Can Fuzzy Clustering Avoid Local Minima and Undesired Partitions?

Balasubramaniam Jayaram and Frank Klawonn

Abstract. Empirical evaluations and experience seem to provide evidence that fuzzy clustering is less sensitive w.r.t. to the initialisation than crisp clustering, i.e. fuzzy clustering often tends to converge to the same clustering result independent of the initialisation whereas the result for crisp clustering is highly dependent on the initialisation. This leads to the conjecture that the objective function used for fuzzy clustering has less undesired local minima than the one for hard clustering. In this paper, we demonstrate that fuzzy clustering does suffer from unwanted local minima based on concrete examples and show how these undesired local minima of the objective function in fuzzy clustering can vanish by using a suitable value for the fuzzifier.

1 Introduction

The aim of cluster analysis is to construct a partition of a given data set into homogenous groups, called clusters. Data objects within a cluster should be similar, whereas data objects assigned to different clusters should differ significantly. The main motivation for the introduction of fuzzy clustering as a generalisation of crisp

Balasubramaniam Jayaram
Department of Mathematics, Indian Institute of Technology Hyderabad,
Yeddumailaram 502205, India
e-mail: jbala@iith.ac.in

Frank Klawonn
Department of Computer Science, Ostfalia University of Applied Sciences,
38302 Wolfenbuettel, Germany
e-mail: f.klawonn@ostfalia.de
and
Bioinformatics and Statistics, Helmholtz Centre for Infection Research,
38124 Braunschweig, Germany
e-mail: frank.klawonn@helmholtz-hzi.de

C. Moewes et al. (Eds.): Computational Intelligence in Intelligent Data Analysis, SCI 445, pp. 31–44.
springerlink.com

or partitioning clustering was to better represent partly overlapping clusters. Data points at the boundary between two clusters should belong partly to both clusters.

Apart from this obvious motivation for fuzzy clustering, it seems that fuzzy clustering is more robust in the sense that the results seem to be less dependent on the initialisation that is required for many clustering algorithms. Since fuzzy clustering is usually based on minimising an objective function by a gradient descent method, this empirical observation suggests the conclusion that the fuzzy versions of crisp clustering algorithms have less local minima in which the clustering algorithm can get stuck.

First investigations in this direction have been described in [14], but without final proofs that local minima of the objective function can really vanish in fuzzy clustering. After a brief review of fuzzy cluster analysis, we provide concrete examples where it can be clearly observed that undesired local minima of the objective function can be ruled out by fuzzy clustering. Although this is a positive result, new problems are introduced by fuzzy clustering when applied to high-dimensional data.

2 From Crisp to Fuzzy Clustering

A simple and common popular approach is the so-called c-means clustering (HCM)[1] [8]. For the HCM algorithm it is assumed that the number of clusters is known or at least fixed, i.e., the algorithm will partition a given data set $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^m$ into c clusters. Since the assumption of a known or a priori fixed number of clusters is not realistic for many data analysis problems, there are techniques based on cluster validity considerations that allow to determine the number of clusters for the HCM algorithm as well. A comparison of methods for determining the number of clusters can be found in [6]. In recent years, resampling or cross-validation techniques [5] are often used to determine the number of clusters. However, the underlying algorithm remains more or less the same, only the number of clusters is varied and the resulting clusters or the overall partition is evaluated. Therefore, it is sufficient to assume for the rest of the paper that the number of clusters is always fixed.

From the purely algorithmic point of view, the c-means clustering can be described as follows. Each of the c clusters is represented by a prototype $v_i \in \mathbb{R}^m$. These prototypes are chosen randomly in the beginning. Then each data vector is assigned to the nearest prototype (w.r.t. the Euclidean distance). Then each prototype is replaced by the centre of gravity of those data assigned to it. The alternating assignment of data to the nearest prototype and the update of the prototypes as cluster centres is repeated until the algorithm converges, i.e., no more changes happen.

This algorithm can also be seen as a strategy for minimising the following objective function:

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

[1] Usually, the algorithm is called k-means. But in fuzzy clustering it is common to use the letter c instead of k for the number of clusters. HCM stand for *H*ard *C*-*M*eans clustering in order to distinguish it from *F*uzzy *C*-*M*eans clustering (FCM).

under the constraints

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

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.

