A Study on Cluster Size Sensitivity of Fuzzy *c*-Means Algorithm Variants

László Szilágyi[1,2,3(✉)], Sándor M. Szilágyi[2,4], and Călin Enăchescu[4]

[1] Faculty of Technical and Human Science of Tîrgu Mureş,
Sapientia - Hungarian Science University of Transylvania, Tîrgu Mureş, Romania
lalo@ms.sapientia.ro
[2] Department of Control Engineering and Information Technology,
Budapest University of Technology and Economics, Budapest, Hungary
[3] University of Canterbury, Christchurch, New Zealand
[4] Department of Informatics, Petru Maior University of Tîrgu Mureş,
Tîrgu Mureş, Romania

Abstract. Detecting clusters of different sizes represents a serious difficulty for all *c*-means clustering models. This study investigates the set of various modified fuzzy *c*-means clustering algorithms within the bounds of the probabilistic constraint, from the point of view of their sensitivity to cluster sizes. Two numerical frameworks are constructed, one of them addressing clusters of different cardinalities but relatively similar diameter, while the other manipulating with both cluster cardinality and diameter. The numerical evaluations have shown the existence of algorithms that can effectively handle both cases. However, these are difficult to automatically adjust to the input data through their parameters.

Keywords: Fuzzy clustering · Cluster size sensitivity · Improved partition · Suppressed partition

1 Introduction

Fuzzy clustering algorithms are unsupervised learning classification algorithms that employ fuzzy membership functions to describe the partition. The fuzzy *c*-means (FCM) algorithm introduced by Dunn [1] and generalized by Bezdek [2] is probably the most popular fuzzy clustering technique, due to its simplicity and the fine partition it usually produces. However, that fine property is relative and conditioned by several aspects. The probabilistic constraint causes several undesired phenomena, including the sensitivity to outlier data, and the multimodality of the fuzzy membership functions produced by FCM [3]. The latter also causes difficulties when the cardinality of clusters differs strongly, or the physical size (diameter) of the clusters is different. This is usually referred to as cluster size sensitivity.

The work of S.M. Szilágyi was supported by the János Bolyai Fellowship Program of the Hungarian Academy of Sciences.

A. Hirose et al. (Eds.): ICONIP 2016, Part II, LNCS 9948, pp. 470–478, 2016.
DOI: 10.1007/978-3-319-46672-9_53

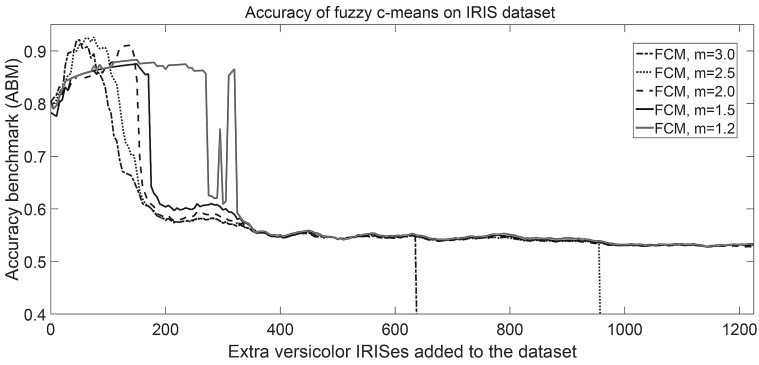

Fig. 1. Limited accuracy of FCM when the input data consist of the IRIS data set extended with synthetic versicolor items.

Problem Formulation. Figure 1 presents some accuracy benchmarks of the FCM algorithm, using a modification of the IRIS data set [4]. The original IRIS data set consists of 150 data vectors, each describing four different physical measures of individual iris flowers. As ground truth, these 150 vectors are divided into three clusters of 50 items each, named after the species of iris flowers: setosa, versicolor, and virginica. The modification consisted in generating further versicolor data vectors by averaging all possible couples of versicolor irises of the original data set. This way we have obtained $50 \times 49/2 = 1225$ further data vectors that are also supposed to belong to the versicolor class. These synthetic data vectors were gradually added to the original set of 150 vectors and fed to FCM at various settings of the fuzzy exponent m. Figure 1 shows us how the accuracy of FCM evolved, plotted against the number of synthetic vectors included into the input data. The formula of the employed benchmark indicator is given in Eq. (13). Obviously, as the number if synthetic data grows, the size of the clusters gets less and less balanced. For example at $m = 3$, 150 extra versicolor items are enough to cause the crash of the algorithm, as it is not able to establish the true boundary between the versicolor and virginica clusters. Further on, around 650 extra versicolor items, the boundary between setosa and versicolor is also mistaken. At lower values of m the accuracy is somewhat better, but there is no possible setting of the FCM algorithm which could accurately handle 400 extra versicolor vectors.

