



**AALBORG UNIVERSITY**  
DENMARK

**Aalborg Universitet**

## **Applying Fuzzy Possibilistic Methods on Critical Objects**

Yazdani, Hossein; Ortiz-Arroyo, Daniel; Choros, Kazimierz; Kwasnicka, Halina

*Published in:*

Proceedings of IEEE 17th International Symposium on Computational Intelligence and Informatics

*DOI (link to publication from Publisher):*

[10.1109/CINTI.2016.7846417](https://doi.org/10.1109/CINTI.2016.7846417)

*Publication date:*

2016

*Document Version*

Peer reviewed version

[Link to publication from Aalborg University](#)

*Citation for published version (APA):*

Yazdani, H., Ortiz-Arroyo, D., Choros, K., & Kwasnicka, H. (2016). Applying Fuzzy Possibilistic Methods on Critical Objects. In Proceedings of IEEE 17th International Symposium on Computational Intelligence and Informatics (pp. 271-276). IEEE Press. DOI: 10.1109/CINTI.2016.7846417

### **General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- ? Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- ? You may not further distribute the material or use it for any profit-making activity or commercial gain
- ? You may freely distribute the URL identifying the publication in the public portal ?

### **Take down policy**

If you believe that this document breaches copyright please contact us at [vbn@aub.aau.dk](mailto:vbn@aub.aau.dk) providing details, and we will remove access to the work immediately and investigate your claim.

# Comparing Fuzzy Possibilistic Methods on Critical Objects

Hossein Yazdani, Daniel Ortiz-Arroyo, Kazimierz Choros, Halina Kwasnicka  
Wroclaw University of Technology, Aalborg University

yazdanihossein@yahoo.com, doa@et.aau.dk, kazimierz.choros@pwr.edu.pl, halina.kwasnicka@pwr.edu.pl

**Abstract**—Providing a flexible environment to process data objects is a desirable goal of machine learning algorithms. In fuzzy and possibilistic methods, the relevance of data objects is evaluated and a membership degree is assigned. However, some critical objects have the potential ability to affect the performance of the clustering algorithms if they remain in a specific cluster or they are moved into another.

In this paper we analyze and compare how critical objects affect the behaviour of fuzzy possibilistic methods in several data sets. The comparison is based on the accuracy and ability of learning methods to provide a proper searching space for data objects. The membership functions used by each method when dealing with critical objects is also evaluated. Our results show that relaxing the conditions of participation for data objects in as many partitions as they can, is beneficial.

**Index Terms**—Data Object, Critical Objects, Fuzzy Possibilistic Method, Classification, Clustering, Membership Function.

## I. INTRODUCTION

Due to the growth of data in recent years, a more suitable searching space representation for data objects is necessary. Some data objects such as outliers, may affect the learned data model and its results. Similarly to outliers, critical objects are capable of changing the performance of the learning algorithms in case they treat them as any other data object. Dealing with critical objects in proper ways prevents algorithms from being affected by these objects.

Regardless of the type of learning methods, object recognition uses the most appropriate methodologies in its mining procedures as is done in pattern recognition [12]. The similarity and membership functions employed, should be completely commensurate with the type of data objects analyzed. Processing critical objects should not be treated in the way the algorithms treat regular objects.

In big data applications selecting the most accurate similarity [34] and membership assignments is crucial but we need to deal at the same time with the *curse of dimensionality* when processing big amounts of data. The searching strategies used should be adequate to the dimensionality of the vector and feature spaces. When considering the cost of big data processing is not enough to think only in classifying data objects into some classes, but learning methods should be also able to identify the most crucial objects in each data set. When considering assigning objects to clusters, miners should provide data objects the ability to participate in several clusters [2], [3], [13], [35].

This paper compares different methods for membership assignments in supervised and unsupervised methods. The paper also provides some examples to show the importance of tracking data objects' movement from one cluster to another.

## A. Learning Methods

In supervised machine learning, methods learn a model of the training data they process [21]. Classification is a form of supervised learning that performs a two-step process [23], [26], [32].

