

Implementation Issues in the Fuzzy c-Medians Clustering Algorithm

Paul R. Kersten

Naval Air Warfare Center, Weapons Division
Target Recognition Section, Code 452220D
China Lake, CA 93555, USA
kersten@vislab.navy.mil

Abstract*

The fuzzy *c*-Median (FCMED) clustering algorithm is an alternating optimization (AO) method of solving the fuzzy *c*-Means (FCM) clustering algorithm using the ℓ_1 -norm. This algorithm is more resistant to outliers than the FCM-AO algorithm using the ℓ_2 -norm. The robustness of the FCMED does not come free, since the fuzzy median is the cluster-centering statistic and exact evaluation of the fuzzy median usually involves ordering the sample values. The efficiency of calculating the fuzzy median is an important implementation issue. Two other evaluation methods are considered for the fuzzy median: The first is the remedian, which statisticians use to simplify the estimation of the median. A fuzzy remedian is defined and used to approximate the fuzzy median. The second method finds the root of the derivative of the functional equation defining the fuzzy median. Both approaches are described and illustrated in this paper.

1. Introduction

The FCM clustering algorithm is a partitional clustering algorithm that generalizes the *c*-means clustering algorithm [1,2]. The FCM is really a class of clustering algorithms, which differ based upon the weighting exponent, the norm used in the definition, and the method used to solve for the optimal exemplars and membership values [3]. In the FCM algorithms each data sample belongs to each class or cluster with varying degrees. The distances from the data points to the cluster exemplars or centers determine their memberships in each cluster. When distances are based on the p -norms, then the class can be called the pFCM problem [3]. When an alternating optimization procedure is used the class of algorithms is sometimes denoted FCM-AO. In fact, the

pFCM-AO using the ℓ_1 -metric ($p=1$) with the exemplar estimation based on the fuzzy median is the FCMED clustering algorithm and the exact algorithm and its derivation may be found in [4-7]. Others researchers have considered ℓ_1 objective functionals [3,8,9]. In the FCM-AO using the squared ℓ_2 -norm, the centering statistic is a weighted linear combination of the data where the weights depend upon the data point memberships. In contrast, for the FCMED the optimum centering statistic is a nonlinear estimate called the fuzzy median [4-5] or weighted median [10-12] where the weights also depend on the data point memberships as shown in [7]. Linear statistics are notoriously vulnerable to outliers whereas order statistics such as the median are known to have a high resistance to outliers. This resistance to outliers or robustness is retained by the fuzzy median and makes the FCMED algorithm more robust than the FCM-AO.

The fuzzy median generalizes the median to fuzzy sets where the membership value for each data point is in $[0,1]$. Given a crisp data set $X = \{x_1, \dots, x_N\}$, and the corresponding ordered sample of real data points $x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(N)}$, the median is defined as the center values $x_{(\lfloor N/2 \rfloor + 1)}$ if N is odd, and as the average of the two center points if N is even [13]. In one dimension, the median can be roughly defined as the point where the number of samples at-and-to-the-left and at-and-to-the-right are equal. In a crisp sample, the cardinality and the number of samples are the same since each point has set membership of 1.0. To define a fuzzy data set $X_f = x_1/u_1 + \dots + x_N/u_N$ requires not only the data X but also the memberships $U = \{u_1, u_2, \dots, u_N\}$. The scalar fuzzy cardinality of the fuzzy data set is the sum of the memberships of the points in the set [14] and is not necessarily equal to the number of samples. Then the fuzzy median m is the point on the real axis where the sum of the memberships of the points at-and-to-the-left of m and at-and-to-the-right of m are equal. Like the crisp median, the fuzzy median is not always unique. If at some point in the ordered set, the sum-of-the-cardinalities-

* This research was supported by IR funds from the Naval Air Warfare Center, China Lake, CA. Approved for public release; distribution is unlimited.

to-the-left first exceeds half-the-total fuzzy cardinality of the set, then that point is the fuzzy median. On the other hand, if for each point in some domain interval, the sum-of-cardinalities-to-the-left equals half-the-total cardinality, then the fuzzy median is the middle of the interval. To evaluate the fuzzy median by this method, the data must be ordered and each data point must retain its associated membership value.

