A generalization of Possibilistic Fuzzy C-Means Method for Statistical Clustering of Data

Souad Azzouzi
Mechanical engineering Laboratory,
Faculty of Sciences and techniques,
Sidi Mohamed Ben, Abdellah University
FEZ, Morocco
souad.azzouzi@usmba.ac.ma

Jaouad EL- Mekkaoui
TI Laboratory, Superior School of
Technology, Sidi Mohamed Ben
Abdellah University, Fez, Morocco

Amal Hjouji
TI Laboratory, Superior School of Technology,
Sidi Mohamed Ben Abdellah University,
Fez, Morocco

Ahmed EL Khalfi
Mechanical Engineering Laboratory
Faculty of Science and Techniques,
Sidi Mohamed Ben Abdellah University,
FEZ, Morocco

Received: June 11, 2021. Received: November 17, 2021. Accepted: December 2, 2021. Published: December 17, 2021.

Abstract— The Fuzzy C-means (FCM) algorithm has been widely used in the field of clustering and classification but has encountered difficulties with noisy data and outliers. Other versions of algorithms related to possibilistic theory have given good results, such as Fuzzy C- Means(FCM), possibilistic C-means (PCM), Fuzzy possibilistic C-means (FPCM) and possibilistic fuzzy C-Means algorithm (PFCM).This last algorithm works effectively in some environments but encountered more shortcomings with noisy databases. To solve this problem, we propose in this manuscript, a new algorithm named Improved Possibilistic Fuzzy C-Means (ImPFCM) by combining the PFCM algorithm with a very powerful statistical method. The properties of this new ImPFCM algorithm show that it is not only applicable on clusters of spherical shapes, but also on clusters of different sizes and densities. The results of the comparative study with very recent algorithms indicate the performance and the superiority of the proposed approach to easily group the datasets in a large-dimensional space and to use not only the Euclidean distance but more sophisticated standards norms, capable to deal with much more complicated problems. On the other hand, we have demonstrated that the ImPFCM algorithm is also capable of detecting the cluster center with high accuracy and performing satisfactorily in multiple environments with noisy data and outliers.

Keywords— Fuzzy Clustering, Fuzzy C-Means (FCM), ·Possibilistic C-Means (PCM); Fuzzy Possibilistic C-Means (FPCM), Possibilistic Fuzzy C- Means (PFCM).

I. INTRODUCTION

To classify the data in different groups according to one or more specific criteria, several methods have been adopted to solve this problem, Clustering is one of them, this method uses in most cases the Euclidean distance. There are 2 types of Clustering algorithms, rigid that is based on one object is in a cluster or not with a degree of membership equal to 0 or 1 [1],[2], this type of algorithm has shown its inefficiency in the case of overlapping of two clusters or, in particular, for points belonging to several clusters at the same time, we cite for example the K-Means algorithm. The other type of clustering is Fuzzy Clustering, which is more efficient method, widely used in several fields such as pattern recognition, image processing, also applicable in the security field for face and fingerprint detection, etc..., the principle of this method is simple: several data can belong to different clusters based on multiple membership values from interval [0,1],[3],[4]. Therefore, over the years, many algorithms appeared; the most used being the Fuzzy C- Means (FCM) algorithm [5]. This algorithm was first developed to deal with relational data, derived from Euclidean and other non-Euclidean distances. [6]. FCM algorithm is effective in the case of spherical clusters [7]. However, given the probabilistic type constraint used in FCM, this algorithm encounters the problem of sensitivity to noise and outliers [8],[9]. To correct this weakness, a new algorithm called possibilistic C-Means (PCM) which is based on the possibilistic approach was proposed [10]. Also, the PCM algorithm allows a better treatment of noisy data and outliers. Despite its efficiency, the PCM algorithm has a weak point in its sensitivity of initializations and choices of typing parameters and it can sometimes generate coincident clusters [11], another drawback of this algorithm is that it considers the typicities values but neglects memberships values which is also important as a parameter. Other versions of the PCM algorithm have been improved in order to correct some difficulties encountered in the PCM by modifying the objective function [12],[13],[14],[15]. Although PCM only uses typicalities values, the researchers tried conserving the highlights of FCM and those of PCM while mentioning the importance of membership and typicality values. [16],[17]. Hence, a new complementary model called the Fuzzy possibilistic C-Means (FPCM) clustering algorithm have been proposed, which merges the characteristics of the two algorithms FCM and PCM and allows to optimize them [18].

This algorithm has solved some of the problems encountered in the case of FCM and PCM, but like any algorithm, it has its strengths and also its vulnerabilities. The problem with FPCM is that it imposes a constraint on the typicality values, and also that the possibilistics values are very small when the size of the data set increases, which will lead to the development of a new and more powerful algorithm, called Possibilistic Fuzzy-C Means (PFCM). This new model was first proposed by [16],[19],[20],[21],[22] to simultaneously produce adhesions and typicities, the PFCM is a hybrid combination of the 2 objective functions of the FCM and PCM algorithms, this algorithm has strong points, overcoming the difficulties encountered in FCM, PCM and FPCM, it has solved the major problem of noise sensitivity encountered in the use of FCM, and that of overlap and coincidence of Clusters, which is the main problem of PCM. The PFCM algorithm was also found to be less sensitive to outliers. Since noise data influence the estimation of centroid, the PFCM algorithm simultaneously creates adhesions and typicities for each cluster with usual prototypes or cluster centers [14],[24],[25],[26].

Although the FCM, PCM, FPCM and PFCM algorithms are based on the Euclidean norm, the problem of using other more efficient norms occurs, for example the covariance norm, which creates ellipsoidal clusters that give better results with models and data structures, then a further problem is encountered, that of compactness, which can lead to the loss of several important data for clusters, such as efficient processing of noisy data or clustering of data with clusters of different sizes [27],[28],[29].

Another problem occurred with these algorithms when using the covariance norm is that FCM, PCM, FPCM and PFCM are unable to provide accurate prototypes even with their own data, in this work, we proposed a new more improved algorithm called ImPFCM (Improved possibilistic uzzy C- Means) to overcome these problems [30].

The ImPFCM algorithm uses functions from the norm; it is more efficient to handle complicated cases encountered as an obstacle for PFCM, especially in the case of noisy data. Hence, ImPFCM algorithm finds accurate cluster centers, its objective function uses norm functions, and it is flexible, efficient, and more suitable to different clustering concepts and constraints encountered in the above-mentioned algorithms, it is an algorithm capable of using the covariance norm with convenience.

This new algorithm gives a good solution to many of the most problems encountered with other previously mentioned algorithms.

After the introduction, the manuscript is exposed as follow: we start with the description of many clustering algorithms used frequently in clustering theory in sections II, we will present, our proposed improved ImPFCM algorithm, in section III, Finally, we close this manuscript by section IV with some experiments and results of ImPFCM algorithm and conclusion.

II. LITERATURE

In this section, we review the classical existing algorithms used to clustering data we focus attention of our studies to noisy environment with outliers.

A. *Fuzzy C- Means algorithm (FCM)*

The FCM algorithm is based on the principle of assigning memberships to x_k that are inversely related to the relative distance from x_k to c points prototypes $\{v_i\}$ these represent the cluster centers in the FCM model.

Before the development of the FCM, many algorithms were used such as k-means and C-means, when running these algorithms, the major problem encountered was the treatment of noise and outlying points which can be explained as follows: For two prototypes having the same distance from the center (equidistant), the value of belonging to each Cluster will be identical, whatever the absolute value of the distance between the two centroids (as well as between the other points of the data). Thus, two distant points which belong to the noises but which are the same distance from the Cluster centers, are selected to have equal membership values in the two Clusters, whereas in reality these two points have a very low membership that cancels each other out in relation to one of the Clusters.

In the case of the C -means algorithm, the membership of each data point to all classes is 1, which makes it suffer from the problem of noise sensitivity.

Thus the theory of fuzzy clustering was able to partially solve these problems.

The FCM algorithm is derived from the following optimization problem:

Minimize:

$$J(U,V,X) = \sum_{j=1}^{N}\sum_{i=1}^{c} u_{ij}^{m} \|x_j - v_i\|_{A'}^{2} \sum_{i=1}^{c} u_{ij} = 1, \sum_{i=1}^{N} u_{ij} > 0 \quad (1)$$

Where:

N: is number of data vectors,

$c \in (1, N]$: is number of clusters,

v_i: is center of the i^{th} cluster,

r : is dimension of the data,

x_j: is j^{th} data vector (j^{th} column of the data matrix $X_{r\times N}$), are also the data centers,

m : $1 < m < \infty$ is the degree of fuzziness,

A_{rxr} is a norm matrix.

This algorithm is characterized by a matrix $U = [u_{ij}]$ called the dimensional fuzzy partition matrix $c\, x\, n$, composed of the elements that represent the degree of belonging to the model x_k to each Cluster. m is degree of fuzziness, it measures the efficiency of the FCM algorithm on clustering performance [27].

The FCM algorithm has a very special mode of operation when compared to other partitioning algorithms, it works independently of the number of clusters existing in the data set, the FCM algorithm finds a fuzzy partition in a particular set of data. Furthermore, in the FCM algorithm, the sum of each column in the membership matrix U must be equal to 1, this is a main constraint which is the key element for this algorithm that characterizes it to other Clustering algorithms such as the case of C- means and K- means.

