Generalization of Fuzzy C-Means Based on Neutrosophic Logic

Aboul Ella HASSANIEN[1]*, Sameh H. BASHA[2]*, Areeg S. ABDALLA[2]*

[1] Faculty of Computers and Information, Cairo University, Giza, 12613, Egypt

[2] Faculty of Science, Cairo University, Giza, 12613, Egypt

aboitcairo@gmail.com, SamehBasha@Sci.cu.edu.eg, areeg@sci.cu.edu.eg (**Corresponding authors*)

Abstract: This article presents a New Neutrosophic C-Means (NNCMs) method for clustering. It uses the neutrosophic logic (NL), to generalize the Fuzzy C-Means (FCM) clustering system. The NNCMs system assigns objects to clusters using three degrees of membership: a degree of truth, a degree of indeterminacy, and a degree of falsity, rather than only the truth degree used in the FCM. The indeterminacy degree, in the NL, helps in categorizing objects laying in the intersection and the boundary areas. Therefore, the NNCMs reaches more accurate results in clustering. These degrees are initialized randomly without any constraints. That is followed by calculating the clusters' centers. Then, iteratively, the NNCMs updates the membership values of every object, and the clusters' centers. Finally, it measures the accuracy and tests the objective function. The performance of the proposed system is tested on the six real-world databases: Iris, Wine, Wisconsin Diagnostic Breast Cancer, Seeds, Pima, and Statlog (Heart). The comparison between the two systems shows that the proposed NNCMs is more accurate.

Keywords: Neutrosophic C-Means, Neutrosophic set, Neutrosophic logic, Fuzzy C-Means, Neutrosophic clustering system.

1. Introduction

Clustering analysis is one of the important tasks in the data mining field [16][3], it is the process of categorizing objects into groups (clusters) based on their similarities [17][3]. There are two main clustering techniques, the supervised and the unsupervised. The supervised clustering techniques require human interaction, while the unsupervised clustering techniques do not. The latter ones are popular and are used in organizing unlabeled data objects into clusters so that the objects belonging to the same clusterhave more similarities [12]. Therefore, the term clustering techniques, mostly refers to the unsupervised ones, which can be classified as hard (crisp) clustering and soft (fuzzy) clusteringones. In hard clustering, the clusters have well defined and clear boundaries. An object belongs or does not belong to a cluster. On the other hand, in soft clustering techniques, the object can belong to more than one cluster with partial membership degrees [9][22]. The Fuzzy C-Means (FCM) is the most popular fuzzy unsupervised clustering algorithm. It was introduced by Dunn[11] and was modified by Bezdek [7].

Research on clustering algorithms focuses on improving systems based on the FCM such as Hongbin Dong et al. in [10]. They have proposed a fuzzy clustering method based on evolutionary programming (EPFCM) to improve the FCM algorithm. They encoded the cluster centers as a sequence of real numbers, and they defined the cluster validity indices as a function of the number of the cluster centers. Using the evolutionary algorithm, they searched for the optimum of the validity indices and the best result of clustering. S. Ganapathy et al. in [12] proposed a Novel Weighted FCM clustering method based on the Immune Genetic Algorithm (IGA-NWFCM) which solves the high dimensional multi-class problems and increases the chance of obtaining the optimal value. by applying the immune genetic algorithm. Their proposed system was tested with KDD99 cup data set and reached better classification accuracy. J. Yao et al. in [26] proposed an entropy-based fuzzy clustering method that automatically identifies the number and the initial locations of the cluster centers. It determined the first cluster center as the data point with minimum entropy after calculating the entropy at each data point. Then it removed all the data points having similarity larger than a threshold. To generalize the FCM, Y. Guo and A. Sengur in [13] introduced a neutrosophic clustering algorithm. They developed Neutrosophic C-Means (NCM) algorithm using neutrosophic logic. They used the three neutrosophic components T, I, and F for dealing with uncertain data. The three functions T, I, F are used to find the determinant, ambiguous, and outlier clusters, respectively. Then, they constructed a new objective function. However, their function is complex and time-consuming for more than three clustersand in order to simplify it, they considered only the two closest determinant clusters of each point. Moreover, their algorithm depends on parameters such as the number of classes C, the degree of fuzziness m, a

https://doi.org/10.24846/v27i1y201805

new parameter δ to control the number of objects considered as outliers, ε termination criterion between 0 and 1, and three weight factors. They also added an additional constraint: for each point

$$\sum_{j=1}^{c} Tj + I + F = 1$$

Then, Guo et al. in [1] extended his work in an attempt to generalize the kernel FCM (KFCM), by adding the kernel function to the algorithm.

This paper uses neutrosophic logic instead of the fuzzy logic as a generalization of the Fuzzy C-Means algorithm. In the proposed system, each object belongs to the clusters by the neutrosophic three membership values. These values represent the degree of truth, the degree of indeterminacy, and the degree of falsity. While in the FCM, only the degree of truth is used. Unlike the NCM in [13], the proposed system (NNCMs) does not change the FCM objective function. However, it makes good use of the indeterminacy term in the neutrosophic logic. The proposed system depends on two parameters only: the number of classes C, and the degree of fuzziness m, which is any real number greater than one. Moreover, the NNCMs does not add any restrictions or constraints on the membership functions.

The proposed system was applied to six real-world datasets Iris, Wine, Wisconsin Diagnostic Breast Cancer (WDBC), Seeds, Pima, and Statlog(Heart). The comparison between the FCM and the NNCMs shows that the later performs better for the six real-world datasets. Guo in [13] has applied his NCM on only the Iris dataset and incomparison between the NCM and the NNCMs, the proposed NNCMs shows better results.

The next two sections 2 and 3 present a brief introduction on theNeutrosophic Logic and the Fuzzy C-Means. In section 4, the proposed new Neutrosophic C-Means (NNCMs) is introduced and the different phases of the proposed NNCMs are presented. Then, section 5 presents a numerical example of the NNCMs system. Next section 6 shows theexperimental results obtained and comparisons with previous systems. Also a detailed discussion is introduced. At the end, section 7 concludes the results obtained and discusses some ideas for future work.

2. Neutrosophic Logic

Neutrosophy is considered a new philosophy, which treats the neutralities in data, as well as their interactions with various fields [25]. For every notion or idea<A>, the theory of neutralities considers its opposite or negation <Anti A> as well as a spectrum of neutralities <Neut A> which allows us to study the origin of neutralities for any phenomena or idea. Now, the terms <Neut–A> and <Anti–A>are used to form the term <Non–A>[25]. According to this theory and as a state of equilibrium every thought <A> tends to be neutralized and balanced by both < Anti–A> and <Non–A> ideas [25]. Neutrosophic logic (NL) was developed to represent mathematical models which can deal with uncertainty, vagueness, ambiguity, imprecision just like fuzzy logic. In addition to that, it can treat incompleteness, inconsistency, redundancy, and contradictions in data [21].