It would be a straight forward generalisation of HCM to simply relax the constraints $u_{ij} \in \{0,1\}$ to $u_{ij} \in [0,1]$ in order to obtain a fuzzy version of HCM. However, it turned out that the minimum of the objective function (1) under the constraints (2) is still obtained, when u_{ij} is chosen in the same way as in HCM, i.e. $u_{ij} \in \{0,1\}$, even if we allow $u_{ij} \in [0,1]$. Therefore, an additional parameter w, the so-called fuzzifier, was introduced – first only for the choice $w = 2$ [9] and later on for any $w > 1$ [2] – and the objective function (1) is replaced by

$$f = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^{w} d_{ij}. \tag{3}$$

Note that the fuzzifier w does not have any effects, when we use hard clustering. The fuzzifier $w > 1$ is not subject of the optimisation process and has to be chosen in advance. A typical choice is $w = 2$.

The minimisation of the objective function (3) under the constraints (2) is usually carried out by an alternating optimisation scheme where the membership degrees are updated by

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

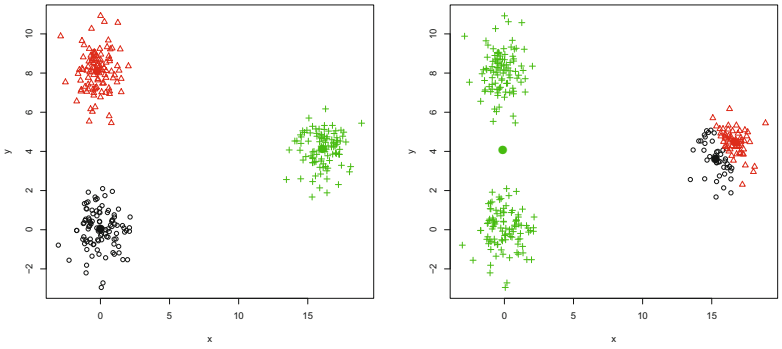
and – in case of the Euclidean distance – the cluster prototypes by

$$v_i = \frac{\sum_{j=1}^{n} u_{ij}^{w} x_j}{\sum_{j=1}^{n} u_{ij}^{w}}. \tag{5}$$

This is the standard fuzzy c-means algorithm (FCM). The update equations (4) and (5) represent the global minimum of the objective function when the corresponding other set of parameters is considered as fixed.

Fig. 1 shows a simple data set with three well-separated clusters. However, in 2,589 out of 10,000 runs with random initialisation, HCM gets stuck in a local minimum of the objective function leading to the undesired clustering result shown in Fig. 1(b) whereas FCM terminates in the correct partition (a) in all 10,000 runs[2]. The reason for the failure of HCM lies in the fact that once a prototype has

[2] The clustering was carried out with the package `cluster` of the statistics software R [19].



(a) HCM works as expected (b) HCM fails!

Fig. 1 A simple two-dimensional data set and an HCM clustering result as it is expected (a). But in about 25% of the runs, HCM gets stuck in a local mininum leading to the partition (b).

'conquered' the two clusters in left-hand side, the other two prototypes will not take any notice of these points anymore.

It is out of the scope of this paper to provide a detailed review on fuzzy clustering as for instance in [3, 12]. It should be noted that there are two parts of the objective function (3) that can be modified or generalised. One the one hand, there is the way how fuzzy membership degrees are incorporated in the objective function. Possibilistic clustering [17] relaxes the constraints (2), leading to an actually undesired global minimum. An improved version of possibilistic clustering, avoiding the problem, has been proposed in [20] for the price of significantly higher computational costs. In [16, 15], the fuzzifier is replaced by more general functions than just a simple power of the membership degrees to overcome certain problems that are introduced by the fuzzifier. One of these problems is discussed at the end of Section 3.

On the other hand, the distance measure can be modified to cover more general cluster shapes. Various approaches have been proposed, for instance to adapt to linear [4, 2] or ellipsoidal [10] clusters, to clusters of different volume [13] or to non-compact shell clusters [18]. Although all these approaches have been published as fuzzy clustering techniques, they have actually nothing specific to do with fuzzy clustering. In principle, one could also use crisp membership degrees for them. The reason why these approaches are exclusively based on fuzzy clustering is probably that the more complex cluster shapes with additional parameters introduce more local minima into the objective function, so that there is a much higher risk to get stuck in an undesired local minimum when hard clustering is applied.

Noise clustering [7] is another example of an approach that is also applicable in the context of hard clustering. An additional noise cluster is introduced to which all data have a fixed (large) distance. In this way, data points that are far away from all clusters will be assigned to the noise cluster and have no longer any influence on other clusters.

