Evaluation Algorithms Based on Fuzzy C-means for the Data Clustering of Cancer Gene Expression

Omar Al-Janabee^{1,*}, Basad Al-Sarray¹

¹University of Baghdad, College of Science, Computer Science Department, Iraq

*Corresponding Author: Omar Al-Janabee

DOI: <https://doi.org/10.52866/ijcsm.2022.02.01.004>

Received October 2021; Accepted January 2022; Available online February 2022

ABSTRACT: The influx of data in bioinformatics is primarily in the form of DNA, RNA, and protein sequences. This condition places a significant burden on scientists and computers. Some genomics studies depend on clustering techniques to group similarly expressed genes into one cluster. Clustering is a type of unsupervised learning that can be used to divide unknown cluster data into clusters. The k-means and fuzzy c-means (FCM) algorithms are examples of algorithms that can be used for clustering. Consequently, clustering is a common approach that divides an input space into several homogeneous zones; it can be achieved using a variety of algorithms. This study used three models to cluster a brain tumor dataset. The first model uses FCM, which is used to cluster genes. FCM allows an object to belong to two or more clusters with a membership grade between zero and one and the sum of belonging to all clusters of each gene is equal to one. This paradigm is useful when dealing with microarray data. The total time required to implement the first model is 22.2589 s. The second model combines FCM and particle swarm optimization (PSO) to obtain better results. The hybrid algorithm, i.e., FCM– PSO, uses the DB index as objective function. The experimental results show that the proposed hybrid FCM–PSO method is effective. The total time of implementation of this model is 89.6087 s. The third model combines FCM with a genetic algorithm (GA) to obtain better results. This hybrid algorithm also uses the DB index as objective function. The experimental results show that the proposed hybrid FCM–GA method is effective. Its total time of implementation is 50.8021 s. In addition, this study uses cluster validity indexes to determine the best partitioning for the underlying data. Internal validity indexes include the Jaccard, Davies Bouldin, Dunn, Xie–Beni, and silhouette. Meanwhile, external validity indexes include Minkowski, adjusted Rand, and percentage of correctly categorized pairings. Experiments conducted on brain tumor gene expression data demonstrate that the techniques used in this study outperform traditional models in terms of stability and biological significance.

Keywords: Clustering; Fuzzy c-means clustering; Genetic algorithm; Particle swarm optimization; Gene expression

1. INTRODUCTION

Recent technological developments have generated massive amounts of data in various fields, including biological science fields, to the extent that analyzing data and extracting results from them have become impossible. This situation is complemented by considerable developments in the field of informatics, which have contributed to the merging of most technologies to analyze big data and provide many systematic solutions, leading to a scientific revolution in many scientific and industrial sectors. Bioinformatics is the extension of information technology to molecular biology. It is useful in comparing genes and sequences in different proteins within an organism or between organisms to identify evolutionary relationships between organisms. In 1978, Paulien Hogeweg coined the term “bioinformatics” to refer to the study of computer processes in biotic systems. To analyze and interpret biological data, scientists use a combination of biology, computer science, information technology, mathematics, and statistics [1], as shown in Figure 1.

FIGURE 1. Bioinformatics

Clustering is an interesting approach for finding similarities in data and grouping similar data. Clustering has been used for decades in many fields, including image processing, data mining, artificial intelligence [2], and genomics research [3]. Clustering is used to divide objects into groups called clusters. Clustering algorithms are generally divided into two classes. The first class is hard clustering, and the second class is soft clustering. In hard clustering, data are divided into groups, such that each data element belongs to only one group. The sums in this type of clustering are as similar as possible to each other and different from those in other groups. In soft clustering, data elements belong to more than one group, and each element is associated with a group of membership levels [4].

Clustering can be done in various ways.

- **Hierarchical clustering (HC):** This technique is used in gene clustering. The results of this technique are affected by noise. HC is also unresponsive to missing data, and it has a problem in providing details, such as number of clusters and trust measures for individual clusters. Points may be clustered by HC on the basis of local decisions, without an opportunity to reexamine the clustering. Dispersed genes make HC algorithms, such as BIRCH [5], CURE [6], and ROCK [7], extremely powerless.

- **Partition-based clustering:** This type of clustering divides data into groups based on their distance. The most used method is Euclidean distance, which finds the minimum distance between points with each available combination and designates the point for a group. This method is dependent on the selected k clusters, such as k -means [8].

- **Fuzzy-based clustering:** This type of clustering allows genes to be grouped partially into more than one cluster. It assigns a membership degree between (0,1), such as fuzzy c-shells (FCS) [9] and fuzzy c-means (FCM) [9], to each gene.

- **Density-based clustering:** This type of clustering detects dense areas in object space, with clusters separating areas with low density. As long as density exceeds a certain threshold, the given cluster will continue to grow in dense regions. Thus, extracting statistically relevant patterns is possible. This clustering technique, which includes density-based spatial clustering of applications with noise (DBSCAN) [10] and ordering points to identify the clustering structure (OPTICS) [11], can be applied to noisy data.

- **Grid-based clustering:** The technique is used for multidimensional data. It creates a grid structure and performs comparisons on grids. Grid-based techniques are fast and have low computational complexity. Examples include CLIQUE [12] and STING [13].

- **Model-based clustering:** This technique uses mathematical models for improvement and determines the suitability of a particular set of data. Model-based clustering techniques find information about the features of each group, with each group representing a concept or class. These techniques include ART [14] and SOM [15].

The remainder of this paper is organized as follows. Section two presented related work on fuzzy gene expression. Section three explains gene expression. Section four describes a microarray. Section five details the methodology. Section six provides the dataset. Section seven presents the result. The study is concluded in the last section.