In all neutrosophic subjects, such asneutrosophy,neutrosophic logic, neutrosophic sets, neutrosophic probability, and neutrosophic statistics, there are three values T, I, and F to represent the truth, indeterminacy, and falsehood, respectively. These neutrosophic components T, I, and F, whether standard or non-standard are real subsets of $]^{-}0,1^{+}[$, where $]^{-}0,1^{+}[$ is the non-standard unit interval [4][18].

Theoretically using the non-standard unit interval $]^{-}0,1^{+}[$ for T, I and F is important. However, in real-world applications, it is difficult to use this non-standard unit interval. Therefore in applications, the non-standard unit interval is replaced by the standard real interval [0,1][4].

As in all soft computing subjects, the NL is very close to human thinking. That is the knowledge that comes from human observation is mostly characterized by imprecise data, as a result of the imprecision of humans [4]. Therefore, NL is perfect in treating problems that have imprecision, uncertainty, partial truth, incompleteness, or inconsistency in data.

The fundamental concepts of neutrosophic set, were introduced by Smarandache in [21][20]. Salama et al. in [2][14][15][19], provide a foundation for mathematically treating the neutrosophic phenomena which exist pervasively in our real world and for building new branches of neutrosophic mathematics.

To formally define the neutrosophic set [25], let X be a space of points (objects), with a generic element x in X. A neutrosophic set A in X is characterized by T_A, I_A and F_A truth, indeterminacy, falsity membership functions. These are real standard or non-standard subsets of $]^{-}0,1^{+}[$ with no restriction on their sum

$T_A, I_A, F_A : \mathrm{X} \to]^{-}0,1^{+}[$, and

$^{-}0 \le supT_A(x) + supI_A(x) + supF_A(x) \le 3^{+}$.

NL has many applications in different computer science areas. S. H. Basha et al. in [5] built a neutrosophic rule-based classification system which generalizes the fuzzy rule-based classification system. NL was used as a tool for representing different forms of knowledge. They extracted the three neutrosophic membership functions from the definition of the fuzzy trapezoidal membership function. The three neutrosophic membership functions were used to generate the “if-then” rules which, in turn, were used for classification. They applied their neutrosophic rule based system on three data sets; Iris, Wine, and Wdbc. Their system achieved more accurate classification rate reached 94.7% on average against 89.5% in the corresponding fuzzy one.

A hybrid classification system based on neutrosophic logic and genetic algorithm [6], extended their work. The genetic algorithm is used for refining the “if-then rules” by applying Michigan approach in order to get the optimal rules. On the same three data sets, the hybrid system applied and achieved a more accurate classification rate reached 98.39% in average against 94.78% in the neutrosophic rule-based classification system.

3. Fuzzy C-Means

When Zadeh introduced fuzzy set theory and logic to handle imprecise, fuzzy, and vague information [27], many applications in different areas such as control systems have been developed. Another important application was to use fuzzy logic in clustering problems to deal with these types of uncertainties [8]. Fuzzy clustering algorithms assign each object a degree of belongingness to clusters calculated by its degrees of membership [8][12].

Fuzzy C-Means (FCM) clustering algorithm extends the fuzzy clustering method. It is simple and the most used among the fuzzy clustering approaches [8].Given object vectors, $X = x_1,...,x_n$, number of clusters, c,where $2 \le c \le n$, the degree of fuzziness, m ≥1, the FCM algorithm determines the degree of the clusters` overlapping, and termination constant, e (maximum iteration number for example). The FCM algorithm consists of the following steps [8]:

Step 1: Randomly initialize the partition matrix U which is a matrix of degrees of membership for every object x_j, j=1,...,n in every cluster i, where i=1,...,c, μ_{ij} represents the value of the degree of membership of j^{th} vector in cluster i.

Initializing U with two constrains:

$$\sum_{i=1}^{c} \mu_{ij} = 1, \forall j > 0. \quad (1)$$

$$0 \le \sum_{j=1}^{n} \mu_{ij} \le n, \forall i > 0. \quad (2)$$

Step 2: Find the initial cluster centers using membership values of the initial partition matrix as inputs. The cluster center vector for cluster i obtained in $(t)^{th}$ iteration is

$$v_i^{(t)} = (\sum_{j=1}^{n} (\mu_{ij}^{(t)})) / \sum_{j=1}^{n} (\mu_{ij}^{(t)})^m; \forall i = 1,2,...,c \quad (3)$$

Step 3: Loop on t to minimize the objective function J, where:

$$J = \sum_{i=1}^{c} \sum_{j=1}^{n} (\mu_{ij})^m d^2(x_j, v_i). \quad (4)$$

where $d^2(x_j, v_i)$ is a measure of the distance between j^{th} object and i^{th} cluster center which may be Euclidean Distance,Maximum Distance, or Minkowski Distance.

Step 3.1. Calculate membership values of each input object j in cluster i, $\mu_{ij}^{(t)}$,

$$\mu_{ij}^{(t)} = \left[\sum_{k=1}^{c} \left(\frac{d(x_j, v_i^{(t-1)})}{d(x_j, v_k^{(t-1)})} \right)^{\frac{2}{m-1}} \right]^{-1} \quad (5)$$

where x_j is a vector of the input object and $v_i^{(t-1)}$ are cluster centers from $(t-1)^{th}$ iteration.

Step 3.2. Calculate the cluster center, v_i^t, of each cluster i at iteration t using the cluster center function in (3). The inputs are the input object matrix, x_j, and the membership values of iteration t are $\mu_{ij}^{(t)}$.

Step 3.3. Stop when the termination condition is satisfied, Otherwise go to Step 2.

4. Generalization of Fuzzy C-Means based on Neutrosophic Logic

The proposed new Neutrosophic C-Means (NNCMs) algorithm is shown in Figure 1. It generalizes the FCM using the NL. In which the membership values of the object are represented by three components the Truth, the Intermediancy, and the Falsity degrees T, I, and F. And they represent the degree of belongingness of an object to a cluster.

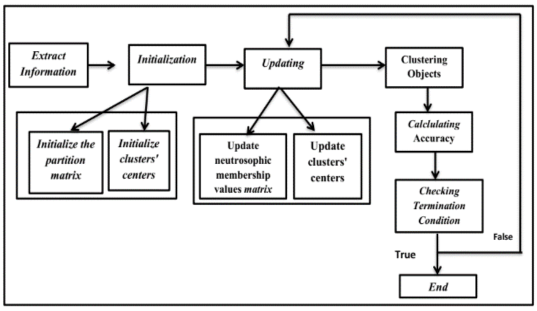
Extract Information
Initialization
Updating
Clustering Objects
Initialize the partition matrix
Initialize clusters' centers
Update neutrosophic membership values matrix
Update clusters' centers
Calculating Accuracy
Checking Termination Condition
False
True
End

Figure 1. New Neutrosophic C-Means Clustering System

In the FCM, the objects may belong to more than one cluster with different membership values that determine their degrees of belongingness. The NNCMs clustering system adds more information. It can determine if an object layes in the boundary area between two clusters. Using the NL, the NNCMs system can determine if the clusters are intersected or not. The proposed system was able to find out that, the IRIS data set, can be divided in only two clusters, instead on three, as one cluster falls in another cluster which was also noticed in [26]. The system was also able to find out the intersected clusters and the overlapped centers. The NNCMs does that without knowing, previously, any information about the data sets. In addition, the FCM randomly initializes the partition matrix with two constraints as in equations (1) and (2). However, in the NNCMs system, there is no need to this constrains.