3 Vanishing of Local Minima

As shown above HCM can get stuck in local minima if the initialisation is not 'proper'. While FCM certainly overcomes many of the lacunae in HCM, a similar problem can also plague FCM. For instance, is it true that FCM does not have any local minima? If it does, what is it that makes FCM come out of this? In this section, we firstly demonstrate that FCM does have undesired local minima and then argue that a proper fuzzifier can reduce the number of local minima in the objective function of FCM and thus help in the proper and faster convergence of FCM.

3.1 Local Minima of FCM

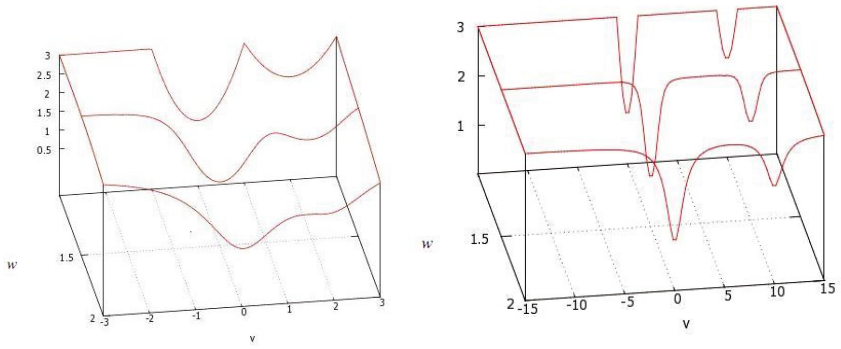
The objective function (3) of FCM is often difficult to visualise – there are too many dimensions (parameters, i.e., prototypes and membership degrees). Hence, let us reduce the dimensions by making the objective function independent of the membership degrees by choosing the optimal values for the membership degrees as in [11] by replacing u_{ij} in (3) by (4).

Taking a similar approach as in [14], let us consider a one-dimensional data set with one cluster at $x = 0$ with k points and one outlier at $x = u$. Clearly, we have just one cluster and let us add a noise cluster [7] to take care of the outlier. Now, the objective function in (3) becomes

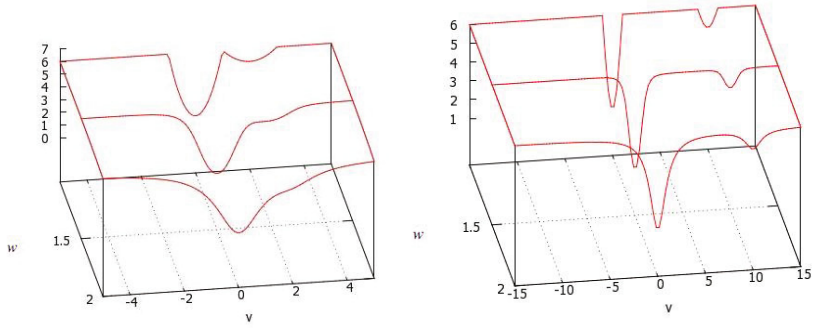
$$f(v) = \frac{k \cdot v^2}{\left(1 + \left(\frac{v^2}{\delta}\right)^{\frac{1}{w-1}}\right)^w} + \frac{(v-u)^2}{\left(1 + \left(\frac{(v-u)^2}{\delta}\right)^{\frac{1}{w-1}}\right)^w} + \frac{k \cdot \delta}{\left(1 + \left(\frac{\delta}{v^2}\right)^{\frac{1}{w-1}}\right)^w} + \frac{\delta}{\left(1 + \left(\frac{\delta}{(v-u)^2}\right)^{\frac{1}{w-1}}\right)^w} \tag{6}$$

where v is the location of the cluster centre and δ is the distance of every point to the noise cluster.

Let us consider Fig. 2, where the location of the cluster centre v is represented on the x-axis and the fuzzifier w on the y-axis. From Fig. 2(a), where $u = 2, k = 2$, i.e., the lone data point is at $x = 2$ and there are $k = 2$ points at $x = 0$, we see that when $w = 1$ there is a clear local minima at $v = 2$ while for the conventionally used value of $w = 2$ we see that the local minima is almost non-existent. Here the noise distance is $\delta = 1$. However, it does not mean that FCM is not plagued by this problem. To see this let us shift the lone data point from $v = 2$ to $v = 10$. As Fig. 2(b) shows, still with $\delta = 1$, we see a clear local minima at $v = 10$. Note that increasing the number of points at $x = 0$ does have an effect in the first case, as is expected, it does not have any effect in the second case, since the lone point is far enough not to be influenced by it. Moreover, note that the local density of data is not in our control in realistic



