1

Fuzzy Cluster Analysis from the Viewpoint of Robust Statistics

Frank Klawonn and Frank Höppner

1.1 Introduction

Fuzzy cluster analysis has been initiated in the beginning of the seventies by Bezdek[1] and Dunn[2]. The ideas were partly motivated by the problems caused by the binary or crisp assignment of data to unique clusters as for instance in the case of the popular c-means clustering algorithm. Handling ambiguous and noisy data in order to overcome these problems was one important issue.

Although such concepts of robustness were part of the motivation for introducing fuzzy clustering, serious attempts to a rigorous analysis of robustness issues in fuzzy clustering have not been made until the mid-nineties.

In this paper, we provide a brief review on robustness issues in fuzzy cluster analysis. We address problems and questions that have not been solved or treated completely so far. But we also would like to draw the attention to those results that are available and that can help in applying the methods of fuzzy clustering in a suitable manner.

We start with an overview on prototype-based clustering, emphasising special forms of fuzzy cluster analysis like noise clustering in section 1.2. In order to keep the paper self-contained, a short detour on issues in robust statistics is needed in section 1.3. Section 1.4 brings together fuzzy cluster analysis and ideas from robust statistics, showing that fuzzy cluster analysis fits quite well into the scheme of robust statistics. In the final conclusions in section 1.5 we address consequences for fuzzy clustering drawn from the robustness considerations and derive possible approaches to improve fuzzy clustering.

1.2 Cluster analysis

Cluster analysis aims at dividing a data set into groups or clusters that consist of similar data. There is a large number of clustering techniques available with different underlying assumptions about the data and the clusters

[1] [1, 3]
[2] [12]

to be discovered. A simple and common popular approach is the so-called c-means clustering as for instance described by Duda and Hart[3]. For the c-means 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 $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^p$ 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 c-means algorithm as well. 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, in this paper we do not consider how to determine the number of clusters and assume a fixed given number of clusters.

1.2.1 Objective function-based clustering

From the purely algorithmic point of view, the c-means clustering approach can be described as follows. Each of the c clusters is represented by a cluster prototype $v_i \in \mathbb{R}^p$, also simply called prototype. These prototypes are chosen randomly or in a suitable fashion in the beginning. Afterwards each data vector is assigned to the nearest prototype (with respect to 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 minimizing the following objective function

$$f \;=\; \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij} d_{ij} \tag{1.1}$$

under the constraints

$$\sum_{i=1}^{c} u_{ij} \;=\; 1 \qquad \text{for all } j = 1, \ldots, n \tag{1.2}$$

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.

Since this is a non-trivial constraint nonlinear optimisation problem with continuous parameters v_i and discrete parameters u_{ij}, there is no obvious analytical solution. Therefore an alternating optimisation scheme, alternatingly optimising one set of parameters while the other set of parameters is considered as fixed, seems to be a reasonable approach for minimizing (1.1). The above mentioned c-means clustering algorithm follows exactly this strategy.

[3] [11]

It should be noted that choosing the (squared) Euclidean distance as a measure for the distance between data vector u_{ij} and cluster i is just one choice out of many. Later on, we will also consider other distance measures and forms of prototypes as they can be found in the overviews by Bezdek et al.[4] or Höppner et al.[5]

The constraint $u_{ij} \in \{0, 1\}$ requires that each data point must be assigned uniquely to one single cluster. In this way, even noisy data points are enforced to be assigned artificially to a unique cluster and thus inflict an error on the cluster prototype of the corresponding cluster. Furthermore, cluster boundaries are very often not sharp and the assignment of a data point close to the boundary between clusters to a unique cluster gives the wrong impression of well-separated clusters.

For this reason, the constraint $u_{ij} \in \{0, 1\}$ is relaxed to $u_{ij} \in [0, 1]$. However, even with this relaxed constraint the minimum of the objective function (1.1) under the general constraint (1.2) is still found at $u_{ij} \in \{0, 1\}$. Therefore, an additional parameter m, the so-called fuzzifier, was introduced by Bezdek[6] and Dunn[7], and the objective function (1.1) is replaced by

$$f = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^{m} d_{ij}. \tag{1.3}$$

Note that the fuzzifier m does not have any effects, when hard clustering, i.e. $u_{ij} \in \{0, 1\}$, is applied. The fuzzifier $m > 1$ is not subject of the optimisation process and has to be chosen in advance. A typical choice is $m = 2$.

The fuzzy clustering approach with the objective function (1.3) under the constraints (1.2) and the assumption $u_{ij} \in [0, 1]$ is called probabilistic clustering, since due to the constraints (1.2) the membership degree u_{ij} can be interpreted as the probability that x_j belongs to cluster i. Nevertheless, due to the fuzzifier, a strict probabilistic interpretation as for instance in the case of expectation maximisation (EM) clustering introduced by Dempster et al.[8] is not possible.

The relaxed constraint $u_{ij} \in [0, 1]$ for fuzzy cluster analysis still leads to a nonlinear optimisation problem, however, in contrast to hard clustering, with all parameters being continuous. The common technique for minimizing this objective function is similar as in hard clustering, alternatingly optimise either the membership degrees or the cluster parameters while considering the other parameter set as fixed.

