Available online at www.sciencedirect.com
ScienceDirect
Fuzzy Sets and Systems ••• (••••) •••–•••
FUZZY sets and systems
www.elsevier.com/locate/fss

Fuzzy *c*-ordered-means clustering

Jacek M. Leski [a,b,*]

[a] *Institute of Electronics, Silesian University of Technology, Akademicka 16, 44-100 Gliwice, Poland*
[b] *Institute of Medical Technology and Equipment, Roosevelt St. 118A, 41-800 Zabrze, Poland*

Received 13 February 2013; received in revised form 10 December 2014; accepted 14 December 2014

Abstract

Fuzzy clustering helps to find natural vague boundaries in data. The fuzzy *c*-means method is one of the most popular clustering methods based on minimization of a criterion function. However, one of the greatest disadvantages of this method is its sensitivity to the presence of noise and outliers in data. This paper introduces a new robust fuzzy clustering method named Fuzzy *C*-Ordered-Means (FCOM) clustering. This method uses both the Huber's M-estimators and the Yager's OWA operators to obtain its robustness. The proposed method is compared to many other ones, e.g.: the Fuzzy *C*-Means (FCM), the Possibilistic Clustering (PC), the fuzzy Noise Clustering Method (NCM), the L_p norm clustering (L_p FCM) ($0 < p < 1$), the L_1 norm clustering (L_1 FCM), the Fuzzy Clustering with Polynomial Fuzzifier (PFCM) and the ε-insensitive Fuzzy *C*-Means (βFCM). To this end experiments on synthetic data with outliers have been performed as well as on data with heavy-tailed and overlapping groups of points in background noise.

Keywords: Fuzzy clustering; Fuzzy *c*-means; ε-Insensitivity; Ordered weighted averaging; Robust methods

1. Introduction

Clustering plays an important role in many engineering fields such as pattern recognition, Web mining, image segmentation, signal processing, system modeling, communication, data mining, and so on. The clustering methods divide a set of N vector observations $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ into c groups denoted $\Omega_1, \Omega_2, \ldots, \Omega_c$ so that the members of the same group are more similar to one another than to the members of the other groups. Generally, clustering methods can be divided into [16]: hierarchical, graph theoretic, decomposing a density function, minimizing a criterion function. In this paper clustering by minimization of a criterion function will be considered.

The traditional clustering methods assume that each data vector belongs to one and the only one class. This approach can be natural for clustering of compact and well-separated groups of data. However, usually clusters overlap, and some data vectors belong partially to two or even more clusters. The fuzzy set theory is a natural way of describing

* Corresponding author at: Institute of Electronics, Silesian University of Technology, Akademicka 16, 44-100 Gliwice, Poland. Tel.: +48 32 2372004, fax: +48 32 2372225.
E-mail address: jacek.leski@polsl.pl.

http://dx.doi.org/10.1016/j.fss.2014.12.007
0165-0114/

this situation. A membership degree of a vector $\mathbf{x}_k$ to the i-th cluster Ω_i is a value from [0, 1] interval, $(u_{ik}) \in [0, 1]$. This idea was first introduced by Ruspini [31] and used by Dunn [14] to construct a fuzzy clustering method based on a criterion function minimization. In [1] Bezdek generalized this approach to an infinite family of fuzzy c-means (FCM) algorithms using a weighting exponent on the fuzzy memberships.

The fuzzy c-means clustering algorithm has successfully been applied to a wide variety of problems [1], e.g. in [19] it was combined with evolutionary computations, in [6,8–10,12] it was applied for complex structures of data such as time trajectories, time series, fuzzy data, in [2,7,11,34] it was applied for interval-valued data and in [38] its connection with spectral clustering was proposed. However, one of its greatest disadvantages is its sensitivity to noise and outliers. In the presence of outliers and/or noise, the computed cluster centers can be placed away from their true values. It results from application of the L_2 norm as a dissimilarity measure. Although it allows for an elegant analytical solution, which requires low computational effort, it is not robust against outliers.

However, since the real data are usually noisy and contain outliers, the clustering methods need to be robust. According to Huber [20], a robust method should have the following properties: (i) it should have a reasonably good accuracy at the assumed model, (ii) small deviations from the model assumptions should impair the performance only by a small amount, (iii) larger deviations from the model assumptions should not cause a catastrophe. In literature there are many robust estimators. In this paper the Huber's M-estimators are of special interest [20]. The robust estimation is related to a strictly theoretical concept of a so-called breakdown point. It is the smallest fraction of outliers that, in the worst case, can cause the estimator catastrophe [5]. The adaptation of the concept of breakdown point to a single cluster, called dissolution point, is presented in [18]. In this paper an idea of isolation robustness is introduced. It is the smallest fraction of outliers that, in the worst case, can join arbitrarily well separated clusters.

Traditionally the 'goodness' of fitting the data to the prototypes in fuzzy clustering can be measured by the least sum of squares criterion, which leads to weighted arithmetic mean, used to aggregate the fit measure for all data points. In 1988 R. Yager proposed Ordered Weighted Averaging (OWA) [35]. In this approach the importance of each of the aggregated points depends on its position after the ordering operation. The OWA class of operators includes for example: min, max, arithmetic mean and median. An overview of the aggregation operators can be found in [15]. Therefore, in general there are two approaches to obtain robustness against outliers: the M-estimators by Huber and the OWA operators by Yager. In 2005 OWA was used as a robust estimator of a location parameter to determine the baseline drift of a biomedical signal in a moving time window [28]. More recently, Yager has proposed OWA-based regression using the power function of the residuals [36].

In literature devoted to fuzzy clustering there is a number of approaches to reduce the effect of outliers, including the Possibilistic Clustering (PC) [23], the fuzzy Noise Clustering Method (NCM) [4], L_p norm clustering $(0 < p < 1)$ (L_p FCM) [17], L_1 norm clustering (L_1 FCM) [21,22], the Fuzzy Clustering with Polynomial Fuzzifier (PFCM) [33] and ε-insensitive loss function clustering (βFCM) [24]. However, the approach based on the OWA operators has not been used for this purpose, yet. Nevertheless, some applications of ordering operation in data clustering have been proposed. In [38] elements of individual columns of a partition matrix are ordered to obtain the fuzzy similarity between data points. In a method called fuzzy clustering with polynomial fuzzifier (PFCM), for each data point its distances to the prototypes are ordered [33]. In the above-mentioned works ordering operation has not been used to achieve resistance to outliers. In this paper, by contrast, for each prototype the data points will be ordered with respect to their distances from the prototype.

It should be mentioned that the robust clustering methods can also used for fuzzy data. An overview of these methods can be found in [10]. They can be divided into the ones that are based on [10]: possibilistic theory, noise approach, metric approach, trimmed approach, influence weighting, semifuzzy approach, evidential theory and order statistics. Despite the fact that the methods introduced in the paper are intended for crisp data, the proposed connection of the Huber's M-estimators and the Yager's OWA operators can also be used together for fuzzy data.

The goal of this paper is to show that the Huber's M-estimators and the Yager's OWA operators can be used together in fuzzy clustering to significantly improve its robustness. In other words, the paper combines the fuzzy c-means clustering with the robust ordered statistics using Huber's M-estimator to develop a new fuzzy clustering method called as Fuzzy C-Ordered-Means clustering (FCOM). The second goal of this work is to investigate the performance of the proposed method when applied to data in the presence of noise and outliers.

This paper is organized as follows: Section 2 presents a short description of clustering methods based on criterion function minimization. The novel clustering algorithms are described in Sections 3 to 6. In Section 7 their investigations on synthetic data with outliers and on data with heavy-tailed groups of points are performed, with the fuzzy

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

c-means, the possibilistic clustering, the fuzzy noise clustering, the L_p norm clustering ($0 < p < 1$), the L_1 norm clustering, the fuzzy clustering with polynomial fuzzifier and the fuzzy ε-insensitive c-means as the reference methods. Finally, conclusions are drawn in Section 8.

2. Clustering by criterion function minimization

A very popular way of data clustering is to define a criterion function (scalar index) that measures the quality of a partition. In fuzzy approach [1] the set of all possible fuzzy partitions of N vectors (p-dimensional) into c clusters is defined by:

$$\mathcal{J}_{fc} = \left\{ \mathbf{U} \in \mathbb{R}_{cN} \left| \underset{\substack{1\le i\le c \\ 1\le k\le N}}{\forall} \; u_{ik} \in [0,1];\; \sum_{i=1}^{c} u_{ik} = 1;\; 0 < \sum_{k=1}^{N} u_{ik} < N \right. \right\}. \tag{1}$$

$\mathbb{R}_{cN}$ denote a space of real $(c \times N)$-dimensional matrices. The fuzzy c-means criterion function has the form [1]:

$$J_m(\mathbf{U}, \mathbf{V}) = \sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik})^m d_{ik}^2, \tag{2}$$

where $\mathbf{U} \in \mathcal{J}_{fc}$, $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_c] \in \mathbb{R}_{pc}$ is a matrix of prototypes and m is a weighting exponent in $[1, \infty)$. The d_{ik} is the inner product induced norm:

$$d_{ik}^2 = \|\mathbf{x}_k - \mathbf{v}_i\|_{\mathbf{A}}^2 = (\mathbf{x}_k - \mathbf{v}_i)^T \mathbf{A} (\mathbf{x}_k - \mathbf{v}_i), \tag{3}$$

where $\mathbf{A}$ is a positive definite matrix. Criterion (2) for $m = 2$ was introduced by Dunn [14]. An infinite family of fuzzy c-means criterions for $m \in [1, \infty)$ was introduced by Bezdek. Using Lagrange multipliers the following theorem can be proved, via obtaining necessary conditions for minimization of (2) [1]:

Theorem 1. *If m and c are fixed parameters, and $I_k, \widetilde{I}_k$ are sets defined as:*

$$\underset{1\le k\le N}{\forall} \quad \begin{cases} I_k = \{i \,|\, 1 \le i \le c;\; d_{ik} = 0\}, \\ \widetilde{I}_k = \{1, 2, \ldots, c\} \setminus I_k, \end{cases} \tag{4}$$

then $(\mathbf{U}, \mathbf{V}) \in (\mathcal{J}_{fc} \times \mathbb{R}_{pc})$ may be globally minimal for $J_m(\mathbf{U}, \mathbf{V})$ only if:

$$\underset{\substack{1\le i\le c \\ 1\le k\le N}}{\forall} \quad u_{ik} = \begin{cases} (d_{ik})^{\frac{2}{1-m}} / [\sum_{j=1}^{c} (d_{jk})^{\frac{2}{1-m}}], & I_k = \emptyset, \\ \begin{cases} 0, & i \in \widetilde{I}_k, \\ [0,1] \; s.t. \; \sum_{i \in I_k} u_{ik} = 1, & i \in I_k, \end{cases} & I_k \neq \emptyset, \end{cases} \tag{5}$$

and

$$\underset{1\le i\le c}{\forall} \quad \mathbf{v}_i = \left[\sum_{k=1}^{N} (u_{ik})^m \mathbf{x}_k \right] \Big/ \left[\sum_{k=1}^{N} (u_{ik})^m \right]. \tag{6}$$

The final partition is a fixed point of (5) and (6), and the solution is obtained from the Picard algorithm. This algorithm is called Fuzzy C-Means (FCM), and can be described as

Algorithm 1.