(a) $u = 2, k = 2$ (b) $u = 10, k = 2$



(c) $u = 2, k = 5$ (d) $u = 10, k = 5$

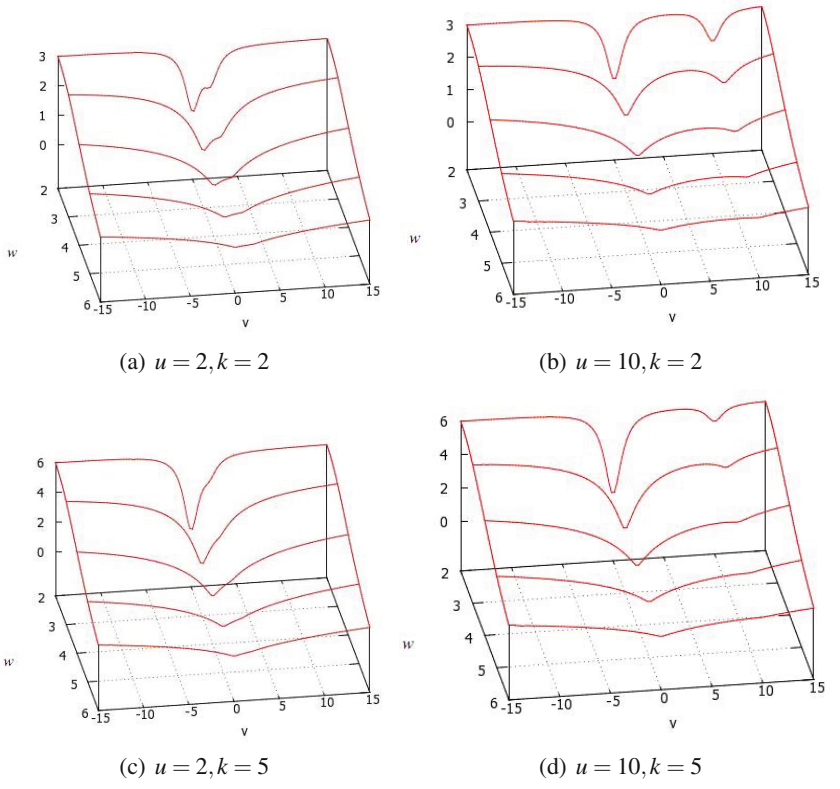
Fig. 2 Plots of the objective function of FCM for $w = 1, 1.5, 2$

situations and hence it is hard to ensure the vanishing of unwanted local minima. In fact, the distribution of the local density of data gives rise to an entirely different problem as is analysed and solved in [16, 14] (see below for more details).

3.2 The Role of the Fuzzifier w

While the generalisation of the membership values u_{ij} from just $\{0, 1\}$ to the whole interval $[0, 1]$ is usually the highlighted aspect of FCM – perhaps even the nomenclature of FCM is also attributable to it – a major role is also played by the so called fuzzifier value 'w' in (3) above.

Firstly, note that even if $u_{ij} \in [0, 1]$ when $w = 1$ we still have hard clustering and FCM is equal to HCM. Secondly, as shown in [16] the value of the fuzzifier w actually controls the amount of overlap among the clusters. Looking at the term u_{ij}^w in (3) as only a particular transformation of u_{ij}, viz., $g(u) = u^w$, it was shown in [14] that suitable transformations g exist that also redeem FCM from the problem of letting their cluster centres be dictated by the local density of the data.



(a) $u = 2, k = 2$ (b) $u = 10, k = 2$

(c) $u = 2, k = 5$ (d) $u = 10, k = 5$

Fig. 3 Plots of the Objective Function of the FCM for $w = 2, 3, 4, 5, 6$

In this work, we show yet another aspect of the fuzzifier, viz., we show that choosing the w value appropriately can make many, if not all, of the local minima vanish and thus help FCM deliver correct results more often than not.

3.3 Suitable w for Vanishing Local Minima

Let us once again consider the scenarios presented in Sec. 3.1. As the Figs. 3(a)–(d) show, by increasing the value of w we see that the unwanted local minima at $v = 2$ and $v = 10$ vanish leading to a correct and, clearly, also a faster convergence. Note also that the local density of the data does not seem to alter the slope of the curve, equivalently the rate of convergence significantly. Thus it is very much applicable to real life data. While it can be seen from Fig. 3 that a value of around $w = 5$ or $w = 6$ seems sufficient to eliminate the unwanted local minima at $v = 2, 10$, it should be emphasised that the scenario considered here is very elementary. In higher dimensions, the value of w required could be much smaller or higher. For instance, see the scenario considerd in Section 3.4 below. As we understand the happenings while $w \to 1$, it is interesting to study the limiting case of $w \to \infty$.