Ordering the data set to evaluate the median has a time complexity of $O(N \log N)$ and a space complexity of $O(N)$. For large values of N found in images, say 16K (128x128 image) or larger, both the time complexity and the space complexity become problems. Instead of trying to order the entire sample, the fuzzy remedian orders samples in batches of size b [15]. So for a sample of size 100 say, ten batches of size 10 reduce the data set to 10 medians. Then the median of the medians, approximates the sample median. The remedian considerably reduces the space complexity, but not without some loss in robustness.

The functional definition of the median does not require the data points be ordered. The crisp median is defined as

$$\text{the solution } m \text{ of } \min_{m \in R} P_{\text{crisp}}(m) = \min_{m \in R} \sum_{k=1}^N |x_k - m| \quad [13, \text{ p. 233}]$$

and the fuzzy median as the solution of

$$\min_{m \in R} P_{\text{fuzzy}}(m) = \min_{m \in R} \sum_{k=1}^N u_k |x_k - m|. \quad \text{In practice, one}$$

finds the root of the functional derivative given

$$\text{respectively by } \sum_{k=1}^N \text{sgn}(x_k - m) = 0 \quad \text{and} \quad \Psi(m) =$$

$$\sum_{k=1}^N u_k \text{sgn}(x_k - m) = 0. \quad \text{Both medians can be found by}$$

numerical search. The search method avoids the space complexity of the median and the remedian by searching for the root m of the non-increasing step function $\Psi(m)$.

$$\text{For } m < \min_k(x_k), \quad \Psi(m) = N_f = \sum_{i=1}^N u_i \quad \text{the fuzzy}$$

cardinality of the data set and for $m > \max_k(x_k)$ it is

$$\Psi(m) = -N_f. \quad \text{So } \Psi(m) \text{ clearly has at least one root and}$$

to find the roots, many search algorithms are available. The bisection method was chosen over gradient search because of the discontinuities in $\Psi(m)$. Bisection also has the advantage of a low space complexity since only a running sum has to be stored.

The rest of this paper is organized as follows: Section 2 describes the FCMED algorithm and gives two examples for small sample sizes to illustrate its robustness. The fuzzy remedian is described in section 3.

The fuzzy remedian is substituted for the fuzzy median in the FCMED algorithm and demonstrated in two examples. Space and time complexity results are discussed. Contained in section 4 is the root search method for approximating the fuzzy median in the FCMED algorithm. The examples of section 3 are used to illustrate this method and a rough time comparison the exemplar calculations for the FCM and the two methods described here for the FCMED. The conclusions are in section 5.

2. The FCMED Algorithm

Consider the data set denoted by $X = \{x_1, x_2, \dots, x_N\}$ of size N , where now each sample $x_i \in R^P$ is a P -dimensional real vector. Assume there are c classes where the set of class exemplars is $v = \{v_1, v_2, \dots, v_c\}$ and the membership matrix is $U = [u_{ik}]$, where $u_{ik} = u_i(x_k) \in [0, 1]$ is the membership of the k -th sample in the i -th class. The special case of the ℓ_2 -FCM-AO clustering algorithm using the squared ℓ_2 -norm is derived by minimizing the objective functional is $J(U, v) = \sum_{k=1}^N \sum_{i=1}^c (u_{ik})^{m_c} (d_{ik})^2$ where $d_{ik} = \|v_i - x_k\|_2$ is the ℓ_2 distance between the i -th exemplar v_i and the k -th data point x_k and m_c is the weighting exponent. The optimal centering statistic for the clusters is a weighted linear combination of the sample points where the weights are formed from the membership functions u_{ik} and these weights are generated using the distances d_{ik} . The ℓ_2 -FCM-AO clustering algorithm is an amazingly practical algorithm that can be used for data exploration, learning, and image segmentation.