The NNCMs seven phases, Figure 1: Extracting Information phase, Initialization phase to initialize the partition matrix and the clusters' centers, Updating phase to update the neutrosophic membership values and clusters' centers, Clustering objects Phase, and Accuracy phase. These seven phases are described in detail in the rest of this section.

4.1 Extract Information phase

In this phase, we extract some important features from the data set, such as number of attributes, number of objects, the minimum and the maximum value for each attribute, and we extract the exact clusters as well.

4.2 Initialize the partition matrix phase

Randomly we initialize the neutrosophic partition matrix U; i.e the vectors μ_{ij} ,the degrees of membership of object j to cluster i.

$$U = \begin{bmatrix} \mu_{1,1} & \mu_{2,1} & \cdots\cdots & \mu_{c,1} \\ \mu_{1,2} & \mu_{2,2} & \cdots\cdots & \mu_{c,2} \\ \cdots\cdots & \cdots\cdots & \cdots\cdots & \cdots\cdots \\ \mu_{1,n} & \mu_{2,n} & \cdots\cdots & \mu_{c,n} \\ & & & \end{bmatrix}$$

where $\mu_{i,j} = \left[T_{i,j}, I_{i,j}, F_{i,j} \right]$ represent the degrees of belongingness, indeterminacy, and not-belongingness respectively. They are real subsets of $]^{-}0,1^{+}[$ with no restriction on their sum. i.e. unlike the FCM there is no need to add any constraints.

4.3 Initialize cluster centers phase

In this phase, we use Initialized neutrosophic partition matrix U and equation (3) to initialize the center value of each cluster.

4.4 Update the neutrosophic membership values phase

Using the initialized neutrosophic partition matrix U, in the $t^{(th)}$ iteration we calculate the new neutrosophic membership values using the cluster centers calculated at iteration $(t-1)^{th}$. In the NNCMs at iteration t^{th}, the neutrosophic membership values have three components and are calculated by:

$$T_{i,j}^{(t)} = \left[\sum_{k=1}^{c} \left(\frac{d(x_j, v_i^{(t-1)})}{d(x_j, v_k^{(t-1)})} \right)^{\frac{2}{m-1}} \right]^{-1} \tag{6}$$

$$, i = 1, \ldots, c, j = 1, \ldots, n.$$

Table 1. Details of each iteration

T	U	V	Cluster	Accuracy
t=1	$\begin{bmatrix} [0.516, 0.130, 0.934] & [0.483, 0.121, 1.0] \\ [0.386, 0.074, 1.0] & [0.613, 0.118, 0.630] \\ [0.368, 0.140, 1.0] & [0.631, 0.240, 0.582] \\ [0.747, 0.667, 0.337] & [0.252, 0.225, 1.0] \\ [0.163, 0.327, 1.0] & [0.836, 0.594, 0.194] \\ [0.124, 0.676, 1.0] & [0.875, 0.096, 0.142] \\ [0.927, 0.112, 0.078] & [0.072, 0.697, 1.0] \\ [0.665, 0.637, 0.501] & [0.334, 0.319, 1.0] \\ [0.451, 0.099, 1.0] & [0.548, 0.120, 0.821] \\ [0.586, 0.109, 0.705] & [0.413, 0.077, 1.0] \end{bmatrix}$	$\begin{bmatrix} 2.861 & 3.573 \\ 3.380 & 4.902 \end{bmatrix}$	A: [9, 6, 5, 3, 2] B: [10, 8, 7, 4, 1]	Number of correct clustered objects is: 6 **Accuracy** is: 60.0%
t=2	$\begin{bmatrix} [0.473, 0.200, 1.0] & [0.526, 0.223, 0.900] \\ [0.325, 0.121, 1.0] & [0.674, 0.252, 0.481] \\ [0.290, 0.219, 1.0] & [0.709, 0.536, 0.408] \\ [0.752, 0.973, 0.329] & [0.247, 0.338, 1.0] \\ [0.108, 0.557, 1.0] & [0.891, 0.218, 0.121] \\ [0.312, 0.403, 1.0] & [0.687, 0.183, 0.455] \\ [0.990, 0.011, 0.009] & [0.009, 0.833, 1.0] \\ [0.788, 0.620, 0.267] & [0.211, 0.431, 1.0] \\ [0.492, 0.194, 1.0] & [0.507, 0.199, 0.971] \\ [0.650, 0.222, 0.537] & [0.349, 0.119, 1.0] \end{bmatrix}$	$\begin{bmatrix} 2.524 & 3.404 \\ 3.652 & 4.969 \end{bmatrix}$	A: [9, 6, 5, 3, 2, 1] B: [10, 8, 7, 4]	Number of correct clustered objects is: 7 **Accuracy** is: 70.0%
t=3	$\begin{bmatrix} [0.399, 0.312, 1.0] & [0.600, 0.468, 0.666] \\ [0.244, 0.198, 1.0] & [0.755, 0.612, 0.323] \\ [0.185, 0.340, 1.0] & [0.814, 0.668, 0.227] \\ [0.737, 0.642, 0.355] & [0.262, 0.552, 1.0] \\ [0.131, 0.911, 1.0] & [0.868, 0.166, 0.151] \\ [0.473, 0.285, 1.0] & [0.526, 0.256, 0.898] \\ [0.993, 0.006, 0.006] & [0.006, 0.939, 1.0] \\ [0.899, 0.193, 0.111] & [0.100, 0.577, 1.0] \\ [0.566, 0.396, 0.766] & [0.433, 0.303, 1.0] \\ [0.728, 0.492, 0.372] & [0.271, 0.183, 1.0] \end{bmatrix}$	$\begin{bmatrix} 2.074 & 3.294 \\ 3.948 & 4.980 \end{bmatrix}$	A: [6, 5, 3, 2, 1] B: [10, 9, 8, 7, 4]	Number of correct clustered objects is: 8 **Accuracy** is: 80.0%
t=4	$\begin{bmatrix} [0.313, 0.452, 1.0] & [0.686, 0.989, 0.456] \\ [0.166, 0.305, 1.0] & [0.833, 0.653, 0.199] \\ [0.089, 0.503, 1.0] & [0.910, 0.195, 0.098] \\ [0.704, 0.493, 0.418] & [0.295, 0.847, 1.0] \\ [0.232, 0.705, 1.0] & [0.767, 0.213, 0.302] \\ [0.591, 0.217, 0.691] & [0.408, 0.314, 1.0] \\ [0.960, 0.0317, 0.040] & [0.039, 0.777, 1.0] \\ [0.966, 0.0479, 0.034] & [0.033, 0.722, 1.0] \\ [0.655, 0.781, 0.524] & [0.344, 0.410, 1.0] \\ [0.799, 0.960, 0.250] & [0.200, 0.261, 1.0] \end{bmatrix}$	$\begin{bmatrix} 1.632 & 3.267 \\ 4.187 & 4.928 \end{bmatrix}$	A: [5, 3, 2, 1] B: [10, 9, 8, 7, 6, 4]	Number of correct clustered objects is: 9 **Accuracy** is: 90.0%
t=5	$\begin{bmatrix} [0.241, 0.584, 1.0] & [0.758, 0.543, 0.317] \\ [0.115, 0.418, 1.0] & [0.884, 0.310, 0.130] \\ [0.032, 0.669, 1.0] & [0.967, 0.050, 0.033] \\ [0.672, 0.430, 0.486] & [0.327, 0.884, 1.0] \\ [0.349, 0.505, 1.0] & [0.650, 0.272, 0.538] \\ [0.667, 0.177, 0.497] & [0.332, 0.357, 1.0] \\ [0.931, 0.051, 0.073] & [0.068, 0.698, 1.0] \\ [0.990, 0.011, 0.009] & [0.009, 0.827, 1.0] \\ [0.732, 0.753, 0.365] & [0.267, 0.485, 1.0] \\ [0.845, 0.559, 0.183] & [0.154, 0.327, 1.0] \end{bmatrix}$	$\begin{bmatrix} 1.325 & 3.290 \\ 4.319 & 4.847 \end{bmatrix}$	A: [5, 3, 2, 1] B: [10, 9, 8, 7, 6, 4]	Number of correct clustered objects is: 9 **Accuracy** is: 90.0%