2. RELATED WORK

In Izakian, H. and Abraham, A. (2011), a hybrid fuzzy clustering method based on FCM and fuzzy particle swarm optimization (PSO) that utilizes the advantages of both algorithms was proposed. Experimental results over six well-known datasets (i.e., Iris, Glass, Cancer, and Wine) demonstrated that the proposed hybrid method is efficient and can

provide highly encouraging results in terms of the quality of the found solution [16]. Chen, S., Xu, Z., and Tang, Y. (2014) developed a hybrid clustering algorithm based on an improved PSO and FCM. Their experimental results showed that the developed method can escape local optima and provide better results [17]. Raichada, N. and Deolekar, R. (2020) presented a new application of FCM–PSO hybridization to clustering analysis. The achievement of FCM–PSO was compared with those of the original PSO clustering and the FCM method; it was determined that the former exhibit better precision and speed. Accordingly, the proposed approach can be considered a strong clustering method, and the hybrid combination provides higher optimization [18]. Poddar and Mukhopadhyay (2020) proposed an EXCLUSIVE and created an environment for analyzing the performance of different clustering techniques on gene expression data. Their program has user-friendly graphical interface for visually and statistically evaluating different clustering algorithms. Some common clustering algorithms have been implemented, including k-means, FCM, and HC; users may easily examine the quality of clustering solutions and observe visual representations of clustering results [19].

3. GENE EXPRESSION

Gene expression is a method through which the encoding of a gene in DNA is used to produce protein [20]. It is the process of converting a gene's information contents into an output protein. Although certain genes, such as those that code for ribosomal and transition RNA, produce RNA as functional output, the vast majority of genes are involved in the cell coding of proteins. Proteins play a variety of roles in biomolecules, structural components, enzymes, antibodies, and other biomolecules. The encoding information of proteins is carried by genes in DNA molecules. Transcription and translation are the two phases of gene expression, which converts genes into proteins. During transcription, a piece of DNA that forms a gene in a DNA molecule is transcribed into the single-stranded RNA sequence known as messenger RNA (mRNA). After several alterations, the mRNA is translated into an amino acid sequence, which is then transformed into a protein, as shown in Figure 2.

FIGURE 2. Gene expression

4. MICROARRAY TECHNOLOGY

The capability to measure differences in gene expression has been one of the goals of biologists for many years. Until recently, however, this goal has been difficult to attain. With thousands of genes in a living organism, the time it will take to extract individual genes and compare them on a normalized scale is daunting, and progress can only be achieved in extremely small steps. With the establishment of the Human Genome Organization in 1989, the demand for a method for measuring differential gene expression increased considerably. A microarray has made simultaneously measuring the expression levels of tens of thousands of genes feasible. Gene arrays are currently widely used in basic biomedical research for mRNA expression profiling. They are also being increasingly used in clinical research to investigate gene expression patterns. Complementary DNA (cDNA) microarrays and oligonucleotide arrays are the two most common forms of microarray experiments. Regardless of the differences in their methods, both types of experiments must include the following three essential processes [21].

1- Chip manufacture: A tiny chip that includes thousands of DNA molecules in grids and is made of chemically treated glass, nylon membrane, or silicon. Each grid cell represents a different DNA sequence.

2- Hybridization and labeling: Two mRNA samples are reverse-transcribed into cDNA, which is then tagged with fluorescent dyes or radioactive isotopes before being used as probes on the chip's hybridized surface.

3- Scanning process: Signal strength is measured through the tagged and hybridized targets.

An oligo chip and a cDNA microarray use the signal force ratio between the test and control samples to determine the expression level for each DNA sequence. Thus, the datasets generated by both technologies exhibit the same biological significance.

5. METHODOLOGY

The section describes clustering, FCM, optimization methods, and the validation process.

5.1 FCM ALGORITHM

FCM is one of the most commonly used partition methods. It allows each data point to belong to a cluster to a certain degree specified by a membership grade. The FCM algorithm introduces the best result for overlapped data. It is relatively better than k-means. The basic idea of this type of clustering algorithms is that FCM allows an object to belong to two or more clusters with a membership grade between zero and one, wherein a data point is assigned depending on its membership degree to each cluster. Consequently, a data point may belong to more than one cluster. Some disadvantages of FCM are as follows: (1) the number of clustering should be determined, (2) easily falls into the local optimum level, and (3) the clustering results are sensitive to the values of the initial parameters. A U matrix with numbers ranging from 0 to 1 is created and used to represent the membership degree between the data and group centers [22, 23]. The membership and cluster centers are adjusted after each iteration by using the following formulas:

$$U_{ij} = \frac{1}{\sum_{k=1}^n \left(\frac{d_{ij}}{d_{ik}} \right)^{\frac{1}{m-1}}} \quad (1)$$

$$V_j = \frac{\sum_{i=1}^n (u_{ij})^m (X_i)}{\left(\sum_{i=1}^n (u_{ij})^m \right)} \quad (2)$$

where U denotes membership; n indicates the number of data points; m is the fuzziness, and $m \in [1, \infty]$; V represents the cluster center; c indicates the number of cluster centers; U_{ij} represents the membership of the data; and $d_{ij} = \|x_i - v_j\|^2$ clarifies the distance between the data and cluster center. The goal of FCM is to minimize the objective function:

$$J(U, V) = \sum_{i=1}^n \sum_{j=1}^c u_{ij}^m d^2(X_i - V_j), \quad (3)$$

where J is the objective function; U is a fuzzy partition matrix; u_{ij} is a membership coefficient of an object in the cluster; $V = [v_1, \dots, v_c]$ is the centroid; m is a fuzzification parameter, and $m \in [1, \infty]$; and d_{ij} is the distance between the data and cluster center.

Algorithm 1: FCM Algorithm

Input: Dataset, c is the number of clusters, m is a fuzzifier, k is an iteration, and ε is the termination criterion between [0,1]

Output: Dataset divided into clusters

Step 1. Initialize the $U = u_{ij}$ matrix.

Step 2. Compute fuzzy membership u_{ij} by using Eq. 1.

Step 3. Find the fuzzy centers v_j by using Eq. 2).

Step 4. If $\max \|U(k+1) - U(k)\| < \varepsilon$, then stop .

Step 5. else $It = It + 1$, proceed to Step 2.

5.2 OPTIMIZATION METHODS

PSO [24, 25] and genetic algorithm (GA) [26] are common evolutionary methods.