The FCM algorithm uses data point belongings that are related to the distance of the data point from the cluster centers. If a data point is at the same distance from the clusters, it will have the same membership value for each cluster. However, to deal with noise and outliers, the FCM algorithm is not the right algorithm, since the existence of at least one outlier can completely affect the result of partitioning in the FCM algorithm [8], which are the weak points of the FCM so in this case, the FCM does not differentiate between noise points or outliers which are also taken into account in the membership values which could cause a big problem later on which influences the final result obtained by the FCM. The second problem is that the FCM algorithm only detects spherical clusters. For non-spherical clusters, the FCM becomes inefficient [9].

B. Possibilistic C- Means algorithm (PCM)

To overcome and correct this weakness FCM algorithm, a new algorithm called PCM based sur possibilistic approach has been proposed [10] and which improves the column sum constraint is equal to 1.

$$M_{fcn} = \left\{U \epsilon\, M_{pcn} : \textstyle\sum_{i=1}^{c} u_{ik} = 1 \ \forall\, k; \sum_{k=1}^{n} u_{ik} > 0,\ \forall\, i\right\}$$

With constraint

$$0 < \sum_{i=1}^{c} u_{ik} \leq c$$

In other words, each element of the k^{th} column can be any number between 0 and 1, provided that at least one of them is positive. Therefore, the PCM has relaxed this constraint of the FCM and has succeeded in solving the problem of noise sensitivity. However, PCM tends to generate coincident clusters and is very sensitive to initializations [20], [21], [22].

The PCM algorithm is characterized by the degree of typicality with respect to the cluster which is more accurate when compared to the membership values interpreted by the FCM.

As the values of typicality with respect to one group do not depend on any of the prototypes of the other clusters [23], the degrees of typicality have been defined to solve this problem, by constructing prototypes characterizing the subcategories of data, taking into account the particularities of the points with respect to the other categories and also the similarities of the members of the category, therefore, the degree of typicality is an effective means of distinguishing the moderately atypical member of the group from the very atypical member. When compared to the FCM, the PCM algorithm relaxes the line sum constraint of the FCM algorithm, the PCM algorithm is characterized by a main constraint which is expressed as follows: each membership value in U can be between 0 and 1 or equal to one of them[24],

i.e., $0 \leq u_{ik} \leq 1$. These values are referred to as the data point types in each group. Thus, the objective function of the PCM algorithm can be formulated as follows:

$$J_{PCM}(V,U,X) = \sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^{m}\, d_{ik}^{2} + \sum_{i=1}^{c} \eta_i \sum_{k=1}^{n} (1 - u_{ik})^{m} \qquad (2)$$

With:

n: the total number of models in a data set,

c: is the number of Clusters,

m: The parameter that defines the fuzziness degree of the partition,

d_{ik}^2: the distance which can be Euclidean or not,

$U = [u_i]$: represents the fuzzy partition of the matrix X,

η_i: the typicity parameter estimated from the data. It is calculated as follows:

$$\eta_i = \frac{\sum_{k=1}^{n} u_{ik}^{m} \|x_k - v_i\|^2}{\sum_{k=1}^{n} u_{ik}^{m}} \qquad (3)$$

With:

n : is the total number of models in a data set

$m \in [1, \infty)$ is a parameter that defines the degree of Fuziness of the partition;

$X = \{x_1, \ldots, x_n\}$ are the characteristics of the data;

$V = \{v_1, \ldots, v_c\}$ represent the Cluster's centroid;

$U = [u_{ik}]$ is a fuzzy matrix partition composed of the degrees of membership of the x_k object of each cluster i.

For the PCM algorithm, the $\mathrm{u_{ik}}$ membership value will be calculated from the following equation:

$$u_{ik} = \left[1 + \frac{d_{ik}^{2}{}^{\frac{1}{m-1}}}{\eta_i}\right]^{-1} \quad (4)$$

d_{ik}^2 is the distance,
η_i is the parameter of typicity.

In the case of the PCM algorithm, the u_{ik} value should be interpreted as the typicity of x_k with respect to the cluster (rather than its cluster membership). For PCM, each line can be interpreted as a possibility of distribution on X.

The PCM algorithm helps to identify outliers and noisy data points. Although the PCM is efficient in dealing with noise, it has a drawback with its sensitivity of initializations and typing parameter choices and can generate coincident clusters [11]. Also, in the PCM algorithm, clusters do not have too much mobility because each data point is classified as a single set at a time, so the PCM produces inaccurate cluster centers when clusters are not of the same size or when the covariance standard is used.

In this way, as it has been observed, several disadvantage exist at the level of the 2 algorithms FCM and PCM, to overcome their weak points, the strong points of the FCM and PCM have been restored, given the importance of the values of typicity and membership, a new model called the Fuzzy possibilistic C-means Clustering FPCM algorithm has been proposed [17] which merges the characteristics of the two algorithms FCM and PCM and allows to optimize them.

C. Fuzzy Possibilistic C- Means algorithm (FPCM)

The objective function of the FPCM also contains membership and typicality values; however, they are represented as follows in equation (5):

$$J_{m,\eta}(U,T;V) = \sum_{i=1}^{c}\sum_{k=1}^{n}\left(u_{ik}^{m} + t_{ik}^{\eta}\right)\|x_k - v_i\|_A^2 \quad (5)$$

Subject to:

$$m > 1, \eta > 1, 0 \le u_{ik}, t_{ik} \le 1 \quad (6)$$

$$\textstyle\sum_{i=1}^{c} u_{ik}\ 1,\ \ \forall\, k \quad (7)$$

$$\sum_{k=1}^{n} t_{ik} = 1 \quad \forall\, k \quad (8)$$

Where m and η are the coefficients of Fuzziness and Typicity respectively.

According to the constraints 6, 7, 8 and the conditions of optimization of the $\sum_{i=1}^{c} u_{ik}$, considering the initial or extreme conditions of $J_{m,\eta}(U,T,V)$ and using Lagrange multipliers we obtain the following equations:

$$u_{ik} = \left[\sum_{j=1}^{c} \frac{d_{ik}^{2}{}^{\frac{2}{m-1}}}{d_{jk}^{2}}\right]^{-1}, 1 \le i \le c; 1 \le k \le n \quad (9)$$

$$t_{ik} = \left[\left[\sum_{j=1}^{c} \frac{d_{ik}^{2}}{d_{ij}^{2}}\right]^{\frac{2}{(\eta-1)}}\right]^{-1} \quad \forall\, i, k \quad (10)$$

$$v_i = \frac{\sum_{k=1}^{n}(u_{ik+}^{m} t_{ik}^{\eta}) x_k}{\sum_{k=1}^{n}(u_{ik+}^{m} t_{ik}^{\eta})}, \forall\, i \quad (11)$$

After running the FPCM algorithm, it was noted that the main problem with the FPCM algorithm is the constraint that corresponds to the sum of all typicality values of all data in the cluster, especially for a large data set [19].

Compared to the FCM, the FPCM algorithm has the same singularity values as the FCM, on the other hand compared to the PCM, the FPCM does not have the sensitivity problem that is a weak point of the PCM, but when the number of data is large, the typicality values (10) will be too low. Thus, typicality values should be scaled up, as in the case of FCM and PCM [25], [26].

Hence, scaling seems to be a palliative way to solve the problem of small values (which is caused by the line sum constraint on T, since scaled values do not have any additional information on data points. Thus, scaling is a good way to correct a mathematical flaw in the FPCM. This FPCM algorithm is unreliable and also has some defects such as FCM and PCM, to overcome this problem a new algorithm has been introduced called Possibilistic Fuzzy C-Mean.

The problem with the FPCM is that it imposes a constraint on the typicities values from the moment when the sum of the typicities values on all data points of a particular cluster is 1, this constraint is relaxed on the typicity values placed normally on the row while keeping the constraint on the membership values placed on the column[27].

D. Possibilistic Fuzzy C- Means algorithm (PFCM)

The PFCM algorithm was initially proposed by [19], it is an algorithm that has some strong points that overcome the difficulties encountered by FCM, PCM and FPCM algorithms. Principally, the PFCM solves the problem of cluster overlap, it should also be noted that this algorithm is less sensitive to outliers. PFCM is a hybrid combination of the two objective functions of the PCM and FCM algorithms. The objective function of the PFCM is as follows:

$$J(U,T;V;X) = \sum_{j=1}^{N}\sum_{i=1}^{c}\left(c_{FCM}u_{ij}^{m} + c_{PCM}t_{ij}^{\eta}\right)\|x_j - v_i\|_A^2 + \sum_{i=1}^{c}\gamma_i\sum_{j=1}^{N}\left(1 - t_{ij}\right)^{\eta} \quad (12)$$

Under the constraints

$$\textstyle\sum_{i=1}^{c} u_{ij} = 1, \sum_{j=1}^{N} u_{ij} > 0 \quad (13)$$

$$c_{FCM} > 0, c_{PCM} > 0, m > 1, \eta > 1, 0 \le u_{ij}, t_{ij} \le 1 \quad (14)$$

The coefficients c_{FCM} et c_{PCM} are weighting values that determine the importance of the typicity values t_{ij} and the membership values u_{ij}.

N is the total number of the data set,
c: is the number of Clusters, $1 < c < n$
The coefficients c_{FCM}, c_{PCM} are constants that respectively define the importance of the membership and typicities values in the objective function.
$m > 1$, $\eta > 1$ and γ_i are constants defined according to the problem by the user.

$T = [t_{ij}]_{cxN}$ is the matrix of typicity values, is considered as an assignment of the typicity of N objects in c Cluster.
$U = [u_{ij}]_{cxN}$ is the matrix of cxN fuzzy partition.
u_{ij}: is the value of the fuzzy membership function for j$^{\text{th}}$ sample belonging to the i^{th} Cluster.
$X = [x_1, x_2, \dots, x_N] \subset \boldsymbol{R^s}$ is the set of data.
S: the space dimension.
$V = [v_1, v_2, \dots, v_c]$ is the matrix of $c\ x\ N$ cluster centers
v_i: is the cluster center.
$d_{ij} = \|x_j - v_i\|$ is the distance between x_j and the center of Cluster v_i.