In unsupervised machine learning no training data is available. Clustering is a form of unsupervised learning that splits data into different groups or clusters by calculating the similarity between the objects contained in a data set [3], [24], [25], [27]. Consider a set of  $n$  objects represented by  $O = \{o_1, o_2, \dots, o_n\}$ , and each data object is typically described by numerical *feature – vector* data that has the form  $X = \{x_1, \dots, x_m\} \subset R^d$ , in  $d$  dimensional search or feature space.

In classification, the data set is divided into two parts: learning set  $O_L = \{o_1, o_2, \dots, o_l\}$  and test set  $O_T = \{o_{l+1}, o_{l+2}, \dots, o_n\}$ . In supervised methods, data objects are classified based on a class label  $x_l$ . A class or cluster is a set of  $c$  values  $\{u_{ij}\}$ , where  $u$  represents a membership value,  $i$  is the  $i^{th}$  object in the data set and  $j$  is the  $j^{th}$  class [37]. Similarity and membership assignment is performed in a similar way in supervised and unsupervised problems, as each data object needs to be compared and assigned a partial or full membership to each class or cluster.

Membership functions use similarity functions to evaluate and categorize data objects [22]. The most common membership functions that learning algorithms use are crisp, fuzzy, probability, possibility and bounded fuzzy possibilistic (BFPM) [6]. In this paper, we briefly explain how most common membership functions assign membership values to data objects [14] – [19].

## B. FCM Algorithm

There are two important types of fuzzy clustering (FCM) algorithms: one is based on the fuzzy partition of sample set and another is on the geometric structure of sample set

of kernel base method [2], [3], [13], [24]. The main FCM algorithm is generally defined as follow:

$$J_m(U, V) = \sum_{i=1}^c \sum_{j=1}^n u_{ij}^m \|X_j - V_i\|_A^2; \quad (1)$$

where  $U$  is the  $(n \times c)$  partition matrix,  $V = v_1, v_2, \dots, v_c$  is the vector of  $c$  cluster centers in  $\mathbb{R}^d$ ,  $m > 1$  is the fuzzification constant, and  $\|\cdot\|_A$  is any inner product  $A$ -induced norm [2], [3], [28], [36].

The rest of the paper is organized as follow. Section II analyzes the most common partitioning methods, comparing their membership functions. Critical objects and their effect on clustering are presented in III. This section also discusses the importance of keeping the searching space more flexible for this kind of objects. Some experimental results on some numerical examples are presented in section IV. Consequently, the conclusion is presented in V.

## II. ANALYSIS OF PARTITIONING METHODS

Crisp, fuzzy, probability, possibilistic and bounded fuzzy possibilistic partitioning are the most common partitioning methods on membership functions [2], [3]. Crisp clusters are non-empty, mutually-disjoint subsets of  $O$ :

$$M_{hcn} = \left\{ U \in \mathbb{R}^{c \times n} \mid u_{ij} \in \{0, 1\}, \forall j, i; \right. \\ \left. 0 < \sum_{i=1}^n u_{ij} < n, \forall j; \sum_{j=1}^c u_{ij} = 1, \forall i \right\} \quad (2)$$

where  $u_{ij}$  is the membership of the object  $o_i$  in cluster  $j$ . If the object  $o_i$  is a member of cluster  $j$ , then  $u_{ij} = 1$ ; otherwise,  $u_{ij} = 0$ . As this method shows, the object  $o_i$  can be a member of only one cluster. Fuzzy or probability clustering is different from crisp clustering, as each object can have partial membership in more than one cluster [1], [8] – [11]. This condition is stated in (3), where data objects may have partial nonzero membership in several clusters, but only full membership in one cluster.

$$M_{fcn} = \left\{ U \in \mathbb{R}^{c \times n} \mid u_{ij} \in [0, 1], \forall j, i; \right. \\ \left. 0 < \sum_{i=1}^n u_{ij} < n, \forall j; \sum_{j=1}^c u_{ij} = 1, \forall i \right\} \quad (3)$$

According to (3), each column of the partition matrix must sum to 1 ( $\sum_{j=1}^c u_{ij} = 1$ ) [2]. Based on this property of fuzzy clustering, as  $c$  becomes larger, the  $u_{ij}$  values must become smaller. In other words, fuzzy methods allow data objects to participate in more than one cluster. Each object can be a full member in one cluster, or being partial member in several clusters.