Step 1. Fix c $(1 < c < N)$, $m \in (1, \infty)$. Initialize $\mathbf{V}^{[0]} \in \mathbb{R}_{pc}$, $j = 1$,
Step 2. Calculate the fuzzy partition matrix $\mathbf{U}^{[j]}$ for j-th iteration using (5),
Step 3. Update the centers for j-th iteration $\mathbf{V}^{[j]} = [\mathbf{v}_1^{[j]}, \mathbf{v}_2^{[j]}, \ldots, \mathbf{v}_c^{[j]}]$ using (6) and $\mathbf{U}^{[j]}$,
Step 4. If $\|\mathbf{V}^{[j]} - \mathbf{V}^{[j-1]}\|_{\mathrm{F}} > \xi$ then $j \leftarrow j + 1$ and go to Step 2 else stop.

$\|\cdot\|_{\mathrm{F}}$ denotes Frobenius norm ($\|\mathbf{V}\|_{\mathrm{F}}^2 = Tr(\mathbf{V}\mathbf{V}^{\top}) = \sum_i \sum_l v_{il}^2$) and ξ is a pre-set parameter. The parameter m influences a fuzziness of the clusters; usually $m = 2$ is chosen.

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

3. Fuzzy c-ordered-means clustering with various dissimilarity measures

The clustering algorithm recalled in the previous section uses a quadratic loss function as a dissimilarity measure between the data and the clusters centers. The reason of using this measure is mathematical, that is, for simplicity and low computational burden. However, this approach is sensitive to noise and outliers. In literature there are many proposals of robust loss functions. For example, the Huber's one is of special interest [20]:

$$\mathcal{L}_{\mathrm{HUB}}(e) = \begin{cases} e^2/\delta^2, & |e| \le \delta, \\ |e|/\delta, & |e| > \delta, \end{cases} \tag{7}$$

where $\delta > 0$ denotes a parameter. Another well-known robust loss function is the logarithmic function:

$$\mathcal{L}_{\mathrm{LOG}}(e) = \begin{cases} 0, & e = 0, \\ \log(1+e^2), & e \neq 0. \end{cases} \tag{8}$$

Using $\mathcal{D}(\mathbf{x}_k, \mathbf{v}_i) = \mathcal{L}(\mathbf{x}_k - \mathbf{v}_i)$ as a dissimilarity measure between the kth datum and the ith prototype, and additional weighting, the fuzzy c-means criterion function (2) takes the form:

$$J(\mathbf{U}, \mathbf{V}) = \sum_{i=1}^{c} \sum_{k=1}^{N} \beta_{ik} (u_{ik})^m \, \mathcal{D}(\mathbf{x}_k, \mathbf{v}_i), \tag{9}$$

where

$$\mathcal{D}(\mathbf{x}_k, \mathbf{v}_i) = \sum_{l=1}^{p} \mathcal{D}(x_{kl}, v_{il}). \tag{10}$$

Now, the set of all possible fuzzy partitions of N vectors (p-dimensional) into c clusters is defined by:

$$\mathcal{J}_{gfc} = \left\{ \mathbf{U} \in \mathbb{R}_{cN} \,\middle|\, \underset{\substack{1 \le i \le c \\ 1 \le k \le N}}{\forall} \; u_{ik} \in [0,1]; \; \sum_{i=1}^{c} \beta_{ik} \, u_{ik} = f_k; \; 0 < \sum_{k=1}^{N} u_{ik} < N \right\}, \tag{11}$$

where $\beta_{ik} \in [0, 1]$ denotes the typicality of the kth datum with respect to the ith cluster; smaller β_{ik} results in a more atypical data. These parameters βs are derived based on the ordering the distances of data from prototypes; for more detailed description see the further part of this work. The $\mathcal{J}_{gfc}$ set is similar to that used in the conditional fuzzy clustering, introduced by Pedrycz in [30] and generalized in [26]. Let us now explain the meaning of f_k parameters from the equality constraints in (11). The f_k parameter can be interpreted as an overall (general) typicality of the kth datum, which depends on typicality of the kth datum with respect to all the clusters. The overall assessment of the typicality of the kth datum is obtained using s-norm S ($\star_S$):

$$\underset{1 \le k \le N}{\forall} \quad f_k = \beta_{1k} \star_S \beta_{2k} \star_S \cdots \star_S \beta_{ck}, \tag{12}$$

which may be linguistically interpreted as the following sentence: "The kth datum is typical IF AND ONLY IF the kth datum is typical with respect to the first cluster OR the kth datum is typical with respect to the second cluster OR $\cdots$ OR the kth datum is typical with respect to the cth cluster". In this paper, the maximum as the s-norm is arbitrarily chosen

$$\underset{1 \le k \le N}{\forall} \quad f_k = \beta_{1k} \vee \beta_{2k} \vee \cdots \vee \beta_{ck}. \tag{13}$$

Appendix A shows that the necessary conditions for minimization of (9) with respect to the elements of the partition matrix can be described as

$$\underset{\substack{1 \le k \le N \\ 1 \le s \le c}}{\forall} \quad u_{sk} = f_k \, \mathcal{D}(\mathbf{x}_k, \mathbf{v}_s)^{\frac{1}{1-m}} \Big/ \left[\sum_{j=1}^{c} \beta_{jk} \, \mathcal{D}(\mathbf{x}_k, \mathbf{v}_j)^{\frac{1}{1-m}} \right]. \tag{14}$$

A more difficult problem is to obtain necessary conditions for prototypes matrix $\mathbf{V}$. Appendix B shows that these conditions can be written in the following form:

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

$$\underset{\substack{1\le i\le c\\ 1\le l\le p}}{\forall} \quad v_{il} = \left[\sum_{k=1}^{N} \beta_{ik}\,(u_{ik})^m h_{ikl}\,x_{kl}\right] \Big/ \left[\sum_{k=1}^{N} \beta_{ik}\,(u_{ik})^m h_{ikl}\right], \tag{15}$$

where h_{ikl} parameters depend on the loss function used (see (B.5)–(B.10)) and residuals. We see that, v_{il} depends on h_{ikl} parameters, but the values of h_{ikl} depend on the obtained residuals $e_{ikl} = x_{kl} - v_{il}$. In turn, the residuals depend on v_{il}. Consequently, criterion function (9) should only be minimized with respect to the prototypes by iteratively reweighting scenario. In the next section, a method of the prototypes location estimation will be described.

4. Fuzzy ordered estimator of location

In this section an algorithm for estimating the location of the lth component of the ith prototype (v_{il}) using iteratively reweighting scenario is presented. Let us denote h_{ikl}, e_{ikl} and v_{il} in the rth iteration as $h_{ikl}^{[r]}$, $e_{ikl}^{[r]}$ and $v_{il}^{[r]}$, respectively. Let $\pi : \{1, 2, \ldots, N\} \to \{1, 2, \ldots, N\}$ be the permutation function.[1] The rank-ordered residuals satisfy the following conditions:

$$\left|e_{i\pi(1)l}^{[r-1]}\right| \le \left|e_{i\pi(2)l}^{[r-1]}\right| \le \left|e_{i\pi(3)l}^{[r-1]}\right| \le \cdots \le \left|e_{i\pi(N)l}^{[r-1]}\right|. \tag{16}$$

Criterion function (B.3) for the rth iteration takes the form

$$g_{il}^{[r]}\left(v_{il}^{[r]}\right) = \sum_{k=1}^{N} \alpha_k\,(u_{i\pi(k)})^m\, h_{i\pi(k)l}^{[r]}\left(e_{i\pi(k)l}^{[r]}\right)^2, \tag{17}$$

where $e_{ikl}^{[r]} = x_{kl} - v_{il}^{[r]}$. The β_{ik} parameters were replaced by parameters α_k. The above criterion is similar to OWA operator [35], with additional weighting and the quadratic function. Let us now explain the meaning of α_k parameters. The α parameters depend on the order of the residuals in the previous ($[r-1]$th) iteration. If α_k parameters fulfill $\alpha_1 \ge \alpha_2 \ge \cdots \ge \alpha_N$, then it is clear that the impact of outliers is reduced by down-weighting the respective residuals. The form of parameters α_k is proposed to be piecewise-linear

$$\alpha_k = \left\{\left[(p_c N - k)/(2p_l N) + 0.5\right] \wedge 1\right\} \vee 0 \tag{18}$$

or sigmoidal

$$\alpha_k = 1 \Big/ \left\{1 + \exp\left[\frac{2.944}{p_a N}(k - p_c N)\right]\right\}, \tag{19}$$

where $\wedge$ and $\vee$ denote *min* and *max* operations, respectively. Both functions, which may be called the weighting functions, are nonincreasing with respect to argument $k \in \{1, 2, \ldots, N\}$. For $k = p_c N$ both functions are equal to 0.5. Parameters $p_l > 0$ and $p_a > 0$ influence their slope. In the case of the piecewise-linear function, for $k \in [p_c N - p_l N, p_c N + p_l N]$ its value linearly decreases from 1 to 0. For the sigmoidal function, for $k \in [p_c N - p_a N, p_c N + p_a N]$ its value decreases from 0.95 to 0.05. In the rest of the work, the functions defined by (18) and (19) are called Sigmoidally-weighted OWA (SOWA) and Piecewise-Linearly-weighted OWA (PLOWA), respectively. If ordering of residuals is not applied, which is equivalent to using Uniformly weighting function for OWA – UOWA ($\alpha_k = 1$ for all k), then we call this case as clustering without ordering (or with no weighting function).

The disadvantage of this approach is the necessity to exchange in each iteration the order of the elements in $\mathbf{U}$ and the elements of $h_{i\pi(k)l}^{[r]}$ what is a time consuming operation. If we denote the inverse function of $\pi(k)$ as $\pi^{-1}(k)$ then (17) may be written as

$$g_{il}^{[r]}\left(v_{il}^{[r]}\right) = \sum_{k=1}^{N} \alpha_{\pi^{-1}(k)}\,(u_{i\pi^{-1}(\pi(k))})^m\, h_{i\pi^{-1}(\pi(k))l}^{[r]}\left(e_{i\pi^{-1}(\pi(k))l}^{[r]}\right)^2, \tag{20}$$

Using the identity $\pi^{-1}(\pi(k)) = k$ the above sum equals

[1] The most formally the permutation function depends on i, l indexes, and iteration index r. Thus, we should used $\pi_{il}^{[r]}(k)$. For the sake of simplicity, all these indexes at the permutation function will be temporarily omitted.