Taking the constraints (1.2) into account by Lagrange functions, the minimum of the objective function (1.3) with respect to the membership degrees is obtained at

[4] [5]
[5] [17]
[6] [1, 3]
[7] [12], only for the specific choice $m = 2$.
[8] [12]

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{kj}} \right)^{\frac{1}{m-1}}}, \tag{1.4}$$

when the cluster parameters, i.e. the distance values d_{ij}, are considered to be fixed. (If $d_{ij} = 0$ for one or more clusters, we deviate from (1.4) and assign x_j with membership degree 1 to the or one of the clusters with $d_{ij} = 0$ and choose $u_{ij} = 0$ for the other clusters i.) For a derivation of equation (1.4), we refer to Bezdek[9].

If the clusters are represented by simple cluster prototypes $v_i \in \mathbb{R}^p$ and the distances d_{ij} are the squared Euclidean distances of the data to the corresponding cluster prototypes as in the hard c-means algorithm, the minimum of the objective function (1.3) with respect to the cluster prototypes is obtained at

$$v_i = \frac{\sum_{j=1}^{n} u_{ij}^m x_j}{\sum_{j=1}^{n} u_{ij}^m}, \tag{1.5}$$

when the membership degrees u_{ij} are considered to be fixed. For a derivation of equation (1.5), we refer again to Bezdek[10]. The cluster prototypes are still the cluster centres. However, using $[0, 1]$-valued membership degrees means that we have to compute weighted cluster centres. The fuzzy clustering scheme using alternatingly equations (1.4) and (1.5) is called fuzzy c-means algorithm (FCM).

1.2.2 Noise clustering and other variants

One of the problems of the above described approach to fuzzy cluster analysis is caused by the constraints specified in equation (1.2) enforcing that each data point must be assigned to the overall degree one to the clusters. As an example consider only two clusters. A data point roughly in the middle between the two clusters will have a membership degree of approximately 0.5 to both cluster, which seems to be a suitable choice, indicating that the data point fits both clusters equally well. However, an outlier, i.e. a data point far away from both clusters, will also have a membership degree of approximately 0.5 to both cluster. Here the membership degree 0.5 means that the outlier fits equally badly to both clusters.

Noise clustering, proposed by Davé[11], tries to solve this problem by introducing an additional noise cluster. All data points have a fixed (large) distance δ to the noise cluster. In this way, data points that are near the border between two clusters still have a high membership degree to both clusters as in probabilistic clustering. But data points that are far away from all clusters will be assigned to the noise cluster and have no longer a considerable membership degree to other clusters.

[9] [3]
[10] [3]
[11] [8]

Besides noise clustering, there are also other approaches to avoid problems caused by the strict probabilistic constraints (1.2). Krishnapuram and Keller[12] introduced possibilistic clustering where the probabilistic constraint is completely dropped and an additional term in the objective function is introduced to avoid the trivial solution $u_{ij} = 0$ for all i, j. However, the aim of possibilistic clustering is actually not to find the global minimum of the corresponding objective function, since this is obtained when all clusters are identical as shown by Timm and Kruse[13].

Another approach that emphasizes a probabilistic interpretation in fuzzy clustering is described by Flores-Sintas et al.[14] where membership degrees as well as membership probabilities are used for the clustering. In this way, some of the problems of the standard FCM scheme can be avoided as well. However, this approach assumes the use of the Euclidean or a Mahalanobis distance and is not suitable for arbitrary cluster shapes as in shell clustering.

Keller[15] introduced additional adaptive weights to reduce the influence of outliers to the clustering results.

A solution to another problem caused by the objective function (1.3) in connection with the constraints (1.2) is discussed by Klawonn and Höppner[16]. Due to equation (1.4), zero membership degrees will never occur, except in the extremely rare case when a data point has zero distance to a cluster prototype. As a consequence, all data points will always influence all cluster prototypes, no matter how well they are covered by any cluster prototype or how far away they are from another cluster prototype. By choosing a small fuzzifier $m > 1$, this effect can be reduced, but not completely eliminated. One of the reasons for introducing the fuzzifier was that the original objective function (1.1) without a fuzzifier would lead to crisp membership degrees, even when the constraint $u_{ij} \in \{0, 1\}$ is relaxed to $u_{ij} \in [0, 1]$. Replacing u_{ij} in the objective function (1.1) by u_{ij}^m to obtain the modified objective function (1.3) means nothing else than to apply a suitable transformation to the u_{ij}. Instead of the transformation $u \mapsto u^m$ based on the fuzzifier m, other transformations $g : [0, 1] \to [0, 1]$ are also possible, for instance

$$g(u) = \alpha u^2 + (1 - \alpha)u \tag{1.6}$$

or

$$g(u) = \frac{1}{e^\alpha - 1} \left(e^{\alpha u} - 1\right). \tag{1.7}$$

In both cases, α is a control parameter similar to the fuzzifier m. These two alternative transformations do not only satisfy suitable general constraints like monotonicity, but lead also to tractable computation schemes for the

[12] [26]
[13] [28]
[14] [13]
[15] [19]
[16] [22, 23]

membership degrees u_{ij}, although they are slightly more complicated than the simple equation (1.4). Both transformations lead to zero membership degree of a data point to a cluster far away from it, at least when the data point is well covered by another cluster.