An alternative partitioning approach is *possibilistic clustering* [4], [6]. In (4) the condition  $\sum_{j=1}^c u_{ij} = 1$  is relaxed by substituting it with  $u_{ij} > 0$ .

$$M_{pcn} = \left\{ U \in \mathbb{R}^{c \times n} \mid u_{ij} \in [0, 1], \forall j, i; \right.$$

$$\left. 0 < \sum_{i=1}^n u_{ij} < n, \forall j; \sum_{j=1}^c u_{ij} > 0, \forall i \right\} \quad (4)$$

In possibilistic methods, data objects have more flexibility to obtain membership degrees. The objects can participate in more clusters as full or partial members. These methods have the advantage that they do not put any constraint on data objects, but there may be some problems in the implementation of the assignment algorithms [5], [6].

Bounded Fuzzy Possibilistic (BFPM) is introduced here to relax all conditions and provide the most flexible (diverse) search space by defining and assigning all constraints and boundaries for membership assignments. In other words, BFPM allows data objects to participate in as many clusters as they can by obtaining the most optimal membership degrees. Data objects have the capability of moving from one cluster to another (mutation) or participating in more clusters with no restriction. BFPM in (5) avoids that decreasing membership degrees are being assigned to data objects, as is done in fuzzy methods. It also defines boundaries and constraints on membership assignments with respect to other clusters to make the algorithm converge easily. This is an issue that other possibilistic methods may have difficulties with.

$$M_{bfpm} = \left\{ U \in \mathbb{R}^{c \times n} \mid u_{ij} \in [0, 1], \forall j, i; \right. \\ \left. 0 < \sum_{i=1}^n u_{ij} < n, \forall j; 0 < 1/c \sum_{j=1}^c u_{ij} \leq 1, \forall i \right\} \quad (5)$$

In conclusion, BFPM allows members to participate in all clusters to obtain full membership degree even in all clusters. Other possibilistic methods have some drawbacks such as offering trivial null solutions [5], [6], and a lack of upper and lower boundaries with respect to each cluster. Based on (2), (3), (4), and (5) it is easy to see that all crisp partitions are subsets of fuzzy partitions, and a fuzzy partition is a subset of a possibilistic partition. And finally, possibilistic partitions are subsets and equals of BFPM partitions if they follow the constraints and conditions presented by BFPM. i.e.,  $M_{hcn} \subset M_{fcn} \subset M_{pcn} \subseteq M_{bfpm}$  [28].

## III. CRITICAL DATA OBJECTS

Membership functions in fuzzy or possibilistic clustering, should allow data objects to participate in several clusters. However, we need to know if this is an advantage or disadvantage, since allowing data objects to participate in more clusters increases the computational complexity of the algorithms. In the following, we explain the reasons why in partitioning methods are important to provide some flexibility. The numerical examples presented show that lack of this consideration makes the membership functions fragile.

Assume  $U = \{u_{ij}(x) \mid x_i \in L_j\}$  is a function that assigns a membership degree to each point  $x_i$  to a line  $L_j$ , where a line

represents a cluster.

Now consider the crossing lines at the origin presented by (6).

$$AX = 0 \quad (6)$$

where matrix  $A$  is a  $n \times d$  coefficient matrix, and  $X$  is an  $d \times 1$  matrix, in which  $n$  is the number of lines and  $d$  is the number of dimensions. More precisely, Eq. (7) describes  $n$  with its different lines as a subspace.

$$\begin{bmatrix} C_{1,1} & C_{1,2} \\ C_{2,1} & C_{2,2} \\ \vdots & \vdots \\ C_{n,1} & C_{n,2} \end{bmatrix} \times \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (7)$$

Eq. (8) shows numerical examples of crossing lines at the origin, where  $X = 0$ ,  $y = 0$ ,  $x = y$ , and  $x = -y$  are some instances of those lines.