5.2.1. PSO

As a stochastic optimization approach, Kennedy was the first to propose PSO. Particles are flown over a multidimensional search space. Each particle tests a possible solution as it moves in the problem space. The particle's best solution (pbest) and the global best solution identified by all the particles (gbest) generate an attractive force that influences the velocity of particles in the search space. Consequently, the particle interacts with all of its neighbors and remembers the best possible position data. After each iteration, if the particle or population discovers a better optimal solution, then (pbest) and (gbest) are adjusted correspondingly. This process is repeated until the desired result is obtained or the computational capability is exhausted [24]. The procedure for this algorithm is provided in Algorithm 2. PSO defines any particle as $X_i = (X_{i1}, X_{i2}, \dots, X_{id})$, where subscript i is the particle number, and k is the maximum number of clusters. The memory of the previous best position (pbest) is represented as $P_i = (P_{i1}, P_{i2}, \dots, P_{id})$, and the velocity along each dimension is $V_i = (V_{i1}, V_{i2}, \dots, V_{id})$. Velocity is updated after iteration. The motion of a particle is affected by its optimum and global best positions.

$$V_i^{(t+1)} = W V_i^t + c_1 r_1 [pbest^{(t)} - X_i^{(t)}] + c_2 r_2 [gbest^{(t)} - X_i^{(t)}] \quad (4)$$

Position updated by

$$X_i^{(t+1)} = X_i^t + V_i^{(t+1)}, \quad (5)$$

where the constants c_1 and c_2 are the learning factor. The memory of PSO is controlled by

Insert equation,

where W is the inertia weight; and r_1 and r_2 are randomly generated values between 0 and 1. $(Pbest - X_i)$ is the difference between the local best P_i of the i th and previous position X_i . Similarly, $(gbest - X_i)$ is the difference between the global best and the previous position X_i .

Algorithm 2 : PSO

Input: p size swarm, w inertia weight, c_1 cognitive coefficient, c_2 social coefficient, $MaxIt$ maximum iteration.

Output: Velocity V_i , position X_i , and clustering results.

Step 1. Create P , and then initialize X_i , V_i , $pbest$, and $gbest$.

Step 2. For each particle, compute the fitness value.

Step 3. For each particle, compute $pbest$; for each swarm, compute $gbest$.

Step 4. Update V_i to all particles by (4).

Step 5. Update X_i to all particles by (5).

Step 6. If the maximum value is reached, then stop; otherwise, proceed to Step 2.

5.2.2. GA

A GA is an optimization and research method based on evolution mechanisms, such as biological genetics and natural selection. This technique is known as an evolutionary algorithm. It is an approach that relies on mimicking the work of nature from a Darwinian perspective. To find a solution or approximate optimization solution, GAs use search techniques. GAs are important and prominent techniques in the search for an optimal choice of set solutions. Genetic manipulation exhibits optimal advantages through successive reproductive processes. It supports traits that have the greatest capability to produce optimal offspring; offspring quality gradually improves by repeating the genetic cycle [26]. Five phases are considered in GA: initial population, fitness function, selection, crossover, and mutation. The procedure for the algorithm is described in Algorithm 3.

Algorithm 3: GA)

Input: np is population size, pc is a crossover, pm is a mutation, and $MaxIt$ is a maximum iteration .

Output: Optimal solution.

Step 1. Randomly initialize pop.

Step 2. Repeat.

Step 2. Evaluate the objective function.

Step 3. Selection.

Step 4. Crossover between the positions of parent1 and parent2.

Step 5. Mutation.

Step 6. If $MaxIt$ is not met, then proceed to Step 2.

5.3 VALIDATION

In this subsection, two types of solution validation criteria are used: internal validity and external validity.

5.3.1. Internal validity

Internal validity indexes are used to assess a clustering solution's quality on the basis of geometric characteristics, such as cluster separation and connectivity. These indexes can be used as a metric for determining which clustering structure is best for a given dataset. The following indexes are implemented with the objective of validating the solution.

A- Jaccard (J) index [27]:

J is a measure of how many objects two sets have in common as a percentage of the total number of objects in both sets. It is used to compare the similarity and diversity of datasets. Its definition is as follows:

$$J = \frac{A}{A + B + C}, \quad (6)$$

where J is the Jaccard similarity coefficient, A is a number of objects in common, B is a number of species unique to the first cluster, and C is a number of species unique to the second cluster.

B- Davies Bouldin (DB) index [28] :

DB is a clustering algorithm evaluation metric. It is used to maximize inter-cluster distance while simultaneously attempting to minimize distance between cluster points. The within-cluster case exhibits a positive correlation, whereas

the between-cluster case presents a negative correlation.

$$DB = \frac{1}{C} \sum_{i=1}^C \max_{j \neq i} \left\{ \frac{d(X_i) + d(X_j)}{d(c_i, c_j)} \right\} \quad (7)$$

where C is the number of clusters, and $d(X_i)$ and $d(X_j)$ denote all samples in clusters to their respective cluster centroid. $d(c_i, c_j)$ indicates the distance between these centroid. To ensure that appropriate clustering is achieved, the DB index's value must be minimized.

C- Dunn index [28]:

It is a clustering algorithm evaluation metric wherein the outcome is determined by the clustered data. The goal is to find clusters that are compact, well separated, and have a small variance between members. A higher Dunn index indicates better clustering.

$$D = \min_{1 \leq i < k} \left\{ \min_{1 \leq j < k, j \neq i} \left\{ \frac{d(c_i, c_j)}{\max_{1 \leq k < K} d(X_k)} \right\} \right\} \quad (8)$$

where $d(c_i, c_j)$ is the distance between two clusters X_i and X_j , $d(X_k)$ represents the intra-cluster distance of a cluster, and c is the number of clusters. A higher Dunn index value indicates compact and well-separated clusters.