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

$$g_{il}^{[r]}\left(v_{il}^{[r]}\right)=\sum_{k=1}^{N}\alpha_{\pi^{-1}(k)}\,(u_{ik})^m\,h_{ikl}^{[r]}\left(e_{ikl}^{[r]}\right)^2=\sum_{k=1}^{N}\breve{\alpha}_k\,(u_{ik})^m\,h_{ikl}^{[r]}\left(e_{ikl}^{[r]}\right)^2, \tag{21}$$

where $\breve{\alpha}_k=\alpha_{\pi^{-1}(k)}$. In sequel, we obtain

$$\breve{\alpha}_{\pi(k)}=\alpha_{\pi(\pi^{-1}(k))}=\alpha_k. \tag{22}$$

Thus, if we have the permutation function π obtained by rank-ordering the residuals, the α_k parameters can easily be calculated using (22) and either (18) or (19). Finally, if we denote the permutation function for the rth iteration as $\pi^{[r]}(k)$, then according to the above result (15) should be replaced by

$$\underset{\substack{1\le i\le c\\1\le l\le p}}{\forall}\quad v_{il}^{[r]}=\left[\sum_{k=1}^{N}\breve{\alpha}_k\,(u_{ik})^m h_{ikl}^{[r]}\,x_{kl}\right]\Big/\left[\sum_{k=1}^{N}\breve{\alpha}_k\,(u_{ik})^m h_{ikl}^{[r]}\right]. \tag{23}$$

where

$$h_{ikl}^{[r]}=\begin{cases}0, & x_{kl}-v_{il}^{[r-1]}=0,\\ \mathcal{L}(x_{kl}-v_{il}^{[r-1]})/(x_{kl}-v_{il}^{[r-1]})^2, & x_{kl}-v_{il}^{[r-1]}\neq 0.\end{cases} \tag{24}$$

On the basis of the above considerations, we obtain an iterative algorithm which can be called as iteratively weighted (or fuzzy) ordered estimator of location v_{il},

Algorithm 2.

Step 1. Initialize $v_{il}^{[0]}=0$. Set the iteration index $r=1$,

Step 2. Calculate residuals $e_{ikl}^{[r-1]}=x_{kl}-v_{il}^{[r-1]}$ and then the coefficients

$$h_{ikl}^{[r]}=\begin{cases}0, & e_{ikl}^{[r-1]}=0\\ \mathcal{L}(e_{ikl}^{[r-1]})/(e_{ikl}^{[r-1]})^2, & e_{ikl}^{[r-1]}\neq 0\end{cases},\quad \text{for } k=1,2,\ldots,N,$$

Step 3. Rank-order the residuals $|e_{i\pi(1)l}^{[r-1]}|\le|e_{i\pi(2)l}^{[r-1]}|\le|e_{i\pi(3)l}^{[r-1]}|\le\cdots\le|e_{i\pi(N)l}^{[r-1]}|$ obtaining the permutation function $\pi(k)$,

Step 4. Calculate $\breve{\alpha}_{\pi(k)}=\alpha_k$ using (18) or (19) or uniform weighting,

Step 5. Update the centers for the rth iteration using (23),

Step 6. If $\|v_{il}^{[r]}-v_{il}^{[r-1]}\|_2^2>\xi$ then $r\leftarrow r+1$ and go to Step 2 else $\beta_{ikl}=\breve{\alpha}_k$, stop.

Remarks. The algorithm enables to obtain the location of the lth component of the ith prototype (v_{il}) and the typicality parameters β_{ikl} for $k=1,2,\ldots,N$ for dissimilarity measures $\mathcal{L}$ used. In Step 6 of the algorithm after the iteration is stopped, on the basis on the permutation function we obtain typicality β_{ikl} of the lth component of the kth datum with respect to estimated location of the lth component of the ith prototype.

5. Fuzzy c-ordered-means clustering algorithm

The effects of Algorithm 2, which is introduced in the previous section, for indices il are: the estimated location of the lth component of the ith prototype (v_{il}) and the parameters β_{ikl} for $k=1,2,\ldots,N$. Let us denote β_{ikl} as typicality of the lth component of the kth datum with respect to estimated location of the lth component of the ith prototype. The overall assessment of the typicality of the kth datum to the ith prototype is obtained using t-norm T ($\star_T$):

$$\underset{\substack{1\le i\le c\\1\le k\le N}}{\forall}\quad \beta_{ik}=\beta_{ik1}\star_T\beta_{ik2}\star_T\cdots\star_T\beta_{ikp}, \tag{25}$$

which may be linguistically interpreted as the following sentence: "The kth datum is typical with respect to the ith cluster IF AND ONLY IF the first component of the kth datum is typical with respect to the first component of the ith cluster AND the second component of the kth datum is typical with respect to the second component of the ith cluster AND $\cdots$ AND the pth component of the kth datum is typical with respect to the pth component of the ith cluster". In this paper, the algebraic product as t-norm is arbitrarily chosen:

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

$$\forall_{\substack{1\le i\le c\\ 1\le k\le N}} \quad \beta_{ik} = \beta_{ik1}\cdot\beta_{ik2}\cdot\ldots\cdot\beta_{ikp}. \tag{26}$$

Thus, the degree of typicality is computed once after the ith prototype location has been determined. In this case, in each particular iteration of the algorithm, the impact of outliers is reduced on the basis of the information from one component of the data only. To use the information from all data components the following modification is proposed. Let us define: $\pi_{il}^{[r]}(k)$ – the permutation function for the rth iteration, the lth component of the ith prototype; $\breve{\alpha}_{ilk}$ – the kth datum weight for estimation of the lth component of the ith prototype.

Now, we obtain an algorithm (which replaces Algorithm 2) that can be called modified iteratively weighted (or fuzzy) ordered estimator of location $\mathbf{v}_i$,

Algorithm 3.

Step 1. Initialize $\mathbf{v}_i^{[0]} = \mathbf{0}$. Set the iteration index $r = 1$,
Step 2. Calculate residuals $e_{ikl}^{[r-1]} = x_{kl} - v_{il}^{[r-1]}$ and then the coefficients $h_{ikl}^{[r]}$, for $k = 1, 2, \ldots, N$ and $l = 1, 2, \ldots, p$ using (24),
Step 3. Rank-order the residuals $|e^{[r-1]}_{i\pi_{il}^{[r-1]}(1)l}| \le |e^{[r-1]}_{i\pi_{il}^{[r-1]}(2)l}| \le |e^{[r-1]}_{i\pi_{il}^{[r-1]}(3)l}| \le \cdots \le |e^{[r-1]}_{i\pi_{il}^{[r-1]}(N)l}|$ obtaining the permutation functions $\pi_{il}^{[r-1]}(k)$ for $l = 1, 2, \ldots, p$,
Step 4. Calculate $\breve{\alpha}_{il\pi_{il}^{[r-1]}(k)} = \alpha_k$ using (18) or (19) or uniform weighting,
Step 5. Calculate $\breve{\alpha}_k = \prod_{l=1}^{p} \breve{\alpha}_{ilk}$,
Step 6. Update the prototype for the rth iteration using (23) for $l = 1, 2, \ldots, p$,
Step 7. If $\|\mathbf{v}_i^{[r]} - \mathbf{v}_i^{[r-1]}\|_2^2 > \xi$ then $r \leftarrow r + 1$ and go to Step 2 else $\beta_{ik} = \breve{\alpha}_k$, stop.

The effects of the above modified algorithm, for index i, are: the estimated location of the all components of the ith prototype $\mathbf{v}_i$ and the parameter β_{ik}, which is a typicality of the kth datum with respect to the estimated location of the ith prototype.

Ultimately, the final form of the proposed Fuzzy C-Ordered-Means clustering (FCOM) can be described

Algorithm 4.

Step 1. Fix c $(1 < c < N)$, $m \in (1, \infty)$. Choose dissimilarity measure. Initialize $\mathbf{V}^{(0)} \in \mathbb{R}_{pc}$, $\beta_{ik} = 1$, $f_k = 1$ and set the iteration index $j = 1$,
Step 2. Calculate the fuzzy partition matrix $\mathbf{U}^{(j)}$ for j-th iteration using (14),
Step 3. Update the centers for j-th iteration $\mathbf{V}^{(j)} = [\mathbf{v}_1^{(j)}, \mathbf{v}_2^{(j)}, \ldots, \mathbf{v}_c^{(j)}]$ using $\mathbf{U}^{(j)}$ and Algorithm 3; (if $j \le 4$ then use uniform weighting UOWA, else either (18) or (19) or UOWA),
Step 4. Update overall typicality parameters f_k using (13),
Step 5. If $\|\mathbf{V}^{(j+1)} - \mathbf{V}^{(j)}\|_{\mathrm{F}} > \xi$ then $j \leftarrow j + 1$ and go to Step 2 else stop.

Remarks. The iterations are stopped as soon as the Frobenius norm in a successive pair of $\mathbf{V}$ matrices is less than ξ where ξ is a pre-set small positive value. In all experiments $\xi = 10^{-5}$ is used. If the initial prototypes are far from their true positions, even the 'good' points can be distant to these prototypes, and as a result can unjustifiably be regarded as outliers. Therefore the condition was added in Step 3 to avoid the ordering operations in the first few iterations.

6. Fuzzy c-ordered-means clustering with ε-insensitive loss functions

It is well known in machine learning that too precise learning on a training set can lead to the so-called overfitting, and in consequence to poor generalization ability (poor performance on previously unseen data points). Tolerating small errors while fitting on a given dataset can improve correctness on the test dataset [25,27,37]. Motivated by the results of statistical learning theory, Vapnik introduced the ε-insensitive loss function. This function disregards errors below some $\varepsilon > 0$, chosen a priori:

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

$$\mathcal{L}_{\varepsilon\mathrm{LIN}}(e) = \begin{cases} 0, & |e| \le \varepsilon, \\ |e| - \varepsilon, & |e| > \varepsilon. \end{cases} \tag{27}$$

Various ε-insensitive loss functions may be considered, including ε-insensitive quadratic, ε-insensitive Huber, and so on. Let us start our considerations from the ε-insensitive quadratic loss

$$\mathcal{L}_{\varepsilon\mathrm{SQR}}(e) = \begin{cases} 0, & |e| - \varepsilon \le 0, \\ (\varepsilon - e)^2, & \varepsilon - e < 0, \\ (\varepsilon + e)^2, & \varepsilon + e < 0. \end{cases} \tag{28}$$

Taking into account the above equation, (B.2) for all $i = 1, 2, \ldots, c;\ l = 1, 2, \ldots, p$, may be written as

$$\begin{aligned} g_{il}(v_{il}) &= \sum_{k=1}^{N} \beta_{ik}\, (u_{ik})^m\, \mathcal{L}_{\varepsilon\mathrm{SQR}}(x_{kl} - v_{il}) \\ &= \sum_{k=1}^{N} \beta_{ik}\, (u_{ik})^m h_{ikl}^{-} (\varepsilon - x_{kl} + v_{il})^2 \\ &\quad + \sum_{k=1}^{N} \beta_{ik}\, (u_{ik})^m h_{ikl}^{+} (\varepsilon + x_{kl} - v_{il})^2, \end{aligned} \tag{29}$$

where h_{ikl}^{-} (h_{ikl}^{+}) are equal to zero for $\varepsilon - x_{kl} + v_{il} \ge 0$ ($\varepsilon + x_{kl} - v_{il} \ge 0$) and 1 otherwise.