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & -1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (8)$$

Before discussing membership assignments by different partitioning methods, we will review one important property of spaces and subspaces from geometry. From a geometrical point of view, each line containing the origin is a subspace of  $R^d$ , where  $d$  is the number of dimensions. Without the origin, each of those lines is not a subspace, since the definition of a subspace comprises the existence of the null vector as a condition, in addition to other properties [20]. Now let us assign membership degrees to origin ( $x_{origin}$ ) with respect to all clusters. It should be noted that removing or decreasing a membership value, with respect to the origin of each cluster, ruins the subspace.

We start assigning membership degrees with crisp methods with respect to four lines presented in (8) as:

$$u'_1(x_{origin}) = 1, u'_2(x_{origin}) = u'_3(x_{origin}) = u'_4(x_{origin}) = 0$$

or

$$u'_2(x_{origin}) = 1, u'_1(x_{origin}) = u'_3(x_{origin}) = u'_4(x_{origin}) = 0$$

or

$$u'_3(x_{origin}) = 1, u'_1(x_{origin}) = u'_2(x_{origin}) = u'_4(x_{origin}) = 0$$

or

$$u'_4(x_{origin}) = 1, u'_1(x_{origin}) = u'_2(x_{origin}) = u'_3(x_{origin}) = 0$$

Where  $u'_1(x_{origin})$  presents the crisp membership degree assigned to origin with respect to the first line. By considering the above membership assignments, all lines can not be subspaces as the membership degree of the origin is zero in some cases. In this scenario, we must ruin almost all data objects subspaces. This may also happen in big data and high

dimensional search spaces [38], [39]. In fuzzy-based clustering we can create clusters using the points in all lines [14] using the condition presented by (3) and ( $\sum_{j=1}^c u_{ij} = 1$ ). According to this condition we calculate the membership degree of the origin with respect to each line [29], [30], [33].

$$u''_1(x_{origin}) = u''_2(x_{origin}) = u''_3(x_{origin}) = u''_4(x_{origin}) = \frac{1}{4}$$

Where  $u''_1(x_{origin})$  presents the fuzzy membership degree assigned to origin with respect to the first line. In fuzzy methods the origin can be a member of all lines, but with less degree of membership. The problem in high dimensional search spaces and big data is that the assigned fuzzy membership degree is at a fraction of the number of clusters  $n$  as  $\frac{1}{n}$ .

In possibilistic methods are more flexible, as objects are able to participate in more clusters with higher membership degrees. The data objects will be able to obtain appropriate degrees if the system is implemented based on the boundaries and constraints imposed by Bounded Fuzzy Possibilistic Method (BFPM), described in next section. In brief, BFPM allows all data objects to fully participate in even all clusters. Using BFPM in the crossing lines example, the membership degrees assigned to origin is calculated as follow:

$$u_1(x_{origin}) = u_2(x_{origin}) = u_3(x_{origin}) = u_4(x_{origin}) = 1$$

The membership degree assigned to the origin with respect to the first line  $u_1(x_{origin})$  and other lines shows that the origin has obtained the right membership degree as a full member of each line.

In conclusion, crisp membership functions are not able to assign membership values to data objects participating in more than one cluster. Fuzzy membership function reduces the membership values assigned to data objects with respect to each cluster. Possibilistic methods are able to assign the right membership degrees if the constraints and boundaries are used in the calculations and implementations.

As another example, let us assume  $U = \{u_{ij}(x)|x_i \in S_j\}$  is the membership function of some professionals who work in one or more sections of a company. Now consider that the company allows the members to participate in more clusters or sections to substitute someone if needed. In this scenario, data objects are able to participate in as many clusters as they may have potential abilities. To assign works and benefits to workers the system should be able to assign the right membership degrees with respect to each cluster. This may be difficult to do using other methods rather than BFPM. On the other side, these critical objects that can play key roles in a company, and might leave if they have not been treated properly.

Let us look at the critical objects from another point of view. Consider  $U = \{u_{ij}(x)|x_i \in C_j\}$  and  $U = \{u_{ij}(x)|x_i \in T_j\}$  as membership functions to assign membership degrees to human cells and transactions on the internet respectively. A system may wrongly classify cells with potential ability to

become cancer cell as members of a normal cluster, if it does not pay attention to its possible participation in the cancer cluster. The same situation happens when some transactions on the internet may be attacked but are clustered as normal transactions.