D- Xie Beni (XB) index [29]:

It is a metric that divides total variance into fuzzy clusters σ on a minimum separation. **Sep** is the number of clusters, Σ . Sep is defined as

$$\sigma = \sum_{k=1}^k \sum_{i=1}^n u_{ki}^m D^2(X_k, c_i) \quad (9)$$

and

$$sep = \min_{k \neq 1} D^2(c_i, c_j) \quad (10)$$

XB is defined as follows:

$$XB = \frac{\sigma}{n * sep} \quad (11)$$

The smaller the σ value, the higher the sep value, the better and more compact the partitioning. Consequently, we must minimize XB to achieve proper clustering.

E- Silhouette index [28]:

It is a measure used to determine how similar an object is to the group to which it belongs compared with other clusters.

$$Si = \frac{(b_i - a_i)}{\max(a_i, b_i)} \quad (12)$$

$$S = \frac{1}{Ci} \sum_{i=1}^n Si \quad (13)$$

where c represents the number of points belonging to the i cluster.

Silhouette ranges from -1 to 1 . A high value indicates that the object is well matched to its cluster but poorly matched to surrounding clusters. A low value indicates that an object is poorly suited to neighboring clusters.

5.3.2. External validity

External validity indexes are used to compare the final solution to the genuine data clustering. When true clustering is known, these indices can be used to compare the efficiency of different clustering techniques. Let **a** be a collection of pairs that belong to the same cluster. In Q and R, **b** is the collection of pairs that belong to the same cluster in Q but to various clusters in R. **c** is the number of pairs that belong to various clusters in Q, but in the same cluster in R. **d** is the collection of pairs that belong to various clusters in Q but the same cluster in R. The external validity indexes are as follows.

A- Minkowski index [30]:

$$M(Q, R) = \sqrt{\frac{b+c}{a+b}} \quad (14)$$

A lower Minkowski index suggests greater fit between Q and R

B- Adjusted Rand index [31]:

$$ARI(Q,R) = \frac{2(a*d - b*c)}{(a+b).(b+d) + (a+c).(c+d)} \quad (15)$$

A higher value of this index indicates the similarity of Q to R

C- Percentage of correctly classified pairs index [32]:

$$P(Q,R) = \frac{((a+d) * 100)}{(a+b+c+d)} \quad (16)$$

6. DATASET

In this research, experiments are conducted on brain tumor, which is considered one of the most dangerous diseases that frequently have fatal outcomes. At present, increasing attention has been given to the study of brain tumors. To compare the benefits of the suggested system's clustering methods, a brain tumor dataset is used in the experiments. This dataset consists of 4000 genes distributed in the form of 400 rows and 10 columns, where rows represent gene name and columns are the samples, i.e., a gene's value at a given point in time [19]. The dataset is selected from the source <http://www.kucse.in/excluvis>. The format of the brain tumor dataset is presented below.

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N
2	W95908	1.5962	0.534	-1.8179	-0.03502	1.8996	0.30639	-0.71784	-0.33849	-1.1351	-0.30056	0.30639	-0.22469	-0.07295
3	AA045003	0.095122	2.0874	0.26687	0.61037	0.85082	-1.485	-0.83232	0.52317	-0.52317	0.24838	-0.83232	-0.76362	1.2974
4	AA044434	0.84243	1.359	0.68745	0.84243	0.32584	-0.8623	-0.13908	0.22253	-0.34571	-0.65566	-1.3789	-1.9471	1.0491
5	W88572	1.1363	0.98137	1.1363	0.36156	0.30991	-0.30991	-0.61982	0.51651	-1.6012	-0.87807	-0.98137	-0.67147	1.6528
6	AA029909	0.48347	1.4065	1.7581	0.83508	-0.04395	-1.0109	-0.17581	-0.96694	-1.3625	-0.30766	-1.2746	0.65927	
7	AA039077	0.23452	1.6489	1.6175	0.61169	0.54883	0.70596	-0.42552	-0.48838	-0.45695	-0.67697	-0.80269	-1.777	-0.73983
8	AA035657	0.64514	1.2043	1.0179	0.64514	0.94334	-0.13763	0.57059	0.75696	-0.88313	-0.95767	-1.3677	-1.6286	-0.80658
9	AA180272	0.28406	1.3116	1.408	0.70151	0.28406	-0.00494	0.99051	0.44462	-1.0646	-1.0967	-1.2252	-1.5463	-0.48661
10	W89011	0.65322	0.6895	2.0322	1.4153	-0.10887	0.10887	-0.14516	-0.52661	-0.65322	-0.87095	-1.4153	-1.2701	-0.10887
11	H15013	1.0642	0.47966	1.1812	1.0642	0.32377	0.36274	0.011991	-0.8454	-1.8977	-0.72848	-1.1572	-0.8454	0.3863
12	N22392	1.0385	1.4728	0.74892	1.0747	0.53176	0.53176	-0.04733	-0.33687	-1.2779	-1.2417	-1.3141	-1.2055	0.025057
13	AA026170	0.75974	1.6334	1.5575	0.5698	0.037987	0.41786	0.26591	-0.18993	-1.5195	-0.98766	-1.0636	-1.0636	-0.41786
14	W89017	1.0516	0.90057	1.7692	1.3538	0.22078	-0.42123	-0.72336	0.68559	-1.4409	-0.42123	-0.78889	-0.98772	1.18302
15	N32247	2.0661	0.71563	1.3158	0.76564	0.16544	-0.08464	-0.58481	-0.78488	-0.8349	-0.28471	-1.035	-1.4351	0.01539
16	T62968	1.6601	0.27039	-0.42445	-0.01572	1.5793	1.047	-0.66969	-1.4054	-1.4871	0.1669	0.31126	-0.17921	-0.79231
17	H28360	1.2298	1.1397	0.41917	0.68938	0.41917	0.68938	-1.112	-1.7425	-1.0219	0.91455	-0.52656	0.10393	-1.2021
18	H16592	1.4802	1.5258	0.61383	0.47704	-0.16135	0.97863	-0.98213	-1.8941	-0.75414	0.15784	-0.34375	-0.66294	-0.43495
19	W78151	0.63132	0.99811	1.4108	0.76886	0.81471	0.86056	-0.42323	-1.9363	-1.0193	-0.56078	0.081119	-0.88173	-0.74418
20	H14996	2.1344	1.4977	-0.22026	-	1.1013	-0.1762	-0.44051	-0.88102	-1.4096	-0.88102	-0.13215	-0.48456	-0.0881
21	H14499	2.0289	1.1227	-0.40357	0.83649	1.1704	0.12107	-0.59435	-1.0713	-1.3575	-0.54665	-0.40357	-0.1651	-0.73743
22	H81336	2.2445	0.65192	-0.42878	0.31064	1.4482	-0.20126	0.19689	-1.282	-1.0544	-0.77005	-0.3719	0.026252	-0.77005
23	H47974	1.9998	1.2105	0.47053	0.47053	0.47053	0.51986	-0.31875	-1.0587	-1.404	0.22388	-1.0094	-1.0094	-0.5654
24	N75026	1.3251	0.92348	0.70034	0.12016	1.95	-0.14762	-0.23688	-1.1741	-1.2634	0.20941	-0.99558	-0.7278	-0.68317
25	H87731	1.3526	0.93393	0.35827	0.35827	1.4049	-0.21738	-0.8977	-1.5257	-1.264	0.096613	-0.95003	-0.63604	0.98626
26	H26264	1.2519	1.1707	-0.20916	0.23726	1.9012	0.23726	-0.57442	-1.6702	-1.2238	0.19668	-0.24975	-0.49325	-0.57442
27	H61273	0.31678	1.6623	-0.49869	0.84682	1.54	0.35755	-0.09096	-1.5996	-1.5996	-0.13173	-0.1725	-0.70255	0.072137
28	H63445	1.2255	1.2255	-0.15064	0.96084	0.96084	0.64327	-0.20357	-1.1033	-2.0031	0.16693	-0.25649	-1.1033	-0.36235
29	W60845	0.68492	1.4069	0.39614	0.011107	1.5513	1.0218	-0.99962	-1.3365	-1.6253	-0.42706	-0.56645	-0.13328	0.011107
30	H60336	1.5265	0.41113	1.3871	0.55055	-0.42543	-0.8437	-0.33248	-1.4014	-1.169	1.4801	0.039325	-0.61133	-0.61133
31	N53427	1.0859	0.55397	1.0041	0.43121	0.59489	-0.63266	-0.96	-1.0828	-1.3283	1.9452	-0.67357	-0.34623	-0.59174