To this end, it is sufficient to consider the following expression that occurs in the denominator of the objective function in (6): $\left(1+\left(\frac{d'}{d''}\right)^{\frac{1}{w-1}}\right)^{w}$, where d', d'' are positive distances of the points from the cluster centres. Since, when either $d'=0$ or $d''=0$ the expression does not come into play, we consider the following equivalent form: $g(w)=\left(1+(\varepsilon)^{\frac{1}{w-1}}\right)^{w}$ for $\varepsilon>0$. Now, it is obvious that $\lim_{w\to\infty} g(w)=\infty$ and hence $\lim_{w\to\infty} f(v)=0$ for every position v of the cluster centre. In other words, every point on the x-axis could be a cluster centre.

3.4 A Slightly More Complex Scenario

Let us consider the following scenario, where we have 2 clusters at $u=2$ and at $w=5$. There are 10 points each at $u=2$ and $w=5$ and 3 'noise' points at $x=0$. Using a noise cluster distance $\delta=1$, we expect the global minima to be at $\overline{v_1}=(2,5)$ and symmetrically at $\overline{v_2}=(5,2)$. Now the objective function becomes a two-variable function $F(\overline{v})=F(r,s)$, the formula of which is quite complex to be given here. However, we do plot F in Figs. 4 and 5 for different values of w. In every case there are clear global minima at $\overline{v_1}, \overline{v_2}$ as expected.

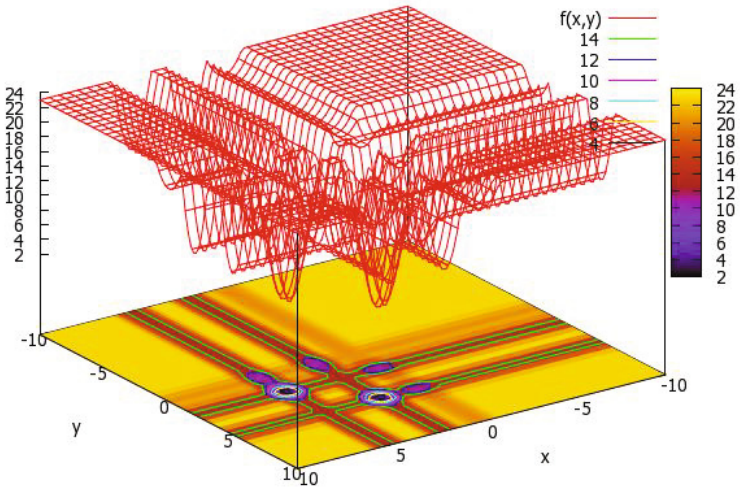
When $w=1$ (HCM), we see from Fig. 4(a) that, apart from the desired minima at $\overline{v_1}, \overline{v_2}$, there are also clear local minima around $(0,2),(2,0),(0,5),(5,0)$ as indicated by the dark blue contour circles. When $w=2$, as is usual for FCM, we see that two of the local minima have vanished and only the local minima around $\overline{v'}=(0,2)$ and $\overline{v''}=(2,0)$ remain. Thus if a cluster centre gets initialised closer to these local minima, it is difficult for these cluster centres to escape from there.

Now let us consider the case when $w=2.3$. It is already clear to see (see Fig. 5(a)) that most of the local minima have vanished and even if cluster centres fall close to the above $\overline{v'}, \overline{v''}$, they can eventually reach one of the global minima. This becomes even more apparent for the case when $w=2.9$ – see Fig. 5(b).