This study intends to provide a comparison of several existing extensions of the FCM algorithm that address the multimodality of the fuzzy membership functions, from the point of view cluster size sensitivity. The literature contains a wide spectrum of such algorithms including the suppressed FCM (s-FCM) [5], and its generalization gs-FCM [6], the FCM algorithm with improved partition (IFP-FCM) [7] and its generalized version GIFP-FCM [8], the FCMA algorithm [9], and the penalized FCM algorithm [10]. Literature also contains two variants of so-called cluster size insensitive FCM algorithms [11,12], which were not

included in the comparison, as in our consideration they have deviated far from the alternative optimization algorithm of the FCM algorithm.

2 Employed Clustering Algorithms

All fuzzy c-means algorithm variants and derivations employed in this study cluster a set of object data $\mathbf{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ into a predefined number of clusters denoted by c, through minimizing a quadratic objective function. Most of these algorithms derived their objective function from the one of FCM by adding certain penalty terms that would modify the behavior of the algorithm. In the following, we briefly present the repository of algorithms involved in this study.

Fuzzy c-means. The FCM algorithm minimizes

$$J_{\mathrm{FCM}} = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^{m} ||\boldsymbol{x}_k - \boldsymbol{v}_i||^2 = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^{m} d_{ik}^2, \tag{1}$$

constrained by the probabilistic condition $\sum_{i=1}^{c} u_{ik} = 1$, where v_i $(i = 1 \ldots c)$ represent the cluster prototypes, u_{ik} $(i = 1 \ldots c,\ k = 1 \ldots n)$ are the fuzzy membership functions that describe the degree to which vector $\boldsymbol{x}_k$ belongs to the cluster Ω_i represented by $\boldsymbol{v}_i$, and $m > 1$ is the fuzzy exponent [2]. The minimization of J_{FCM} is achieved via alternately applying the following partition and cluster prototype update formulas:

$$u_{ik} = \frac{d_{ik}^{-2/(m-1)}}{\sum_{j=1}^{c} d_{jk}^{-2/(m-1)}} \qquad \begin{matrix} \forall i = 1 \ldots c \\ \forall k = 1 \ldots n \end{matrix}, \tag{2}$$

$$\boldsymbol{v}_i = \frac{\sum_{k=1}^{n} u_{ik}^{m} \boldsymbol{x}_k}{\sum_{k=1}^{n} u_{ik}^{m}} \qquad \forall i = 1 \ldots c. \tag{3}$$

Fuzzy c-means with Improved Partition. The fuzzy c-means algorithm with improved partition were introduced by Höppner and Klawonn [7], and generalized by Zhu et al. [8]. In its generalized form (GIFP-FCM), the algorithm minimizes

$$J_{\mathrm{GIFP-FCM}} = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^{m} d_{ik}^2 + \sum_{k=1}^{n} a_k \sum_{i=1}^{c} u_{ik} (1 - u_{ik}^{m-1}), \tag{4}$$

subject to the probabilistic constraint, where a_k $(k = 1 \ldots n)$ are penalty terms. The optimization is achieved via alternately applying the partition update formula

$$u_{ik} = \frac{(d_{ik}^2 - a_k)^{-1/(m-1)}}{\sum_{j=1}^{c} (d_{jk}^2 - a_k)^{-1/(m-1)}} \qquad \begin{matrix} \forall i = 1 \ldots c \\ \forall k = 1 \ldots n \end{matrix}, \tag{5}$$

and the prototype update formula, which is identical with Eq. (3). The values of a_k terms are chosen at the beginning of each loop using the formula $a_k = \omega \times \min\{d_{ik}^2, i = 1 \ldots c\}$, with $\omega \in [0.9, 0.99]$.