Appendix C shows that for ε-insensitive dissimilarity measures the necessary conditions for minimization of criterion (9) with respect to the prototypes can be expressed in the form similar to (23)

$$\underset{\substack{1\le i\le c \\ 1\le l\le p}}{\forall} \quad v_{il}^{[r]} = \left[\sum_{k=1}^{2N} \breve{\alpha}_k\, (u_{ik})^m h_{ikl}^{[r]}\, {}_e x_{kl}\right] \Big/ \left[\sum_{k=1}^{2N} \breve{\alpha}_k\, (u_{ik})^m h_{ikl}^{[r]}\right]. \tag{30}$$

where ${}_e x_{kl}$, $k = 1, 2, \ldots, 2N$ denote the extended data, defined as follows: ${}_e x_{kl} = x_{kl} - \varepsilon$ for $k = 1, 2, \ldots, N$ and ${}_e x_{kl} = x_{(k-N)l} + \varepsilon$ for $k = N+1, N+2, \ldots, 2N$,

The $\breve{\alpha}_k$ parameters should be calculated in a different way, if compared to the one presented in the previous sections. Let a distance from the insensibility zone be $q_{ikl}^{[r-1]} = -({}_e e_{ikl}^{[r-1]} \wedge {}_e e_{i(N+k)l}^{[r-1]} \wedge 0)$ for $k = 1, 2, \ldots, N$ and let the permutation function for the rank-ordered q_{ikl}s be $\pi^{[r-1]}$ for $[r-1]$th iteration. Let $N_\gamma^{[r-1]}$ denote the number of q_{ikl}s equal to zero, i.e., $q_{i\pi^{[r-1]}(k)l} = 0$ for $k = 1, 2, \ldots, N_\gamma^{[r-1]}$. For the rth iteration $\breve{\alpha}_{\pi^{[r-1]}(k)}^{[r]} = 1$ for $k = 1, 2, \ldots, N_\gamma^{[r-1]}$ and takes a form of piecewise-linear (18) or sigmoidal (19) function for $k = N_\gamma^{[r-1]} + 1, N_\gamma^{[r-1]} + 2, \ldots, N$:

$$\breve{\alpha}_{\pi^{[r-1]}(k)}^{[r]} = \begin{cases} 1, & k = 1, 2, \ldots, N_\gamma^{[r-1]}, \\ \alpha_{\frac{(N-1)k - N N_\gamma^{[r-1]}}{N - N_\gamma^{[r-1]} - 1}}, & k = N_\gamma^{[r-1]} + 1, N_\gamma^{[r-1]} + 2, \ldots, N. \end{cases} \tag{31}$$

Indeed, the $\breve{\alpha}_k$ parameters should also be 'doubled', i.e., $\breve{\alpha}_{N+k} = \breve{\alpha}_k$, for $k = 1, 2, \ldots, N$. Now, we ready to describe an algorithm that can be called modified iterative weighted (or fuzzy) ordered estimator of location $\mathbf{v}_i$ with ε-insensitive loss,

Algorithm 5.

Step 1. Initialize $v_{il}^{[0]} = 0$. Set the iteration index $r = 1$,
Step 2. Calculate residuals ${}_e e_{ikl}^{[r-1]}$ and then the coefficients $h_{ikl}^{[r]}$, for $k = 1, 2, \ldots, 2N$ and $l = 1, 2, \ldots, p$ using (C.6),
Step 3. Rank-order the distances $q_{ikl}^{[r-1]}$ from the insensibility zone to obtain the permutation functions $\pi_{il}^{[r-1]}(k)$ for $l = 1, 2, \ldots, p$,
Step 4. Calculate $\breve{\alpha}_{il\pi_{il}^{[r-1]}(k)}$ using (31), and UOWA or PLOWA or SOWA weighting function,
Step 5. Calculate $\breve{\alpha}_k = \prod_{l=1}^{p} \breve{\alpha}_{ilk}$,
Step 6. Update the prototype for rth iteration using (30) for $l = 1, 2, \ldots, p$,
Step 7. If $\|\mathbf{v}_i^{[r]} - \mathbf{v}_i^{[r-1]}\|_2^2 > \xi$ then $r \leftarrow r + 1$ and go to Step 2 else $\beta_{ik} = \breve{\alpha}_k$, stop.

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

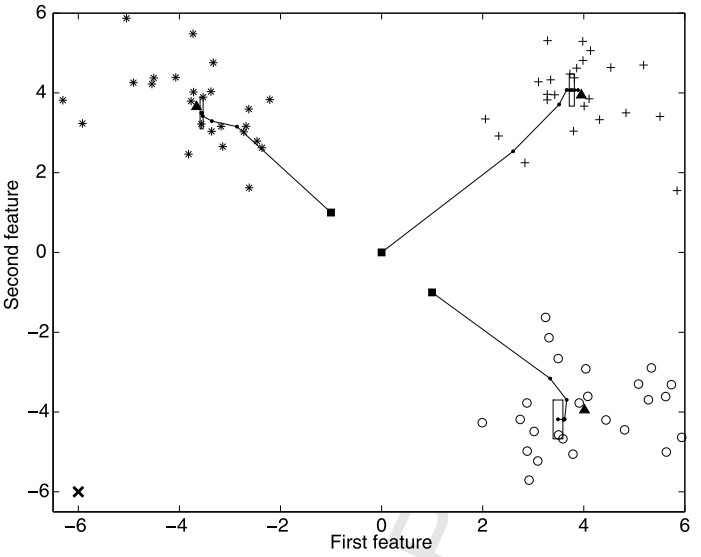

Fig. 1. Performance of the βFCM method with $\beta = 0.8$ for 15 outliers.

Finally, the clustering algorithm named Fuzzy C-Ordered-Means clustering (FCOM) with ε-insensitive loss function can be described

Algorithm 6.

Step 1. Fix c $(1 < c < N)$, $m \in (1, \infty)$. Choose ε-insensitive dissimilarity measure. Initialize $\mathbf{V}^{(0)} \in \mathbb{R}_{pc}$, $\beta_{ik} = 1$, $f_k = 1$ and set the iteration index $j = 1$,

Step 2. Calculate the fuzzy partition matrix $\mathbf{U}^{(j)}$ for j-th iteration using (14),

Step 3. Update the centers for j-th iteration $\mathbf{V}^{(j)} = [\mathbf{v}_1^{(j)}, \mathbf{v}_2^{(j)}, \ldots, \mathbf{v}_c^{(j)}]$ using $\mathbf{U}^{(j)}$ and Algorithm 5 (if $j \leq 4$ then use uniform weighting UOWA, else either (18), (31) or (19), (31) or UOWA),

Step 4. Update parameters f_k using (13),

Step 5. If $\|\mathbf{V}^{(j+1)} - \mathbf{V}^{(j)}\|_{\mathrm{F}} > \xi$ then $j \leftarrow j + 1$ and go to Step 2 else stop.

7. Numerical experiments

In this section the proposed method is compared to the following reference ones: the Fuzzy C-Means (FCM), the Possibilistic Clustering (PC), the fuzzy Noise Clustering Method (NCM), the L_p norm clustering (L_p FCM) $(0 < p < 1)$, the L_1 norm clustering (L_1 FCM), the Fuzzy Clustering with Polynomial Fuzzifier (PFCM) and the ε-insensitive Fuzzy C-Means (βFCM). To this end experiments on synthetic data with outliers have been performed as well as on data with heavy-tailed and overlapping groups of points in background noise. In all experiments for FCM, NCM, PCM, PFCM, L_1 FCM, L_p FCM ($p = 0.5$), βFCM and FCOM the weighting exponent $m = 2$ was used. The iterations were stopped as soon as the Frobenius norm of the successive $\mathbf{V}$ matrices difference was less than 10^{-4} for FCM, NCM, PCM, PFCM, L_1 FCM, L_p FCM and FCOM, and 10^{-2} for βFCM. The following values: $\varepsilon = 0.5$, $\alpha = 6.0$, $\beta = 1.0$, $\delta = 1.0$ were used for loss functions (C.7)–(C.13), and $p_c = 0.5$, $p_l = 0.2$, $p_a = 0.2$ for weighting functions (18), (19). For the NCM [4] $\delta = 0.5$ was used. For the PCM method η_i's were estimated using (9) from [23] ($K = 1$). For the PFCM method [33] β equals 0.5. For the computed terminal prototypes, we measured the performance of clustering by the Frobenius norm of the difference between the true centers matrix and the terminal prototypes matrix. All experiments were run in the MATLAB environment. The linear optimization with constraints in βFCM was performed using the MATLAB 'linprog' procedure.

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

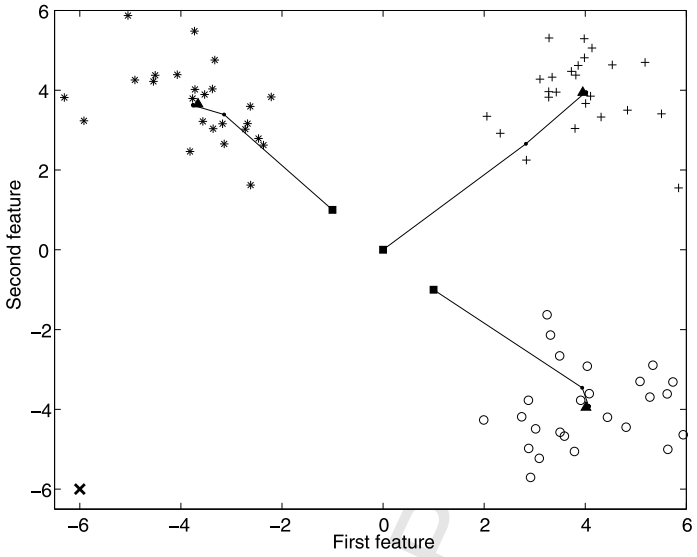

Fig. 2. Performance of the FCOM method with the quadratic loss function and the SOWA ordering function for 25 outliers.

7.1. Synthetic data with varying number of outliers

The purpose of this experiment was to investigate the sensitivity to outliers of the FCM, NCM, PCM, PFCM, L_1 FCM, L_p FCM, βFCM and the FCOM methods. The two-dimensional (two features vectors) data set, presented in Fig. 1, consists of three well-separated groups (each of 25 points), and a varying number of outliers located at the point $(-6, -6)$. The number of outliers varies from 0 (no outliers) to 30 (the number of outliers exceeding the cardinality of the clusters). The true clusters centers, calculated without outliers, are marked by triangles. The tested methods were initialized using prototypes: $(-1, 1)$, $(0, 0)$ and $(1, -1)$, marked by squares on Fig. 1. Like as in [24] $\beta = 0.8$ was used for the βFCM method.

Fig. 1 illustrates the performance of the βFCM method for 15 outliers and $\beta = 0.8$. In this figure we can observe the traces of the prototypes calculated in the successive iterations. We can see that the prototypes terminate near the true clusters centers, but some influence of the outliers is already noticeable. The rectangles visualize the calculated insensitivity regions. Fig. 2 illustrates the performance of the FCOM method for 25 outliers. We can notice that despite the large number of outliers the prototypes terminate near the true clusters centers. Fig. 3 illustrates the performance of NCM for 25 outliers. We can see that one of the prototypes terminates near the true cluster center, but for the other two prototypes the unfavorable influence of the outliers is significant.