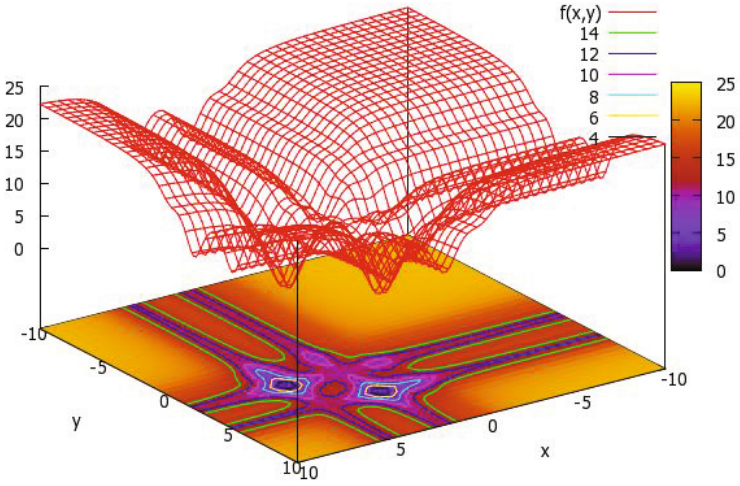
It is also interesting to note that for larger values of w, in fact, for $w=4$ or $w=5$ we see that all the local minima vanish and just one global minimum appears around the centroid of the whole data set, as is expected – see Figs. 6(a) & (b).

3.5 FCM Problems with High-Dimensional Data

The above examples seem to suggest that the fuzzifier will always lead to less local minima. However, for high-dimensional data, fuzzy clustering suffers from the so-called curse of dimensionality [1]. In higher dimensions, standard distances like the Euclidean distance seem to lose their power to distinguish between points. Fig. 7 is adopted from [21] where the following example is considered. Clusters are uniformly distributed on the surface of a hypersphere. Then the objective function of FCM is drawn along one axis only by moving the prototypes from the centre of the sphere along the radii to the cluster centres. Surprisingly, there is a local minimum of the objective function when all prototypes are positioned in the centre of