Suppressed Fuzzy c-means Algorithms. Suppressed FCM (s-FCM) [5] was not introduced through the objective function it minimizes. Instead of that, s-FCM by definition performs an extra step between the application of Eqs. (2) and (3) that modifies the partition according to the formula:

$$u_{ik} \leftarrow \begin{cases} \alpha u_{ik} & \text{if } i \neq w_k \\ 1 - \alpha + \alpha u_{ik} & \text{if } i = w_k \end{cases} \qquad \begin{matrix} \forall i = 1 \dots c \\ \forall k = 1 \dots n \end{matrix}, \tag{6}$$

where $w_k = \arg\min_j \{d_{jk}^2, j = 1 \dots c\}$, and $\alpha \in [0, 1]$ is the so-called suppression rate. Suppressed FCM also received several generalization schemes denoted by gs-FCM [6], which made the suppression rate context sensitive, that is, dependent on k. Such algorithms also have a single extra parameter varying in the interval $[0, 1]$ which governs the choice of suppression rates α_k $(k = 1 \dots n)$. Based on previous performance analysis, in this study we have chosen to include the gs-FCM algorithm of type ξ, which defines the suppression rate as: $\alpha_k = [1 - (sin\frac{\pi u_{wk}}{2})^{\xi}][1 - u_{wk}]^{-1}$, with parameter $\xi \in [0, 1]$, where u_{wk} stands for the largest fuzzy membership function value of vector $\boldsymbol{x}_k$ provided by the FCM algorithm. Further details of the algorithm can be found in [6]. The objective function optimized by all suppressed FCM algorithms is given in [13].

FCMA by Miyamoto and Kurosawa. The FCMA algorithm optimizes

$$J_{\text{FCMA}} = \sum_{i=1}^{c} \sum_{k=1}^{n} \alpha_i^{1-m} u_{ik}^m d_{ik}^2, \tag{7}$$

subject to the probabilistic constraint of the fuzzy memberships, and of the extra parameters α_i $(i \dots c)$: $\sum_{i=1}^{c} \alpha_i = 1$ [9]. The minimization of J_{FCMA} is achieved via alternately applying the partition updating formula:

$$u_{ik} = \frac{\alpha_i d_{ik}^{-2/(m-1)}}{\sum_{j=1}^{c} \alpha_j d_{jk}^{-2/(m-1)}} \qquad \begin{matrix} \forall i = 1 \dots c \\ \forall k = 1 \dots n \end{matrix}, \tag{8}$$

the prototype update formula given in Eq. (3), and the extra formula:

$$\alpha_i = \frac{\sqrt[m]{\sum_{k=1}^{n} u_{ik}^m d_{ik}^2}}{\sum_{j=1}^{c} \sqrt[m]{\sum_{k=1}^{n} u_{jk}^m d_{jk}^2}} \qquad \forall i = 1 \dots c. \tag{9}$$

Penalized FCM by Yang. The Penalized FCM (PFCM) algorithm optimizes

$$J_{\text{PFCMA}} = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^m [d_{ik}^2 - \lambda \log \alpha_i], \tag{10}$$

subject to the probabilistic constraint of the fuzzy memberships, and of the extra parameters α_i $(i \dots c)$: $\sum_{i=1}^{c} \alpha_i = 1$ [10]. The minimization of J_{PFCM} is achieved via alternately applying the partition updating formula:

$$u_{ik} = \frac{[d_{ik}^2 - \lambda \log \alpha_i]^{-1/(m-1)}}{\sum_{j=1}^{c} [d_{jk}^2 - \lambda \log \alpha_j]^{-1/(m-1)}} \qquad \begin{matrix} \forall i = 1 \dots c \\ \forall k = 1 \dots n \end{matrix}, \tag{11}$$

the prototype update formula given in Eq. (3), and the extra formula:

$$\alpha_i = \frac{\sum_{k=1}^{n} u_{ik}^m}{\sum_{j=1}^{c} \sum_{k=1}^{n} u_{jk}^m} \qquad \forall i = 1 \ldots c. \tag{12}$$