The membership values u_{ij} have the same meaning as those used in the FCM algorithm. Similarly, the values of typicity t_{ij} have the same interpretations as those defined in the PCM model.

The constants $c_{FCM}\, et\, c_{PCM}$ must respects the following constraint:

$c_{FCM} + c_{PCM} = 1$, and establish the importance of the membership value u_{ij} and the typicity value t_{ij}, Therefore, if we reduce the importance (weight) of the membership value u_{ij} this necessarily forces us to reduce the importance of typicity to the same extent, it is also restrictive. Moreover, to guarantee optimal typicity it will depend on the importance of the value of b. So, by restricting $c_{FCM} + c_{PCM} = 1$, the flexibility of the model is compromised.

If $c_{PCM} = 0$ et $\gamma_i = 0$ for every i, then equation (12) is reduced to a FCM optimization problem, on the other hand if $c_{PCM} = 0$, then in this case we will be dealing with a PCM optimization problem. We will see further on that if $c_{PCM} = 0$ even if we don't set $\gamma_i = 0$ for all i, (12) becomes implicitly equivalent to the FCM model. As the FPCM, placed under the regular conditions of C-means optimization problems, we obtain the first order of the necessary conditions for the extrema of J, if we put $\|x_j - v_i\|_A > 0$ for any $i, j, m, \eta > 1$, and X contains at least c distinct data points, then $(U, T, V) \in M_{fcn} \times M_{pcn} \times \Re^p$ minimize J if and only if:

$$u_{ij} = \left[\sum_{k=1}^{c}\left(\frac{(\|\vec{x}_j - \vec{v}_i\|_A^2)}{(\|\vec{x}_j - \vec{v}_k\|_A^2)}\right)^{\frac{1}{m-1}}\right]^{-1}, \ 1 \le i \le c; \qquad (15)$$

k : the iteration index

$$t_{ij} = \left(1 + \left(\frac{c_{PCM}}{\gamma_i}(\|\vec{x}_j - \vec{v}_i\|_A^2)\right)^{\frac{1}{\eta-1}}\right)^{-1}, 1 \le i \le c;\ 1 \le j \le N \quad (16)$$

$$v_i = \frac{\sum_{j=1}^{N}\left(c_{FCM} u_{ij}^m \left(\|\vec{x}_j - v_i\|_A^2\right) + c_{PCM} t_{ij}^\eta \left(\|\vec{x}_j - v_i\|_A^2\right)\right)\vec{x}_j}{\sum_{j=1}^{N}\left(c_{FCM} u_{ij}^m \left(\|\vec{x}_j - v_i\|_A^2\right) + c_{PCM} t_{ij}^\eta \left(\|\vec{x}_j - v_i\|_A^2\right)\right)}, 1 \le i \le c \quad (17)$$

The FCM, PCM, FPCM and PFCM algorithms are well adapted models to the Euclidean norm, on the other hand the covariance norm creates ellipsoidal clusters which give better results with the models and data structures.

Another problem for the Euclidean norm occurs, when the units of the data rows are different, which shows that the Euclidean norm is not well adapted, whereas the covariance norm gives better results[28],[29],[30].

A second problem occurs in the maximum compactness obtained by minimizing the equations (2) and (12) to obtain a simple and linear update equation for the calculation of prototypes $\mathrm{u_{ik}}$ et t_{ik}, if we use $D_{ikA} = \|x_k - v_i\|_A$ instead of $\|x_k - v_i\|_A^2$ this requires the resolution of a non-linear equation for the prototypes, however if we use $(\|x_k - v_i\|^2)$ this can lead to the loss of several important cluster data in the case of processing noisy data or clustering data with clusters of different sizes [28].

Other problems encountered with these algorithms when using the covariance norm is that FCM, PCM, FPCM and PFCM are not able to provide accurate prototypes even with clean data, in this paper, we propose a new more general algorithm called ImPFCM to solve some of this shortcomings

III. Methodology

To make the PFCM more general and efficient, we proposed a new algorithm called Improved Possibilistic Fuzzy C -Means ImPFCM this algorithm is well performing.

The objective function of the IMPFCM is as follows:

$$J(U,T,V;X) = \sum_{j=1}^{N}\sum_{i=1}^{c}\left(c_{FCM} u_{ij}^m f_{i,FCM}\left(\|\vec{x}_j - \vec{v}_i\|_A^2\right) + c_{PCM} t_{ij}^\eta f_{i,PCM}\left(\|\vec{x}_j - \vec{v}_i\|_A^2\right)\right) + \sum_{i=1}^{c}\gamma_i \sum_{j=1}^{N}(1 - t_{ij})^\eta \qquad (18)$$

under the constraints

$$\sum_{i=1}^{c} u_{ij} = f_j, \sum_{j=1}^{N} u_{ij} > 0 \qquad (19)$$

The objective function of the algorithm ImPFCM is almost the same as the one presented in the case of the PFCM, the only existing difference is the replacement of the term $\|x_k - v_i\|_A^2$ in equation (12) by the two terms $f_{i,FCM}(\|x_k - v_i\|_A^2)$ and $f_{i,PCM}(\|x_k - v_i\|_A^2)$**.**

As we remark, our proposed ImPFCM objective function algorithm is flexible and more scalable efficaciously to the different concepts and constraints of clustering encountered in the previously mentioned algorithms, and which is also able to easily use another norm, for example, covariance norm.

The optimization of equation (16) is performed using Lagrange multipliers for which the following function is optimized.
Using Lagrange multipliers, we optimize equation (16) and obtain the following optimized function:

$$J^*(U,T,V;X) = \sum_{j=1}^{N}\sum_{i=1}^{c}\left(c_{FCM} u_{ij}^m f_{i,FCM}\left(\|\vec{x}_j - \vec{v}_i\|_A^2\right) + c_{PCM} t_{ij}^\eta f_{i,PCM}\left(\|\vec{x}_j - \vec{v}_i\|_A^2\right)\right) + \sum_{i=1}^{c}\gamma_i \sum_{j=1}^{N}(1 - t_{ij})^\eta + \sum_{j=1}^{N}\lambda_j\left(\sum_{i=1}^{c} u_{ij} - f_j\right) \qquad (20)$$

The value of the condition f_j is designed to prevent large clusters from pulling the centers of smaller ones, which is discussed in detail in [28], so we calculate the cluster centers as follows:

$$\frac{\partial J^*(U,T,V;X)}{\partial \vec{v}_i} = \sum_{j=1}^{N}\left(c_{FCM}u_{ij}^{m}f'_{i,FCM}\left(\|\vec{x}_j-\vec{v}_i\|_A^2\right)+c_{PCM}t_{ij}^{\eta}f'_{i,PCM}\left(\|\vec{x}_j-\vec{v}_i\|_A^2\right)\right)(A+A^T)(\vec{x}_j-\vec{v}_i)=0 \Rightarrow$$

$$\vec{v}_i^{k} = \frac{\sum_{j=1}^{N}\left(c_{FCM}u_{ij}^{m}f'_{i,FCM}\left(\|\vec{x}_j-\vec{v}_i^{k}\|_A^2\right)+c_{PCM}t_{ij}^{\eta}f'_{i,PCM}\left(\|\vec{x}_j-\vec{v}_i^{k}\|_A^2\right)\right)\vec{x}_j}{\sum_{j=1}^{N}\left(c_{FCM}u_{ij}^{m}f'_{i,FCM}\left(\|\vec{x}_j-\vec{v}_i^{k}\|_A^2\right)+c_{PCM}t_{ij}^{\eta}f'_{i,PCM}\left(\|\vec{x}_j-\vec{v}_i^{k}\|_A^2\right)\right)} \quad (21)$$

The ImPFCM algorithm converges when $\|\mathrm{U}^{(k+1)}-\mathrm{U}^{(k)}\| \le \varepsilon$ with:

k: index of the iteration

ε: is a predefined threshold.

The membership values $\mathrm{u_{ij}}$ and the typicity values $\mathrm{t_{ij}}$ are calculated as follows:

u_{ij} is calculated from the following derivation:

$$\frac{\partial J^*(U,T,V;X)}{\partial u_{ij}} = c_{FCM}mu_{ij}^{m-1}f_{i,FCM}\left(\|\vec{x}_j-\vec{v}_i\|_A^2\right)+\lambda_j=0 \Rightarrow u_{ij}=\left(\frac{-\lambda_j}{c_{FCM}mf_{i,FCM}\left(\|\vec{x}_j-\vec{v}_i\|_A^2\right)}\right)^{\frac{1}{m-1}}, \sum_{k=1}^{c}u_{kj}=f_j \Rightarrow$$

$$u_{ij} = \left[\sum_{k=1}^{c}\left(\frac{f_{i,FCM}\left(\|\vec{x}_j-\vec{v}_i\|_A^2\right)}{f_{k,FCM}\left(\|\vec{x}_j-\vec{v}_k\|_A^2\right)}\right)^{\frac{1}{m-1}}\right]^{-1} f_j \quad (22)$$

t_{ij} is calculated from the following derivation:

$$\frac{\partial J^*(U,T,V;X)}{\partial t_{ij}} = c_{PCM}\eta t_{ij}^{\eta-1}f_{i,PCM}\left(\|\vec{x}_j-\vec{v}_i\|_A^2\right)-\gamma_i\eta(1-t_{ij})^{\eta-1}=0 \Rightarrow$$

$$t_{ij} = \left(1+\left(\frac{c_{PCM}}{\gamma_i}f_{i,PCM}\left(\|\vec{x}_j-\vec{v}_i\|_A^2\right)\right)^{\frac{1}{\eta-1}}\right)^{-1}, \quad (23)$$

$$\gamma_i = K\frac{\sum_{j=1}^{N}u_{ij}^{m}\|\vec{x}_j-\vec{v}_i\|_A^2}{\sum_{j=1}^{N}u_{ij}^{m}} \quad (24)$$

IMPFCM is initialized by γ_i, U, V which are calculated from the PFCM algorithm.