7. PROPOSED METHOD

The proposed system of fuzzy-based clustering for gene expression data,

7.1 PREPROCESSING

This phase is implemented by applying a series of initial processing procedures to the data before they are used to improve data quality and remove noise. Before applying the analysis, the gene dataset must be preprocessed to eliminate outliers, missing values, and any incorrect data. This stage consists of the following substages.

- Unnecessary columns are deleted. In this step, the dataset contains unnecessary columns that do not contain information that can help us in the clustering process. Thus, these columns are deleted.
- The missing values are filled. Missing values are processed by obtaining the average values for each column and using them in place of the missing values or by replacing them with adjacent values.

7.2 CLUSTERING ALGORITHMS**7.2.1. FCM algorithm**

After the data processing stage, the clustering process is implemented using some clustering algorithms. The second algorithm selected is FCM, which is used in grouping genes within the required clusters. Algorithm 4 presents the steps in FCM.

Algorithm 4 : Steps in FCM clustering

Input: Data after preprocessing.

Output: A list of cluster centers is produced.

Step 1. Load data.

Step 2. Determine the number of clusters.

Step 3. Apply the FCM algorithm depending on the Euclidean distance (Eq. 3).

FIGURE 3. System diagram

Step 4. Calculate the internal validity indexes, which include J (Eq. 6), DB (Eq. 7), Dunn (Eq. 8), XB (Eq. 11), and silhouette (Eq. 13).

Step 5. Calculate the external validity indexes, which include Minkowski (Eq.14), adjusted Rand (Eq.15), and percentage of correctly classified pairs (Eq.16).

7.2.2. FCM-PSO

FCM, which extracts cluster centers, is paired with PSO to produce a hybrid technique for addressing the fuzzy clustering problem, particularly for large sizes. When the problem becomes huge, the FCM algorithm may produce an unequal distribution of data, making finding an optimal solution within a reasonable length of time challenging. We demonstrate that by seeding the starting swarm with the outcome of the c-means approach, the PSO algorithm can provide good or near-optimal solutions within a reasonable amount of time. The proposed object function is the DB index, as shown in Algorithm 5.

Algorithm 5: Hybrid FCM-PSO

Input: Dataset, k, nPop, VarMin, VarMax, MaxIt, w, wdamp, c_1 , c_2 .

Output: Data clustering.

Step 1. Select the number of population and initial by using FCM.

Step 2. Calculate pbest and gbest.

Step 3. Ensure the lower- and upper-bound variables.

Step 4. It = 1:MaxIt

Step 5. Update velocity by using Eq. 4.

Step 6. Update position by using Eq. 5.

Step 7. Evaluation of the objective function by using Eq. 7 .

Step 8. Update personal pbest and gbest.

Step 9. If It = MaxIt, then proceed to Step 10; else, return to Step 4.

Step 10. Plot (data, best position, global position, k).

7.2.3. FCM-GA

At this stage, FCM, in which the centers of clusters are extracted, is combined with GA, in which the clustering results are optimized. The proposed objective function is the DB index, as shown in Algorithm 6.

Algorithm 6: Hybrid FCM-GA

Input: Dataset, number of chromosomes

Output: Data clustering.

Step 1. Select the number of chromosomes and initial by using FCM.

Step 2. It = 1:Max.It

Step 3. Select the chromosomes by performing roulette wheel selection.

Step 4. Crossover between the positions of parent₁ and parent₂.

Step 5. Evaluate the objective function by using Eq. 7 .

Step 6. Generate random number and use in mutation.

Step 7. If It = Max.It, then proceed to Step 8; else, It = It+1 and return to Step 2.

Step 8. Plot (data, best position, U, k).

