Computational Journal of Mathematical and Statistical Sciences
4(xxx), xxx–xxx
DOI:10.21608/cjmss.2025.363038.1130
https://cjmss.journals.ekb.eg/

Research article

Unsupervised Feature Selection via Fuzzy C-Means Clustering and Binary Atom Search Algorithm

Hanadi D. Saleem[1], **Talal F. Hussein**[1], **Fatima M. Hasan**[1,*], **Omar S. Qasim**[1]

[1] Department of Mathematics, College of Computer Science and Mathematics, University of Mosul, Mosul 41002, Iraq; hanadidawood@uomosul.edu.iq, talal.math@uomosul.edu.iq, omar.saber@uomosul.edu.iq.

* **Correspondence:** fatima.zamzoom@uomosul.edu.iq

Abstract: This paper presents a proposed algorithm that combines the Binary Atomic Search (BAS) algorithm and the Fuzzy C-means Method (FCM) technique, called BAS-FCM. The proposed algorithm, BAS-FCM, is based on two-stage data processing. The first stage involves selecting important features from the datasets and discarding unimportant ones using the BAS algorithm, which relies on a proposed fitness function to test feature sets and select the best ones for classification and clustering. The second stage involves classifying an unsupervised datasets using FCM. This innovative combination leverages both features to improve accuracy, efficiency, and usefulness on high-dimensional datasets. Experiments on five different datasets demonstrate that the proposed method consistently achieves higher scores for silhouettes with fewer selected features compared to the default clustering technique. These results demonstrate the effectiveness and robustness of the proposed BAS-FCM method compared to the classical method, making it a promising approach for improving the performance of clustering-based feature selection.

Keywords: Fuzzy C-means; Binary Atom Search; Feature Selection; Clustering.
Mathematics Subject Classification: 03B52, 54C65, 68T10.
Received: 24 February 2025; Revised: 2 June 2025; Accepted: 15 June 2025; Online: 21 June 2025.

1. Introduction

Feature selection (FS) is a very important method of removing some irrelevant features from most of the feature sets and obtaining the most unique information in aggregate data, it aims to rank order ranking in accuracy depending on the characteristics of the complete problem Feature selection can be handled as a combinatorial optimization case It is assumed that classification accuracy and the

computational cost of benefit extraction are well balanced by fitness functions representing classifiers any or other standard accuracy [1].

Fuzzy C-means (FCM) clustering is an unsupervised device learning set of rules used for grouping data factors into clusters, wherein every point can belong to multiple cluster with varying stages of membership [2]. Unlike tough clustering methods like K-approach, which assigns each data factor to a single cluster, FCM assigns membership values between zero and 1 to symbolize the diploma of affiliation with every cluster. The algorithm iteratively updates cluster centers and club values primarily based on minimizing a price characteristic, normally the sum of weighted distances. It's broadly implemented in picture segmentation, sample reputation, and statistics evaluation in which boundaries among companies are uncertain. Fuzzy logic is widely applied in various domains, analysis for COVID-19, and the optimization of machine learning models see [3, 4, 5].

Atom search optimization (ASO) is a natural optimization algorithm based on atomic motion correlation principles. It mimics the behavior of atoms interacting through attractive and repulsive forces [6]. In the algorithm, the candidate solutions are treated as atoms, and their positions in the search space are updated based on their interactions, resulting in an optimal solution ASO makes efficient use of these atomic forces for search and are used effectively in the search area, making it suitable for solving complex optimization problems. It has applications in areas such as industrial design, resource allocation, and machine learning optimization [7], [8].

In this study, we propose a new unsupervised selection method that exploits FCM clustering and BASO properties. The aim of our approach is to select the most appropriate features that contribute to a better understanding of the internal structure of the data, thereby increasing both clustering accuracy and computational efficiency.

This is how the remainder of the paper is structured: Section 2 describes fuzzy C-means clustering, feature selection, and atom search optimization. Atom search algorithm has been obtained in Section 3. In Section 4 the proposed BAS-FCM algorithm is describe. In Section 5 the all results obtained are included. The most important general conclusions and discussion are cleared in Section 6.