The FCMED is a pFCM-AO ($p=1$) since it optimizes the objective functional given by [3] $J(U, v) = \sum_{k=1}^N \sum_{i=1}^c (u_{ik})^{m_c} d_{ik}$ where $d_{ik} = \sum_{j=1}^P |x_k(j) - v_i(j)| = \|x_k - v_i\|_1$ is the ℓ_1 metric. Alternating optimization of $J(U, v)$ with respect to the membership values u_{ik} and with respect to the prototypes v_i , yields the FCMED clustering algorithm[5,7].

The FCMED algorithm is:

1. Fix c , the number of classes such that $c \in \{2, \dots, N-1\}$ and choose the ℓ_1 metric in R^P . Fix the weighting exponent $m_c \in [1, \infty)$. Initialize the membership matrix denoted by $U^{(0)} \in M_{fc}$ where V_{cN} is the vector space of $c \times N$ matrices and $M_{fc} =$

- $$\left\{ U \in V_{CN} \mid u_{ik} \in [0,1] \forall i,k; \sum_{i=1}^c u_{ik} = 1 \forall k; 0 < \sum_{k=1}^N u_{ik} < N \forall i \right\}.$$
2. Construct the c exemplars v_i for $i = 1, \dots, c$ by finding the fuzzy median for each class. Each exemplar is p -dimensional so $v_i(j)$ is found for $j = 1, \dots, p$, using the membership functions given by u_{ik} and the p -th component of x_k .
3. Update the memberships u_{ik} with $u_{ik} = \frac{1}{1 + \left[\sum_{j=1}^c \left(\frac{d_{ik}}{d_{jk}} \right)^{1/(m_c - 1)} \right]}$, provided of course that none of the d_{jk} are zero. In the latter case, the u_{ik} are assigned differently [1, p. 66].
4. Compare the last two membership matrices, $U^{(l)}$ and $U^{(l+1)}$. When they are sufficiently close, terminate the algorithm; otherwise, go to step 2.

The alternating optimization scheme is the same as the FCM-AO since one alternately calculates the cluster centers from the memberships and then calculates the memberships using the cluster centers. The optimum cluster-center is the fuzzy median [7], which replaces the weighted mean in the ℓ_2 -FCM-AO algorithms, and the membership function of step 3 is quite similar, although not identical, to the ℓ_2 -FCM-AO algorithm. As explained in section 1, the fuzzy median can be calculated by sorting the sample values. But the space and time complexity of this calculation is higher than the weighted mean used in the FCM. The time complexity for each exemplar v_i is $O(pN \log N)$ since for each of the p -dimensions of the sample vectors it takes $O(N \log N)$ operations to sort the data. There are c classes so the time complexity for Step 2 is $O(cpN \log N)$. The space complexity is $O(N)$, which for large data sets like images can be quite onerous. Although the time complexity is high to order the data set, once ordered the computation of the fuzzy median is considerably simplified.

The FCMED is illustrated for two small-sample data sets: first a two class problem with antipodal Gaussian clusters and second a two class problem with antipodal Cauchy cluster centers. The former is a benign test for both the ℓ_2 -FCM-AO and the FCMED, and the latter is a heavy-tailed, outlier-latent example designed to stress both the ℓ_2 -FCM-AO and the FCMED algorithms. To standardize the comparison, all examples initialize the exemplars at ± 1 in the second component of the starting vector, and zero in all other components. The Gaussian clusters are five dimension clusters with centers at $(\pm 1.27, 0, 0, 0, 0)$ with identity covariance matrices. Two hundred samples for each class make exact fuzzy median

calculations quite feasible and the weighting exponent is set to 1.25. The resultant trajectories for the Gaussian clusters are shown in figure 1.

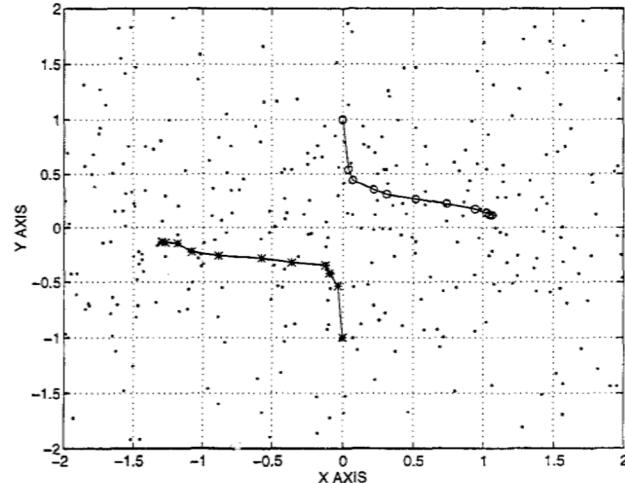


Figure 1. FCMED trace with two Gaussian clusters.