8. RESULTS

8.1 RESULTS OF FCM

Table 1. Results of clustering based on the FCM algorithm with internal validation when the range of k is 2–5.

k	Objective Function	J Index	DB Index	Dunn Index	XB Index	Silhouette Index
2	225.1607	0.65367	1.513	-1.2272	0.9246	0.4297
3	136.0838	0.81448	1.1675	-2.0319	0.7977	0.4589
4	95.4698	0.92429	1.1682	-1.3623	0.7897	0.4305
5	71.0966	0.95868	1.1977	-1.3987	0.7681	0.4841

Table 1: Results obtained using the FCM algorithm with internal validation. In the case of using the J index, the best result is obtained at Cluster 5. In the case of using the DB index, the best result is achieved at Cluster 3. When using the Dunn index, the best result is in Cluster 2. In the case of using the XB index, the best result is in Cluster 5. Finally, when using the Silhouette index, the best result is achieved in Cluster 5.

Table 2. Result of clustering based on the FCM algorithm with external validation when the range of k is 2–5.

k	Minkowski	Adjusted Rand	Percentage
2	1.1261	-0.021474	49.411
3	1.0144	0.003378	50.091
4	1.1201	-0.021033	49.441
5	1.0337	0.009423	50.07

Table 2: Results obtained using the FCM algorithm with external validation. In the case of using the Minkowski index, the best result is in Cluster 3. In the case of using the adjusted Rand index, the best result is in Cluster 5. Finally, when using the percentage index, the best result is in Cluster 3.

8.2 RESULT OF FCM-PSO

This subsection provides the implementation results of the hybrid FCM-PSO algorithm. The hybrid algorithm uses the DB index as objective function. The experimental results show that the proposed hybrid FCM-PSO method is effective. The total time of implementation of the model is 89.6087 s when using the DB objective function.

Table 3. Result of clustering based on FCM-PSO with internal validation when the range of k is 2–5.

k	Objective Function	J Index	DB Index	Dunn Index	XB Index	Silhouette Index
2	0.96154	0.37903	1.5122	-1.2279	0.71436	0.43015
3	0.91572	0.54506	1.152	-2.1226	0.47293	0.45979
4	0.93952	0.1616	1.1617	-1.3628	0.66683	0.43346
5	0.9165	0.30479	1.1977	-1.3987	0.64241	0.48419

Table 3: Results obtained using the FCM-PSO clustering algorithm with internal validation. In the case of using the J index, the best result is in Cluster 3. In the case of using the DB index, the best result is also in Cluster 3. When using the Dunn index, the best result is in Cluster 2. In the case of using the XB index, the best result is in Cluster 3. Finally, when using the Silhouette index, the best result achieved is in Cluster 5.

FIGURE 4. Clustering via FCM when k=2–5

Table 4. Result of clustering based on FCM–PSO with external validation when the range of k is 2–5.

k	Minkowski Index	Adjusted Rand Index	Percentage Index
2	1.1261	–0.02143	49.411
3	1.0631	–0.02194	49.387
4	1.1489	–0.00977	49.657
5	1.0245	–0.01245	49.689

Table 4: Results obtained using the FCM–PSO clustering algorithm with external validation. In the case of using the Minkowski index, the best result is in Cluster 5. In the case of using the adjusted Rand index, the best result is in Cluster 4. Finally, when the percentage index is used, the best result is in Cluster 5.

FIGURE 5. Clustering via FCM–PSO when k=2–5

8.3 RESULT OF FCM–GA

This subsection presents the implementation results of the hybrid FCM–GA algorithm. This hybrid algorithm uses the DB index objective function. The experimental results show that the proposed hybrid FCM–GA is effective. The total time of implementation is 50.8021 s when using the DB index objective function.

Table 5. Result of clustering based on FCM–GA with internal validation when the range of k is 2–5.

k	Objective Function	J Index	DB Index	Dunn Index	XB Index	Silhouette Index
2	0.96154	0.16432	1.513	–1.2272	1.1037	0.42979
3	0.91572	0.5478	1.1543	–2.1084	1.0033	0.45943
4	0.93952	0.60194	1.1556	–1.3586	0.79425	0.43559
5	0.92004	0.30476	1.1977	–1.3987	0.35958	0.48419

Table 5: Results obtained using the FCM–GA clustering algorithm with internal validation. In the case of using the J index, the best result is in Cluster 4. In the case of using the DB index, the best result is in Cluster 3. When the Dunn index is used, the best result is in Cluster 2. In the case of using the XB index, the best result is in Cluster 5. Finally, when the silhouette index is used, the best result is achieved in Cluster 5.

Table 6. Result of clustering based on FCM–GA with external validation when the range of k is 2–5.

k	Minkowski Index	Adjusted Rand Index	Percentage Index
2	1.1261	−0.021474	49.411
3	1.006	0.004018	50.191
4	1.047	−0.027107	49.18
5	1.0653	−0.020107	49.438

Table 6: Results obtained using the FCM–GA clustering algorithm with external validation. In the case of using the Minkowski index, the best result is in Cluster 3. When the adjusted Rand index is used, the best result is achieved in Cluster 3. Finally, when the percentage index is used, the best result is achieved in Cluster 3.

FIGURE 6. Clustering via FCM–GA when k=2–5

9. COMPRESSION RESULT

9.1 INTERNAL VALIDITY

In this section, the results of using internal validity measures are reviewed and compared with the proposed aggregation algorithms on the dataset.

Table 7. Comparison of the algorithms when using the J index and k=2–5.

k	FCM	FCM–PSO	FCM–GA
2	0.65367	0.37903	0.16432
3	0.81448	0.54506	0.5478
4	0.92429	0.1616	0.60194
5	0.95868	0.30479	0.30476

Table 7: Comparison of the results obtained using the J index. The best result is achieved using the FCM algorithm when the iteration is 5. When using the FCM–PSO algorithm, the best result is achieved when the iteration is 2. When using the FCM–GA algorithm, the best result is obtained when the iteration is 2.