1.2.3 Other cluster prototypes

So far, we have only considered modifications concerning the membership degrees u_{ij}, but have not touched the cluster prototypes and the related distances d_{ij} in the objective function (1.3). Gustafson and Kessel[17] extended the cluster prototypes by covariance matrices, so that clusters could not only have the shape of (hyper-)spheres, but of ellipsoids.

Bock[18] and later on Bezdek[19] introduced clusters in the form of affine subspaces. The corresponding clustering algorithm is called fuzzy c-varieties algorithm (FCV). A cluster prototypes describes an r-dimensional hyperplane

$$v_i + < e_{i,1}, \ldots, e_{i,r} >= \left\{ y \in \mathbb{R}^p \mid (\exists t \in \mathbb{R}^r) \left(y = v_i + \sum_{s=1}^{r} t_s e_{i,s} \right) \right\}, \quad (1.8)$$

defined by a point v_i and r (orthogonal) vectors $e_{i,1}, \ldots, e_{i,r}$ spanning the hyperplane. The distance of a data point x_j to the cluster prototype is the difference between the squared lengths of the vector $(x_j - v_i)$ and its projection to the hyperplane associated with the cluster prototype. This is the same as the squared distance of x_j to the hyperplane. The distance is zero if and only if the point x_j belongs to the hyperplane.

There are many other cluster shapes that can be described by suitable cluster prototypes and an adequate distance function. In principle, almost any cluster shape would be possible, however, for the price that the computations for the parameters of the prototypes become extremely complicated. Since the clustering algorithms are usually based on an iteration scheme in which the membership degrees and the cluster prototypes are updated alternatingly, it is highly recommended that there exists an explicit solution for the optimal cluster prototypes, assuming the membership degrees to be fixed. Cluster prototypes have even been extended to boundaries of geometric shapes like circles or ellipses. These techniques are called shell clustering. For overviews we refer to Krishnapuram et al.[20] and Klawonn et al.[21].

A detailed discussion of different cluster shapes is not the topic of this paper. Nevertheless, it is important to notice that more complex cluster prototypes lead to two significant problems. The objective function (1.3)

[17] [14]
[18] [6]
[19] [3]
[20] [25]
[21] [24]

tends to have more local minima, leading to bad clustering results, i.e. the alternating optimisation strategy gets stuck in a local minimum, although the data set might contain clear cluster structures. In addition to the problem of local minima, the clustering result is also more sensitive to noise and outliers. In order to discuss these topics, we briefly introduce some fundamental notions from robust statistics in the following section.

1.3 Notions from robust statistics

Classical statistics mainly focuses on procedures that are optimal – for instance in terms of efficient estimators – given the model assumptions are correct. For example, assuming that a sample comes from a normal distribution, the most efficient estimator for the expected value is the mean value. The same applies to the least squares method for linear regression. As long as the model assumption[22]

$$y_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_k x_{ik} + \varepsilon_i = x_i^\top \beta + \varepsilon_i \tag{1.9}$$

where the ε_i are independent normal distributions with zero mean and the same variance for all i and the β_i are the unknown regression coefficients.

However, it is well known that even single outliers can have extreme influence on the mean value or on the estimation of the coefficients of the regression function. Robust statistics deals with such problems.

1.3.1 Robustness

Classical statistics assumes that the data represent independent samples from the same distribution F_{model}, the "model". Robust statistics assumes that the data are partly corrupted, i.e. the ideal model distribution F_{model} is mixed with an unknown noise distribution.

$$F \;=\; (1-\varepsilon) \cdot F_{\text{model}} + \varepsilon \cdot F_{\text{random}}. \tag{1.10}$$

The aim of robust statistics is the development of methods that perform well even under the imperfect conditions (1.10). For an overview on robust statistics and related methods, we refer to Huber[23] and Hoaglin et al[24].

1.3.2 Resistance

Of course, even robust statistics cannot cope with the situation when the influence of the noise distribution F_{random} in equation (1.10) becomes too

[22] Note that each data point $(x_{i1}, \ldots, x_{ik})$ has been extended by the constant component $x_{i0} = 1$ in the last part of the equation in order to simplify the notation.
[23] [18]
[24] [15]

strong. Nevertheless, methods from robust statistics try to cope with as much distortion from the noise distribution as possible. One way to analyse robust methods is to consider F_{random} is "random noise". However, it is not always clear for every model what random noise means. Another way to investigate robust methods is to consider the influence of single data points and of extreme outliers.

The influence curve shows, how a single data point added to the data will change the estimation of the model parameters. Influence curves are very helpful to analyse the influence of data points to single parameters of a model. The breakdown point is the proportion of extreme outliers that can be included in the data set without (drastically) changing the estimation of the model parameters. For instance, the mean has a breakdown point of zero, since a single extreme outlier $x \to \infty$ will also let the mean tend to infinity. In contrast, the median has a breakdown point of (almost) 50%, because the median depends only on the point or two points in the middle of the ordered data.