3 Results and Discussion

The algorithms enumerated in Sect. 2 underwent thorough numerical evaluation using two different scenarios. The first test employed the IRIS data set with synthetic extension of a centrally located versicolor cluster, as described in Sect. 1. Data vectors were initially normalized, namely the values in each dimension of the feature vectors were linearly mapped upon the $[0, 1]$ interval. In case of all algorithms, we investigated the conditions of relatively good accuracy, meaning that the majority of setosa, versicolor, and virginica irises are assigned to three different clusters. In order to characterise the accuracy with a single numerical value, we propose the accuracy benchmark index defined as:

$$\mathrm{ABM} = \max_{p \in P_c} \left\{ \sum_{i=1}^{c} \frac{|\Lambda_i \cap \Omega_{p(i)}|^2}{|\Lambda_i| \times |\Omega_{p(i)}|} \right\}, \tag{13}$$

where Λ_i stands for the class i according to the grand truth, Ω_i represents the cluster with index i, P_c is the set of all possible permutations of numbers $1, 2, \ldots, c$, and $|\Psi|$ stands for the cardinality of set Ψ. ABM can range from 0 to 1: the maximum value indicates perfect separation of the ground true classes, while any deviance from the perfect separation is penalized.

Figure 2 exhibits the obtained ABM characteristics for various algorithms and settings, plotted against the number of synthetic versicolor irises added to the input data. All FCM algorithms with improved or suppressed partition have their limit around 350-400 synthetic vectors, above which they fail to distinguish the three clusters. On the other hand, for FCMA and PFCM there exists such a setting which can provide acceptable accuracy even in case of 1225 extra items. However, finding automatically these settings is not a trivial job. Table 1 shows some examples of confusion matrices obtained during the tests. The algorithms are ranked according to their performance. The fact that FCMA can stand at the top ($m = 1.5$) or the bottom ($m = 2$) of the ranking, clearly justifies the importance of well chosen parameter values. PFCM has an optimal λ value for both $m = 1.5$ and $m = 2$, but that is impossible to guess. There is no wide interval for λ, where PFCM has the ideal behavior, see Table 2.

While the first test employed classes of different cardinality without significantly changing the diameter of expected clusters, in the second example we have a different case. Let us define two collections of two-dimensional vectors, centered in $(-2, 0)$ and $(2, 0)$. The first circular group has a fixed radius of $r_1 = 1$ unit, and contains 100 randomly generated vectors distributed with uniform density. On the other hand, for the second circular group we will gradually change the radius r_2 from 1 to 3, while keeping the density of vectors constant.

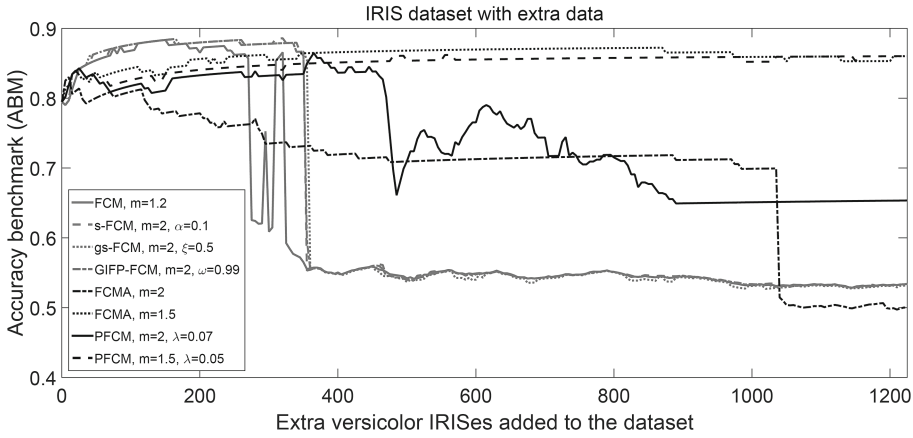

Fig. 2. Accuracy benchmark values obtained by various tested algorithm plotted against the number of extra versicolor irises.