Table 8: Comparison of the results obtained using the DB index. The best result is achieved using the FCM algorithm when the iteration is 3. In the case of using the FCM–PSO algorithm, the best result is achieved when the iteration is 3. When using the FCM–GA algorithm, the best result is obtained when the iteration is 3.

Table 8. Comparison of the algorithms when using the DB index and k=2-5.

k	FCM	FCM-PSO	FCM-GA
2	1.513	1.5122	1.513
3	1.1675	1.152	1.1543
4	1.1682	1.1617	1.1556
5	1.1977	1.1977	1.1977

Table 9. Comparison of the algorithms when using the Dunn index and k= 2-5.

k	FCM	FCM-PSO	FCM-GA
2	-1.2267	-1.2279	-1.2272
3	-2.0387	-2.1226	-2.1084
4	-1.2631	-1.3628	-1.3586
5	-1.3987	-1.3987	-1.3987

Table 9: Comparison of the results obtained using the Dunn index. The best result is achieved using the FCM algorithm when the iteration is 2. In the case of using the FCM-PSO algorithm, the best result is obtained when the iteration is 2. In the case of using the FCM-GA algorithm, the best result is achieved when the iteration is 2.

Table 10. Comparison of the algorithms when using the XB index and k=2- 5.

k	FCM	FCM-PSO	FCM-GA
2	0.89574	0.71436	1.1037
3	0.79734	0.47293	1.0033
4	0.67385	0.66683	0.79425
5	0.76808	0.64241	0.35958

Table 10: Comparison of the results obtained when using the XB index. When using the FCM algorithm, the best result is achieved when the iteration is 4. When the (FCM-PSO) algorithm is used, the best result is achieved when the iteration is 3. When using the FCM-GA algorithm, the best result is obtained when the iteration is 5.

Table 11. Comparison of the algorithms when using the silhouette index and k=2- 5.

k	FCM	FCM-PSO	FCM-GA
2	0.42979	0.43015	0.42979
3	0.45893	0.45979	0.45943
4	0.42096	0.43346	0.43559
5	0.48419	0.48419	0.48419

Table 11 Comparison of the results obtained when using the silhouette index. The best result is achieved using the FCM algorithm when the iteration is 5. When the FCM-PSO algorithm is used, the best result is obtained when the iteration is 5. When using the FCM-GA algorithm, the best result is realized when the iteration is 5.

Table 12. Comparison of the algorithms when using the Minkowski index and k= [2-5].

k	FCM	FCM-PSO	FCM-GA
2	1.1261	1.1261	1.1261
3	1.0144	1.0631	1.006
4	1.1201	1.1489	1.047
5	1.0337	1.0245	1.0653

Table 12 Comparison of the results obtained when using the Minkowski index. The best result using the FCM algorithm is achieved when the iteration is 3. When using the FCM-PSO algorithm, the best result is obtained when the iteration is 5. In the case of using the FCM-GA algorithm, the best result is achieved when the iteration is 3.

Table 13: Comparison results obtained using the adjusted Rand index. The best result of the FCM algorithm is obtained when the iteration is 5, that of the FCM-PSO algorithm is achieved when the iteration is 4, and that of the FCM-GA algorithm is obtained when the iteration is 3.

Table 13. Comparison of the algorithms when using the adjusted Rand index and k=2– 5.

k	FCM	FCM-PSO	FCM-GA
2	-0.0214	-0.0214	-0.0214
3	0.0033	-0.0219	0.0040
4	-0.0210	-0.0097	-0.0271
5	0.0094	-0.0124	-0.0201

Table 14. Comparison of the algorithms when using the percentage index and k=2– 5.

k	FCM	FCM-PSO	FCM-GA
2	49.411	49.411	49.411
3	50.23	49.387	50.191
4	49.416	49.657	49.18
5	50.3	49.689	49.438

Table 14: Comparison of the results obtained using the percentage index. The FCM algorithm achieves the best result when the iteration is 5. In the case of using the FCM-PSO algorithm, the best result is obtained when the iteration is 5. In the case of using the FCM-GA algorithm, the best result is achieved when the iteration is 3.

10. DISCUSSION

Through research on gene expression, a conclusion was drawn that gene expression exhibits overlaps and that the solid assembly process does not detect overlapping genes, thus, the use of flexible assembly algorithms to perform the assembly process is required. One of these algorithms is FCM. This algorithm is sensitive to initialization and easily falls into the local optimum. Therefore, evolution strategies are used to improve results and avoid falling into the local minimum.

When the J index is used as a measure for internal validation, the best algorithm when k=2,3,4,5 is FCM. If the measure used for internal validation is DB, then the best algorithm when k=2, 3 is FCM-PSO and FCM-GA when k=4 . When k=5 is equal, the result for all algorithms is equal. If the measure used for internal validation is the Dunn index, then the best algorithm when k=2 is FCM. When k=3, the best result is FCM-GA. When k=4, the best result is FCM. The result for all algorithms is equal when k=5. If the measure used for internal validation is XB, then the best algorithm when k=2,3,4 is FCM-PSO. Meanwhile, the best algorithm when k=5 is FCM-GA. When the measure used for internal validation is the silhouette index, the best algorithm when k=2, 3, 4 is FCM-PSO. Finally, the result for all algorithms is equal when k=5.

11. CONCLUSION

Data on gene expression mask the basic knowledge necessary to understand the biological mechanism that occurs in a single organism in relation to its environment. These data prevents ambiguity, inaccuracies, and noise. To extract useful information regarding gene activity in relation to different systemic conditions, three evolutionary algorithms were compared through gene analysis by using a tumor dataset. The proposed algorithms exhibit the capability to solve many defects inherent in traditional methods. In our future work, we will expand this research in the following directions. First, we will conduct more experiments on different datasets. Second, we will work with biologists to obtain more information about microarray data.

ACKNOWLEDGEMENTS

The first author thanks the reviewers for their useful suggestions that improved the presentation of this paper.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

- [1] D. Dembele and P. Kastner, "Fuzzy C-means method for clustering microarray data," *Bio informatics*, vol. 19, pp. 973–980, 2003.
- [2] R. Xu and D. Wunsch, "Survey of clustering algorithms," *IEEE Trans. Neural Networks*, vol. 16, no. 3, pp. 645–678, 2005.