In the case of the PFCM, based on the equation (15) if $\|\vec{\mathrm{x}}_{\mathrm{j}}-\vec{\mathrm{v}}_{\mathrm{k}}\|_A^2=0$, we will not be able to calculate the membership value u_{ij} neither the typicality value t_{ij}, which in our case will be indeterminate values. On the other hand for the IMPFCM algorithm this problem does not occur, so if we impose the condition $\mathrm{f_{i,FCM}(x)} \neq 0, \mathrm{f_{i,PCM}(x)} \neq 0 \quad \forall \mathrm{x}, \mathrm{i}$, in (22), (23) then the values u_{ij} and $\mathrm{t_{ij}}$ will always be well determined.

In equation (23) if indeed $c_{PCM}=0$, then $\mathrm{t_{ij}}=1$, therefore the term $\sum_{i=1}^{c}\gamma_i\sum_{j=1}^{N}(1-t_{ij})^{\eta}$ in equation (16) will be equal to 0 and equation (16) will become expressed in terms of the Euclidean norm and will have the following form :

$$J(U,T,V;X) = \sum_{j=1}^{N}\sum_{i=1}^{c}\left(c_{FCM}u_{ij}^{m}f_{i,FCM}\left(\|\vec{x}_j-\vec{v}_i\|_A^2\right)\right), (25)$$

under contraints $\sum_{i=1}^{c}u_{ij}=f_j, \sum_{j=1}^{N}u_{ij}>0$

Since the identity norm matrix (Euclidean distance) is inappropriate for data sets where the lines have different physical units and interpretations, then as we have already noticed, this equation will subsequently pose a problem in the case cited. This leads us to use the covariance norm matrix in the ImPFCM.

For optimal results, the matrix covariance norm was used which gives a dimensionless distance.

The ImPFCM algorithm is based on FCM and PFCM in it's initialization and execution, so the steps of ImPFCM are as follows:

1- The FCM is applied.

2- V and γ_i are calculated to initialize PFCM.

3-After the V's and γ_i's obtained from the PFCM that will be used to initialize the ImPFCM to obtain the most accurate cluster centers.

4- $\mathbf{T}$ et U are calculated from (22) and (23) considering $f_{i,FCM}(x)=f_{i,PCM}(x)=x \ \forall i.$

Once these matrices are computed, we notice that they are less sensitive to noise points than those obtained in the PFCM case from (15) because the noise contributions on the cluster centers are reduced using (17).

In other hand, in the case of PFCM, the minimization of the equation

$$J(U,T,V;X) = \sum_{j=1}^{N}\sum_{i=1}^{c}\left(c_{FCM}u_{ij}^{m}+c_{PCM}t_{ij}^{\eta}\right)\|\vec{x}_j-\vec{v}_i\|_A^2 + \sum_{i=1}^{c}\gamma_i\sum_{j=1}^{N}(1-t_{ij})^{\eta}, \sum_{i=1}^{c}u_{ij}=1, \sum_{j=1}^{N}u_{ij}>0 \quad (26)$$

gives us a greater compactness, the clustering will be impacted by the noise which influences the quality of the results obtained.

Therefore, the functions $f_{i,FCM}(x)$ et $f_{i,PCM}(x)$ help to overcome this problem, they are effective as long as they allow to dampen the noise on the prototypes,

The exponential function $exp\left(-\|\vec{x}_j-\vec{v}_i\|_A^2/L_i^2\right)$ where L_i^2 is the characteristic width of the i^{th} Cluster, very appropriate in the case of high noise or distant points.

The choice of the functions $\mathrm{f_{i,FCM}(x)}$ and $f_{i,PCM}(x)$ influences the convergence of equation (16), that's right why we have to choose them well and avoid functions with low convergence rate which give a too high computation time and wrong results.

However, there are other functions that have also shown their ability to attenuate the impact of noise on cluster centers and that can be used instead of the exponential function, for example:

$$f_{i,FCM}\left(\|\vec{x}_j-\vec{v}_i\|_A^2\right) = f_{i,PCM}\left(\|\vec{x}_j-\vec{v}_i\|_A^2\right) = \frac{1}{1+x}, \ \tan(x), \ \frac{1}{1+x^2}, \ \frac{1}{1+e^{-x}}, ln(1+x) \ , \ e^{-x} \quad (27)$$

In order to avoid high computational time, it should be noted that the type of functions used affects the convergence of equations (16)-(17) and functions with low convergence rates should be avoided for the reason cited above.

IV. EXPERIMENTS AND RESULTS

A. *Comparative study of FCM, PFCM, ImPFCM and algorithms*

In this section, we focus on the experimental analysis and testing ImPFCM performance algorithm, we need to compare it with FCM and PFCM algorithms using simple dataset containing noise and taken from [19] and presented as following in Table I:

Table I. Details of the six test datasets

	DATA 1	DATA 2	DATA 3	DATA 4	DATA 5	DATA 6
No. of clusters	3	4	6	10	6	7
No. of data points	240	460	480	800	675	714
No. of noise	2400	3000	1440	2400	3000	3000

The six datasets are presented as below in Fig. 1. With various ratios of noise points to the actual data points:

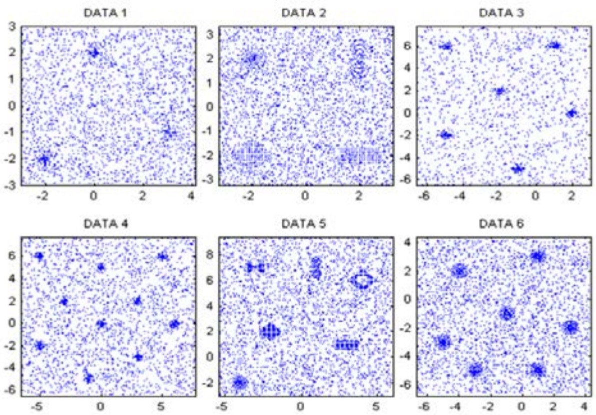


Fig. 1 The Six Datasets used for testing proposed clustering algorithms

To compare the performance of the three algorithms FCM, PFCM and ImPFCM, in terms of exactness to detect more precisely the clusters centers, we perform a clustering on these 6 datasets, we start with data 1.

***For DATA 1:** Based on the results obtained in Table II and Table III, we can deduce that FCM algorithm is not efficient to find the cluster center exactly, because it detects the approximate location of the prototypes near the potential clusters, while the PFCM algorithm reduces the noise effects and finds the approximate clusters centers, the ImPFCM method localizes the centers precisely compared to FCM and PFCM algorithms, Fig. 2.

Table II. Matrix of initial and FCM, PFCM, ImPFCM cluster centers

$V_{Initial}$	V_{FCM}	V_{PFCM}	V_{ImPFCM}
$\begin{bmatrix}-2.00 & 0.00 & 3.00\\ -2.00 & 2.00 & -1.00\end{bmatrix}$	$\begin{bmatrix}-1.0782 & 0.1895 & 2.3895\\ -1.2529 & 1.4256 & -0.7364\end{bmatrix}$	$\begin{bmatrix}-2.0142 & 0.0131 & 2.9561\\ -1.9970 & 2.0092 & -0.9842\end{bmatrix}$	$\begin{bmatrix}\mathbf{-2.0020} & \mathbf{0.0050} & \mathbf{2.9992}\\ \mathbf{-1.9899} & \mathbf{2.0008} & \mathbf{-0.9972}\end{bmatrix}$

Table III. Matrix of performance and error of FCM, PFCM, ImPFCM algorithms

Δ_{FCM}	E_{FCM}
$[0.6423 \quad 0.4261 \quad 0.3273]$	0.3691
Δ_{PFCM}	E_{PFCM}
$[[0.0047 \quad 0.0080 \quad 0.0081]]$	0.0043
Δ_{ImPFCM}	E_{ImPFCM}
$[0.0025 \quad 0.0040 \quad 0.0021]$	0.0024