In figure 2, the FCMED algorithm converges to the antipodal cluster centers located at $[\pm 1.27, 0]$ of the two dimensional Cauchy sample. The ℓ_2 -FCM-AO converges to the cluster centers for the Gaussian distribution but not for the Cauchy [6].

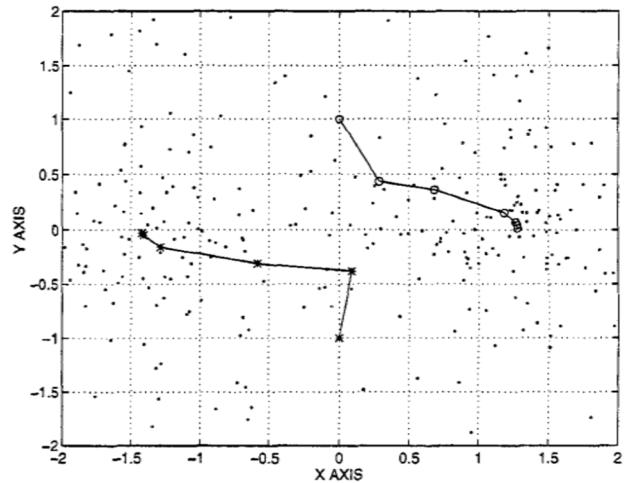


Figure 2 FCMED trace with two Cauchy clusters. Outliers not shown.

3. The Fuzzy Remedian

What makes the remedian so attractive is its conceptual simplicity and its well defined properties [15]. For $N = b^k$ samples, the remedian is defined by iteratively

applying the median to the data in batches of size b , called the *base*. Conceptually, one sweeps through the samples calculating b^{k-1} medians of size b . The same sweep is made through these derived medians yielding b^{k-2} medians (the medians of the medians). This process continues until there is a single median remaining, called the *remedian*. However, for $N < b^k$, the computation also requires the calculation of a *weighted median* [15]. This section describes a program for arbitrary b and k , that has been developed to calculate a fuzzy remedian for any $N \leq b^k$ [16]. The program is a modified depth-first search algorithm.

The remedian can be calculated without storing the entire data sequence or waiting until the entire sequence is available. For $N = b^k$ samples, the space complexity of the remedian is $O(bk)$, which for large sample sizes is small in comparison to the space complexity of the median $O(b^k)$. For *fixed* b , if the batch time complexity of the median is $O(b \log(b))$, then the time complexity of the remedian is $O\left(\frac{b}{b-1}(b^k - 1)\log(b)\right) = O(N \log(N^{1/k})) = O(N \log N)$.

The remedian is equivariant with respect to any monotone transformation. Unlike the mean and the median, the remedian is *not* invariant with respect to the permutation transformations. However, for independent and identically distributed random variables (i.i.d.r.v.'s) the remedian is "nearly" permutation invariant since the sequence of least-favorable outlier patterns needed to break down the remedian rapidly approaches zero in probability as the sample size N increases. The remedian is also a weakly consistent estimator and converges in law to a limiting distribution. The finite-sample breakdown of any statistic is the smallest fraction of outliers needed to destroy the capability of the statistic to estimate the desired quantity. For the remedian, the breakdown is $\varepsilon_N^* = \lceil b/2 \rceil^k / n = (\lceil b/2 \rceil / b)^k$ which tends to 0 as k increases since $\lceil b/2 \rceil / b < 1$ [15]. For example, for $k=4$, $N=10,000$, and $b=10$, the remedian breakdown is near 6%, which is far less than the near 50% breakdown of the median [17]. However, the 6% breakdown can occur only when the outliers occupy a specific subset of the samples [15]. For random samples, this is a low probability event, so that worst case breakdown is not probable for large N .