$$I_{i,j}^{(t)}=\left[\sum_{k=1}^{c}\left(\frac{d(x_j,v_i^{(t-1)})}{d(v_k^{(t-1)},v_l^{(t-1)})}\right)^{\frac{2}{m-1}}\right]^{-1} \quad (7)$$

$$,i=1,\ldots,c,j=1,\ldots,n,l=1,\ldots,c.$$

$$F_{i,j}^{(t)}=\left[\sum_{k=1}^{c}\left(\frac{d(x_j,v_i^{(t-1)})}{\max d(x_j,v_k^{(t-1)})}\right)^{\frac{2}{m-1}}\right]^{-1} \quad (8)$$

$$,i=1,\ldots,c,j=1,\ldots,n.$$

In order to calculate the distance between any object and the center of a cluster, we may use any distance metric or similarity measure. Here, we have used the Euclidean metric.

4.5 Update cluster centers phase

In this phase we calculate the cluster centers at iteration t using the updated neutrosophic membership values, $\mu_{ij}^{(t)}$, and equation (3).

4.6 Clustering objects phase

At the end of each iteration, here, we determine the cluster to which the object belongs by using the neutrosophic membership values.

4.7 Accuracy phase

In this phase, we compare the suggested clusters' objects, obtained from "Clustering objects" phase, with the exact clusters' objects using a set of the testing data. Then, we compute the confusion matrix; a matrix that is used to describe the performance of a clustering algorithm. Then,we calculate the total accuracy, precision, recall, and specificity for each class.

5. Numerical Example

Here, we explain a simple numeric example to illustrate the steps of the NNCMs system. If we have ten objects with two attributes categorized in two clusters A and B as following:

$$\{<0.2,5,A>,<0.5,2,A>,<1,3,A>,<2.5,6,A>,$$
$$<3.2,3,A>,<3.3,4,B>,<3.5,5,B>,<4.5,5,B>,$$
$$<6,3,B>,<6.2,6,B>\}$$

Step 1: Randomly the neutrosophic partition matrix U is initialized:

$$U=\begin{bmatrix} [0.733,0.884,0.536] & [0.267,0.116,0.711] \\ [0.244,0.564,0.597] & [0.864,0.436,0.403] \\ [0.246,0.894,0.805] & [0.838,0.172,0.257] \\ [0.527,0.709,0.24] & [0.473,0.435,0.76] \\ [0.55,0.162,0.883] & [0.517,0.838,0.117] \\ [0.212,0.412,0.256] & [0.788,0.618,0.828] \\ [0.384,0.596,0.658] & [0.616,0.5,0.424] \\ [0.041,0.468,0.405] & [0.959,0.532,0.688] \\ [0.297,0.52,0.094] & [0.703,0.48,0.924] \\ [0.759,0.075,0.953] & [0.363,0.924,0.047] \\ & \end{bmatrix}$$

Step 2: Initializing the cluster centers using the partition matrix U and equation (3):

$$V=\begin{pmatrix} 3.164 & 4.870 \\ 3.028 & 3.807 \end{pmatrix}$$

Step 3: Looping until the stability of the clusters. Table 1 shows the iterations.

6. Experimental results

All the experiments are performed using Intel(R)_Core(TM)2_Duo_CPU_T6400_@_200GHz, 2.00GHz Frequency, 300GBRam, 250GB Hard Drive, and Windows 8. And all the algorithms are self-coded using java.

6.1 Datasets

Since we have limited accessibility to private datasets, all tests are done on public datasets.The performance of the proposed new Neutrosophic C-Means (NNCMs) clustering system is studied using the six famous real-world databases Iris, Wine, Wdbc, Seeds, Pima, and Statlog(Heart),from the UCI Machine Learning Repository website. We have chosen these data sets because of their different characteristics. Table 2 presents some details about these databases.

Table 2. Details Of Six UCI Datasets

DataSet Name	Number Of Sampling	Number Of Features	Number Of Clusters
Iris	150	4	3
Wine	178	13	3
Wdbc	569	32	2
Seeds	210	7	3
Pima	768	8	2
Statlog(Heart)	270	13	2

6.2 Assessment methods

Accuracy is famous and commonly used measure in classification and clustering where,

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}, \text{ where:}$$

		Actual Result	
		In Cluster	Not In Cluster
Predicted Result	In Cluster	TP (true positive)	FP (false positive)
	Not In Cluster	FN (false negative)	TN (true negative)

With the imbalanced data, the accuracy measure is not enough. The Precision, Sensitivity, and Specificity are the most used where, the Precision= $\frac{TP}{TP+FP}$, the Sensitivity(or recall)= $\frac{TP}{TP+FN}$, and the Specifcity= $\frac{TN}{TN+FP}$ [24].