\

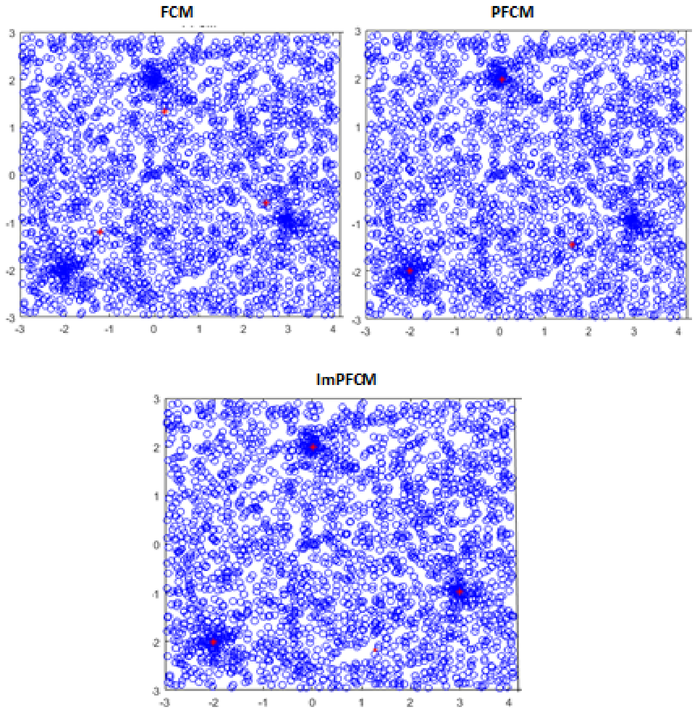


Fig. 2 Results of clustering DATA 1 using FCM, PFCM and ImPFCM, c=3, m=2

*For DATA 2

Table IV. Matrix of initial and FCM, PFCM, ImPFCM cluster centers for data 2

$V_{Initial}$	$\begin{bmatrix} -2.00 & -2.00 & 2.00 & 2.00 \\ -2.00 & 2.00 & -2.00 & 2.00 \end{bmatrix}$
V_{FCM}	$\begin{bmatrix} -1.6755 & -1.6401 & 1.6238 & 1.6812 \\ -1.6747 & 1.6322 & -1.6496 & 1.6600 \end{bmatrix}$
V_{PFCM}	$\begin{bmatrix} -2.0658 & -1.9500 & 1.9301 & 1.9270 \\ -2.0881 & 1.9295 & -2.0936 & 1.9141 \end{bmatrix}$
V_{ImPFCM}	$\begin{bmatrix} -2.0101 & -1.9976 & 1.9572 & 1.9392 \\ -2.0003 & 1.9813 & -2.0968 & 1.9350 \end{bmatrix}$

Table V. Matrix of performance and error of FCM, PFCM, ImPFCM algorithms for data 2

Results for	DATA2	DATA2
FCM	Δ_{FCM}	E_{FCM}
	$[0.2566 \quad 0.2877 \quad 0.2772 \quad 0.2409]$	0.1331
PFCM	Δ_{PFCM}	E_{PFCM}
	$[0.0259 \quad 0.0471 \quad 0.0560 \quad 0.0345]$	0.0212
ImPFCM	Δ_{ImPFCM}	E_{ImPFCM}
	$[0.0231 \quad 0.0453 \quad 0.0531 \quad 0.0327]$	0.0130

According to the results provided in Table IV and Table V, we can observe that the ImPFCM algorithm successfully detects the cluster centers accurately, while the PFCM algorithm performs well then FCM algorithm, but the ImPFCM algorithm performs satisfactorily despite the noisy environment.

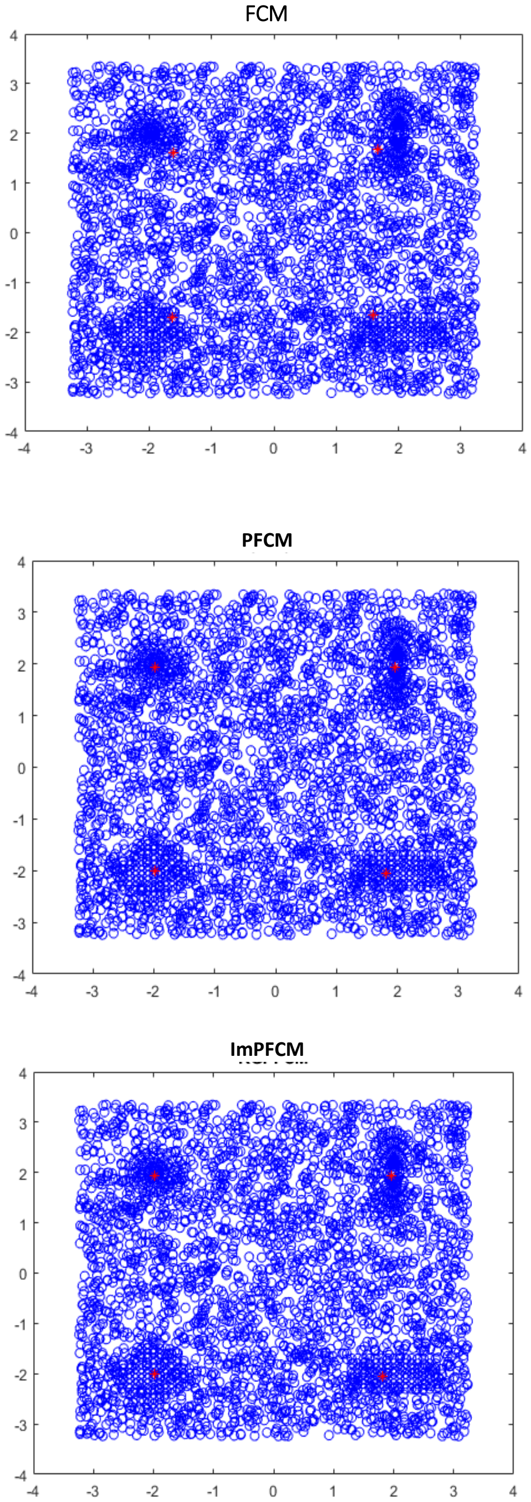


Fig. 3. Results of clustering DATA 2 using FCM, PFCM and ImPFCM, c=4, m=2

In data 2 clustering results, from Table IV, Table V and Fig.3, it's clear that ImPFCM algorithm outperformed in comparison with FCM and PFCM algorithms and the ImPFCM error is 88% smaller than FCM error. In addition, the ImPFCM error is 13% smaller than PFCM error.

***For Data 3:** Cluster centers and FCM, PFCM, ImPFCM algorithms performances for data 3 are presented

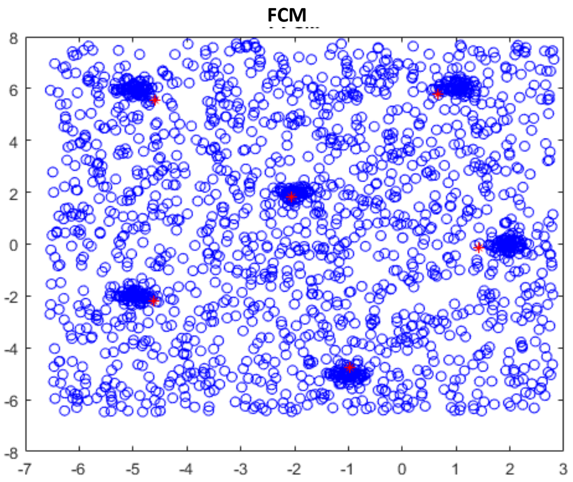
Table VI. Matrix of initial and FCM, PFCM, ImPFCM algorithms

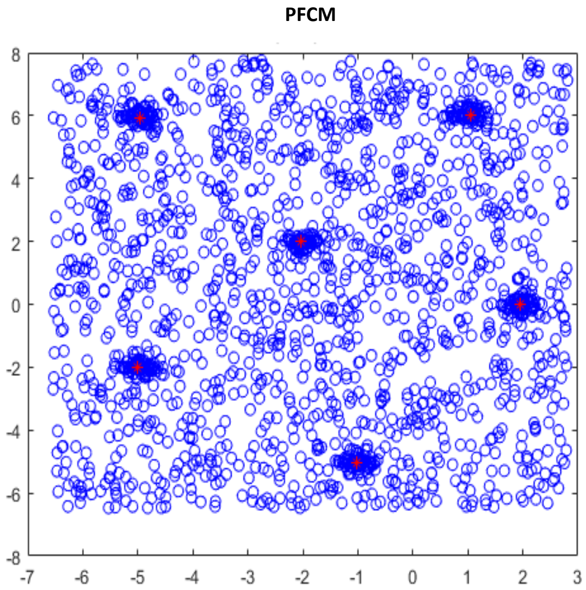
$V_{Initial}$	$\begin{bmatrix} -5.00 & -5.00 & -2.00 & -1.00 & 1.00 & 2.00 \\ -2.00 & 6.00 & 2.00 & -5.00 & 6.00 & 0.00 \end{bmatrix}$
V_{FCM}	$\begin{bmatrix} -4.8803 & -4.6551 & -1.7201 & -0.9354 & 0.7622 & 1.4856 \\ -2.1958 & 5.2001 & 1.7596 & -4.4536 & 5.6987 & -0.31 \end{bmatrix}$
V_{PFCM}	$\begin{bmatrix} -4.9902 & -4.9532 & -2.0436 & -1.0512 & 1.0487 & 1.9873 \\ -1.8793 & 5.8745 & 1.8963 & -5.0589 & 6.0487 & 0.0100 \end{bmatrix}$
V_{ImPFCM}	$\begin{bmatrix} -5.0001 & -4.9800 & -2.0115 & -1.0012 & 1.0322 & 1.9902 \\ -2.0010 & 5.9933 & 1.9765 & -5.0004 & 6.0000 & 0.0001 \end{bmatrix}$

Table VII. Matrix of performance and error of FCM, PFCM, ImPFCM algorithms

Results for	**DATA 3**	**DATA3**
FCM	Δ_{FCM}	E_{FCM}
	$[0.0841 \quad 0.1775 \quad 0.0481 \quad 0.1085 \quad 0.1154 \quad 0.1562]$	0.0627
PFCM	Δ_{PFCM}	E_{PFCM}
	$[0.0067 \quad 0.0175 \quad 0.0092 \quad 0.0115 \quad 0.0196 \quad 0.0160]$	0.0060
ImPFCM	Δ_{ImPFCM}	E_{ImPFCM}
	$[0.0060 \quad 0.0150 \quad 0.0080 \quad 0.0101 \quad 0.0160 \quad 0.0143]$	0.0044

Hence, for data 3, with a number of cluster centers equal to 6, and from the obtained tests results in Table.VI, Table VII, Fig.4, we can conclude that the ImPFCM algorithm is more performing to identifying exactly the cluster centers versus the FCM and PFCM algorithm, and, the ImPFCM error is 87% less than that of FCM and 19% less than PFCM error.