In this paper we will mainly focus on resistance consideration concerning fuzzy cluster analysis.

1.3.3 M-estimators and robust regression

Before we view fuzzy cluster analysis from the viewpoint of robust statistics, we need another notion from robust statistics, the so-called M-estimators. An M-estimator for a model parameter or vector of model parameters θ is based on minimizing a suitable error function indicating how well the choice of θ fits the data. It is sufficient to consider the case of linear regression here.

Given a data set of measured values[25] $(x_1, y_1), \ldots, (x_n, y_n)$, the aim is to determine a linear model

$$y_i = b_0 + b_1 x_{i1} + \ldots + b_k x_{ik} + e_i = x_i^\top b + e_i \tag{1.11}$$

defined by the coefficient vector b and to minimize the errors e_i.

The objective function to be minimized is

$$\sum_{i=1}^{n} \rho(e_i) \;=\; \sum_{i=1}^{n} \rho(y_i - x_i^\top b) \tag{1.12}$$

where ρ is a suitable error measure.

A suitable error measure should at least satisfy the following properties:

$$\rho(e) \geq 0 \tag{1.13}$$

$$\rho(0) = 0 \tag{1.14}$$

$$\rho(e) = \rho(-e) \tag{1.15}$$

$$\rho(e_i) \geq \rho(e_j) \text{ if } |e_i| \geq |e_j|. \tag{1.16}$$

[25] Note that the x_i can be vectors.

Parameter estimation (here the estimation of the parameter vector b) based on an objective function of the form (1.12) and an error measure satisfying (1.13)–(1.16) is called an M-estimator. The classical least squares approach is based on the quadratic error, i.e. $\rho(e) = e^2$. Table 1.1 provides the error measure ρ for the classical least squares method as well as for two approaches from robust statistics.

Method	$\rho(e)$
Least squares	e^2
Huber	$\begin{cases} \frac{1}{2}e^2 & \text{if } \lvert e\rvert \le k, \\ k\lvert e\rvert - \frac{1}{2}k^2 & \text{if } \lvert e\rvert > k. \end{cases}$
Bisquare	$\begin{cases} \frac{k^2}{6}\left(1-\left(1-\left(\frac{e}{k}\right)^2\right)^3\right), & \text{if } \lvert e\rvert \le k, \\ \frac{k^2}{6}, & \text{if } \lvert e\rvert > k. \end{cases}$

Table 1.1. Error measures ρ for different approaches.

In order to understand the more general setting of an error measure ρ satisfying (1.13)–(1.16), it is useful to consider the derivative of the error measure $\psi = \rho'$.

Taking the derivatives of the objective function (1.12) with respect to the parameters b_i, we obtain a system of $(k+1)$ linear equations

$$\sum_{i=1}^{n} \psi_i(y_i - x_i^\top b)x_i^\top = 0. \tag{1.17}$$

Defining $w(e) = \psi(e)/e$ and $w_i = w(e_i)$, (1.17) can be rewritten in the form

$$\sum_{i=1}^{n} \frac{\psi_i(y_i - x_i^\top b)}{e_i} \cdot e_i \cdot x_i^\top = \sum_{i=1}^{n} w_i \cdot (y_i - x_i^\top b) \cdot x_i^\top = 0. \tag{1.18}$$

Solving this system of linear equations corresponds to solving a standard least squares problem with (non-fixed) weights in the form

$$\sum_{i=1}^{n} w_i e_i^2. \tag{1.19}$$

However, the weights w_i depend on the residuals e_i, the residuals depend on the coefficients b_i and the coefficients depend on the weights. Therefore, it is in general not possible to provide an explicit solution to the system of equations. Instead, the following iteration scheme is applied.

1. Choose an initial solution $b^{(0)}$, for instance the standard least squares solution setting all weights $w_i = 1$.

2. In each iteration step t, calculate the residuals $e^{(t-1)}$ and the corresponding weights $w^{(t-1)} = w\left(e^{(t-1)}\right)$ determined by the previous step.
3. Compute the solution of the weighted least squares problem $\sum_{i=1}^{n} w_i e_i^2$, i.e.

$$b^{(t)} = \left(X^\top W^{(t-1)} X\right)^{-1} X^\top W^{(t-1)} y. \tag{1.20}$$

This iterative algorithm shows an obvious resemblance with the alternating optimisation scheme of fuzzy clustering. The weights for robust regression play a similar role as the membership degrees in fuzzy clustering and the regression coefficient correspond to the parameters of the cluster prototypes.

Method	$w(e)$
Least squares	1
Huber	$\begin{cases} 1 & \text{if } \lvert e\rvert \le k, \\ k/\lvert e\rvert, & \text{if } \lvert e\rvert > k. \end{cases}$
Bisquare	$\begin{cases} \left(1 - \left(\frac{e}{k}\right)^2\right)^2 & \text{if } \lvert e\rvert \le k, \\ 0, & \text{if } \lvert e\rvert > k. \end{cases}$

Table 1.2. The computation of the weights for the corresponding approaches.