6.3 Evaluation results and discussion

The proposed NNCMs clustering system generalizes the FCM clustering system and gives better and more accurate results. Figure 2 shows the comparison of the total accuracy in both NNCMs and FCM for the datasets. As shown in this comparison, the proposed system gives more detailed information about the data and the clusters. This is a result of using the indeterminacy. It helps in getting more accurate results and gives more details about the clusters and the data. For example, the object $< 7.0, 3.2, 4.7, 1.4, Iris - versicolorin >$ in the Iris data set is wrongley categorized in both FCM and NNCMs. However the NNCMs explains the reason behind that. The NNCMs shows that the two clusters are interleaved. The FCM categorizes this object with degrees 0.1776, 0.3992, and 0.4231 to clusters 1, 2, and 3, respectively. i.e. it belongs to cluster 3, which is incorrect and the FCM was not able to explain why. However, the NNCMs calculates the [T, I, F] degrees as $[0.0446, 0.1345, 1.0]$, $[0.4544, 0.7296, 0.0981]$, $[0.5009, 0.6618, 0.0890]$ in clusters 1, 2, and 3 respectively. Since the F degree of belongingness to cluster 1 is equal to 1.0, we conclude that it does not belong to cluster one. And the truth and indeterminacy degrees of this object are close. Looking at other objects with the same case, we found out that clusters 2 and 3 should be unified in one cluster, a fact that was found before in [26].

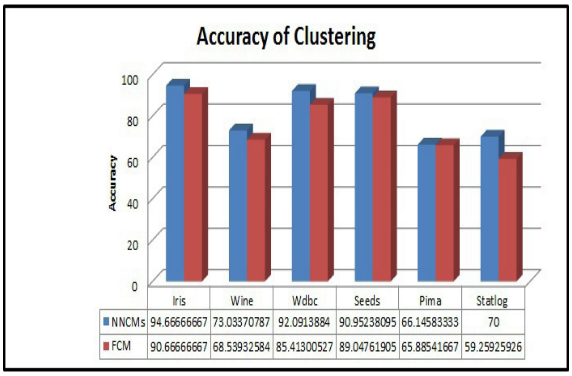

Figure 2. Accuracy of Clustering for the six real-world databases in NNCMs and FCM

Tables 3, 4, 5, 6, 7, and 8 show the actual cluster labels and the achieved clusters for thedatasets in both NNCMs and the corresponding FCM.

A comparison of the Precision, Sensitivity, and Specificity for each class of the datasets in both the NNCMs and the FCM clustering systems is shown in Figures 4, 6, 8, 10, 12, and 14. Since the NNCMs uses the truth, indeterminacy, and falsity degrees, the time rate of clustering in the NNCMs is generally higher compared to the FCM. In worest cases, the NNCMs uses 7 seconds while the FCM uses 5 seconds. Which is a reasonable price for obtaining more accurate and better details of the data. In order to evaluate our results we use the Principal Component Analysis (PCA) technique [23] as a visualization of high-dimensional datasets to draw the six datasets in two dimension-spaces.

Table 3. Comparison between results of NNCMs, NCM, and FCM for Iris Dataset

		Clusters in NNCMs			Clusters in NCM			Clusters in FCM		
		------	---------	---------	-------	---------	---------	--------	---------	---------
		setosa	versicolor	virginica	setosa	versicolor	virginica	setosa	versicolor	virginica
Actual cluster label	Setose	50	0	0	50	0	0	50	0	0
	Versicolor	0	44	6	0	47	8	0	47	3
	Virginica	0	2	48	0	2	37	0	11	39

Table 4. Comparison on Wine DataSet

		Clusters in NNCMs			Clusters in FCM		
		C 1	C 2	C 3	C 1	C 2	C 3
Actual cluster label	C 1	50	0	9	45	0	14
	C 2	4	49	18	1	50	20
	C 3	1	16	31	0	21	27

Table 5. Comparison on Wdbc DataSet

		Clusters in NNCMs		Clusters in FCM	
		- - - - - - ---	- - - - - - ---	---------	---------
		Class M	Class B	Class M	Class B
Actual cluster	Class M	175	37	130	82
	Class B	8	349	1	356

Table 6. Comparison on Seeds DataSet

		Clusters in NNCMs			Clusters in FCM		
		-----	-----	------	------	----	------
		C 1	C 2	C 3	C 1	C 2	C 3
Actual cluster label	C 1	95	3	8	59	2	9
	C 2	6	64	0	10	60	0
	C 3	2	0	68	2	0	68

Table 7. Comparison on Pima DataSet

		Clusters in NNCMs		Clusters in FCM	
		------	------	-------	-------
		C 0	C 1	C 0	C 1
Actual cluster label	C 0	405	95	404	96
	C 1	165	103	166	102

Table 8. Comparison on Statlog (Heart) DataSet

		Clusters in NNCMs		Clusters in FCM	
		-------	- - - - - - ---	-------	-------
		C 1	C 2	C 1	C 2
Actual cluster label	C 1	105	45	98	52
	C 2	36	84	58	62

6.3.1 Iris Dataset

As shown in Figure 3, the Iris dataset contains three classes Setosa, Versicolour, and Virginica. The Setosa is completely separated from the other two. Therefore, the Precision, Sensitivity, and Specificity measures of the Setosa class are 100 %, Figure 4. The second and third classes are so close. Yet, the data set describes them as twoclasses. As a result, the NNCMs gets precision 95% for the Versicolour and 88% for the Virginica. However, if they were combined the NNCMs system would have reached 99%. The FCM, for the second and third classes, reaches a precision of only 81% for the Versicolourand 92% for the Virginica, which are less than the NNCMs result.