- [3] D. X. Jiang, C. Tang, and A. D. Zhang, "Cluster analysis for gene expression data: A survey," *IEEE Trans. Know. and Data Eng.*, vol. 16, no. 11, pp. 1370–1386, 2004.
- [4] E. Naghieh and Y. Peng, "Microarray Gene Expression Data Mining: Clustering Analysis Review," 2009.
- [5] D. Guo, J. Chen, Y. Chen, and Z. Li, "LBIRCH: an improved BIRCH algorithm based on link," *Proceedings of the 2018 10th International Conference on Machine Learning and Computing*, pp. 74–78, 2018.
- [6] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern recognition letters*, vol. 31, no. 8, pp. 651–666, 2010.
- [7] T. Velmurugan and T. Santhanam, "A survey of partition based clustering algorithms in data mining: An experimental approach," *Information Technology Journal*, vol. 10, no. 3, pp. 478–484, 2011.
- [8] S. Na, L. Xumin, and G. Yong, "Research on k-means clustering algorithm: An improved k-means clustering algorithm," *2010 Third International Symposium on intelligent information technology and security informatics*, pp. 63–67, 2010.
- [9] F. Klawonn, R. Kruse, and H. Timm, "Fuzzy shell cluster analysis," in *Learning, networks and statistics*, pp. 105–119, Springer, 1997.
- [10] K. Khan, S. U. Rehman, K. Aziz, S. Fong, and S. Sarasvady, "DBSCAN: Past, present and future," *The fifth international conference on the applications of digital information and web technologies*, pp. 232–238, 2014.
- [11] D. Djenouri, R. Laidi, Y. Djenouri, and I. Balasingham, "Machine learning for smart building applications: Review and taxonomy," *ACM Computing Surveys (CSUR)*, vol. 52, no. 2, pp. 1–36, 2019.
- [12] X. Yu, G. Yu, and J. Wang, "Clustering cancer gene expression data by projective clustering ensemble," *PloS one*, vol. 12, no. 2, 2017.
- [13] D. Shae, K. W. Becker, P. Christov, D. S. Yun, A. K. Lytton-Jean, S. Sevimli, . . Wilson, and J. T, "Endosomolytic polymersomes increase the activity of cyclic dinucleotide STING agonists to enhance cancer immunotherapy," *Nature nanotechnology*, vol. 14, no. 3, pp. 269–278, 2019.
- [14] A. Baraldi and E. Alpaydin, "Constructive feedforward ART clustering networks. I. IEEE transactions on neural networks, 13, 645-661," 2002.
- [15] T. Kohonen, "Neural Modeling," in *Self-Organizing Maps*, pp. 71–104, Springer, 2001.
- [16] H. Izakian and A. Abraham, "Fuzzy C-means and fuzzy swarm for fuzzy clustering problem," *Expert Systems with Applications*, vol. 38, no. 3, pp. 1835–1838, 2011.
- [17] N. U. Raichada and R. Deolekar, "Enhancing Clustering Techniques with Hybrid Optimization," *2020 IEEE India Council International Subsections Conference (INDISCON)*, pp. 146–150, 2020.
- [18] S. Chen, Z. Xu, and Y. Tang, "A hybrid clustering algorithm based on fuzzy c-means and improved particle swarm optimization," *Arabian Journal for Science and Engineering*, vol. 39, no. 12, pp. 8875–8887, 2014.
- [19] S. Poddar and A. Mukhopadhyay, "EXCLUVIS: A MATLAB GUI Software for Comparative Study of Clustering and Visualization of Gene Expression Data," 2020.
- [20] Z. Wang and V. Palade, "Multi-objective evolutionary algorithms based interpretable fuzzy models for microarray gene expression data analysis," *2010 IEEE International Conference on Bioinformatics and Biomedicine*, pp. 308–313.
- [21] S. Bandyopadhyay, U. Maulik, and Wang, *Analysis of biological data: a soft computing approach*, vol. 3. World Scientific, 2007.
- [22] J. C. Bezdek, "FUZZY-MATHEMATICS IN PATTERN CLASSIFICATION," 1973.
- [23] J. Bezdek, R. Ehrlich, and W. Full, "FCM: the fuzzy c-means clustering algorithm," *Comput Geosci*, vol. 10, pp. 191–203, 1984.
- [24] J. Kennedy, "Swarm intelligence," in *Handbook of nature-inspired and innovative computing*, pp. 187–219, Springer, 2006.
- [25] T. Cura, "A particle swarm optimization approach to clustering," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1582–1588, 2012.
- [26] M. Mahalakshmi, P. Kalaivani, and E. K. Nesamalar, "A review on genetic algorithm and its applications," *International Journal of Computing Algorithm*, vol. 2, no. 2, pp. 415–423, 2013.
- [27] A. Hardy and J. Baune, "Clustering and validation of interval data," in *Selected Contributions in Data Analysis and Classification*, pp. 69–81, Springer, 2007.
- [28] E. Rendón, I. Abundez, A. Arizmendi, and E. M. Quiroz, "Internal versus external cluster validation indexes," *International Journal of computers and communications*, vol. 5, no. 1, pp. 27–34, 2011.
- [29] X. L. Xie and G. Beni, "A validity measure for fuzzy clustering," *IEEE Transactions*, vol. 13, no. 8, pp. 841–847, 1991.
- [30] A. Ben-Hur and I. Guyon, "Detecting stable clusters using principal component analysis," *Functional genomics*, pp. 159–182, 2003.
- [31] K. Y. Yeung and W. L. Ruzzo, "Principal component analysis for clustering gene expression data," *Bioinformatics*, vol. 17, no. 9, pp. 763–774, 2001.
- [32] P. K. Tripathi, S. Bandyopadhyay, and S. K. Pal, "Multi-objective particle swarm optimization with time variant inertia and acceleration coefficients," *Information sciences*, vol. 177, no. 22, pp. 5033–5049, 2007.