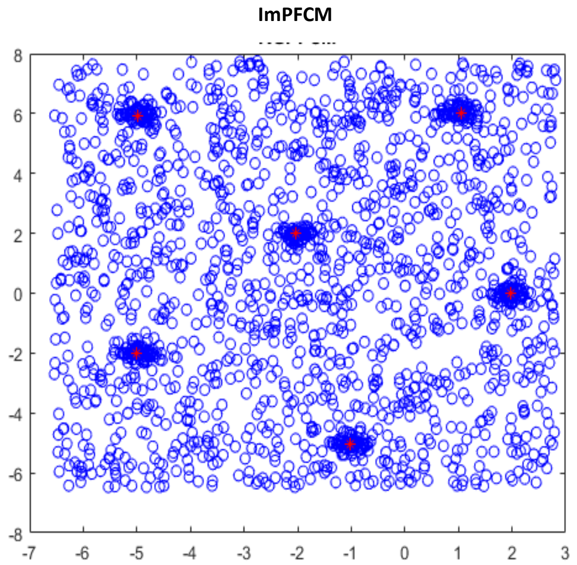


Fig. 4 Result of clustering DATA 3 using FCM, PFCM and ImPFCM, c=6, m=2 for data 3

***For Data5:** FCM, PFCM, ImPFCM algorithms Cluster centers and performances are presented in following Table IIX and Table IX. Fig.5.

Once again with six clusters, we observed that DATA 5 is clustered Fig. V. When using the ImPFCM algorithm, we notice that the cluster centres found are identical to the actual centres, Table IIX, Table IX, thus the ImPFCM accurately detected all six cluster centeres. Thereby, the ImPFCM again demonstrates superiority and performance over both the FCM and PFCM algorithms. As we remark, the ImPFCM error is 92% less than that of FCM, and 25% less than PFCM error.

Table IIX. Matrix of initial and FCM, PFCM, ImPFCM cluster centers for data 5

$V_{Initial}$	$\begin{bmatrix} -4.00 & -3.00 & -2.00 & 1.00 & 3.00 & 4.00 \\ -2.00 & 7.00 & 2.00 & 7.00 & 1.00 & 6.00 \end{bmatrix}$
V_{FCM}	$\begin{bmatrix} -3.2001 & -2.5018 & -1.8655 & 1.0912 & 2.8014 & 3.6007 \\ -1.0400 & 6.5289 & 2.1127 & 6.3904 & 0.4855 & 5.6498 \end{bmatrix}$
V_{PFCM}	$\begin{bmatrix} -3.9951 & -2.9501 & -2.0001 & 1.0002 & 2.9951 & 3.9856 \\ -1.9704 & 7.0189 & 2.0387 & 7.0001 & 0.9917 & 5.9800 \end{bmatrix}$
V_{ImPFCM}	$\begin{bmatrix} -4.0001 & -2.9912 & -2.0003 & 1.0004 & 2.9978 & 3.9850 \\ -1.9842 & 7.0002 & 2.0014 & 7.0000 & 1.0125 & 5.9564 \end{bmatrix}$

Table IX. Performance and error Matrix of FCM, PFCM, ImPFCM algorithms

Results for	**DATA5**	**DATA5**
FCM	Δ_{FCM}	E_{FCM}
	$[0.3547 \quad 0.1078 \quad 0.1003 \quad 0.1577 \quad 0.1260 \quad 0.0853]$	0.0833
PFCM	Δ_{PFCM}	E_{PFCM}
	$[0.213 \quad 0.0420 \quad 0.0212 \quad 0.0161 \quad 0.0080 \quad 0.0336]$	0.0082
ImPFCM	Δ_{ImPFCM}	E_{ImPFCM}
	$[0.200 \quad 0.0411 \quad 0.0201 \quad 0.0145 \quad 0.0071 \quad 0.0221]$	0.0074

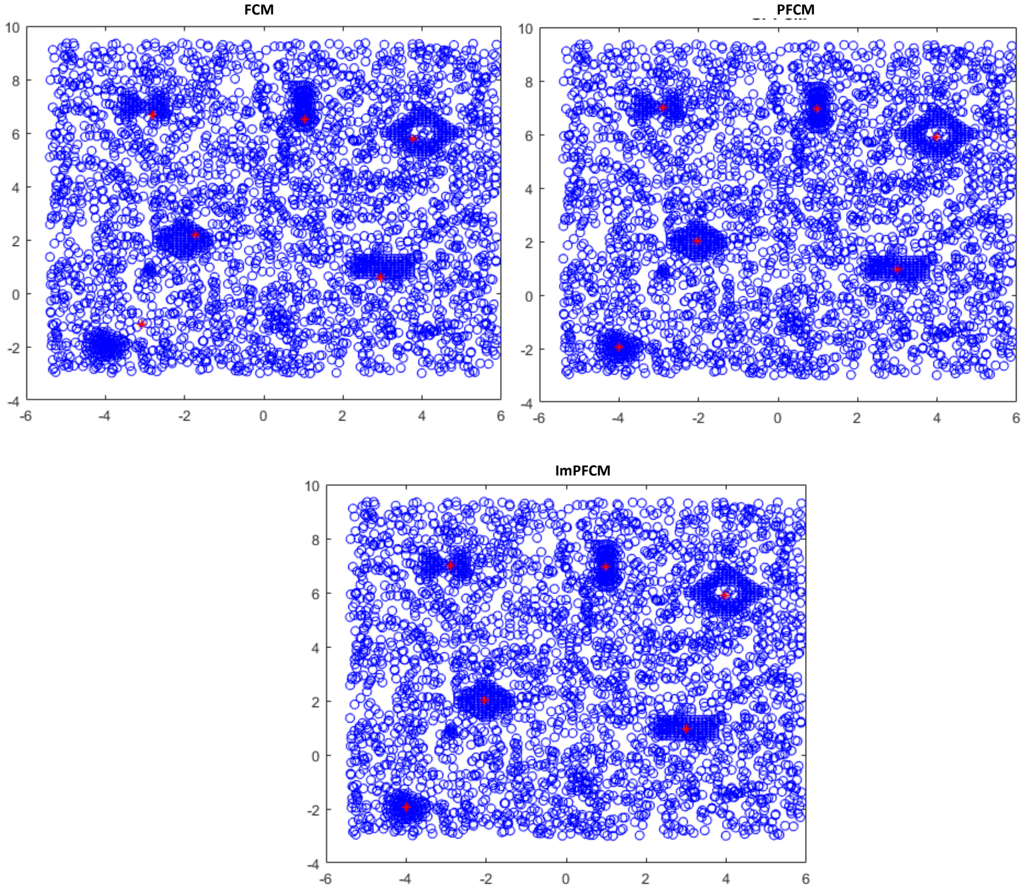


Fig. 5 Result of clustering DATA 5 using FCM, PFCM and ImPFCM algorithms, c=6, m=2

In order to investigate the ImPFCM algorithm performance in various environments, we scope the execution of the ImPFCM algorithm to other types of databases, and accordingly, we perform clustering on 6 clean and noiseless datasets and 13 real datasets using the FCM, PFCM and ImPFCM algorithms and then we compare them in both number of iterations and execution time, results are presented in following (Fig.6, Table X):

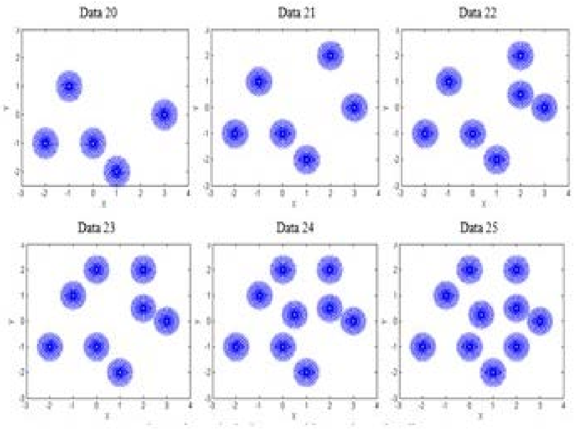


Fig. 6 Six synthetic data sets with no noise and outliers

Table X. Number of iterations and runtime (seconds) of the FCM, PFCM and ImPFCM algorithms.

			Number of Iteration			Runtime		
	Data Set	c	FCM	PFCM	ImPFCM	FCM	PFCM	ImPFCM
1	DATA1	3	22	39	39	3.210	12.946	12.946
2	DATA2	4	19	57	63	1.800	15.835	16.065
3	DATA3	6	14	39	39	1.127	12.946	12.946
4	DATA4	10	19	14	22	1.815	5.076	8.066
5	DATA5	6	31	24	32	4.186	8.341	11.875
6	DATA6	7	17	16	26	1.412	6.028	9.415
7	Climate Model Simulation Crashes	2	22	14	32	3.204	5.063	11.305
8	Connectionist Bench	2	18	42	50	1.540	12.950	15.678
9	Energy Efficiency	2	32	20	28	4.311	7.986	10.543
10	Fertility	2	21	16	32	3.020	6.043	11.592
11	Glass Identification	6	37	22	45	4.371	3.312	14.217
12	Haberman Survival	2	40	30	57	4.778	11.145	15.862
13	Heart Disease Cleveland	4	51	43	74	5.154	13.011	18.027
14	Ionosphere	2	53	7	35	5.416	3.084	12.087
15	IRIS	3	60	24	41	5.621	8.431	13.100
16	Pima Indians Diabetes	2	26	43	150	3.523	13.009	24.284
17	Seeds	3	33	27	46	4.157	10.041	13.632
18	Wine	3	27	37	100	3.623	12.015	20.321
19	Wisconsin Prognostic Breast Cancer	2	44	12	36	5 .167	5.032	12.524
20	DATA7	5	22	11	24	3.200	4.841	8.580
21	DATA8	6	26	17	19	3.567	6.517	6.957
22	DATA9	7	19	18	20	1.945	6.110	7.340
23	DATA10	8	29	20	23	3.713	7.954	7.520
24	DATA11	9	32	25	28	4.165	8.579	9.730
25	DATA12	10	44	28	32	5.221	10.722	11.678

In a further extension of the study to other types of databases, the FCM, PFCM and ImPFCM algorithms were applied to 25 synthetic and real datasets and compared in terms of execution time and number of iterations, as detailed below.