The effects of the FCM, the βFCM and the FCOM methods investigation for varying number of outliers are presented in Fig. 4. It shows that for a few outliers (from 1 to 4) the terminal prototypes calculated by the FCM method are closer to the true centers than for the βFCM method. But for a greater number of outliers the terminal prototypes errors are smaller for βFCM than for the FCM method. For example, the terminal prototypes errors for βFCM and 23 outliers are comparable with the errors for FCM and 5 outliers. In the case of the FCM method the errors of the prototypes calculation suddenly increase with the grooving number of outliers. The FCM method performance is catastrophically deteriorated for 20 outliers. For the βFCM the performance is catastrophically deteriorated for the number of outliers equals to 24, that is, for the number of outliers approximately equal to the cardinality of data points in each cluster. However, even for smaller number of outliers, both for FCM and βFCM rather serious increase of errors is visible. The FCOM method without ordering has also a catastrophic deterioration of its performance for the number of outliers greater than twenty, but this number depends on the applied loss function. For example in the case of Huber's function we obtain a reasonable result for the number of outliers up to 27! The worst result is obtained for the quadratic loss function. Much better results are achieved with the use of the ordering functions. They allowed to make the errors of prototypes calculation be approximately equal for the number of outliers varying from 0 to 30! The best results were achieved for the SQR, HUB and LOG loss functions. Similar results were obtained for both ordering functions, but usually the SOWA ordering function gives a slightly better results. We can conclude that the

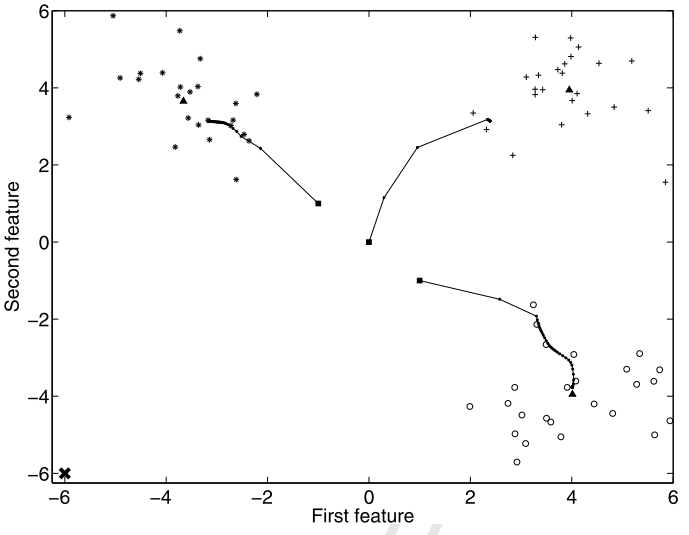

Fig. 3. Performance of NCM for 25 outliers.

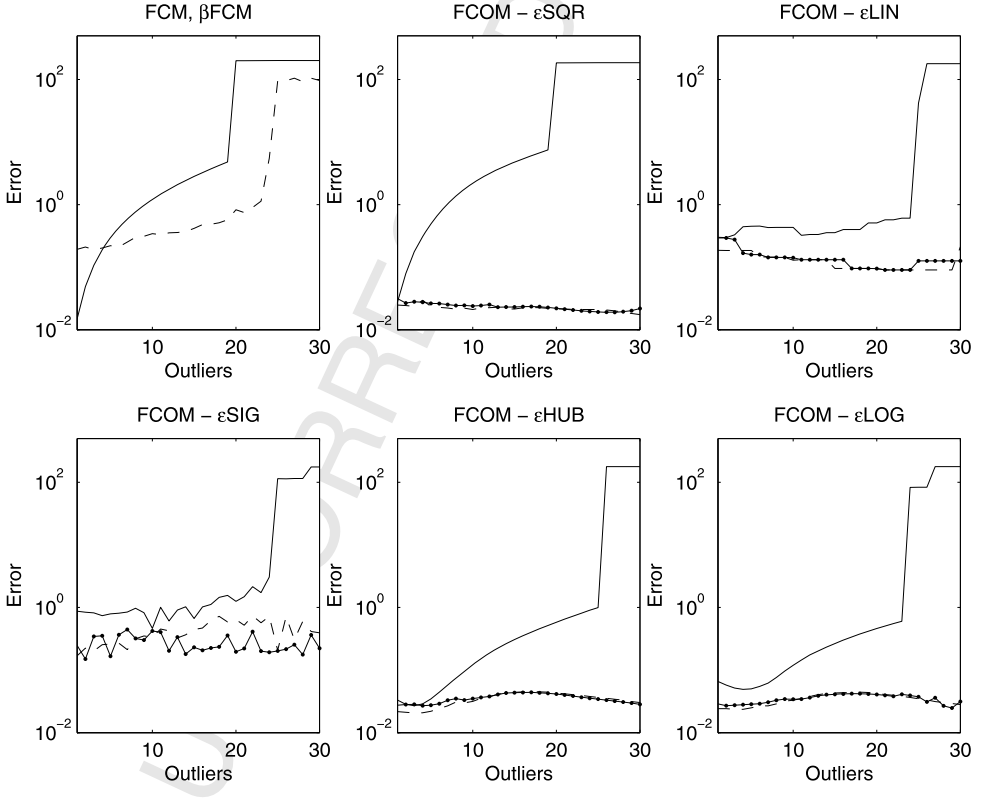

Fig. 4. The Frobenius norm of the clusters centers errors for various numbers of outliers in the synthetic data from Fig. 1. FCM and βFCM are presented in the upper left subplot with solid and dashed lines, respectively. In the other subplots FCOM without ordering, with SOWA ordering and with PLOWA ordering is presented with solid line, dashed lines and solid lines with point markers, respectively.

best performance of the FCOM method is obtained for the SQR loss function and the SOWA ordering function. The experiments show that the FCOM method leads to reasonable results even if the number of outliers is bigger than the number of data in each cluster! But we know that, in a case of approximately equal number of bad (outliers) and good data points, no method distinguishes the good points from the bad ones. The results were caused by the location of

Please cite this article in press as: J.M. Leski, Fuzzy *c*-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

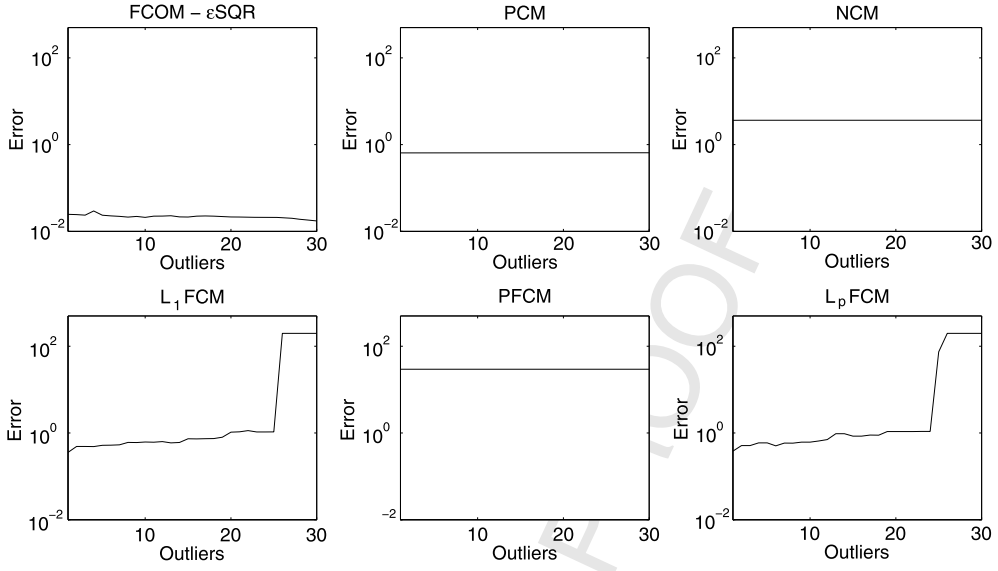

Fig. 5. The Frobenius norm of the clusters centers errors for various numbers of outliers in the synthetic data from Fig. 1. FCOM method with εSQR loss function and SOWA ordering is presented in the upper left subplot for comparison.

the initial prototypes, which were located a bit closer to the groups of data than to the outliers. This shows that for the appropriate initial prototypes, the FCOM method is resistant to the number of outliers exceeding the number of "good" data!!! Results of the PCM, NCM, L_1 FCM, PFCM, L_p FCM methods investigation for varying number of outliers are presented in Fig. 5. For L_1 FCM and L_p FCM the errors of the prototypes calculation suddenly increase with the growing number of outliers. Their performance is catastrophically deteriorated for 25 outliers, that is for the number of outliers approximately equal to the cardinality of data points in each cluster. However, even for smaller number of outliers, these methods cause errors which are comparable with those of the FCM method (see Fig. 4). For the PCM, NCM and PFCM methods extraordinary resistance to an increasing number of outliers can be observed. For these methods no influence of additional outliers is visible in the figure. However, we can notice that for this resistance these methods pay with big errors for small number (or absence) of outliers. In such conditions, the errors of these methods are much higher than those of the traditional FCM method!!! However, a robust method should have a reasonably good accuracy at the assumed model with no outliers. It should also be noted that, regardless of the number of outliers, errors are greater for these methods than for the FCOM method proposed in this work.

7.2. *Heavy-tailed and overlapping groups of data in background noise*

The purpose of this experiment was to compare the usefulness of the FCM, NCM, PCM, PFCM, L_1 FCM, L_p FCM, βFCM and the FCOM methods for clustering of heavy-tailed and overlapping groups of data. The two-dimensional data set, presented in Fig. 6, consists of two overlapping groups of points. Each group, with cardinality 50, was generated by a pseudorandom generator with the t-distribution. The inverse of the cumulative distribution function method was applied to generate the random numbers with this distribution. The true centers of the clusters are marked by triangles. The tested methods were initialized with prototypes $(-250, 800)$ and $(25, 800)$, marked by squares. The βFCM algorithm was tested with parameter β equal to 0.8. Background noise with the uniform distribution was added to the data. The cardinality of noise data was chosen as the following fraction of the "good" data: 0%, 25% (i.e. 25 noise points), 50% (i.e. 50 noise points), 75% (i.e. 75 noise points) and 100% (i.e. 100 noise points).

In Table 1 the Frobenius norm of the clusters centers deviations from the true centers is presented. Taking this table into account, several observations can be made. First of all, it should be noted that, irrespective of the fraction of the background noise, the FCOM method leads to smaller terminal prototypes errors than other reference methods. In the case when no additional background noise is added, the FCOM method also leads to better results with respect to the other methods. It also must be noted that, despite of the fraction of the background noise, the best results are obtained for the FCOM method with liner loss function and the PLOWA ordering (only slightly worse results are obtained for

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

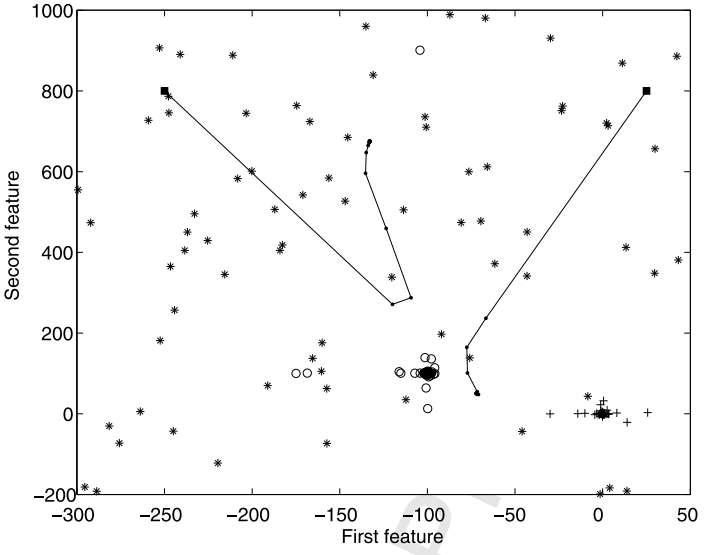

Fig. 6. Performance of the FCM method on the heavy-tailed and overlapping clusters of points in background noise. The cardinality of the noise data is 75% of the "good" data.