2. The feature selection

Following data extraction, the most pertinent features for the classification job are obtained through feature selection. All of the features in the vast feature space of the archived datasets are not useful for categorization [9, 10]. In order to increase accuracy and streamline the procedure, the most essential features are chosen. Several feature selection techniques from the embedded category, wrappers, and filters are assessed. The best three techniques are then chosen based on accuracy from each category (filters, wrappers, and embedding). The accuracy criterion is set at 80%, which indicates that an SVM classifier is used to assess the features chosen by each feature selection method once the top three methods from each category have been chosen. Lastly, those attributes will be chosen as a final [11].

2.1. Fuzzy C-means (FCM)

Fuzzy C-Means Clustering (FCM) is an unsupervised technique. In 1973, Dunn presented the clustering algorithm, and in 1981, Bezdek refined it (Höppner, Klawonn, Kruse, & Runkler, 2000). and is an iterative clustering algorithm built on differences between groups when data specify groups with different numbers of members It allows you to do so [12]. FCM is a clustering algorithm that

extends the classic K-Means algorithm to allow for soft clustering and is particularly useful when dealing with data that may not have clear boundaries between clusters or when a data point belongs to multiple clusters same Deserves It has applications in various fields such as pattern recognition, image segmentation and data mining.

The FCM algorithm can be summarized as follows:

- **Step 1:** Initialization: Start by initializing the following parameters:
  - Number of clusters (C)
  - Fuzziness parameter (M): Determines the degree of fuzziness in the cluster assignments. Typical values range between 1.5 and 2.5.
- **Step 2:** Initialize the membership μ randomly using equation:

$$\sum_{j=1}^{c} M_{ij} = 1, \tag{2.1}$$

  where:

  C: Number of clusters.

  M: is the degree of membership of x_i in the cluster j .
- **Step 3:** Update cluster centers: Calculate centroid using equation:

$$C_j = \frac{\sum_{i=1}^{D} (M_{ij})^m * x_i}{\sum_{i=1}^{D} (M_{ij})^m}, \tag{2.2}$$

  where:

  D : the number of data points.

  C_j: is the center of jth cluster.

  M_{ij} : is the degree of membership of x_i in the cluster j.

  X_i is the ith data point.
- **Step 4:** Calculate dissimilarly between the data points and centroid using the euclidean distance:

$$D_i = \sqrt{(x_i - c_1)^2 + (y_i - c_2)^2}. \tag{2.3}$$

- **Step 5:** Update the new membership using the equation:

$$M_{ij} = \frac{1}{\sum_{k=1}^{c} (\frac{d_{ij}}{d_{kj}})^{\frac{2}{m-1}}}, \tag{2.4}$$

  where:

  d_{ij}: is the distance of ij^{th}.

  d_{kj}: is the distance of kj^{th}.

- **Step 6:** Repeat steps 3, 4 and 5: Iterate the process of updating cluster centers and membership grades until convergence. Convergence happens when the change in cluster centers between iterations falls below a predetermined threshold or when the maximum number of iterations is reached.
- **Step 7:** Output: The final cluster centers represent the centroids of the clusters, and the membership grades indicate the degree of belongingness of each data point to each cluster.

3. Atom Search Algorithm

One of the most current descriptive characteristics of a novel metaheuristic algorithm put forth by Zhao et al. in 2019 is ASO [13]. Complex atomic analysis uses interaction potentials, geometric constraint potentials, and potential functional characteristics to simulate fundamental ideas in the theory of motion of those atoms and molecular dynamics. Atoms with the two vectors of velocity and position under control are known as solvation set (ASO) atoms [14]. The principle of motion states that every atom is constantly moving. The following is a mathematical expression for an atom's velocity:

$$a_i = \frac{F_i + G_I}{m_i}, \tag{3.1}$$

Represents the force interaction by f, constraint force represents by g, The mass of the atom is denoted by m. The strength of interaction between the(ith & jth) atom is determined using Lennard Jones ($L-J$) potentials as :

$$F_{ij}^{d}(t) = \frac{24\varepsilon(t)}{\sigma(t)}\left[2\left(\frac{\sigma(t)}{r_{ij}(t)}\right)^{13} - \left(\frac{\sigma(t)}{r_{ij}(t)}\right)^{7}\right]\frac{r_{ij}(t)}{r_{ij}^{d}(t)}, \tag{3.2}$$

and

$$F_{ij}^{'}(t) = \frac{24\varepsilon(t)}{\sigma(t)}\left[2\left(\frac{\sigma(t)}{r_{ij}(t)}\right)^{13} - \left(\frac{\sigma(t)}{r_{ij}(t)}\right)^{7}\right], \tag{3.3}$$