-Data sets 1 to 6 are the six synthetic noisy data sets studied above and shown in Fig. 1, Fig.3, Fig.4, and Fig.5.

-Data sets 7 to 19 are real data taken from the UCI Machine Learning Repository.

-Data sets 20 to 25 are the synthetic noise-free shown in Fig. 6 that contain different numbers of clusters to study the algorithms performance when there are large numbers of clusters in the data.

To examine the reliability of our ImPFCM algorithm, we started by measuring the efficiency of ImPFCM with thirteen real-world datasets with a known number of clusters, and then including previously known datasets such as: Climate Model Simulation Crashes, Connectionist Bench, Glass ID, Energy Efficiency, Fertility, Glass ID, Haberman Survival, Cleveland Heart Disease, Ionosphere, IRIS, Pima Indian Diabetes, Seeds, Wine, and Wisconsin Prognostic Breast Cancer, which are all taken from the UCI Machine Learning Repository.

In the case of real datasets, distributed between 7 and 19, and due to outliers, the FCM algorithm remains unable to detect cluster centres characterised by their large size, but in this case the PFCM finds dense clusters, but the ImPFCM algorithm is more efficient on this point compared to the PFCM algorithm, so it can detect large clusters with higher densities and presents remarkable accuracy results compared to the PFCM and FCM algorithms.

In another hand, there is no noise or outliers in the 20-25 datasets and their clusters have the same sizes. Therefore, the ImPFCM algorithm easily finds the actual cluster centers of these sets and also iterates only twice because the datasets are clean, and due to the interactions between the clusters Fig.VI, it corrects the insignificant displacements of the cluster centers calculated by the PFCM algorithm.

Based on the results in Table X, it deduced that the PFCM and FCM algorithms require relatively less execution time and iterations than the ImPFCM algorithm, this is due to the projection into another space and the non-linear cluster center update equation, two major reasons that influence the execution time of the ImPFCM algorithm,

On the other hand, according to what it noticed, sometimes, even if there is no noise or outliers in the data, FCM and PFCM algorithms remain limited in this case and cannot find the exact cluster centers due to the interactions between clusters, ImPFCM algorithm limits or cancels these interactions due to function introduced in the objective function and finds the cluster centers precisely. Hence, we can deduce that ImPFCM algorithm detects the cluster centers more precisely. We can deduce that, when using ImPFCM algorithm, although we lost some time but we gain in terms of accuracy.

B. The impact of Cluster sizes and density to ImPFCM algorithm

The quality of the clustering is indicated by the density, so a good clustering is equivalent to a high density which means that the cluster centers calculated by the algorithm are located in dense regions of the data. Therefore, we have a h igher density rho, when there are many data points around each cluster center, which is expressed using the following equation.

$$\rho = \frac{\sum_{j=1}^{n}\sum_{i=1}^{c} u_{ij}^{m} d_{ij}^{2}}{\sum_{j=1}^{n}\sum_{i=1}^{c} u_{ij}^{m}} \tag{28}$$

In order to evaluate the impact of different cluster sizes on the cluster centers, we notice that the cluster size influences the location of the cluster centers. Thus, when the cluster sizes are varied, the cluster centers are misplaced, and this even in noise-free datasets, because the larger clusters attract the cluster centers to their side. For this reason, if we experiment with the FCM, PFCM, and ImPFCM algorithms on a noise-free dataset composed of two clusters of different sizes as shown in Fig.7, we notice that, the FCM algorithm places the two cluster centers in the larger cluster. On the other hand, the PFCM algorithm identifies the center of the largest cluster accurately and the center of the smallest one as well, the ImPFCM algorithm, is initialized by the cluster centers computed by the PFCM algorithm, this algorithm also shows its efficiency in this point, cancelling the mutual interactions of the clusters and finding the real location of the cluster c.

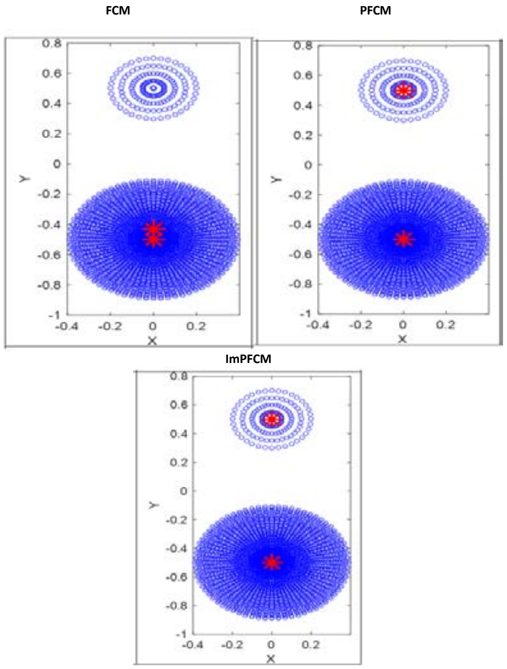
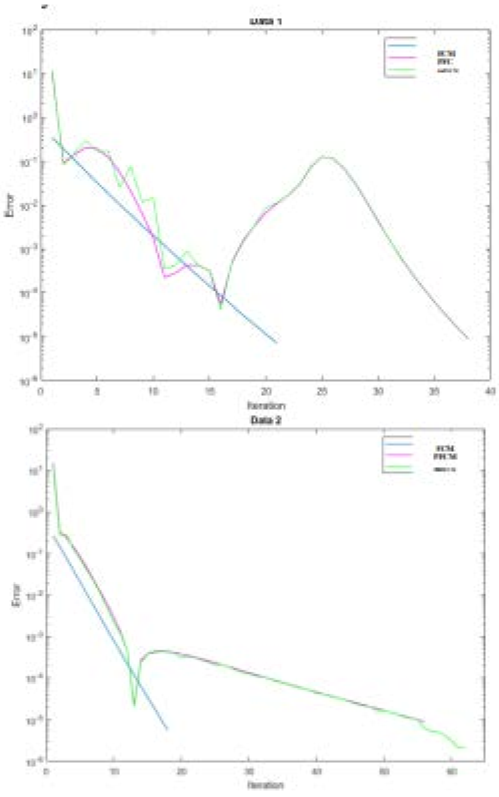


Fig. 7 Clustering the data with different cluster sizes by FCM, PFCM, ImPFCM algorithms.

C. Computational cost of the ImPFCM algorithm

Although the ImPFCM algorithm shows its performance and effectiveness to detect precisely cluster centers, this algorithm in its execution, is greedy in terms of consumption of execution time, and convergence, and finding the precise center of the clusters requiring a higher computational cost compared to the FCM algorithm, and this is due to the nonlinear nature of the equations contained in the objective function and their update, the high execution time of the ImPFCM algorithm compared to FCM, is caused by the complex and excessive computations for convergence, which is not the case for FCM and also because the functions are linear and do not require additional calculations. On the other hand, compared to algorithm PFCM, we notice that for testing FCM , PFCM and ImPFCM algorithms to some data set for example chosen Data 1 to data 6, it is noted that when the program start and the number of iteration was higher in threshold 7, Fig.8. the ImPFCM algorithm has relatively a small error of convergence compared to PFCM algorithm that explain a high precision to find exactly a real clusters centers, finding accurate cluster centers in presence of noise and outliers and finding accurate cluster centers when sizes of the clusters are considerably different, and this is due to using statistical method, so we can deduce that the ImPFCM algorithm improves PFCM and FCM algorithms and works satisfactorily.