Table 1.2 lists the formulae for the weights in the regression scheme based on the error measures listed in table 1.1.

Figure 1.1 shows the graph of the error measure and the weighting function for the standard least squares approach. The error measure ρ increases in a quadratic manner with increasing distance. The weights are always constant. This means that extreme outliers will have full influence on the regression coefficients and can corrupt the result completely.

In the more robust approach by Huber the change of the error measure ρ switches from a quadratic increase for small errors to a linear increase for larger errors. As a result, only data points with small errors will have the full influence on the regression coefficients. For extreme outliers the weights tend to zero. This is illustrated by the corresponding graphs in figure 1.2.

The bisquare approach is even more drastic than Huber's approach. For larger errors the error measure ρ does not increase at all, but remains constant. As a consequence, the weights for outliers drop to zero when they are too far away from the regression curve. This means that extreme outliers have no influence on the regression curve at all. The corresponding graphs for the error measure and the weights are shown in figure 1.3.

1.4 Robustness issues in fuzzy clustering

In the previous section, a relation between M-estimators and cluster analysis has already been established where the membership degrees in fuzzy cluster

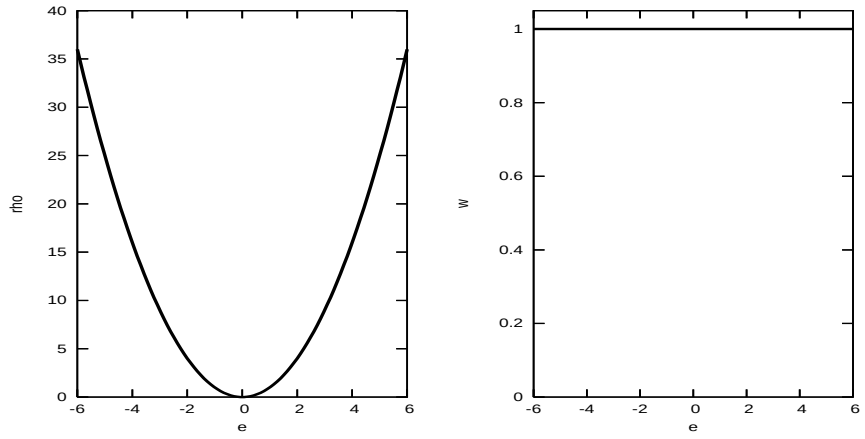

Fig. 1.1. The error measure ρ and the weight w for the standard least squares approach.

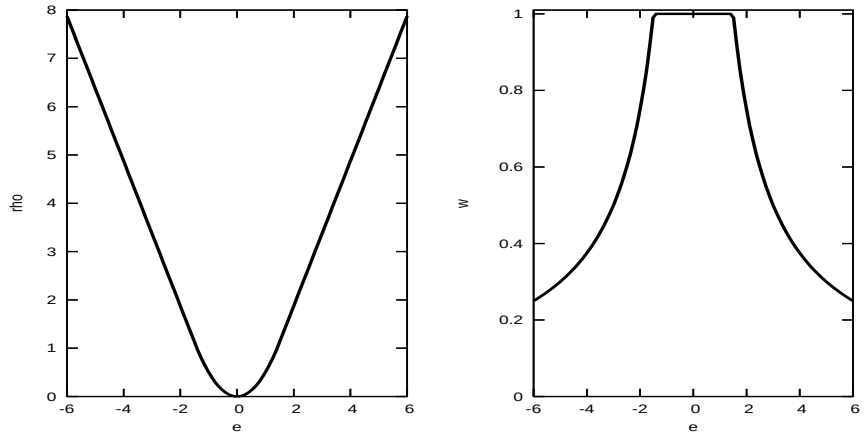

Fig. 1.2. The error measure ρ and the weight w for Huber's approach.

analysis take the part of the weights in robust regression. In this section, we take a closer look at this connection and other robustness issues in fuzzy clustering. The next subsection first provides an exact correspondence between a special case of the fuzzy clustering algorithm FCV and robust regression.

1.4.1 A simple fuzzy regression model

Let us consider the special case of FCV with a single cluster in combination with noise clustering. This means that the single cluster represents a linear regression function. It can be shown easily that the cluster results from weighted least squares regression with the membership degrees to the power

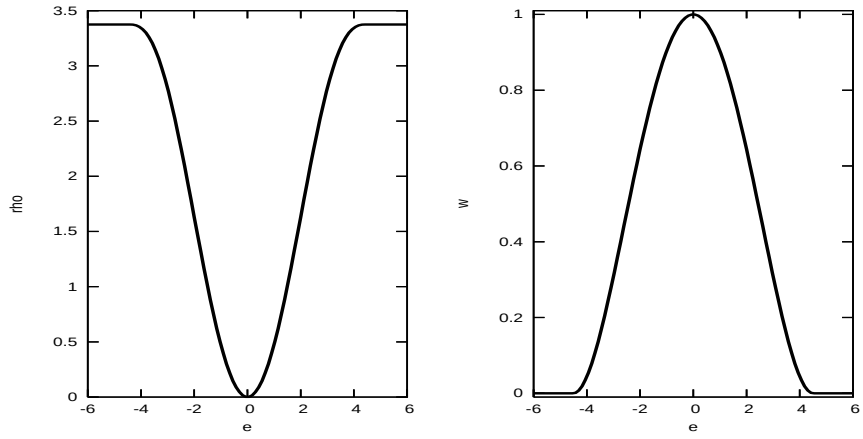

Fig. 1.3. The error measure ρ and the weight w for the bisquare approach.