Table 1. Confusion matrices obtained by various algorithms on the IRIS data set with 50, 150, 350, and 850 extra versicolor irises. Zeros are omitted

Algorithm and parameters	Ground truth	IRIS + 50			IRIS + 150			IRIS + 350			IRIS + 850		
		Ω_1	Ω_2	Ω_3	Ω_1	Ω_2	Ω_3	Ω_1	Ω_2	Ω_3	Ω_1	Ω_2	Ω_3
FCMA	setosa	50			50			50			50		
$m = 1.5$	versicolor		100			200			400			900	
	virginica		17	33		20	30		18	32		18	32
PFCM	setosa	50			50			50			50		
$m = 1.5$	versicolor		100			200			400			900	
$\lambda = 0.05$	virginica		18	32		20	30		20	30		20	30
PFCM	setosa	50			50			50			50		
$m = 2.0$	versicolor		100			200			400			899	1
$\lambda = 0.07$	virginica		18	32		23	27		22	28		45	5
s-FCM / gs-FCM	setosa	50			50			50			50		
$m = 2.0$ / $m = 2.0$	versicolor		100			200			396	4		488	412
$\alpha = 0.1$ / $\xi = 0.5$	virginica		14	36		14	36		14	36		2	48
GIFP-FCM	setosa	50			50			50			50		
$m = 2.0$	versicolor		99	1		200			395	5		474	426
$\omega = 0.99$	virginica		14	36		14	36		14	36		2	48
FCM	setosa	50			50			50			50		
$m = 1.2$	versicolor		99	1		200			208	192		484	416
	virginica		14	36		14	36		3	47		2	48
FCMA	setosa	50			50			49	1		48	2	
$m = 2.0$	versicolor		100			200			400			900	416
	virginica		21	29		27	23		35	15		38	12

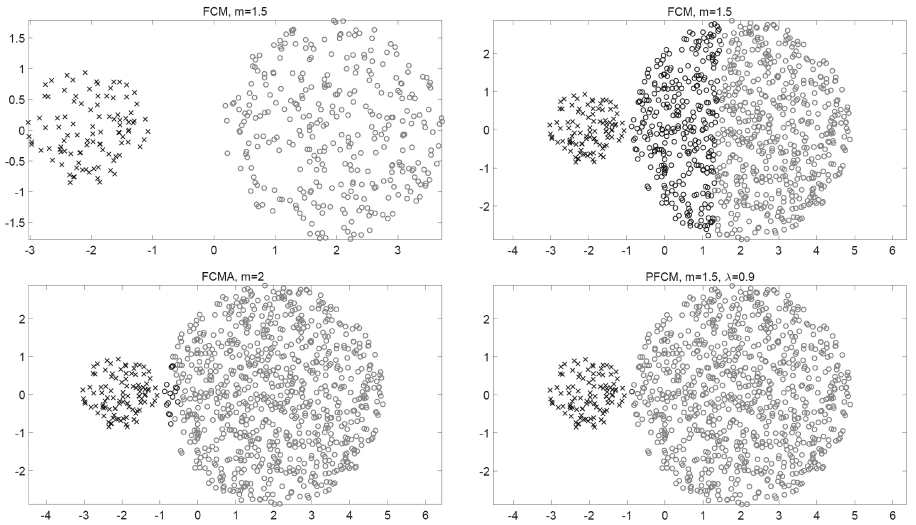

Fig. 3. Case of two clusters of different sizes: at $r_2 = 1.5$ even FCM can be accurate, but a more significant difference in cluster sizes require more sophisticated solutions, which are possible with FCMA and PFCM.

Table 2. Behavior of various tested algorithms in case of two clusters of different sizes

Algorithm	Parameters		Maximum r_2 with perfect accuracy	Misclassifications at $r_2 = 2$ out of 500	Misclassifications at $r_2 = 2.5$ out of 725	Misclassifications at $r_2 = 2.9$ out of 941
	m	other				
FCM	2.0		1.7	33	178	316
FCM	1.2		1.9	14	153	302
s-FCM	2.0	$\alpha = 0.1$	1.9	6	141	304
gs-FCM	2.0	$\xi = 0.8$	1.9	10	137	289
GIFP-FCM	2.0	$\omega = 0.9$	1.9	10	141	296
GIFP-FCM	1.5	$\omega = 0.9$	1.9	7	146	305
FCMA	2.0		2.7	0	0	16
FCMA	1.5		2.6	0	0	150
PFCM	2.0	$\lambda = 0.8$	2.4	0	4	127
PFCM	2.0	$\lambda = 0.9$	2.6	0	0	36
PFCM	2.0	$\lambda = 1.0$	2.1	0	5	11
PFCM	1.5	$\lambda = 0.8$	2.7	0	0	23
PFCM	1.5	$\lambda = 0.9$	2.8	0	0	1
PFCM	1.5	$\lambda = 1.0$	2.4	0	2	12