All these examples demonstrate why we should not prevent data objects from participating in as many clusters as they can [7]. On the other hand the method should be able to treat the critical objects in the proper way [7].

#### IV. EXPERIMENTAL RESULTS

In order to evaluate the accuracy and functionality of the proposed method on critical objects and data sets, we implement the BFPCM algorithm. Its accuracy is compared with other fuzzy and possibilistic methods on some data sets "Iris", "Pima Indians", "Yeast", "MAGIC", "Dermatology", and "Libras" shown by Table I.

TABLE I  
MULTI DIMENSIONAL DATA SETS

Dataset	Attributes	No. Objects	Clusters
Iris	4	150	3
Pima Indians	8	768	2
Yeast	8	1299	4
MAGIC	11	19200	2
Dermatology	34	358	6
Libras	90	360	15

##### A. BFPM Algorithm

BFPM provides a flexible searching space for data objects, additionally to clustering them as accurately as possible.

---

##### Algorithm 1 BFPM Algorithm

---

**Input:** X, c, m

**Output:** U, V

**Initialize V;**

**while**  $\max_{1 \leq k \leq c} \{ \|V_{k,new} - V_{k,old}\|^2 \} > \varepsilon$  **do**

$$u_{ij} = \left[ \sum_{k=1}^c \left( \frac{\|X_i - v_j\|}{\|X_i - v_k\|} \right)^{\frac{2}{m-1}} \right]^{\frac{1}{m}}, \quad \forall i, j \quad (9)$$

$$V_j = \frac{\sum_{i=1}^n (u_{ij})^m x_i}{\sum_{i=1}^n (u_{ij})^m}, \quad \forall j; \quad \left( 0 < \frac{1}{c} \sum_{j=1}^c u_{ij} \leq 1 \right). \quad (10)$$

**end while**

---

Equations (9) and (10) show how the algorithm calculates  $(u_{ij})$  and how the prototypes  $(v_j)$  will be updated in each iteration. The algorithm runs as long as the condition:

$$\max_{1 \leq k \leq c} \{ \|V_{k,new} - V_{k,old}\|^2 \} > \varepsilon$$

is true. The value assigned to  $\varepsilon$  is a predetermined constant that varies based on the type of objects and clustering problems.

U is the  $(n \times c)$  partition matrix,  $V = v_1, v_2, \dots, v_c$  is the vector of  $c$  cluster centers in  $\mathbb{R}^d$ ,  $m$  is the fuzzification

constant, and  $\|\cdot\|_A$  is any inner product A-induced norm [10], [11], and Euclidean distance function presented by 11.

$$D_E = \sqrt{\sum_{i=1}^d |X_i - Y_i|^2} \\ = \sqrt{(X_1 - Y_1)^2 + (X_2 - Y_2)^2 + \dots + (X_d - Y_d)^2} \quad (11)$$

where  $d$  is the number of features or dimensions, and  $X$  and  $Y$  are two different objects in  $d$  dimensional search space.

Table II shows a comparison of results between fuzzy, possibilistic, and BFPM algorithms. As table shows, BFPM works better than the other algorithms on presented data sets to cluster data objects.

Using the results obtained by BFPM, we will use a graphical

TABLE II  
COMPARED ACCURACY RATE WITH CONVENTIONAL FCM AND PCM ALGORITHMS THAT USE EUCLIDEAN DISTANCE FUNCTION.

Method-FCM	Iris	Pima Indian	Yeast	MAGIC
FCM	88.6%	74%	67.4%	54%
PCM	89.4 %	59.8%	32.8%	62%
BFPM	97.33%	99.9%	67.71%	100.0%

method to identify critical objects from data sets.

Figures 1 and 2 plot 150 data objects from Iris and MAGIC data sets and their membership degrees obtained by fuzzy methods. Upper points illustrate the membership degree obtained by data objects in their own cluster. The lower points show the membership degree that each data object has in its closest cluster.

Fig. 1. Fuzzy plot for Iris data objects for their own and the closest cluster.