ε represents the potential depth, length scale is σ, and the distance between two atoms is represented by r, dimension represents by d, and the current iteration is represented by t. We note that Eq.(3.3) is not immediately applicable to the optimization activities. As a result, an updated version and the following form have been created.

$$F_{ij}^{'}(t) = -\eta(t)\left[2\left(h_{ij}(t)\right)^{13} - \left(h_{ij}(t)\right)^{7}\right], \tag{3.4}$$

η We can report the following since it reflects the depth function that arranges the area of attraction and repulsion:

$$\eta(t) = \alpha(t)\left(1-\frac{t-1}{T}\right)^{3} e^{-\frac{20t}{T}}, \tag{3.5}$$

α is the depth of weight symbol, while T stands for the maximum number of iterations. The following expression can therefore be used to calculate function h:

$$h_{ij}(t) = \begin{cases} h_{\min} & \text{if} \quad \frac{r_{ij}(t)}{\sigma(t)} < h_{\min} \\ h_{\max} & els \text{ if} \quad \frac{r_{ij}(t)}{\sigma(t)} \succ h_{\max} \\ \frac{r_{ij}(t)}{\sigma(t)} & \text{otherwise} \end{cases} \tag{3.6}$$

where$h_{\min}$ and $h_{\max}$ represent the lowest and upper boundaries of h to 1.1 and 2.4, respectively. $\sigma(t)$ is determined using the following equation and represents the length scale:

$$\sigma(t) = \left\| x_{ij}(t), \frac{\sum_{j\in k} x_{ij}(t)}{k(t)} \right\|_2, \tag{3.7}$$

and

$$\begin{cases} h_{\min} = g_0 + g(t), \\ h_{\max} = u, \end{cases}, \tag{3.8}$$

K is a subset of atoms, and g_0 and μ are equal to 1.1 and 2.4 with the best fitness value. As indicated by the following equation, g stood for the drift factor, which primarily controls the ratio of exploration to exploitation:

$$g(t) = 0.1\times \sin\left(\frac{\pi}{2}\times\frac{t}{T}\right). \tag{3.9}$$

The following is the definition of the equation for the total interaction force:

$$F_i^d = \sum_{j\in k} r_j F_{ij}^d(t), \tag{3.10}$$

r is a random number in the interval [0, 1]. In accordance with Newton's third law,the jth atom of the same binary interaction has a force that is reversed to the force on the *ith* atom:

$$F_{ij}(t) = -F_{ij}(t), \tag{3.11}$$

Moreover, the constraint force and the geometric constraint of the atom are defined as follows:

$$\theta_i = \left[|x_i(t) - x_{best}(t)|^2 - b_{i,best}^2 \right], \tag{3.12}$$

$$G_i^d = -\lambda(t)\nabla\theta_i^d(t) = -2\lambda(t)\left(x_i^d(t) - x_{best}^d(t)\right), \tag{3.13}$$

x_{best} represents the best atom's position, $b_{i,best}$ is the length of the fixed link between the atoms (*ith* & *best*) , represents the Lagrangian multiple by λ. The constraint force is defined as follows when 2λ is substituted by λ:

$$G_i^d = \lambda(t)\left(x_{best}^d(t) - x_i^d(t)\right). \tag{3.14}$$

The definition of the lagrangian multiple is as follows:

$$\lambda(t) = \beta e^{-\frac{20t}{T}}, \tag{3.15}$$

β stands for the multiplier weight. Ultimately, the acceleration of the atom is stated as follows:

$$a_i^d(t) = \frac{F_i^d(t)}{m_i^d(t)} + \frac{G_i^d(t)}{m_i^d(t)},$$
$$= -\alpha(1 - \frac{t-1}{T})^3 e^{-\frac{20t}{T}} \sum_{j \in k} \frac{r_j 2\left(h_{ij}(t)\right)^{13} - \left(h_{ij}(t)\right)^7}{m_i(t)} \cdot \frac{\left(x_j^d(t) - x_i^d(t)\right)}{\left\|x_i(t), x_j(t)\right\|_2} + \beta e^{-\frac{20t}{T}} \frac{x_{best}^d(t) - x_i^d(t)}{m_i(t)}, \tag{3.16}$$