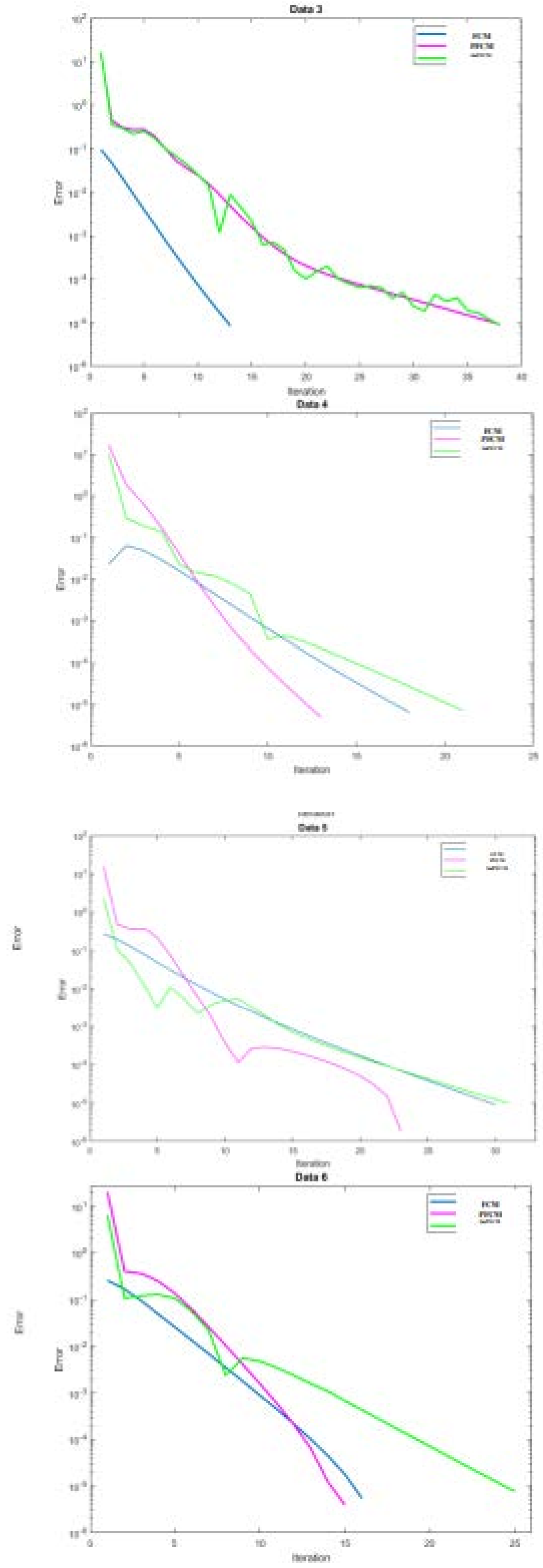


Fig. 8 Error expressed as a function of no. iterations required of FCM, PFCM and ImPFCM algorithms for different Data

V. CONCLUSION

In summary, the main contribution of the present paper is to improve performance of the well-known FCM and PFCM algorithms by proposing the ImPFCM algorithm. As mentioned in this manuscript and in other researches done previously, the existence of noise in the data, the precise determination of the center of the clusters, the outliers, the density, the different sizes of the clusters, the interactions between the clusters, have always constituted great obstacles for the clustering method, Therefore, several algorithms have been developed to solve these problems, for example as mentioned in this paper among others the algorithms FCM, PCM, FPCM, PFCM these last ones have not been able to determine the real centers of the clusters, and have not given satisfactory results, So in order to overcome these difficulties, a new algorithm named ImPFCM has been proposed which represent an improved version of PFCM, subsequently, it was also shown in this work that the proposed ImPFCM algorithm is more efficient and accurate, when the data is noisy and can accurately calculate the real centers of the clusters when the size of the clusters is significantly different and when there are interactions between these clusters, and, In particular, the ImPFCM algorithm finds the dense regions of the data more precisely when compared to the FCM and PFCM algorithms, which is indicated by the density of the clustering results, and it converges faster than the ImPFCM algorithm. Furthermore, the results of the comparative study algorithms indicates the performance and efficiency of ImPFCM algorithm to easily cluster the data sets in a space with a high dimension and to use not only Euclidean distance but more sophisticated norms able to deal with much more complicated problems. On the other hand, the ImPFCM algorithm performs satisfactorily even when the covariance norm matrix is used, in which case the FCM, PCM, PFCM algorithms fails to find accurate prototypes even in clean data, while the ImPFCM is insensitive to the cluster size and the type of covariance norm matrix, and work effectively in different environments with noisy data and outliers.

Acknowledgment

The author gratefully acknowledge the all members of scientific team which participated in this research.

References

[1] Cebeci, Z. and Yildiz, F., 2015. Comparison of k-means and fuzzy c-means algorithms on different cluster structures. Journal of Agricultural Informatics, 6, 13-23J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] Bora, D. J. and Gupta, A. K., 2014. A comparative study between fuzzy clustering algorithm and hard clustering algorithm. International Journal of Computer Trends and Technology, 10, 108-113.

[3] Zadeh, L., 1965. Fuzzy sets. Information and Control, 8, 338-353.

[4] Cebeci, Z., Kavlak, A.T. and Yıldız, F., 2017. Validation of fuzzy and possibilistic clustering results. International Artificial Intelligence and Data Processing Symposium, IEEE.

[5] Bezdek, J. C., 1981. Pattern Recognition with Fuzzy Objective Function Algorithms. New York: Plenum.

[6] R.N. Dave, S. Sen, Robust Fuzzy Clustering of Relational Data, IEEE Transactions on F uzzy Systems, 10 (2002) 713-727).

[7] Dzenan Gusic, Sanela Nesimovic, Automatization in Vague Database Relations via Lukasiewicz Fuzzy Implication Operator, pp. 445-459, WSEAS Transactions on Systems and Control, Volume 14, 2019.

[8] Şanlı, K. and Apaydın, A., 2006. Robust kümeleme yöntemleri. Anadolu Üniversitesi Bilim ve Teknoloji Dergisi, 7, 33-39.

[9] Ozdemir, O. and Kaya, A., 2018. K-medoids and fuzzy c-means algorithms for clustering CO2 emissions of Turkey and other OECD countries. Applied Ecology and Environmental Research, 16, 2513-2526.

[10] Krishnapuram, R. and Keller, J., 1993. A possibilistic approach to clustering. IEEE Transactions on Fuzzy Systems, 1, 98-110.

[11] (Barni et al.), M. Barni, V. Cappellini and A. Mecocci, "Comments on A Possibilistic Approach to Clustering", IEEE Trans. Fuzzy Systems, vol. 4, No. 3, 1996, pp. 393-396

[12] H. Timm, C. Borgelt, C. Doring, and R. Kruse, "Fuzzy cluster analysis with cluster repulsion," presented at the Euro. Symp. Intelligent Technologies (EUNITE), Tenerife, Spain, 2001.

[13] H. Timm and R. Kruse, "A modification to improve possibilistic fuzzy cluster analysis," presented at the IEEE Int. Conf. Fuzzy Systems,FUZZ-IEEE' 2002, Honolulu, HI, 2002).

[14] H. Timm, C. Borgelt, C. Doring, and R. Kruse, "An extension to possibilistic fuzzy cluster analysis," Fuzzy Sets Syst., vol. 2004, pp. 3–16).

[15] E. E. Gustafson and W. C. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," in Proc. IEEE Conf. Decision and Control, San Diego, CA,1979, pp. 761–766.

[16] N. R. Pal, K. Pal, and J. C. Bezdek, "A New Hybrid c-Means Clustering Model", Proceedings of the IEEE International Conference On Fuzzy Systems, vol. 1, 2004, pp. 179-184.

[17] N. R. Pal, K. Pal, and J. C. Bezdek, "A mixed c-means clustering model,"in IEEE Int. Conf. Fuzzy Systems, Spain, 1997, pp. 11–21.

[18] (Jafar, M. O. A. and Sivakumar, R., 2012. A study on possibilistic and fuzzy possibilistic c- means clustering algorithms for data clustering. International Conference on Emerging Trends in Science, Engineering and technology, 90-95.).

[19] Pal, N. R., Pal, K., Keller, J. M. and Bezdek, J. C., 2005. A possibilistic fuzzy c-means clustering algorithm. IEEE Transactions on Fuzzy Systems, 13, 517-530.

[20] Dzenan Gusic, Adis Alihodzic, Sanela Nesimovic, On Some Applications of h-generated Fuzzy Implications, pp. 490-507, WSEAS Transactions on Systems and Control, Volume 15, 2020.

[21] Snejana Yordanova, Milen Slavov, Georgi Prokopiev, Disturbance Compensation in Fuzzy Logic Control of

Level in Carbonisation Column for Soda Production, pp. 64-72, WSEAS Transactions on Systems and Control, Volume 15, 2020,

[22] Yassine Zahraoui, Mohamed Akherraz, Chaymae Fahassa, Induction Motor Performance Improvement using Twelve Sectors DTC and Fuzzy Logic Speed Regulation, pp. 47-56, WSEAS Transactions on Systems and Control, Volume 15, 2020

[23] (V.N. Vapnik, Statistical Learning Theory, Wiley, 1998), ISBN: 978-0-471-03003-4.

[24] Dzenan Gusic, Adis Alihodzic, Sanela Nesimovic, On Some Applications of h-generated Fuzzy Implications, pp. 490-507, WSEAS Transactions on Systems and Control, Volume 15, 2020.

[25] Snejana Yordanova, Milen Slavov, Georgi Prokopiev, Disturbance Compensation in Fuzzy Logic Control of Level in Carbonisation Column for Soda Production, pp. 64-72, WSEAS Transactions on Systems and Control, Volume 15, 2020,

[26] Yassine Zahraoui, Mohamed Akherraz, Chaymae Fahassa, Induction Motor Performance Improvement using Twelve Sectors DTC and Fuzzy Logic Speed Regulation, pp. 47-56, WSEAS Transactions on Systems and Control, Volume 15, 2020.

[27] Şahinli, F., 1999. Fuzzy set theory to cluster analysis approach. Master Thesis, Gazi University Science, Sciences Institute, Ankara, 119.

[28] L. Lin, P.W. Huang, C.H. Kuo, Y.H. Lai, A size-insensitive integrity-based fuzzy c-means method for data clustering, Pattern Recognition, 47 (2014) 2042-2056.

[29] Francesco A. Bharathi Sankar Ammaiyappan, R. Seyezhai,"Implementation of Fuzzy Logic Control based MPPT for Photovoltaic System with Silicon Carbide (SiC) Boost DC-DC Converter", WSEAS Transactions on Systems and Control, vol. 16, pp. 198-215, 2021

[30] Abdullah J. H. Al Gizi,"PLC Fuzzy PID Controller of MPPT of Solar Energy Converter", WSEAS Transactions on Systems and Control, vol. 16, pp. 1-20, 2021.