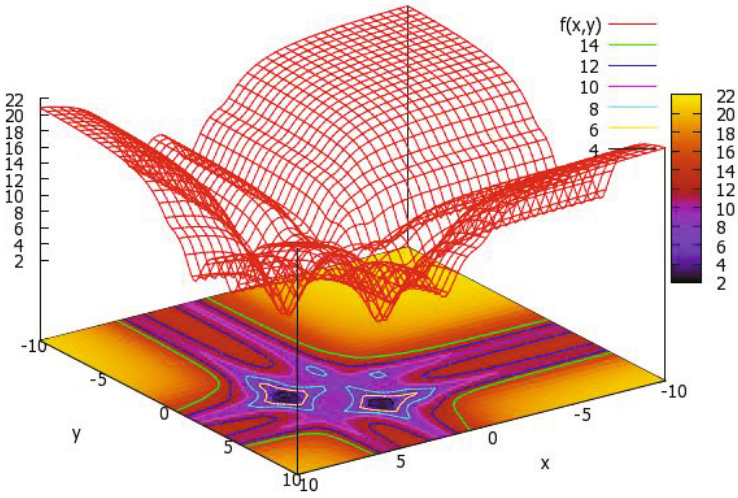


(a) $w = 1$, Local minima around $(0,2),(2,0),(0,5),(5,0)$

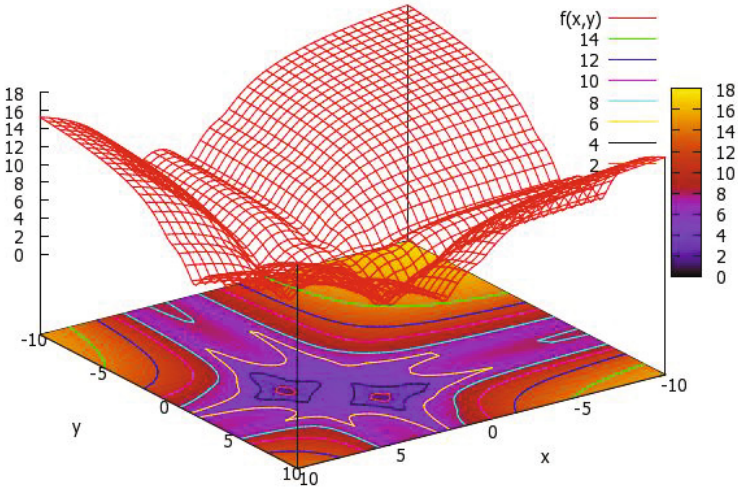


(b) $w = 2$, Local minima around $(0,2),(2,0)$

Fig. 4 Plots of the objective function $F(\overline{v})$ of FCM for $w = 1,2$

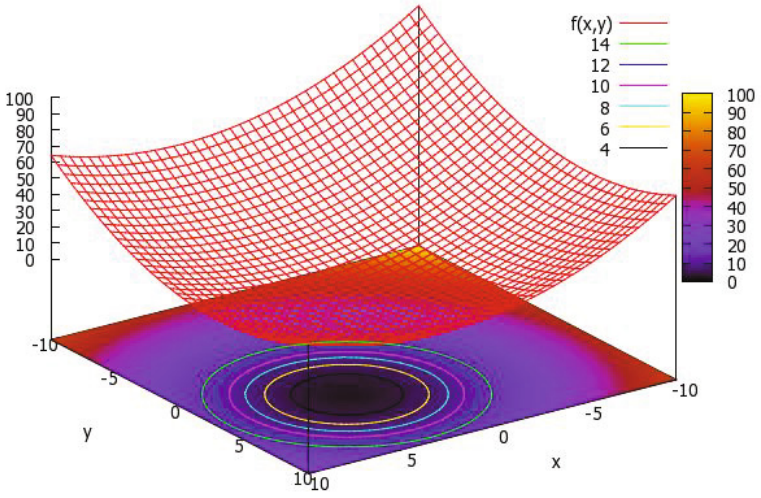


(a) $w = 2.3$

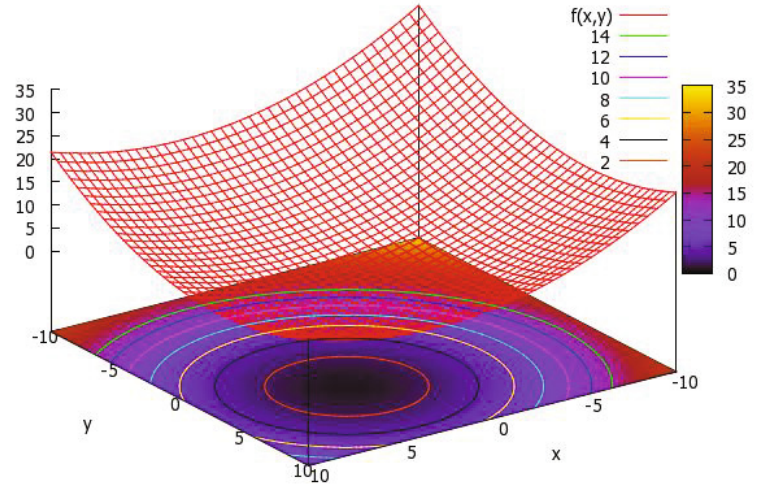


(b) $w = 2.9$, All the local minima have vanished

Fig. 5 Plots of the objective function $F(\overline{v})$ of FCM for $w = 2.3, 2.9$



(a) $w = 4$



(b) $w = 5$

Fig. 6 Plots of the objective function $F(\overline{v})$ of FCM for $w = 4, 5$ - one global minimum appears around the centroid of the whole data set

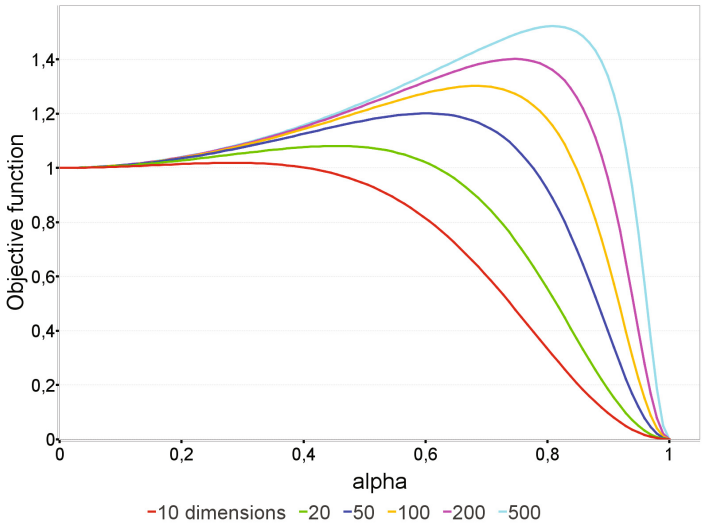


Fig. 7 An undesired local minimum at the centre 0 for high-dimensional data (the bottom curve is for dimension $m = 10$ and the top curve is for $m = 500$)

the sphere and the danger of getting stuck in this local minimum increases with number of dimensions. HCM does not have this specific problem although it has its own problems with high-dimensional data, see for instance [21] and the references therein.

One can escape from this problem of FCM with high-dimensional data by either choosing a fuzzifier very close to 1 or by using a generalised fuzzifier function as proposed in [16].