of m as weights. The membership degree of a data point x to the single cluster is given by

$$u = \frac{1}{1 + \left(\frac{d^2}{\delta}\right)^{\frac{1}{m-1}}} \tag{1.21}$$

where d is proportional to the distance of x to the cluster, m is the fuzzifier and δ is the noise distance. The membership degree to the noise cluster is $1 - u$. Figure 1.4 shows this curve. The weight is given by $w = u^m$.

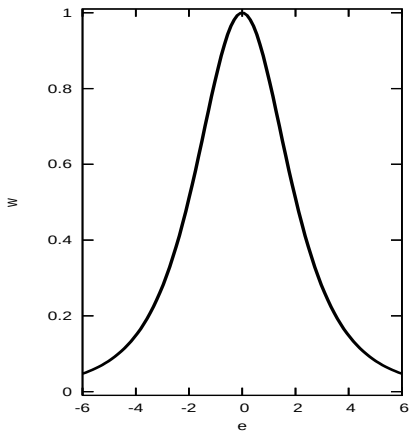

Fig. 1.4. The membership degree for FCV.

It neither corresponds to the Huber nor to the bisquare weight curve in figures 1.2 and 1.3, respectively. In contrast to the Huber weight curve, the

weight one is only assumed for a residual or error of zero. But like the Huber weight curve, it only approaches zero for larger residuals, but does never reach zero. In this aspect it differs strongly from the bisquare weight curve. Before we continue our investigations on weight curves for clustering in more general terms, we consider general convergence aspects of the alternating optimisation scheme.

1.4.2 Convergence issues and the avoidance of local minima

It was shown by Bezdek[26] and later on in the corrected paper by Bezdek et al.[27] that the fuzzy c-means algorithm does always converge to a local minimum or, in the worst case, to a saddle point of the objective function (1.3). The convergence conditions were further elaborated and generalised to other algorithms by Höppner and Klawonn[28]. Nevertheless, the problem of local minima remains. This is not a specific problem of fuzzy clustering, but already a problem for classical hard c-means clustering. Figure 1.5 illustrates this problem by a very simple example, how c-means clustering can get stuck in a local minimum. The indicated partition of the data into clusters does not correspond to our intuition. The cluster prototypes are marked by crosses. The problem here is that the prototype on the right-hand side covers two clusters and the other two prototypes have to compete for data in one cluster. However, since all data points in the two clusters in the right-hand side of the figure are closer to the single prototype on the right-hand side, the other two prototypes will never "take any notice" of these data points and cannot be attracted by them.

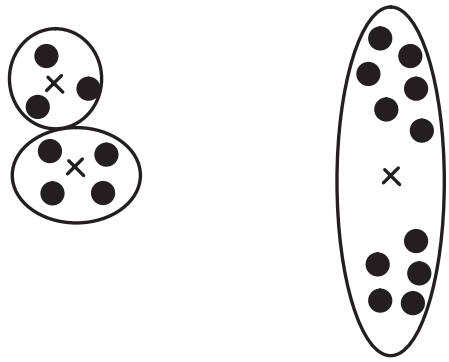

Fig. 1.5. An undesired local minimum for c-means clustering.

[26] [2]
[27] [4]
[28] [16]

In this case, fuzzy clustering might even be able to overcome this problem. Since the membership degrees in fuzzy clustering are (almost) never zero, the two prototypes on the left-hand side will at least be slightly attracted by the data points on the right-hand side, so that fuzzy clustering is able to escape certain local minimum. Klawonn[29] has demonstrated that at least in certain settings the introduction of the fuzzifier can smooth out undesired local minimum in the objective function (1.1).

Nevertheless, the introduction of the fuzzifier can also lead to new problems. For instance, in the case when clusters of highly different densities exist. Then the large number of data points from a very dense cluster will still attract the cluster prototypes of other clusters, even if the dense cluster is well covered by a cluster prototype. We will come back to this problem later on.

1.4.3 Fuzzy clustering and M-estimators

Although it was very often empirically claimed that fuzzy clustering is more robust, it took more than twenty years for the first in depth investigations of robustness properties of fuzzy clustering initiated among others by Nasraoui and Krishnapuram[30] and carried out in more detail by Davé and R. Krishnapuram[31]. These authors and Choi and Krishnapuram[32] have established relations between fuzzy clustering – especially noise and possibilistic clustering – and M-estimators and also W-estimators, another class of estimators from robust statistics. In subsection 1.4.1 we have demonstrated on which concepts the relation between robust estimators and fuzzy clustering is based. For further analysis, the objective function (1.3) was generalised by the above mentioned authors to the form

$$f = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^{m} \rho(d_{ij}). \tag{1.22}$$

It would lead to far to discuss all details here and we refer to the original works. Instead, in the next subsection we want to point out some problems that still remain and are caused by the fuzzifier.