Table 1
The Frobenius norm of clusters centers errors for the heavy-tailed and overlapping clusters of points in background noise. The noise cardinality is given as the fraction of 'good' data.

Method		Fraction of noise data				
		0%	25%	50%	75%	100%
FCM		71.85	$5.08 \cdot 10^5$	$4.61 \cdot 10^5$	$4.77 \cdot 10^5$	$5.08 \cdot 10^5$
PCM		$2.01 \cdot 10^4$	$2.15 \cdot 10^5$	$2.37 \cdot 10^5$	$2.37 \cdot 10^5$	$2.37 \cdot 10^5$
NCM		$2.10 \cdot 10^4$	$1.27 \cdot 10^6$	$1.12 \cdot 10^6$	$1.12 \cdot 10^6$	$1.12 \cdot 10^6$
L_1 FCM		6.16	$7.48 \cdot 10^4$	$1.56 \cdot 10^5$	$2.42 \cdot 10^5$	$3.83 \cdot 10^5$
PFCM		$4.75 \cdot 10^4$	$3.38 \cdot 10^5$	$1.01 \cdot 10^6$	$1.03 \cdot 10^6$	$1.03 \cdot 10^6$
L_p FCM		0.34	0.48	0.54	0.83	0.97
βFCM		0.0876	$4.22 \cdot 10^5$	$3.98 \cdot 10^5$	$4.60 \cdot 10^5$	$4.97 \cdot 10^5$
FCOM–εSQR	UOWA	85.93	$5.09 \cdot 10^5$	$4.60 \cdot 10^5$	$4.77 \cdot 10^5$	$5.08 \cdot 10^5$
	PLOWA	8.53	$4.27 \cdot 10^5$	$4.14 \cdot 10^5$	$4.09 \cdot 10^5$	$4.81 \cdot 10^5$
	SOWA	8.50	$4.75 \cdot 10^5$	$3.88 \cdot 10^5$	$4.28 \cdot 10^5$	$4.98 \cdot 10^5$
FCOM–εLIN	UOWA	0.06	0.23	0.68	1.44	2.26
	PLOWA	0.06	0.23	0.67	0.10	0.13
	SOWA	0.06	0.23	0.67	0.11	0.14
FCOM–εHUB	UOWA	0.04	0.43	1.29	2.74	4.29
	PLOWA	0.02	0.06	0.07	0.13	0.29
	SOWA	0.02	0.06	0.07	0.15	0.31
FCOM–εSIG	UOWA	0.29	1.23	2.78	5.12	7.38
	PLOWA	0.21	0.44	0.43	0.56	0.68
	SOWA	0.21	0.44	0.44	0.59	0.74
FCOM–εLOG	UOWA	0.01	$1.97 \cdot 10^5$	$1.97 \cdot 10^5$	$1.97 \cdot 10^5$	$1.97 \cdot 10^5$
	PLOWA	0.02	$1.97 \cdot 10^5$	$1.97 \cdot 10^5$	$1.97 \cdot 10^5$	$1.97 \cdot 10^5$
	SOWA	0.02	$1.97 \cdot 10^5$	$1.97 \cdot 10^5$	$1.97 \cdot 10^5$	$1.97 \cdot 10^5$

the SOWA ordering). We can also see that the FCM method terminated far away from the true centers, both when the background noise was added, and when no noise was added at all. In the presence of background noise, the terminal prototypes errors of the FCOM method with LOG loss function (the worst case) are approximately 2-times smaller than the errors of the FCM and βFCM methods. Among the reference methods, the best results were achieved by the

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

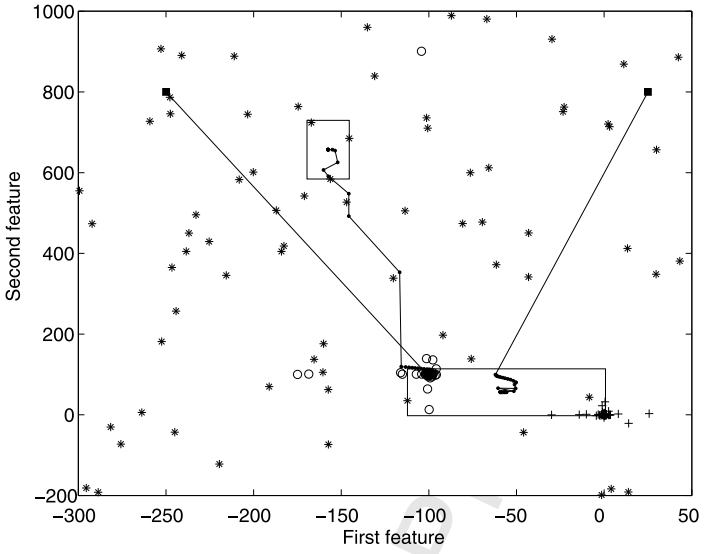

Fig. 7. Performance of the βFCM method with $\beta = 0.8$ on the heavy-tailed and overlapping clusters of points in background noise. The cardinality of the noise data is 75% of the "good" data.

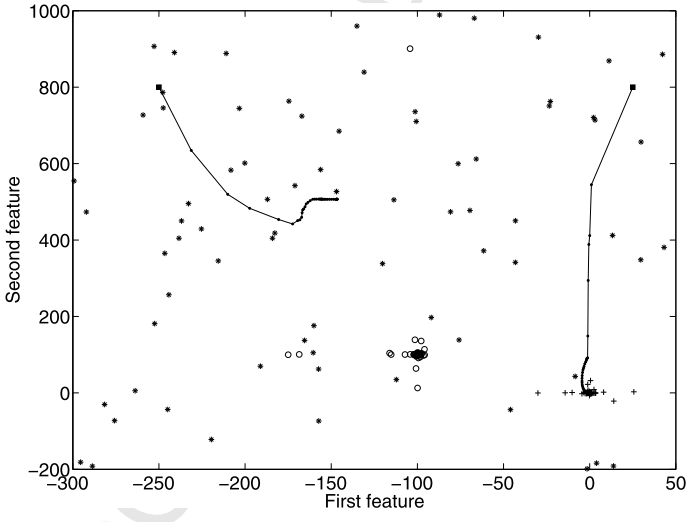

Fig. 8. Performance of the L_1 FCM method on the heavy-tailed and overlapping clusters of points in background noise. The cardinality of the noise data is 75% of the "good" data.

L_p FCM one; however, it produced larger errors than the method proposed in this work. We can conclude that this numerical example shows the usefulness of the new method proposed to clustering of the heavy-tailed, non-Gaussian data with overlapping groups of points in background noise.

The performance of the FCM method for 75 point background noise is illustrated in Fig. 6. We can see that the prototypes terminate in wrong places. Figs. 7 and 8 illustrate the performance of the βFCM and L_1 FCM methods, respectively, for 75 point background noise. In this figure the traces of the prototypes and the terminal insensitivity regions calculated are also shown. Performance of the FCOM method for 100 (!!!) point background noise is illustrated in Fig. 9. If these figures are taken into account, it can easily be noticed that the FCOM method terminates after a few iterations whereas much larger number of the iterations are needed by the other methods.

Finally, the running times of the methods tested, when applied to the heavy-tailed, non-Gaussian data with overlapping groups of points in 100 point background noise, are presented in Table 2. We can notice that the running time of the proposed clustering algorithm is approximately 10-times greater than that of the fuzzy c-means algorithm, and

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

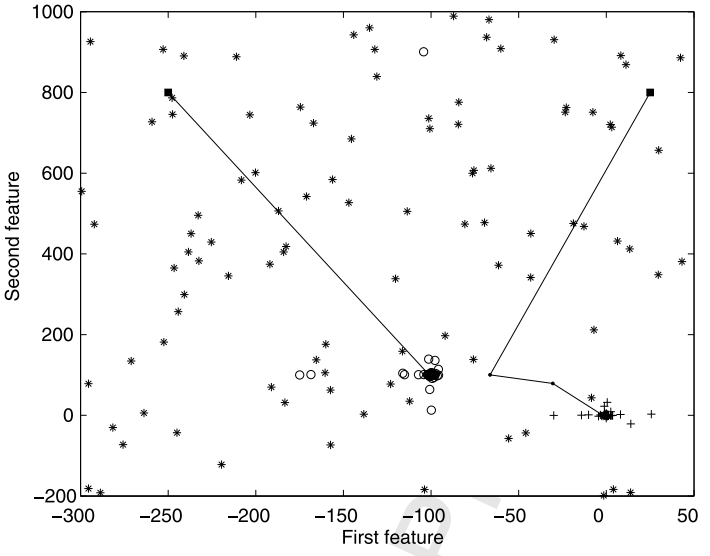

Fig. 9. Performance of the FCOM method with the εLIN loss function and PLOWA ordering function on the heavy-tailed and overlapping clusters of points in background noise. The cardinality of the noise data is 100% of the "good" data.

Table 2
Comparison of the clustering methods running times.

Method	Running time [s]
FCM	0.047
FCOM (εSQR, PLOWA)	0.51
βFCM	0.13
PCM	0.64
NCM	3.64
L_1 FCM	1.05
PFCM	29.49
L_p FCM	74.21

approximately 4-times greater if compared to that of the βFCM algorithm. However, compared to the PCM, NCM and PFCM methods, the proposed one is very competitive. This is due to the smaller number of iterations required for its convergence.

8. Conclusions

Real world data can be noisy and can contain outliers. Therefore the clustering methods need to be robust. Among the approaches to obtain robustness against outliers two the distinct ones can be considered: application of the M-estimators by Huber and of the ordered weighted averaging operators by Yager. Thus this paper combines the fuzzy c-means clustering with the ordered weighted averaging and the Huber's M-estimator. The developed FCOM clustering is based on various dissimilarity measures (including the ε-insensitive one) and the ordering of models residuals. The method is introduced as the problem of a constrained minimization of the criterion function. The necessary conditions for obtaining local minimum of the criterion function are shown. The existing L_1 norm clustering method and ε-insensitive fuzzy clustering method can be obtained as special cases of the method developed. Comparison of FCOM to the traditional fuzzy c-means, the possibilistic clustering, the fuzzy noise clustering, the L_p norm clustering, the L_1 norm clustering, the fuzzy clustering with polynomial fuzzifier and the ε-insensitive fuzzy c-means shows the competitiveness of the method proposed when applied the data with outliers and to the data with heavy-tailed and overlapping groups of points in background noise.

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

Acknowledgements

The author is grateful to the anonymous referees for their constructive comments that have helped to improve the paper. The work was performed using the infrastructure supported by POIG.02.03.01-24-099/13 grant: GeCONiI–Upper Silesian Center for Computational Science and Engineering.