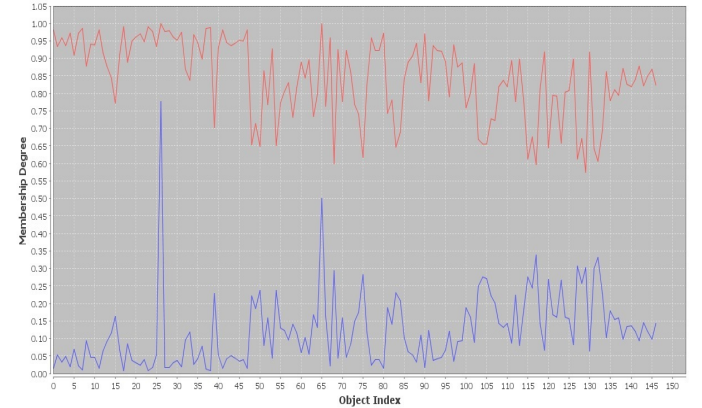


Fig. 2. Fuzzy plot for MAGIC data objects for their own and the closest cluster.

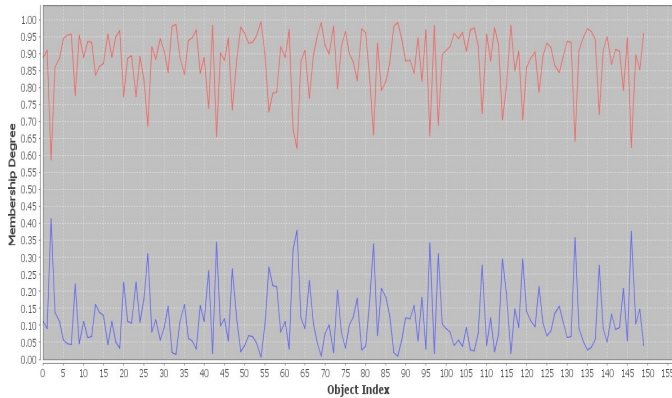


Fig. 3. BFPM plot for Iris data objects for their own and the closest cluster.

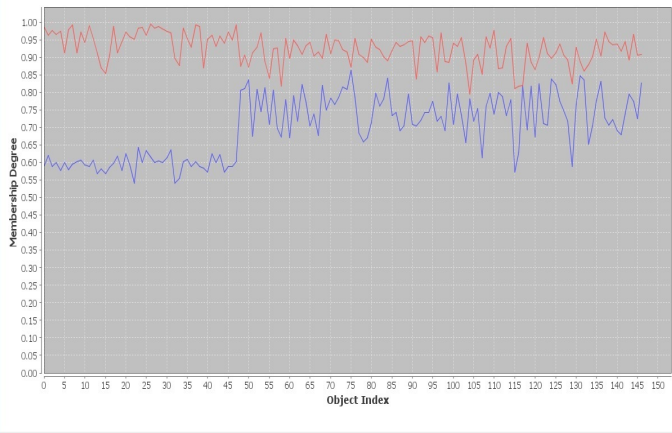
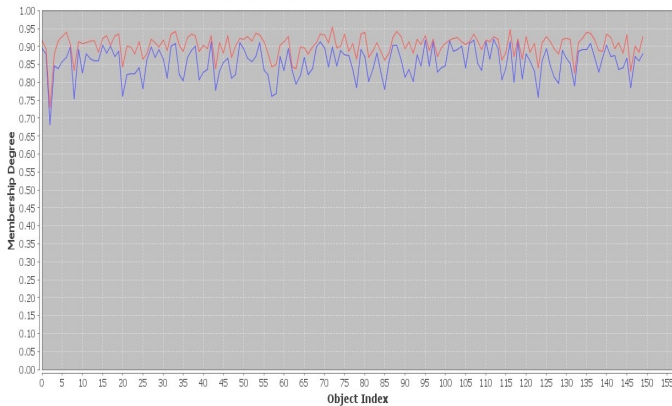


Fig. 4. BFPM plot for MAGIC data objects for their own and the closest cluster.