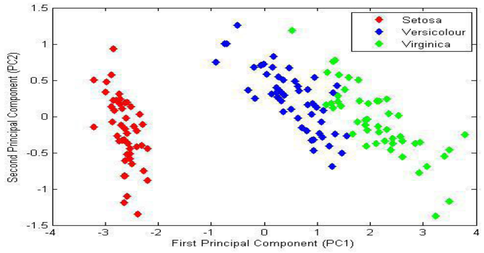

Figure 3. Iris Dataset

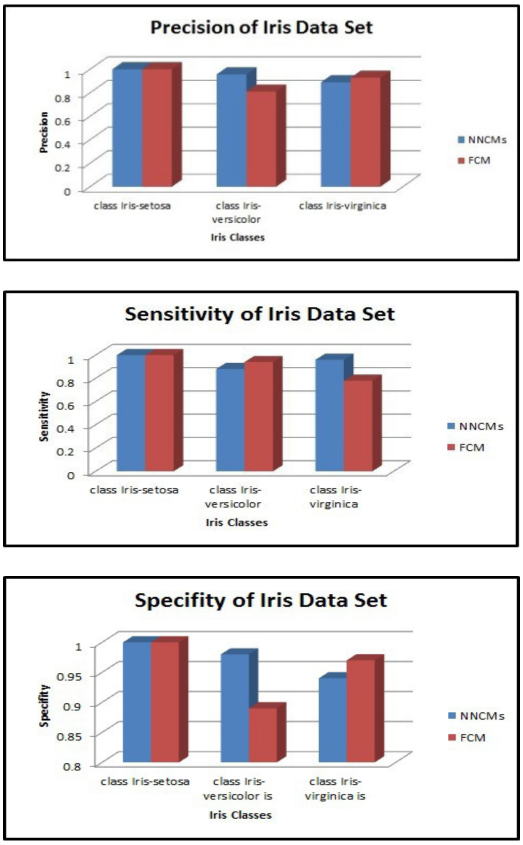

Figure 4. Precision, Sensitivity, and Specificity for the Iris Dataset

As shown in Table 3, all the clustering algorithms achieve 100% for the Setosa class, which isnatural as explained eailer. The NNCMs system proved to be more accurate, it misclassified only 8 object while Guo-NCM [13] misclassified 10 objects.

6.3.2 Wine Dataset

As shown in Figure 5, the Wine dataset contains three classes C1, C2, and C3, which are not completely separated. Most objects of C1 are separated from the other two classes. But some objects in C2 and C3 lie near the centers of C2 and C3, as these two classes are overlapped and their centers are very close. Therefore, using the indeterminacy term, here, gives better results. And since the data set treats C2 and C3 as two different classes, the NNCMs reaches precision of 75.3% for C2 and 53.4% for C3, compared to 70.4% for C2 and 44.2% for C3 in the FCM. That is overall the NNCMs is more accurate than the FCM, Figure 6.

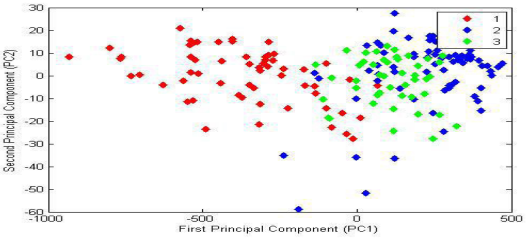

Figure 5. Wine Dataset

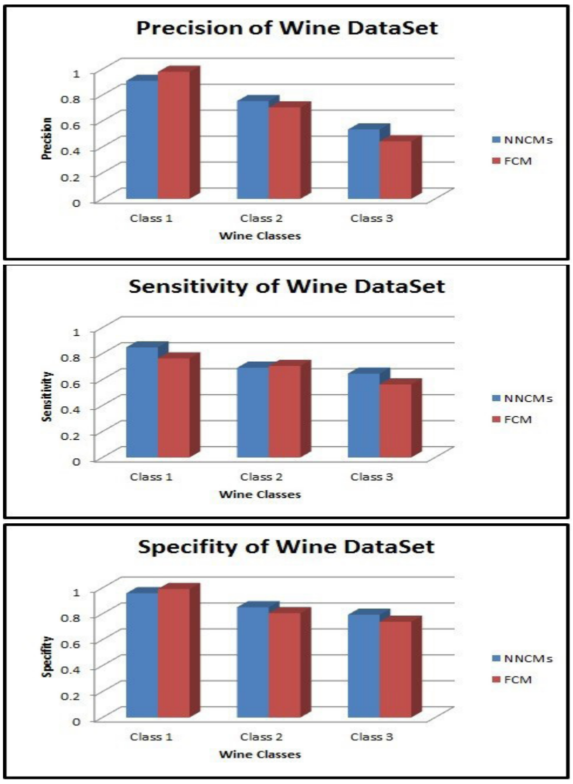

Figure 6. Precision, Sensitivity, and Specificity for Wine Dataset

6.3.3 Wdbc Dataset

As shown in Figure 7, Wdbc dataset contains two classes class M, and class B, where class B is a subset of class M. Also, using the intermediary term, here, gives better results. Since the data set treats them as two different ones, actually there are objects that belong to both classes and also nearby the center of each class, It is difficult to determine the class of these objects correctly. The NNCMs determined the class successfully for objects in the class B that are far from the center. As shown in Figure 8, the NNCMs reachesa precision 95.6% for class M and 90.4% for class B. However, the FCM reaches a precision of 99.2% for class M and 81.2% for class B. The FCM considers most of the data belongs to class M, therefore, it gets a higher precision for M but a much worse precision for B. The NNCMs is more robust, it gives a better accuracy, reaching 92.09% against 85.41% in FCM.

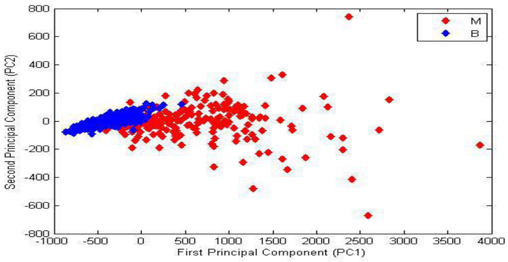

Figure 7. Wdbc Dataset

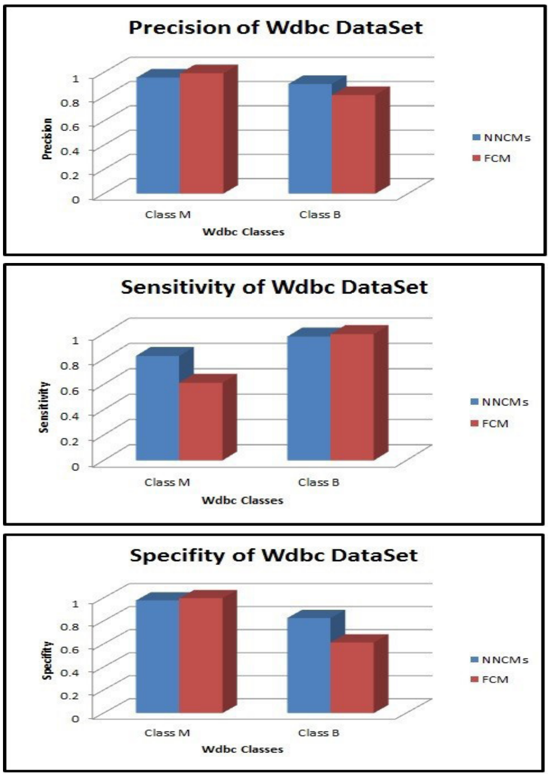

Figure 8. Precision, Sensitivity, and Specificity for Wdbc Dataset

6.3.4 Seeds Dataset

As shown in Figure 9, the Seeds dataset contains three classes C1, C2, and C3, where C1 and C2 are intersected, just like C1 and C3. As shown in Figure 10, that results in getting higher values for the Precision, Sensitivity, and Specifity for C2 and C3 in both NNCMs and FCM. However, NNCMs is better as its accuracy reaches 90.95% against 89.04% in FCM.

