell University (1973).
- [4] J. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.
- [5] G. McLachlan, D. Peel, *Finite Mixture Models*, Wiley, New York, 2000.
- [6] M. Sugeno, T. Yasukawa, A fuzzy-logic-based approach to qualitative modelling, *IEEE Transaction on Fuzzy Systems* 1 (1993) 7–31.
- [7] M. Berthold, C. Borgelt, F. Höppner, F. Klawonn, *Guide to Intelligent Data Analysis: How to Intelligently Make Sense of Real Data*, Springer, London, 2010.
- [8] F. Klawonn, R. Kruse, Derivation of fuzzy classification rules from multi-dimensional data, in: *Advances in Intelligent Data Analysis*, The International Institute for Advanced Studies in Systems Research and Cybernetics, Windsor, Ontario, 1995, pp. 90–94.
- [9] M. Setnes, Supervised fuzzy clustering for rule extraction, *IEEE Transaction on Fuzzy Systems* 8 (2000) 416–424.
- [10] D. Nauck, F. Klawonn, R. Kruse, *Neuro-Fuzzy Systems*, Wiley, Chichester, 1997.
- [11] F. Klawonn, V. Chekhtman, E. Janz, Visual inspection of fuzzy clustering results, in: J. Benitez, O. Cordón, F. Hoffmann, R. Roy (Eds.), *Advances in Soft Computing: Engineering Design and Manufacturing*, Springer, London, 2003, pp. 65–76.
- [12] E. Ruspini, A new approach to clustering, *Information and Control* 15 (1969) 22–32.

- [13] J. Bezdek, J. Keller, R. Krishnapuram, N. Pal, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*, Kluwer, Boston, 1999.
- [14] F. Höppner, F. Klawonn, R. Kruse, T. Runkler, *Fuzzy Cluster Analysis*, Wiley, Chichester, 1999.
- [15] C. Borgelt, Resampling for fuzzy clustering, *Int. Journal of Uncertainty, Fuzziness and Knowledge-based Systems* 15 (2007) 595–614.
- [16] J. C. Bezdek, R. J. Hathaway, Convergence of alternating optimization, *Neural, Parallel Sci. Comput.* 11 (4) (2003) 351–368.
- [17] E. E. Gustafson, W. C. Kessel, Fuzzy clustering with a fuzzy covariance matrix, in: *Proc. 18th IEEE Conference on Decision and Control (IEEE CDC, San Diego)*, IEEE Press, Turku, 1979, pp. 761–766.
- [18] H. Bock, Clusteranalyse mit unscharfen Partitionen, in: H. Bock (Ed.), *Klassifikation und Erkenntnis: Vol. III: Numerische Klassifikation*, IN-DEKS, Frankfurt, 1979, pp. 137–163.
- [19] R. Krishnapuram, H. Frigui, O. Nasraoui, Possibilistic shell clustering algorithms and their application to boundary detection and surface approximation – part 1 & 2, *IEEE Transactions on Fuzzy Systems* 1 (1995) 29–60.
- [20] R. N. Davé, Characterization and detection of noise in clustering, *Pattern Recognition Letters* 12 (1991) 406–414.
- [21] D. Arthur, S. Vassilvitskii, k-means++: The advantages of careful seeding, in: *Proceedings of the eighteenth annual ACM-SIAM Symposium on Discrete Algorithms*, Society for Industrial and Applied Mathematics, Philadelphia, 2007, pp. 1027–1035.
- [22] B. Jayaram, F. Klawonn, Can fuzzy clustering avoid local minima and undesired partitions?, in: C. Moewes, A. Nürnberger (Eds.), *Computational Intelligence in Intelligent Data Analysis*, Springer, Berlin, 2012, pp. 31–44.
- [23] H. Marchand, A. Martin, R. Weismantel, L. Wolsey, Cutting planes in integer and mixed integer programming, *Discrete Applied Mathematics* 123 (2002) 397–446.
- [24] J. Peng, Y. Wei, Approximating k-means-type clustering via semidefinite programming, *SIAM Journal on Optimization* 18 (2007) 186–205.
- [25] R. Winkler, F. Klawonn, R. Kruse, Fuzzy C-Means in high dimensional spaces, *Fuzzy System Applications* 1 (2011) 1–17.
- [26] C. Borgelt, Objective functions for fuzzy clustering, in: C. Moewes, A. Nürnberger (Eds.), *Computational Intelligence in Intelligent Data Analysis*, Springer, Berlin, 2012, pp. 3–16.

- [27] R. Krishnapuram, J. Keller, A possibilistic approach to clustering, *IEEE Transactions on Fuzzy Systems* 1 (1993) 98–110.
- [28] H. Timm, R. Kruse, A modification to improve possibilistic fuzzy cluster analysis, in: *Proc. 2002 IEEE Intern. Conf. on Fuzzy Systems, (FUZZ-IEEE'02)*, IEEE, Honolulu, 2002, pp. 1460–1465.
- [29] F. Klawonn, F. Höppner, What is fuzzy about fuzzy clustering? understanding and improving the concept of the fuzzifier, in: M. R. Berthold, H.-J. Lenz, E. Bradley, R. Kruse, C. Borgelt (Eds.), *Advances in Intelligent Data Analysis, Vol. V*, Springer, Berlin, 2003, pp. 254–264.
- [30] S. Miyamoto, K. Umayahara, Fuzzy clustering by quadratic regularization, in: *Proc. IEEE Int. Conf. on Fuzzy Systems/IEEE World Congress on Computational Intelligence Vol. 2*, IEEE, Piscataway, NJ, 1998, pp. 1394–1399.
- [31] K. Honda, H. Ichihashi, Regularized linear fuzzy clustering and probabilistic PCA mixture models, *IEEE Transactions on Fuzzy Systems* 13 (2005) 508–516.