Implementation of the remedian is not easy. If the values of base b and the power k are hard coded into the program, then the algorithm can implemented as a set of nested do loops [15]. However, once both b and k become input parameters, then more sophisticated data

structures are needed. While the remedian applies to crisp data streams, the fuzzy remedian applies to fuzzy data streams. A fuzzy data stream is a sequence of pairs $\{x_k, u_i(x_k)\}$, where the first term is the sample value and the second term is the membership of that sample in the i -th fuzzy set. When the memberships of the samples are identical and $N = b^k$, then the fuzzy remedian reduces to the crisp remedian since the fuzzy median reduces to the median. A fuzzy remedian is constructed by replacing the base- b medians with the base- b fuzzy medians. A non-recursive graph-search implementation is described in [16].

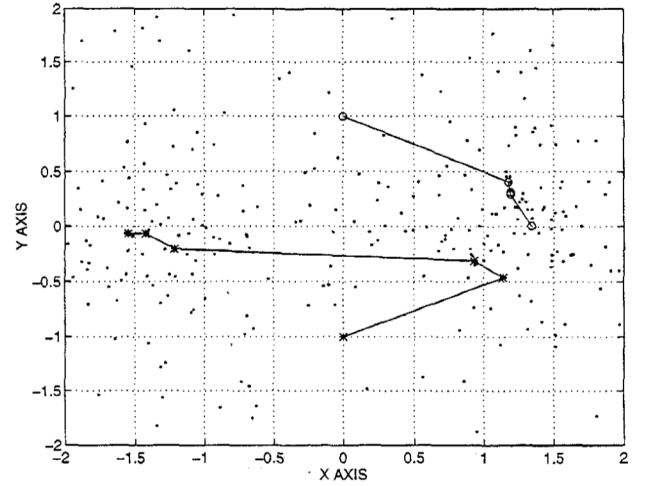


Figure 3. Remedian-FCMED trace with Cauchy clusters and data ordered by class.

Replacing the fuzzy median in the FCMED with the fuzzy remedian (remedian-FCMED) yields some surprising results. In the Cauchy example with 400 samples, the batch size is taken as 8. Unless the "outliers" are occupying a special set of sample points, then the remedian should illustrate the same robustness as the median with high probability [15]. Statistically, this means that the data is sufficiently randomized. This effect is evident in the FCMED trace of the Cauchy example as shown in figure 3, where the samples are presented first from class 1 and then from class 2. Note again the starting vectors are $[0, \pm 1]$. The presentation sequence is *not* random since the x-coordinate values resemble a step function, which produces a poor first estimate in the trace. Shuffling the simulated data so the presentation is randomized, improves the performance considerably, as illustrated in figure 4. Comparing figure 2 to figure 4, it is easy to see a loss of accuracy in the left cluster center using the fuzzy remedian (true centers are $[\pm 1.27, 0]$).

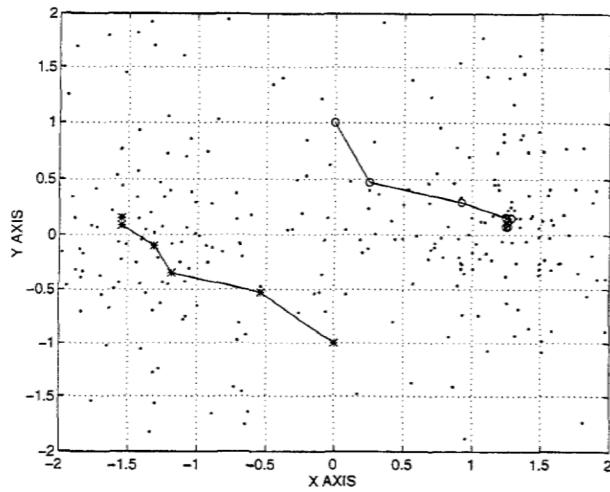


Figure 4. Remedian-FCMED trace with Cauchy clusters and data shuffled.