Appendix A

If $\mathbf{V} \in \mathbb{R}_{cp}$ is fixed, then columns of $\mathbf{U}$ are independent, and the minimization of (9) can be performed term by term:

$$J(\mathbf{U}, \mathbf{V}) = \sum_{k=1}^{N} g_k(\mathbf{U}), \tag{A.1}$$

where

$$\underset{1\le k\le N}{\forall} \quad g_k(\mathbf{U}) = \sum_{i=1}^{c} \beta_{ik} (u_{ik})^m \, \mathcal{D}(\mathbf{x}_k, \mathbf{v}_i). \tag{A.2}$$

The Lagrangian of (A.2) with constraints from (11) is:

$$\underset{1\le k\le N}{\forall} \quad G_k(\mathbf{U}, \lambda_k) = \sum_{i=1}^{c} \beta_{ik} \, (u_{ik})^m \, \mathcal{D}(\mathbf{x}_k, \mathbf{v}_i) - \lambda_k \left[\sum_{i=1}^{c} \beta_{ik} \, u_{ik} - f_k \right], \tag{A.3}$$

where λ_k is the Lagrange multiplier. Setting the Lagrangian's gradient to zero we obtain:

$$\underset{1\le k\le N}{\forall} \quad \frac{\partial G_k(\mathbf{U}, \lambda_k)}{\partial \lambda_k} = \sum_{i=1}^{c} \beta_{ik} \, u_{ik} - f_k = 0, \tag{A.4}$$

and

$$\underset{\substack{1\le k\le N \\ 1\le s\le c}}{\forall} \quad \frac{\partial G_k(\mathbf{U}, \lambda_k)}{\partial u_{sk}} = m\beta_{sk}(u_{sk})^{m-1}\mathcal{D}(\mathbf{x}_k, \mathbf{v}_s) - \lambda_k \beta_{sk} = 0. \tag{A.5}$$

From (A.5) we get

$$u_{sk} = \left(\frac{\lambda_k}{m} \right)^{\frac{1}{m-1}} \mathcal{D}(\mathbf{x}_k, \mathbf{v}_s)^{\frac{1}{1-m}}. \tag{A.6}$$

From (A.6), (A.4) we obtain:

$$\left(\frac{\lambda_k}{m} \right)^{\frac{1}{m-1}} \sum_{j=1}^{c} \beta_{jk} \, \mathcal{D}(\mathbf{x}_k, \mathbf{v}_j)^{\frac{1}{1-m}} = f_k. \tag{A.7}$$

Combination of (A.6) and (A.7) yields:

$$\underset{\substack{1\le k\le N \\ 1\le s\le c}}{\forall} \quad u_{sk} = f_k \, \mathcal{D}(\mathbf{x}_k, \mathbf{v}_s)^{\frac{1}{1-m}} \Big/ \left[\sum_{j=1}^{c} \beta_{jk} \, \mathcal{D}(\mathbf{x}_k, \mathbf{v}_j)^{\frac{1}{1-m}} \right]. \tag{A.8}$$

Let us define:

$$\underset{1\le k\le N}{\forall} \quad \begin{cases} \mathcal{I}_k = \{ i \,|\, 1 \le i \le c; \ \mathcal{D}(\mathbf{x}_k, \mathbf{v}_i) = 0 \}, \\ \widetilde{\mathcal{I}}_k = \{1, 2, \ldots, c\} \setminus \mathcal{I}_k, \end{cases} \tag{A.9}$$

If $\mathcal{I}_k \neq \emptyset$, then the choice of $u_{ik} = 0$ for $i \notin \mathcal{I}_k$ and $\sum_{i\in\mathcal{I}_k} \beta_{ik} \, u_{ik} = f_k$ for $i \in \mathcal{I}_k$ results in minimization of the criterion (9), because elements of the partition matrix are zeros for non-zero dissimilarities, and non-zero for zero dissimilarities.

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

Appendix B

Combination of (9) and (10) yields:

$$J(\mathbf{U},\mathbf{V}) = \sum_{i=1}^{c}\sum_{k=1}^{N}\beta_{ik}\,(u_{ik})^m \sum_{l=1}^{p}\mathcal{D}(x_{kl},v_{il}) = \sum_{i=1}^{c}\sum_{l=1}^{p} g_{il}(v_{il}), \tag{B.1}$$

where

$$g_{il}(v_{il}) = \sum_{k=1}^{N}\beta_{ik}\,(u_{ik})^m\,\mathcal{D}(x_{kl},v_{il}) = \sum_{k=1}^{N}\beta_{ik}\,(u_{ik})^m\,\mathcal{L}(x_{kl}-v_{il}), \tag{B.2}$$

and can be called the weighted (or fuzzy) estimator of location v_{il}.

Our problem of criterion (9) minimization with respect to the prototypes can be decomposed into $c \cdot p$ minimization subproblems (defined by (B.2) for $i = 1, 2, \ldots, c;\ l = 1, 2, \ldots, p$). We can write (B.2) in the following way

$$g_{il}(v_{il}) = \sum_{k=1}^{N}\beta_{ik}\,(u_{ik})^m\,h_{ikl}(x_{kl}-v_{il})^2, \tag{B.3}$$

where

$$h_{ikl} = \begin{cases} 0, & e_{ikl} = 0, \\ \mathcal{L}(e_{ikl})/(e_{ikl})^2, & e_{ikl} \neq 0. \end{cases} \tag{B.4}$$

and $e_{ikl} = x_{kl} - v_{il}$. Thus, through the proper selection of the h_{ikl} parameters we may change different loss functions to the quadratic loss ($h_{ikl} \in \mathbb{R}^+ \cup \{0\}$). For example, the absolute or linear (LIN) error function is easy to obtain by taking

$$h_{ikl} = \begin{cases} 0, & e_{ikl} = 0, \\ 1/|e_{ikl}|, & e_{ikl} \neq 0. \end{cases} \tag{B.5}$$

Many other loss functions may easily be obtained:

- HUBer (HUB) with parameter $\delta > 0$

$$h_{ikl} = \begin{cases} 1/\delta^2, & |e_{ikl}| \leq \delta, \\ 1/(\delta|e_{ikl}|), & |e_{ikl}| > \delta. \end{cases} \tag{B.6}$$

- SIGmoidal (SIG) with parameters $\alpha, \beta > 0$

$$h_{ikl} = \begin{cases} 0, & e_{ikl} = 0, \\ 1/\{e_{ikl}^2[1+\exp(-\alpha(|e_{ikl}|-\beta))]\}, & e_{ikl} \neq 0. \end{cases} \tag{B.7}$$

- SIGmoidal-Linear (SIGL) with parameters $\alpha, \beta > 0$

$$h_{ikl} = \begin{cases} 0, & e_{ikl} = 0, \\ 1/\{|e_{ikl}|[1+\exp(-\alpha(|e_{ikl}|-\beta))]\}, & e_{ikl} \neq 0. \end{cases} \tag{B.8}$$

- LOGarithmic (LOG)

$$h_{ikl} = \begin{cases} 0, & e_{ikl} = 0, \\ \log(1+e_{ikl}^2)/e_{ikl}^2, & e_{ikl} \neq 0. \end{cases} \tag{B.9}$$

- LOG-Linear (LOGL)

$$h_{ikl} = \begin{cases} 0, & e_{ikl} = 0, \\ \log(1+e_{ikl}^2)/|e_{ikl}|, & e_{ikl} \neq 0. \end{cases} \tag{B.10}$$

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

Assuming temporarily the constancy of h_{ikl} and setting the g_{il} gradient to zero we obtain:

$$\underset{\substack{1\le i\le c\\ 1\le l\le p}}{\forall} \quad \frac{\partial g_{il}(v_{il})}{\partial v_{il}} = -2\sum_{k=1}^{N} \beta_{ik}\,(u_{ik})^m\, h_{ikl}\,(x_{kl}-v_{il}) = 0. \tag{B.11}$$

From (B.11) we get:

$$\underset{\substack{1\le i\le c\\ 1\le l\le p}}{\forall} \quad v_{il} = \left[\sum_{k=1}^{N} \beta_{ik}\,(u_{ik})^m h_{ikl}\,x_{kl}\right] \Big/ \left[\sum_{k=1}^{N} \beta_{ik}\,(u_{ik})^m h_{ikl}\right]. \tag{B.12}$$

Appendix C

The ε-insensitive quadratic loss function may be decomposed into two asymmetric quadratic loss functions, where:

$$h_{ikl}^{-} = \begin{cases} 0, & \varepsilon - x_{kl} + v_{il} \ge 0, \\ \mathcal{L}(\varepsilon - x_{kl} + v_{il})/(\varepsilon - x_{kl} + v_{il})^2, & \varepsilon - x_{kl} + v_{il} < 0, \end{cases} \tag{C.1}$$

and

$$h_{ikl}^{+} = \begin{cases} 0, & \varepsilon + x_{kl} - v_{il} \ge 0, \\ \mathcal{L}(\varepsilon + x_{kl} - v_{il})/(\varepsilon + x_{kl} - v_{il})^2, & \varepsilon + x_{kl} - v_{il} < 0, \end{cases} \tag{C.2}$$

Let ${}_e e_{ikl}$ for $i = 1, 2, \ldots, c;\ k = 1, 2, \ldots, 2N,\ l = 1, 2, \ldots, p$ be the extended residuals defined as follows: ${}_e e_{ikl} = \varepsilon - x_{kl} + v_{il}$ for $k = 1, 2, \ldots, N$ and ${}_e e_{ikl} = \varepsilon + x_{(k-N)l} - v_{il}$ for $k = N+1, N+2, \ldots, 2N$. The fitting of the kth datum by the ith prototype is represented by the kth and the $(N+k)$th element of ${}_e e_{ikl}$. If both ${}_e e_{ikl}$ and ${}_e e_{i(N+k)l}$ are greater than or equal to zero, then the kth datum falls in the insensibility zone (with respect to the lth component). If ${}_e e_{ikl}$ (or ${}_e e_{i(N+k)l}$) is less than zero, then the kth datum is above (or below) the insensibility zone and should be penalized. Now (C.1) and (C.2) may be written in one formula

$$\underset{1\le k\le 2N}{\forall} \quad h_{ikl} = \begin{cases} 0, & {}_e e_{ikl} \ge 0, \\ \mathcal{L}({}_e e_{ikl})/({}_e e_{ikl})^2, & {}_e e_{ikl} < 0, \end{cases} \tag{C.3}$$

and (29) takes the following form

$$g_{il}(v_{il}) = \sum_{k=1}^{2N} \beta_{ik}\,(u_{ik})^m h_{ikl}({}_e e_{ikl})^2, \tag{C.4}$$

where the membership (u_{ik}) and the typicality (β_{ik}) values were 'doubled' for each kth data point, i.e., $u_{i(N+k)} = u_{ik}$, $\beta_{i(N+k)} = \beta_{ik}$, for $k = 1, 2, \ldots, N$, (because each data point is also 'doubled' in criterion function (C.4)). Considerations similar to those conducted at Appendix B lead to the following result

$$\underset{\substack{1\le i\le c\\ 1\le l\le p}}{\forall} \quad v_{il}^{[r]} = \left[\sum_{k=1}^{2N} \breve{\alpha}_k\,(u_{ik})^m h_{ikl}^{[r]}\,{}_e x_{kl}\right] \Big/ \left[\sum_{k=1}^{2N} \breve{\alpha}_k\,(u_{ik})^m h_{ikl}^{[r]}\right]. \tag{C.5}$$

where ${}_e x_{kl}$, $k = 1, 2, \ldots, 2N$ denote the extended data, defined as follows: ${}_e x_{kl} = x_{kl} - \varepsilon$ for $k = 1, 2, \ldots, N$ and ${}_e x_{kl} = x_{(k-N)l} + \varepsilon$ for $k = N+1, N+2, \ldots, 2N$, and

$$\underset{1\le k\le 2N}{\forall} \quad h_{ikl}^{[r]} = \begin{cases} 0, & {}_e e_{ikl}^{[r-1]} \ge 0, \\ \mathcal{L}({}_e e_{ikl}^{[r-1]})/({}_e e_{ikl}^{[r-1]})^2, & {}_e e_{ikl}^{[r-1]} < 0. \end{cases} \tag{C.6}$$