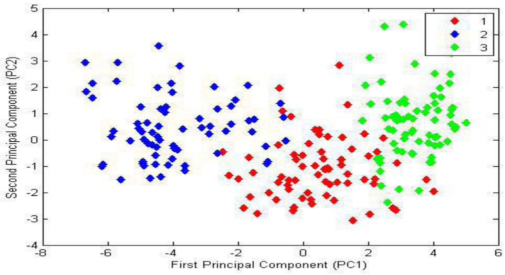

Figure 9. Seeds Dataset

6.3.5 Pima Dataset

As shown in Figure 11. Pima dataset contains two classes C0, and C1, where the two classes are interleaved and may have overlapped centers. It is difficult to identify them separately. Therefore, using the indeterminacy term, here, gives better results. Since the data set treats them as two different classes. As shown in Figure 12, the NNCMs reaches a precision of 71.0% for C0 and 52.0% for C1 compared to 70.8% for C0 and 51.5% C1 in FCM.

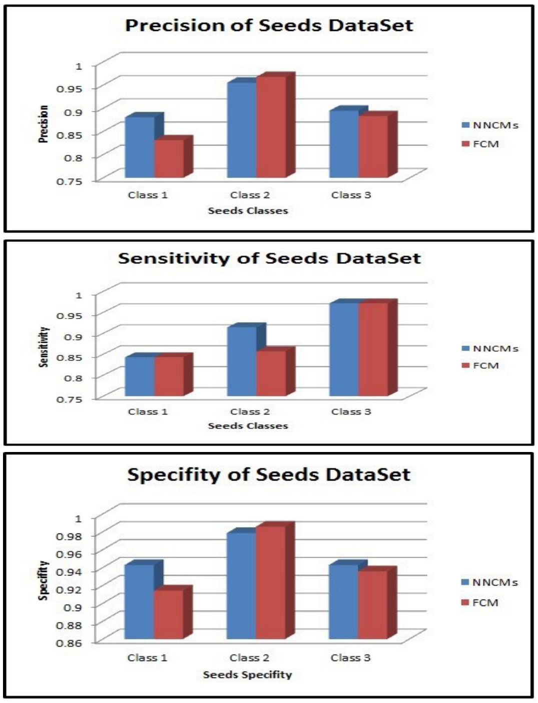

Figure 10. Precision, Sensitivity, and Specifity for Seeds Datasets

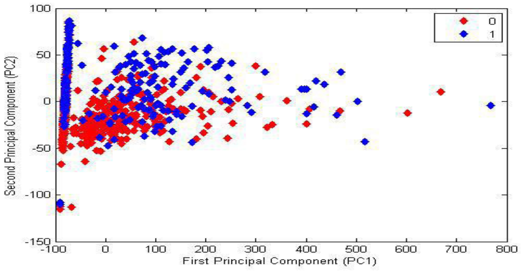

Figure 11. Pima Dataset

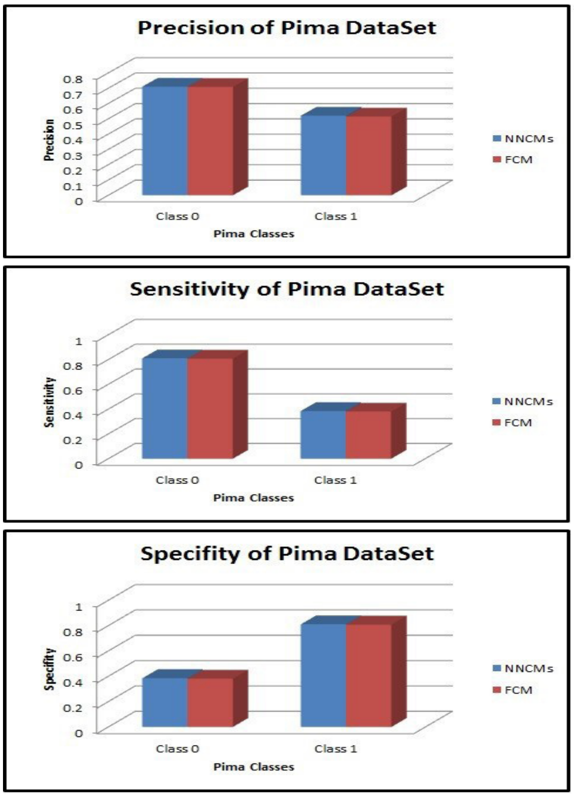

Figure 12. Precision, Sensitivity, and Specifity for Pima Dataset

6.3.6 Statlog(Heart) Dataset

As shown in Figure 13, Statlog(Heart) dataset contains two classes C1, and C2, where these classes are overlapped. They are very difficult to identify from each other. The data set treats them as two different classes. As shown in Figure 14, the NNCMs reaches a precision of 74.4% for C1 and 65.1% for C2, compared to 62.8% for C1 and 54.3% C2 in FCM.

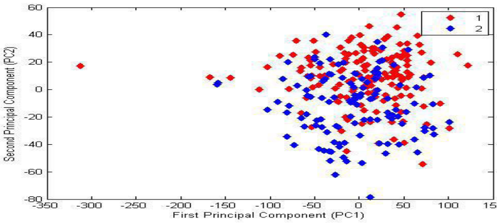

Figure 13. Statlog(Heart) Dataset

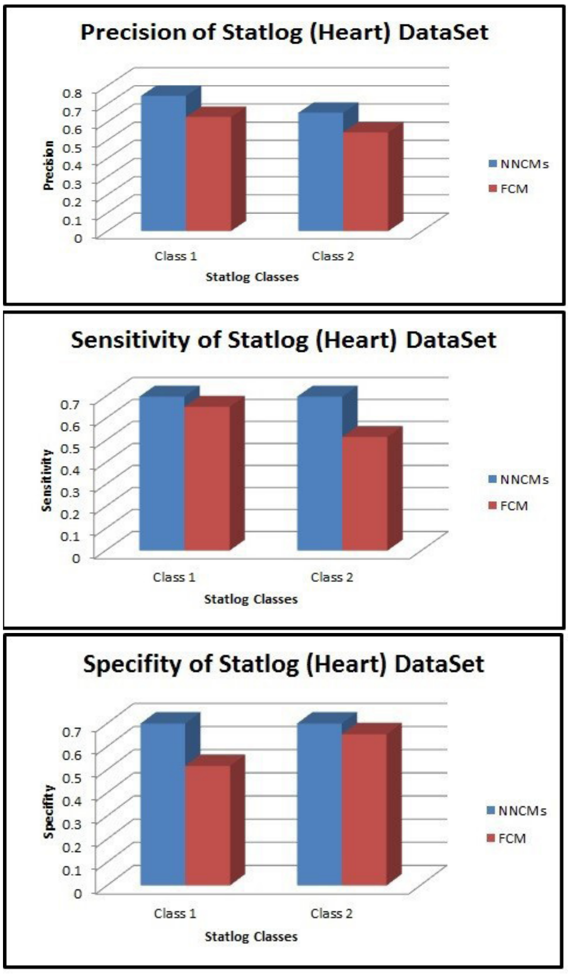

Figure 14. Precision, Sensitivity, and Specificity for Statlog(Heart)