4 Conclusions

The answer to the initial question whether fuzzy clustering can avoid local minima is partly positive. For low-dimensional data, local minima can vanish by a suitable choice of the fuzzifier. For high-dimensional data, additional local minima can be introduced. The choice of the fuzzifier can be crucial for the avoidance of local minima. How to choose an appropriate value for the fuzzifier will be investigated in a future work.

Acknowledgements. This work was done during the visit of the first author to the Department of Computer Science, Ostfalia University of Applied Sciences under the fellowship provided by the Alexander von Humboldt Foundation. Part of this work was supported by DIeTY grant on Cyber Physical Systems–13(6)/2010-CC&BT.

References

[1] Aggarwal, C.C., Hinneburg, A., Keim, D.A.: On the Surprising Behavior of Distance Metrics in High Dimensional Space. In: Van den Bussche, J., Vianu, V. (eds.) ICDT 2001. LNCS, vol. 1973, pp. 420–434. Springer, Heidelberg (2000)

[2] Bezdek, J.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York (1981)

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

[4] Bock, H.: Clusteranalyse mit unscharfen partitionen. In: Bock, H. (ed.) Klassifikation und Erkenntnis. Numerische Klassifikation, vol. III, pp. 137–163. INDEKS, Frankfurt (1979)

[5] Borgelt, C.: Resampling for fuzzy clustering. Int. Journal of Uncertainty, Fuzziness and Knowledge-based Systems 15(5), 595–614 (2007)

[6] Chiang, M.M.-T., Mirkin, B.: Experiments for the Number of Clusters in K-Means. In: Neves, J., Santos, M.F., Machado, J.M. (eds.) EPIA 2007. LNCS (LNAI), vol. 4874, pp. 395–405. Springer, Heidelberg (2007)

[7] Davé, R.N.: Characterization and detection of noise in clustering. Pattern Recognition Letters 12, 406–414 (1991)

[8] Duda, R., Hart, P.: Pattern Classification and Scene Analysis. Wiley, New York (1973)

[9] Dunn, J.: A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. Cybernetics and Systems 3(3), 32–57 (1973)

[10] Gustafson, E.E., Kessel, W.C.: Fuzzy clustering with a fuzzy covariance matrix. In: Proc. 18th IEEE Conference on Decision and Control IEEE CDC, San Diego, pp. 761–766. IEEE Press, Turku (1979)

[11] Höppner, F., Klawonn, F.: A contribution to convergence theory of fuzzy c-means and its derivatives. IEEE Transactions on Fuzzy Systems 11, 682–694 (2003)

[12] Höppner, F., Klawonn, F., Kruse, R., Runkler, T.: Fuzzy Cluster Analysis. Wiley, Chichester (1999)

[13] Keller, A., Klawonn, F.: Adaptation of cluster sizes in objective function based fuzzy clustering. In: Leondes, C. (ed.) Intelligent Systems: Technology and Applications. Database and Learning Systems, vol. IV, pp. 181–199. CRC Press, Boca Raton (2003)

[14] Klawonn, F.: Fuzzy clustering: Insights and a new approach. Mathware and Soft Computing 11, 125–142 (2004)

[15] Klawonn, F., Höppner, F.: An alternative approach to the fuzzifier in fuzzy clustering to obtain better clustering results. In: Proc. 3rd Eusflat Conference, pp. 730–734 (2003)

[16] Klawonn, F., Höppner, F.: What Is Fuzzy about Fuzzy Clustering? Understanding and Improving the Concept of the Fuzzifier. In: Berthold, M., Lenz, H.-J., Bradley, E., Kruse, R., Borgelt, C. (eds.) IDA 2003. LNCS, vol. 2810, pp. 254–264. Springer, Heidelberg (2003)

[17] Krishnapuram, R., Keller, J.: A possibilistic approach to clustering. IEEE Transactions on Fuzzy Systems 1, 98–110 (1993)

[18] Krishnapuram, R., Frigui, H., Nasraoui, O.: Possibilistic shell clustering algorithms and their application to boundary detection and surface approximation – part 1 & 2. IEEE Transactions on Fuzzy Systems 1, 29–60 (1995)

[19] R Development Core Team (2009) R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2009), http://www.R-project.org
[20] Timm, H., Kruse, R.: A modification to improve possibilistic fuzzy cluster analysis. In: Proc. 2002 IEEE Intern. Conf. on Fuzzy Systems (FUZZ-IEEE 2002), pp. 1460–1465. IEEE, Honolulu (2002)
[21] Winkler, R., Klawonn, F., Kruse, R.: Fuzzy C-Means in high dimensional spaces. Fuzzy System Applications 1, 1–17 (2011)