1.4.4 Resistance properties of fuzzy cluster analysis

It is quite obvious that standard fuzzy clustering is not at all resistant to extreme outliers. For FCM, for extreme outliers x with $\| x \| \to \infty$, the distance to all cluster prototypes of such outliers will also tend to infinity. In this case, we can see from update equation (1.4) for the membership degrees

[29] [20, 21]
[30] [27]
[31] [9]
[32] [7]

that the membership degrees for the outliers will all converge to $1/c$. So the outliers with their (almost) infinite distance will also draw the cluster prototypes away from the data.

The situation changes when noise clustering is applied. Let us again consider a data point x_j with $\| x_j \| \to \infty$ and its influence on cluster prototype i. Apart from the normalising nominator in equation (1.5), its contribution to the location of the prototype is $u_{ij}x_j$. Let us just consider the length $\| u_{ij}x_j \|$ of this vector. For a finite prototype v_i and for x_j with large norm $\| x_j \|$, we have, assuming c ordinary clusters and one noise cluster with noise distance δ

$$\| x_j - v_i \| \approx \| x_j \| = \sqrt{d_{ij}} \tag{1.23}$$

when d_{ij} denotes the squared Euclidean distance. This implies

$$\lim_{\|x_j\|\to\infty} \| u_{ij}x_j \| = \lim_{\|x_j\|\to\infty} u_{ij}\sqrt{d_{ij}} \tag{1.24}$$

$$= \lim_{\|x_j\|\to\infty} \frac{\sqrt{d_{ij}}}{\sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{kj}}\right)^{\frac{1}{m-1}} + \left(\frac{d_{ij}}{\delta}\right)^{\frac{1}{m-1}}} \tag{1.25}$$

$$= \lim_{d\to\infty} \frac{\sqrt{d}}{c + \left(\frac{d}{\delta}\right)^{\frac{1}{m-1}}} \tag{1.26}$$

$$= \lim_{d\to\infty} \frac{1}{\frac{c}{\sqrt{d}} + \delta^{\frac{-1}{m-1}} d^{\frac{3-m}{2m-2}}} \tag{1.27}$$

$$= \lim_{d\to\infty} \delta^{\frac{1}{m-1}} d^{\frac{m-3}{2m-2}} \tag{1.28}$$

$$= \begin{cases} 0 & \text{if } 1 < m < 3, \\ \sqrt{\delta} & \text{if } m = 3, \\ \infty & \text{if } m > 3. \end{cases} \tag{1.29}$$

This implies that for a fuzzifier smaller than 3, the noise cluster will prevent the other clusters from being corrupted by extreme outliers. However, for a fuzzifier larger than 3, even the noise cluster cannot protect the other clusters from being inflicted by extreme outliers.

No matter, wether a noise cluster is introduced or not, due to equation (1.4), outliers and all other data will still have an influence on all clusters. In terms of robust statistics, this is very much in the spirit of Hubert's error measure. The influence of outliers is gradually reduced, but never completely reduced to zero. The more drastic bisquare approach, removing the influence of outliers completely, can only be achieved when the simple fuzzifier transformation $g(u) = u^m$ is replaced by generalised transformations as mentioned in equations (1.6) and (1.7). In this case, extreme outliers will be covered by the noise cluster completely and have zero membership degree to all other clusters.

1.5 Conclusions

Robustness issues have been neglected in fuzzy cluster analysis for quite a long time and still have not been investigated and exploited in full detail. Especially the problem of non-zero membership degrees for all – outliers as well as data points from other clusters – caused by equation (1.4) has not been a serious issue until recently. Two approaches might be needed:

(a) On the one hand it is reasonable not to neglect outliers completely as for instance in Hubert's approach in robust regression and in the case of the standard fuzzifier in fuzzy clustering. When outliers are ignored completely or membership degrees are set to absolutely zero, this can easily lead to the problems illustrated in figure 1.5 and the danger of getting stuck in local minima of the objective function.

(b) On the other hand, for larger data sets and especially for clusters with different densities, the avoidance of zero membership degrees leads to undesired results. In this case, the global minimum of the objective function might not coincide with the intuitive partition into clusters.

In this sense, it seems reasonable – apart from making sure to find a good initialisation for clustering – to start the clustering procedure in terms of approach (a) in order give each cluster a chance to "see" all data in the beginning. But for better resistance and robustness purposes in the later stage of the clustering procedure it might be advisable to switch to approach (b) to remove the influence of outliers completely as well as to stop data from dense clusters to influence other cluster prototypes. Little work has been carried out in this direction so far.