Thus, for the ε-insensitive quadratic loss we have

$$h_{ikl}^{[r]} = \begin{cases} 0, & {}_e e_{ikl}^{[r-1]} \ge 0, \\ 1, & {}_e e_{ikl}^{[r-1]} < 0. \end{cases} \tag{C.7}$$

Many other ε-insensitive loss functions may also easily be obtained:

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

- Vapnik or ε-insensitive LINear (εLIN)

$$h_{ikl}^{[r]} = \begin{cases} 0, & {}_e e_{ikl}^{[r-1]} \geq 0, \\ -1/{}_e e_{ikl}^{[r-1]}, & {}_e e_{ikl}^{[r-1]} < 0. \end{cases} \tag{C.8}$$

- ε-Insensitive HUBer (εHUB) with parameter $\delta > 0$

$$h_{ikl}^{[r]} = \begin{cases} 0, & {}_e e_{ikl}^{[r-1]} \geq 0, \\ 1/\delta^2, & 0 > {}_e e_{ikl}^{[r-1]} \geq -\delta, \\ -1/(\delta|{}_e e_{ikl}^{[r-1]}|), & {}_e e_{ikl}^{[r-1]} < -\delta. \end{cases} \tag{C.9}$$

- ε-Insensitive SIGmoidal (εSIG) with parameters $\alpha, \beta > 0$

$$h_{ikl}^{[r]} = \begin{cases} 0, & {}_e e_{ikl}^{[r-1]} \geq 0, \\ 1/(({}_e e_{ikl}^{[r-1]})^2(1+\exp(\alpha({}_e e_{ikl}^{[r-1]}+\beta)))), & {}_e e_{ikl}^{[r-1]} < 0. \end{cases} \tag{C.10}$$

- ε-Insensitive SIGmoidal-Linear (εSIGL) with parameters $\alpha, \beta > 0$

$$h_{ikl}^{[r]} = \begin{cases} 0, & {}_e e_{ikl}^{[r-1]} \geq 0, \\ -1/({}_e e_{ikl}^{[r-1]}(1+\exp(\alpha({}_e e_{ikl}^{[r-1]}+\beta)))), & {}_e e_{ikl}^{[r-1]} < 0. \end{cases} \tag{C.11}$$

- ε-Insensitive LOGarithmic (εLOG)

$$h_{ikl}^{[r]} = \begin{cases} 0, & {}_e e_{ikl}^{[r-1]} \geq 0, \\ \log(1+({}_e e_{ikl}^{[r-1]})^2)/({}_e e_{ikl}^{[r-1]})^2, & {}_e e_{ikl}^{[r-1]} < 0. \end{cases} \tag{C.12}$$

- ε-Insensitive LOG-Linear (εLOGL)

$$h_{ikl}^{[r]} = \begin{cases} 0, & {}_e e_{ikl}^{[r-1]} \geq 0, \\ -\log(1+({}_e e_{ikl}^{[r-1]})^2)/{}_e e_{ikl}^{[r-1]}, & {}_e e_{ikl}^{[r-1]} < 0. \end{cases} \tag{C.13}$$

In the robust statistics many measures of robustness are defined [29], such as a sensitivity curve, an influence function and a gross-error sensitivity. If we denote estimate of v_{il} obtained by iterating (C.5) for the sample $x_1, x_2, \ldots, x_N$ as $\hat{v}_{il}(x_1, x_2, \ldots, x_N)$ then the sensitivity curve is a difference $\hat{v}_{il}(x_0, x_1, x_2, \ldots, x_N) - \hat{v}_{il}(x_1, x_2, \ldots, x_N)$ as a function of the outlier x_0 location. The influence function of an estimator is an asymptotic version of the sensitivity curve, when the sample with F distribution contains a small fraction γ of identical outliers x_0

$$\mathrm{IF}(x_0, F) = \lim_{\gamma \to 0^+} \frac{\hat{v}_{il}((1-\gamma)F + \gamma\delta_{x_0}) - \hat{v}_{il}(F)}{\gamma}, \tag{C.14}$$

where $\hat{v}_{il}((1-\gamma)F + \gamma\delta_{x_0})$ denotes the estimate for a sample with F distribution contaminated by a small fraction of identical outliers x_0. The gross-error sensitivity is defined as $\max_{x_0} |\mathrm{IF}(x_0, F)|$. The above-mentioned measures concern a crisp set of samples. For a fuzzy set of samples (with memberships not equal one) such convenient measures have not been defined yet.

Uncited references

[3] [13] [32]

References

[1] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1982.
[2] F. De Carvalho, C.P. Tenorio, Fuzzy k-means clustering algorithms for interval-valued data based on adaptive quadratic distances, Fuzzy Sets Syst. 161 (3) (2010) 2978–2999.
[3] R. Coppi, P. D'Urso, P. Giordani, Fuzzy and possibilistic clustering models for fuzzy data, Comput. Stat. Data Anal. 56 (2012) 915–927.

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007

[4] R.N. Davé, Characterization and detection of noise in clustering, Pattern Recognit. Lett. 12 (11) (1991) 657–664.
[5] R.N. Davé, R. Krishnapuram, Robust clustering methods: a unified view, IEEE Trans. Fuzzy Syst. 5 (2) (1997) 270–293.
[6] P. D'Urso, Fuzzy clustering for data time arrays with inlier and outlier time trajectories, IEEE Trans. Fuzzy Syst. 13 (5) (2005) 583–604.
[7] P. D'Urso, P. Giordani, A robust fuzzy k-means clustering model for interval valued data, Comput. Stat. 21 (2006) 251–269.
[8] P. D'Urso, R. Massari, Fuzzy clustering of human activity patterns, Fuzzy Sets Syst. 215 (2013) 29–54.
[9] P. D'Urso, L. De Giovanni, R. Massari, D. Di Lallo, Noise fuzzy clustering of time series by the autoregressive metric, Metron 71 (2013) 217–243.
[10] P. D'Urso, L. De Giovanni, Robust clustering of imprecise data, Chemom. Intell. Lab. Syst. 136 (2014) 58–80.
[11] P. D'Urso, L. De Giovanni, R. Massari, Trimmed fuzzy clustering for interval-valued data, Adv. Data Anal. Classif. (2014), in press.
[12] P. D'Urso, Fuzzy clustering, in: C. Hennig, M. Meila, F. Murtagh, R. Rocci (Eds.), Handbook of Cluster Analysis, Chapman & Hall, 2015, in press.
[13] R.O. Duda, P.E. Hart, Pattern Classification and Scene Analysis, John Wiley & Sons, New York, 1973.
[14] J.C. Dunn, A Fuzzy relative of the ISODATA process and its use in detecting compact well-separated cluster, J. Cybern. 3 (3) (1973) 32–57.
[15] J. Fodor, M. Roubens, Fuzzy Preference Modelling and Multicriteria Decision Support, Kluwer, Dordrecht, 1994.
[16] K. Fukunaga, Introduction to Statistical Pattern Recognition, Academic Press, San Diego, 1990.
[17] R.J. Hathaway, J.C. Bezdek, Generalized fuzzy c-means clustering strategies using L_p norm distances, IEEE Trans. Fuzzy Syst. 8 (5) (2000) 576–582.
[18] C. Henning, Dissolution point and isolation robustness: robustness criteria for general cluster analysis methods, J. Multivar. Anal. 99 (2008) 1154–1176.
[19] D. Horta, I.C. de Andrade, R.J. Campello, Evolutionary fuzzy clustering of relational data, Comput. Sci. 412 (42) (2011) 5854–5870.
[20] P.J. Huber, Robust Statistics, Wiley, New York, 1981.
[21] K. Jajuga, L_1-norm based fuzzy clustering, Fuzzy Sets Syst. 39 (1) (1991) 43–50.
[22] P.R. Kersten, Fuzzy order statistics and their application to fuzzy clustering, IEEE Trans. Fuzzy Syst. 7 (6) (1999) 708–712.
[23] R. Krishnapuram, J.M. Keller, A possibilistic approach to clustering, IEEE Trans. Fuzzy Syst. 1 (1) (1993) 98–110.
[24] J.M. Leski, Towards a robust fuzzy clustering, Fuzzy Sets Syst. 137 (2) (2003) 215–233.
[25] J.M. Leski, Neuro-fuzzy system with learning tolerant to imprecision, Fuzzy Sets Syst. 138 (2) (2003) 427–439.
[26] J.M. Leski, Generalized weighted conditional fuzzy clustering, IEEE Trans. Fuzzy Syst. 11 (6) (2003) 709–715.
[27] J.M. Leski, An ε-margin nonlinear classifier based on if–then rules, IEEE Trans. Syst. Man Cybern., Part B, Cybern. 34 (1) (2004) 68–76.
[28] J.M. Leski, N. Henzel, ECG baseline wander and powerline interference reduction using nonlinear filter bank, Signal Process. 85 (2) (2005) 781–793.
[29] R.A. Maronna, R.D. Martin, V.J. Yohai, Robust Statistics: Theory and Methods, John Wiley and Sons, New York, 2006.
[30] W. Pedrycz, Conditional fuzzy clustering in the design of radial basis function neural network, IEEE Trans. Neural Netw. 9 (4) (1998) 601–612.
[31] E.H. Ruspini, A new approach to clustering, Inf. Control 15 (1) (1969) 22–32.
[32] J.T. Tou, R.C. Gonzalez, Pattern Recognition Principles, Adison-Wesley, London, 1974.
[33] R. Winkler, F. Flawonn, R. Kruse, Fuzzy clustering with polynomial fuzzifier connection with M-estimators, Appl. Comput. Math. 10 (2011) 146–163.
[34] C. Xu, P. Zhang, B. Li, D. Wu, H. Fun, Vague c-means clustering algorithm, Pattern Recognit. Lett. 34 (5) (2013) 505–510.
[35] R.R. Yager, On ordered weighted averaging aggregation operators in multicriteria decision making, IEEE Trans. Syst. Man Cybern. 18 (1) (1988) 183–190.
[36] R.R. Yager, OWA operators in regression problems, IEEE Trans. Fuzzy Syst. 18 (1) (2010) 106–113.
[37] V. Vapnik, Statistical Learning Theory, Wiley, New York, 1998.
[38] F. Zhao, H. Liu, L. Jiao, Spectral clustering with fuzzy similarity measure, Digit. Signal Process. 21 (6) (2011) 701–709.

Please cite this article in press as: J.M. Leski, Fuzzy c-ordered-means clustering, Fuzzy Sets and Systems (2014), http://dx.doi.org/10.1016/j.fss.2014.12.007