4. Root Search for Fuzzy Median

By searching for a root of $\Psi(m)$, one avoids ordering the data to find the median. The bisection method can find the root in 16-20 iterations. The summation $\Psi(m)$ must be evaluated at each iteration and the upper and lower bounds of domain of X are needed to start the search. Roots exist because $\Psi(m)$ is a non-increasing function of m ; however, the uniqueness and thus the accuracy of the root depend upon the example. If the root is unique, the accuracy can be easily calculated. If the root is *not* unique, $\Psi(m)=0$ over a region of the domain, then the accuracy of the root is unknown, unless the extent of the zero region is known. Defining the zero region requires ordering the data locally about the root, which is time consuming. The bisection method will *not* solve the extent of the zero region, since it stops the first time it hits any point in the zero region. Since the summation $\Psi(m)$ is independent of the data presentation order, so is the bisection method. For the Cauchy example, figure 5 is the FCMED trace using the root method (root-FCMED), which is surprisingly similar to the true FCMED trace of figure 2. The error tolerance on the bisection routine is 1.e0-5 and the maximum number of iterations is 20. Even so, there are many instances when the root does not meet the tolerance during the creation of this trace because of the large outliers in the Cauchy sample.

So far, the examples have included only small sample data files up to 400 samples. These examples illustrated several features of the two approximating methods for the

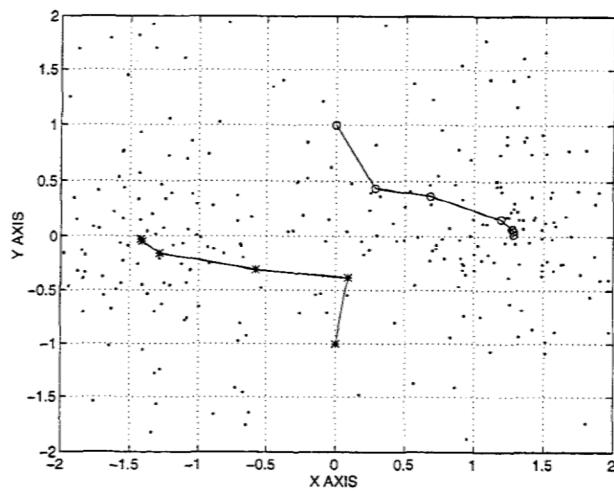


Figure 5. Root-FCMED trace with Cauchy clusters.

fuzzy median. For large sample sizes (2.0×10^5), a profile of the cpu usage was run for the ℓ_2 -FCM-AO and FCMED. For this rough profile, the ℓ_2 -FCM-AO exemplar estimation routines cpu usage was about 1.6 seconds / iteration. Exemplar estimation for the three FCMED algorithms took about twice the cpu time - 3.1 seconds/iteration. Total cpu usage for each program was: 300 seconds for the FCM, 800 seconds for the bisection-FCMED, 900 seconds for the remedian-FCMED, and 1100 seconds for the exact-FCMED. Again, these times are very rough estimates. A surprising result was obtained with a 128x128, 256-level grayscale image when clustering with only a small number of integer intensities in the images. Here, the exact-FCMED ran *faster* than the ℓ_2 -FCM-AO because once the data is ordered, the calculation of the fuzzy median is fast and the ordering itself is faster since there are fewer interchanges. In addition, the calculation of the membership matrix is faster since more exemplars were exactly equal to the image values, so no complicated membership update calculation was required for step 3 of the algorithm. So for integer-valued domains, the rough complexity estimates and simulation profiles for real-valued domains are not valid. For large real-valued samples where the space complexity is a concern, the bisection-FCMED is the best choice since it is faster than the exact-FCMED and its has low space complexity. For small sample sizes, the exact FCMED is the best choice. To this point in the testing, no example has been found where the remedian-FCMED has the best performance.