References

1. J.C. Bezdek, "Fuzzy Mathematics in Pattern Classification", Ph.D. Thesis, Applied Math. Center, Cornell University, Ithaca, 1973
2. J.C. Bezdek, "A Convergence Theorem for the Fuzzy ISODATA Clustering Algorithm", IEEE Trans. Pattern Analysis and Machine Intelligence 2, pp. 1-8, 1980
3. J.C. Bezdek, "Pattern Recognition with Fuzzy Objective Function Algorithms", Plenum Press, New York, 1981
4. J.C. Bezdek, R.H. Hathaway, M.J. Sabin, W.T. Tucker, "Convergence Theory for Fuzzy c-Means: Counterexamples and Repairs", IEEE Trans. Systems, Man, and Cybernetics 17, pp. 873-877, 1987
5. J.C. Bezdek, J. Keller, R. Krishnapuram, N.R. Pal, "Fuzzy Models and Algorithms for Pattern Recognition and Image Processing", Kluwer, Boston 1999
6. H.H. Bock, "Clusteranalyse mit unscharfen Partitionen", in: H.H. Bock (ed.), "Klassifikation und Erkenntnis: Vol. III: Numerische Klassifikation", INDEKS, Frankfurt, pp. 137-163, 1979
7. Y. Choi, R. Krishnapuram, "Fuzzy and Robust Formulation of Maximum-Likelihood-Based Gaussian Mixture Decomposition", in: IEEE Conference on Fuzzy Systems, New Orleans, pp. 1899-1905, 1996
8. R.N. Davé, "Characterization and Detection of Noise in Clustering", Pattern Recognition Letters 12, pp. 657-664, 1991
9. R. N. Davé, R. Krishnapuram, "Robust Clustering Methods: A Unified View", IEEE Trans. on Fuzzy Systems 5, pp. 270-293, 1997
10. A. Dempster, N. Laird, D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm", Journal of the Royal Statistical Society: Series B 39, pp. 1–38, 1977
11. R. Duda, P. Hart, "Pattern Classification and Scene Analysis", Wiley; New York 1973
12. J.C. Dunn, "A Fuzzy Relative of the ISODATA Process and its Use in Detecting Compact, Well Separated Clusters", Journ. Cybern. 3, pp. 32-57, 1973
13. A. Flores-Sintas, J.M. Cadenas, F. Martin, "Membership Functions in the Fuzzy c-Means Algorithm", Fuzzy Sets and Systems 101, pp. 49-58, 1999

14. D.E. Gustafson, W.C. Kessel, "Fuzzy Clustering with a Fuzzy Covariance Matrix", Proc. IEEE CDC, San Diego, pp. 761-766, 1979
15. D.C. Hoaglin, F. Mosteller, J.W. Tukey, "Understanding Robust and Exploratory Data Analysis", Wiley, New York, 2000
16. F. Höppner, F. Klawonn, "A Contribution to Convergence Theory of Fuzzy c-Means and its Derivatives", IEEE Trans. on Fuzzy Systems 11, pp. 682-694, 2003
17. F. Höppner, F. Klawonn, R. Kruse, T. Runkler, "Fuzzy Cluster Analysis", Wiley, Chichester 1999
18. P.J. Huber, "Robust Statistics", Wiley, New York, 2004
19. A. Keller, "Fuzzy Clustering with Outliers", Proc. NAFIPS 2000: 19th International Conference of the North American Fuzzy Information Processing Society, Atlanta, pp. 143-147 , 2000
20. F. Klawonn, "Fuzzy Clustering: Insights and a New Approach", Mathware and Soft Computing 11 (2004), 125-142
21. F. Klawonn, "Understanding and Controlling the Membership Degrees in Fuzzy Clustering", in: M. Spiliopoulou, R. Kruse, C. Borgelt, A. Nürnberger, W. Gaul (eds.), "From Data and Information Analysis to Knowledge Engineering", Springer, Berlin, pp. 446-453, 2006
22. 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 V", Springer, Berlin, pp. 254-264, 2003
23. F. Klawonn, F. Höppner, "An Alternative Approach to the Fuzzifier in Fuzzy Clustering to Obtain Better Clustering Results", in: Proc. 3rd Eusflat Conference, Zittau, pp. 730-734, 2003
24. F. Klawonn, R. Kruse, H. Timm, "Fuzzy Shell Cluster Analysis", in: G. Della Riccia, H.J. Lenz, R. Kruse (eds.), "Learning, Networks and Statistics", Springer, Wien, pp. 105-120, 1997
25. R. Krishnapuram, H. Frigui, O. Nasraoui, "Fuzzy and Possibilistic Shell Clustering Algorithms and Their Application to Boundary Detection and Surface Approximation – Part 1 & 2", IEEE Trans. on Fuzzy Systems 3, pp. 29-60, 1995
26. R. Krishnapuram, J. Keller, "A Possibilistic Approach to Clustering", IEEE Trans. on Fuzzy Systems 1, pp. 98-110, 1993
27. O. Nasraoui, R. Krishnapuram, "Crisp Interpretation of Fuzzy and Possibilistic Clustering Algorithms", in: 3rd European Congress on Intelligent Techniques and Soft Computing, vol. 3, Aachen, pp. 1312-1318, 1995
28. H. Timm, R. Kruse, "A Modification to Improve Possibilistic Fuzzy Cluster Analysis", in: FUZZ-IEEE'02: Proc. of the 2002 IEEE International Conference on Fuzzy System Volume 2, pp. 1460-1465, 2002