Figures 3 and 4 plot the same data objects from Iris and MAGIC data sets and their membership degrees obtained by BFPM method. The lower points indicate how much ability data objects have to move from their own cluster to the closest

cluster. By considering the vertical axes, we see that data objects obtained higher membership values with respect to each cluster in BFPM when compared to the fuzzy clustering case. This indicates that some objects will not change the performance of the algorithm if they are placed in its closest cluster instead of the one they were placed in. Additionally, we may identify critical objects as those that have a high but different degree of membership in several clusters. Moving these objects to a different cluster will change the performance of the algorithm. Contrarily, in fuzzy methods data objects are completely separated and it will be difficult to identify critical objects. Multi-objective optimization methods may be needed [31] for this.

## V. CONCLUSION

The paper evaluated learning methods with different criteria, such as their accuracy in classifying the objects. We highlighted the importance that learning methods should be able to allow data objects to participate in as many clusters as they can. The paper discussed critical objects and its influence in learning methods. Providing the most accurate searching space for this kind of object is another concept presented in this paper. Fuzzy, possibilistic and bounded fuzzy possibilistic learning methods and their membership assignments were compared with each other on some data sets. Obtained results from these methods were analysed about its ability to extracting the most critical objects. Results from bounded fuzzy possibilistic method show that this method is able to obtain better results in classifying data objects, besides providing the most flexible search space for data objects.

## REFERENCES

- [1] L.A. Zadeh, "Fuzzy Sets As A Basis For A Theory Of Possibility" North-Holland Publishing Company Fuzzy Sets and Systems 1, 1978.
- [2] L. Parsons, E. Haque, H. Liu, "Subspace Clustering for High Dimensional Data: A Review", *Sigkdd Explorations*, Vol. 6, pp. 90, 2004.
- [3] H. P. Kriegel, P. Kroger, A. Zimek, "Clustering High-Dimensional Data: A Survey on Subspace Clustering, Pattern -Based Clustering, and Correlation Clustering," in *ACM Transactions on Knowledge Discovery from Data*, Vol. 3, 2009.
- [4] R. Krishnaouram, J. M. Keller, "A Possibilistic Approach to Clustering," in *IEEE Transactions on Fuzzy Systems*, Vol. 1, 1993.
- [5] M. Barni, V. Cappellini, A. Mecocce, "Comments on "A Possibilistic Approach to Clustering"," *IEEE, Transaction on Fuzzy Systems*, Vol. 4, No. 3, 1996.
- [6] D.T. Anderson, J. C. Bezdek, M. Popescu, J. M. Keller "Comparing Fuzzy, Probabilistic, and Possibilistic Partitions," in *IEEE Transactions On Fuzzy Systems*, Vol. 18, No. 5, 2010.
- [7] H. Yazdani, D. Ortiz-Arroyo, H. Kwasnicka, "Issues On Critical Objects in Mining Algorithms", in *IEEE, AIPR*, pp. 53-58 , 2016.
- [8] L.A. Zadeh, "Toward Extended Fuzzy Logic- A First Step", in Elsevier, *Fuzzy Sets and Systems*, 3175-3181, 2009.
- [9] R. Timothy J. "Fuzzy Logic with Engineering Applications," Wiley 2nd Edition, Aug, 2004.
- [10] R. Xu, D. Wunsch, "Clustering," in *IEEE Press Series on Computational Intelligence*, 2009.
- [11] N. R. Pal, K. Pal, J. M. Keller, J. C. Bezdek, "A Possibilistic Fuzzy c-Means Clustering Algorithm ," in *IEEE Transactions On Fuzzy Systems*, Vol. 13, No. 4, 2005.
- [12] J. Bezdek, "Pattern Recognition with Fuzzy Objective Function Algorithm," in New York: Plenum, 1981.
- [13] T. C.Havens, R. Chitta, A. K. Jain, R. Jin, "Speed-up of Fuzzy and Possibilistic Kernel c-Means for Large-Scale Clustering," in *IEEE international Conference on Fuzzy Systems*, 2011.