7. Conclusions and Future Work

The proposed Neutrosophic C-Means (NNCMs) Clustering system generalizes the Fuzzy C-Means (FCM) Clustering system. It introduces a clustering system based on the neutrosophic logic (NL), inwhich, three degrees of membership are used to describe the belongingness of an object to a class. The NNCMs performs better clustering than both the FCM and the neutrosophic C-means (NCM). Even in hard cases, in which the data sets are overlapped and should be unified, as suggested by former researcher, and the FCM gives higher values according to the measures used. Yet, the NNCMs gives more details about the clusters and its overall performance is better. This is a natural result of using the intermediancy term used in NL.

In the future work, to increase the convergence speed, we aim to build a hybrid system between the NNCMs and one of the evolutionary techniques.

REFERENCES

1. Akbulut, Y., Sengur, A., Guo, Y. & Polat, K. (2017). KNCM: Kernel Neutrosophic c-Means Clustering, *Applied Soft Computing*, *52*, 714-724.

2. Alblowi, S., Salama, A. & Eisa, M. (2014). New concepts of neutrosophicsets, *International Journal of Mathematics and Computer Applications Research* (*IJMCAR*), *4*(1), 59-66.

3. Amaricai, A. (2017). Design Trade-offs in Configurable FPGA Architectures for K-Means Clustering, *Studies in Informatics and Control*, *26*(1), 43-48, ISSN 1220-1766.

4. Ansari, Q., Biswas, R. & Aggarwal, S. (2013). Neutrosophic classifier: An extension of fuzzy classifer, *Applied Soft Computing*, *13*(1), 563-573.

5. Basha, S., Abdalla, A. & Hassenian, A. (2016). GNRCS: Hybrid Classification System based on Neutrosophic Logic and Genetic Algorithm. In *12th International Computer Engineering Conference* (*ICENCO*), Egypt (pp. 53-58).

6. Basha, S., Abdalla, A. & Hassenian, A. (2016). NRCS: Neutrophic Rule-based Classification System. In *Proceedings of SAI Intelligent Systems Conference* (*IntelliSys*) *2016*, *Lecture Notes in Networks and Systems*, *15* (pp. 627–639).

7. Bezdek, J. C., Ehrlich, R. & Full, W. (1984). FCM: The fuzzy c-means clustering algorithm, *Computers and Geosciences*, *10*(2-3), 191-203.

8. Celikyilmaz, A. & Türksen, I. B. (2009). *Modeling Uncertainty with Fuzzy Logic: With Recent Theory and Applications*. Springer Publishing Company.

9. Chattopadhyay, S., Pratihar, D. & Sarkar, S. (2011). A comparative study of fuzzy c-means algorithm and entropy-based fuzzy clustering algorithms, *Computing and Informatics*, *30*(4), 701-720.

10. Dong, H., Dong, Y., Zhou, C., Yin, G. & Hou, W. (2009). A fuzzy clustering algorithm based on evolutionary programming, *Expert Systems with Applications*, *36*(9), 11792-11800.

11. Dunn, J. C. (1973). A Fuzzy Relative of the ISODATA Processand Its Use in Detecting Compact Well-Separated Clusters, *Cybernetics and Systems*, *3*, 32-57.

12. Ganapathy, S., Kulothungan, K., Yogesh, P. & Kannan, A. (2012). A Novel Weighted Fuzzy C –Means Clustering Based on Immune Genetic Algorithm for Intrusion Detection. In *Proceeding Engineering, Elsevier*, *38* (pp.1750-1757).

13. Guo, Y. & Sengur, A. (2015). NCM: Neutrosophic c-means clustering algorithm, *Pattern Recognition*, *48*, 2710-2724.

14. Hanafy, I., Salama, A. & Mahfouz, K. (2012). Correlation of neutrosophicdata, *International Refereed Journal of Engineering and Science* (*IRJES*), *1*(2), 33-39.

15. Hanafy, I., Salama, A. & Mahfouz, K. (2013). Neutrosophic classical events and its probability, *International Journal of Mathematics and Computer Applications Research* (*IJMCAR*), *3*(1), 171-178.

16. Lu, Y., Ma, T., Yin, C., Xie, X., Tian, W. & Zhong, S. (2013). Implementation of the Fuzzy C-Means Clustering Algorithm in Meteorological Data, *International Journal of Database Theory and Application*, *6*(6), 1-18.

17. Nazari, M., Shanbehzadeh, J. & Sarrafzadeh, A. (2013). Fuzzy C-means based on Automated Variable Feature Weighting. In *Proceedings of the International MultiConference of Engineers and Computer Scientists 2013*, *1* (pp. 25-29), Hong Kong.

18. Robinson, A. (1996). *Non-Standard Analysis, Princeton Landmarks in Mathematics and Physics series*.

19. Salama, A. & Alblowi, S. (2012). Generalized neutrosophic set and generalized neutrosophic spaces, *Journal Computer Sci. Engineering*, *2*(7), 129-132.

20. Smarandache, F. (2005). Neutrosophic set, a generialization of the intuituionistics fuzzy sets, *International Journal of Pure and Applied Mathematics*, *24*, 287-297.

21. Smarandache, F. (2003). *A Unifying Field in Logics: Neutrosophic Logic. Neutrosophy, Neutrosophic Set, Neutrosophic Probability*, 3rd edition. American Research Press.

22. Suganya, R. & Shanthi, R. (2012). Fuzzy C-Means Algorithm – A Review, *International Journal of Scientific and Research Publications*, *2*(11), 440-442.

23. Tharwat, A. (2016). Principal component analysis – a tutorial, *International Journal of Applied Pattern Recognition*, *3*(3), 197-240.

24. Tharwat, A., Moemen, Y. & Hassenian, A. (2017). Classification of toxicity effects ofb*IoT*ransformed hepatic drugs using whale optimized support vector machines, *Journal of Biomedical Informatics*, *68*, 132-149.

25. Wang, H., Smarandache, F., Sunderraman, R. & Zhang, Y. (2005). *Interval Neutrosophic Sets and Logic: Theory and Applications in Computing*, Hexis, Neutrosophic Book Series.

26. Yao, J., Dash, M., Tan S. & Liu, H.(2000) Entropybased fuzzy clustering and fuzzy modeling, *Fuzzy Sets and Systems*, *113*(3), 381-388.

27. Zadeh, L. A. (1965). Fuzzy sets, *Information and Control*, *8*, 338-353.