We will investigate, how the tested algorithms react to different sized clusters, and what circumstances or settings are required to achieve best accuracy. Figure 3 shows some examples of the clustering outcome, while Table 2 exhibits numerical information of the best performance achieved by each algorithm. FCM can provide two accurately separated clusters up to $r_2 = 1.7$, while improved and suppressed partitions can extend this behavior up to $r_2 = 1.9$. To achieve perfect separation of the two groups at $r_2 > 2$, one needs to turn to PFCM or FCMA. In the extreme case of $r_2 = 2.9$, the best result is achieved by PFCM with a single misclassification. Although the best outcome is provided by PFCM, FCMA can also be useful because it has no extra parameter compared to FCM, so it is much easier to tune than PFCM.

4 Conclusions

This study performed a comparative analysis of several modified and enhanced fuzzy c-means clustering in terms of sensitivity to cluster sizes. Two numerical tests were proposed, to assess the behavior of the algorithms both in case of clusters with different cardinality but relatively similar diameter, and in case of clusters that differ in both cardinality and diameter. The best performance was achieved by FCMA and PFCM, but even these are difficult to automatically tune to the input data. All other tested algorithms are significantly less effective.

References

1. Dunn, J.C.: A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. Cybern. Syst. **3**(3), 32–57 (1973)
2. Bezdek, J.C.: Pattern recognition with fuzzy objective function algorithms. Plenum, New York (1981)
3. Komazaki, Y., Miyamoto, S.: Variables for Controlling Cluster Sizes on Fuzzy c-Means. In: Torra, V., Narukawa, Y., Navarro-Arribas, G., Megías, D. (eds.) MDAI 2013. LNCS, vol. 8234, pp. 192–203. Springer, Heidelberg (2013)
4. Anderson, E.: The irises of the Gaspe peninsula. Bull. Am. Iris Soc. **59**, 2–5 (1935)
5. Fan, J.L., Zhen, W.Z., Xie, W.X.: Suppressed fuzzy c-means clustering algorithm. Patt. Recogn. Lett. **24**, 1607–1612 (2003)
6. Szilágyi, L., Szilágyi, S.M.: Generalization rules for the suppressed fuzzy c-means clustering algorithm. Neurocomput. **139**, 298–309 (2014)
7. Höppner, F., Klawonn, F.: Improved fuzzy partition for fuzzy regression models. Int. J. Approx. Reason. **5**, 599–613 (2003)
8. Zhu, L., Chung, F.L., Wang, S.: Generalized fuzzy c-means clustering algorithm with improved fuzzy partition. IEEE Trans. Syst. Man Cybern. B. **39**, 578–591 (2009)
9. Miyamoto, S., Kurosawa, N.: Controlling cluster volume sizes in fuzzy c-means clustering. In: SCIS and ISIS, Yokohama, Japan, pp. 1–4 (2004)
10. Yang, M.S.: On a class of fuzzy classification maximum likelihood procedures. Fuzzy Sets Syst. **57**(3), 365–375 (1993)

11. Noordam, J., Van Den Broek, W., Buydens, L.: Multivariate image segmentation with cluster size insensitive fuzzy *c*-means. Chemom. Intell. Lab. Syst. **64**(1), 65–78 (2002)
12. Lin, P.L., Huang, P.W., Kuo, C.H., Lai, Y.H.: A size-insensitive integrity-based fuzzy *c*-means method for data clustering. Patt. Recogn. **47**(5), 2024–2056 (2014)
13. Szilágyi, L.: A Unified Theory of Fuzzy c-Means Clustering Models with Improved Partition. In: Torra, V., Narukawa, T. (eds.) MDAI 2015. LNCS, vol. 9321, pp. 129–140. Springer, Heidelberg (2015)