- [14] S. H. Cha, "Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions", *International Journal Of Mathematical Models And Methods In Applied Sciences*, Issue 4, Vol. 1, 2007.
- [15] J. Looman, J.B. Campbell, "Adaptation of Sorensen's K For Estimating Unit Affinities In Prairie Vegetation," *Ecology*, Vol. 41, No. 3, 1960.
- [16] M. DEZA, E. DEZA, "Encyclopaedia of Distances", 2014.
- [17] E.F. Krause, "Taxicab Geometry An Adventure in Non-Euclidean Geometry," Dover, 1987.
- [18] X. Chen, X. Li, B. Ma, P. M.B. Vitanyi, "The Similarity Metric," *IEEE Transactions On Information Theory*, Vol. 50, No. 12, 2004. R. Cordeiro de Amorim,
- [19] D.G. Gavin, W.W. Oswald, E.R. Wahl, J.W. Williams, "A Statistical Approach To Evaluating Distance Metrics And Analog Assignments For Pollen Records," *Quaternary Research*, Vol. 60, Issue 3, pp. 356-367, 2003.
- [20] G. Strang, "Introduction to Linear Algebra", Wellesley-Cambridge Press, 2003.
- [21] T. Mitchell, "Machine Learning", Mc-Graw Hill, 1997.
- [22] H. Yazdani, H. Kwasnicka, "New Similarity Functions", *IEEE, AIPR*, pp. 46-52, 2016.
- [23] B. Taskar, E. Segal, D. Koller, "Probabilistic classification and clustering in relational data," In *Proc. Int. Joint Conf. Artificial Intelligence (IJCAI01)*, pp. 870878, Seattle,WA, 2001.
- [24] H. C. Huang, Y. Y. Chuang, C. S. Chen, "Multiple Kernel Fuzzy Clustering," *IEEE Transactions On Fuzzy Systems*, 2011.
- [25] X. Wang, Y. Wang, L. Wang, "Improving Fuzzy c-means Clustering Based on Feature-weight Learning," in *Elsevier B. V.* 2004.
- [26] C. Borgelt, "Prototype-based Classification and Clustering." *Magdeburg*, 22 June 2005.
- [27] F. Yang, T. Sun, C. Zhang, "An Efficient Hybrid Data Clustering Method Based on K-Harmonic Means and Particle Swarm Optimization," in *Elsevier Ltd*, 2009.
- [28] R. J. Hathaway, J. C. Bezdekb. "Extending fuzzy and probabilistic clustering to very large data sets," *Elsevier*, 2006.
- [29] B. Kosko "Fuzziness Vs. Probability," *International Journal, General Systems*, Vol. 17, pp. 211-240, 1990.
- [30] J. J. Kli, B. Yuan. "Fuzzy Sets and Fuzzy Logic Theory and Applications," *Printice Hall PTR*, 1995.
- [31] H. Yazdani, D. Ortiz-Arroyo, H. Kwasnicka, Multi-objective Particle Swarm Optimization Using Fuzzy Logic *Springer Int. Conf. Computer and Computational Intelligence*, pp. 224-233, 2011.
- [32] H. Yazdani, H. Kwasnicka, "Fuzzy Classification Method in Credit Risk," in *Springer Int. Conf. Computer and Computational Intelligence*, pp. 495-505, 2012.
- [33] G. Chen, T. Tat Pham, "Introduction to Fuzzy Sets, Fuzzy Logic and Fuzzy Control Systems," *CRC Press*, 2000.
- [34] P. Kumar, A. Johnson, "On A Symmetric Divergence Measure And Information Inequalities," *Journal of Inequalities in Pure And Applied Mathematics*, Vol. 6, Issue 3, Article 65, 2005.
- [35] T. F. Cox, M.A.A Cox, "Multidimensional Scaling," *Chapman and Hall/CRC 2nd Edition*, 2001.
- [36] "An Empirical Evaluation of Different Initialization on the Number of K-means Iterations," in *Lecture notes on Artificial Intelligence*, 2012.
- [37] J. Han, M. Kamber, "Data mining Concepts and Techniques," in *Elsvier Inc.* 2006.
- [38] L. Wang, C. Leckie, R. Kotagiri, J. Bezdek, "Approximate Pairwise Clustering For Large Data Sets Via Sampling Plus Extension", *Elsevier, Pattern Recognition*, pp. 222-235, 2010.
- [39] C. W. Tsai, C. F. Lai, H. C. Chao, A. V. Vasilakos "Big Data Analytics: A Survey", *Springer, Journal of Big Data*, 2015.