5. Conclusions

The application of the FCMED clustering algorithm to large data sets may require the approximation of the fuzzy median to speed the algorithm or to meet storage constraints. Two approximations to the fuzzy median have been described: the fuzzy remedian and the root-FCMED. Both the remedian-FCMED and the root-FCMED were described and illustrated with examples. Rough cpu timing profiles show that exemplar estimation in the ℓ_2 -FCM-AO is twice as fast as any implementation of the FCMED. For total cpu usage, the ℓ_2 -FCM-AO is four times faster than the exact-FCMED. For large samples sets, the total cpu usage of the root-FCMED appears to be faster and more accurate than the remedian-FCMED, which in turn is faster than the exact-FCMED. The remedian-FCMED traces strongly depend on the ordering of the samples, which can even lead to incorrect solutions. In contrast, the root-FCMED is independent of the sample presentation order. Although the remedian is a consistent estimate provided the samples are i.i.d.r.v.'s, consistency is only a limiting result, so the accuracy of the fuzzy remedian is not guaranteed. Accuracy results for the root-FCMED can only be estimated if the root is unique. Multiple roots might be a problem in certain circumstances, most notably if the membership values are restricted to some lattice. For integer-valued data sets, like grayscale images, the time complexity analysis does not apply and in fact, can be totally reversed from the real-valued analysis. Extensive testing is needed to quantify the performance differences between FCMED implementations and the ℓ_2 -FCM-AO and to reveal other performance issues for both real-valued and integer-valued data sets.

References

- [1] J. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.
- [2] J. Bezdek, "Pattern Recognition with Fuzzy Sets and Neural Nets," Tutorial Notes, Second IEEE International Conference on Fuzzy Systems, IEEE, 1993.
- [3] L. Bobrowski and J. Bezdek, "c-means clustering with the L_1 and L_∞ norms," *IEEE Trans. Syst., Man, Cybern.*, vol. 21, no. 3, pp. 545-554, 1991.
- [4] P. Kersten, "Fuzzy Robust Statistics for Application to the Fuzzy c-Means Clustering Algorithm, NUWC-NPT Technical Report 10,298, 1 December 1993.
- [5] _____. "Fuzzy Order Statistics and Its Application to Fuzzy Clustering, Naval Air Warfare Center Weapons Division, Technical Paper 8274, August 1995.
- [6] _____. "The Fuzzy Median and the Fuzzy MAD," *Proceedings of ISUMA-NAFIPS'95*, College Park, Maryland, September 17-20, 1995, pp. 85-88.
- [7] _____. "Fuzzy Order Statistics and Their Application to Fuzzy Clustering, Resubmitted to *IEEE Transaction on Fuzzy Systems*, November 1996.
- [8] R. Hathaway and J. Bezdek, "Optimization of Clustering Criteria by Reformulation," *IEEE Transactions on Fuzzy Systems*, Vol. 3, No. 2, pp. 241-245, May 1995.
- [9] Jajuga, K., " L_1 -norm based fuzzy clustering," *Fuzzy Sets and Systems*, Vol 39, 1991, pp. 43-50.
- [10] K. Engermann, H. Miller, and R. Yager, "Decision Making Using the Weighted Median Applied to Risk Management," *Proceedings of ISUMA-NAFIPS'95*, College Park, Maryland, USA, September 17-20, 1995, pp. 9-13.
- [11] E. Dougherty and J. Astola, *An Introduction to Nonlinear Image Processing*, SPIE Optical Engineering Press, Volume TT 16, Bellingham, Washington, 1994
- [12] D. Birkes and Y. Dodge, *Alternative Methods of Regression*, Wiley, New York, 1993.
- [13] R. Randles, and D. Wolfe, *Introduction to The Theory of Nonparametric Statistics*, John Wiley, New York, 1979.
- [14] G. Klir and B. Yuan, "Fuzzy Sets and Fuzzy Logic, Theory and Applications," Prentice-Hall, Upper Saddle River, New Jersey, 1995.
- [15] P. Rousseeuw and G. Bassett, Jr. "The Remedian: A Robust Averaging Method for Large Data Sets," *Journal of the American Statistical Association*, March 1990, Vol. 85, No. 409, pp. 97-104.
- [16] P. Kersten, "The Fuzzy Remedian," Naval Air Warfare Center Weapons Division, China Lake, Technical Paper, NAWCWPNS TP 8301, May 1996.
- [17] F. R. Hampel, P. J. Rousseeuw, E. M. Ronchetti, and W. A. Stahel. *Robust Statistics, The Approach Based on Influence Functions*, John Wiley, New York, 1986.