m represents the atom's mass, computed from the following equation:

$$M_i(t) = e^{-\frac{Fit_j(t) - Fit_{best}(t)}{Fit_{worst}(t) - Fit_{best}(t)}}, \tag{3.17}$$

$$m_i(t) = \frac{M_i(t)}{\sum_{j=1}^N M_j(t)}, \tag{3.18}$$

Fit symbolizes the importance of fitness. Examining the problem of minimization, Fit_{best} & Fit_{worst}defined as:

$$Fit_{best} = \min_{i=1}^{N} Fit(t), \tag{3.19}$$

$$Fit_{worst} = \max_{i=1}^{N} Fit(t). \tag{3.20}$$

The atom's position and variety are updated the d with the following expression:

$$V_i^d(t+1) = r_1 V_i^d(t) + a_i^d(t), \tag{3.21}$$

$$X_i^d(t+1) = X_i^d(t) + V_i^d(t+1), \tag{3.22}$$

$X_i$and v_i represent the ith atom's velocity and position, acceleration by a, d represents the search space dimension, represents r_1 a random number in [0, 1], current iteration denoted by t [15].

In ASO, to control the exploration and exploitation phase the best atoms number in the subgroup K is used.

$$K(t) = N - (N-2)\sqrt{\frac{t}{T}}, \tag{3.23}$$

represents N, the population's number of atoms. Initially, increasing K improves exploration. At the conclusion of the cycle, we see that a lower ensures higher exploitation. The ASO pseudo code is displayed in Algorithm 1 [6].

3.1. Binary Atom Search Algorithm

Atoms move throughout the search space using the actual continuous field in the continuous form of the atomic search optimization. Atoms only work with two numbers in binary optimization, either (0) or (1) [6, 9]. Accordingly, we can find a technique to reverse the direction or change it from (0) or (1) utilizing the atom's velocity. The transfer function was effectively applied in binary versions of the bat algorithm, tree growth algorithm, gravitational search algorithm, and antlion optimizer in some of the first studies to improve the particle swarm. As established in past works, one of the most helpful

Algorithm 1 The ASO algorithm's pseudo-code

Input: N, T, α, and β
Start
  Initialize the atoms' location and velocity at random
  Calculate the fitness of each atom
  Identify x_{best} as the best atom
while stopping criterion t is not met **do**
  for $i = 1$ to number of atoms n **do**
    Determine the atom's mass using equations (3.17) and (3.18)
    Identify subset of atoms K as in equation (3.19)
    Use equation (3.2) to calculate the interaction force f
    Determine the constraint force g using equation (3.14)
    Calculate acceleration using equation (3.16)
    Update the atoms' velocity using equation (3.17)
    Update the location of the atoms using equation (3.18)
    if current atom is better than x_{best} **then**
      Update x_{best}
    end if
  end for
end while
Output: Global best atom

tools is the transfer function, which translates the continuous optimization method to the binary version [16, 17].

The transfer functions of the atoms are provided across the binary search space. ASO includes a transport feature for a binary version. The search agent's position can be updated using Eq. (3.25) [18], or the probability can be converted into a binary representation.

$$X_i^d(t+1) = \begin{cases} 1 & \text{if} \quad rand \prec T(V_i^d(t+1)) \\ 0 & \text{if} \quad rand \geq T(V_i^d(t+1)) \end{cases} \tag{3.24}$$

$$X_i^d(t+1) = \begin{cases} 1 - X_i^d(t) & \text{if} \quad rand \prec T(V_i^d(t+1)) \\ X_i^d(t) & \text{if} \quad rand \geq T(V_i^d(t+1)) \end{cases} \tag{3.25}$$

$X_i^d(t)$ and $V_i^d(t)$ represents speed & position. If the random vector is large, we need a maximum value of $T(V)$ to change the position of the atom. The atom can hold its position even when $T(V)$ is low when the rand is low. In addition, when the value of the rand depreciates, the position of the atom changes [19]. From the above, it can be seen that additional atomic positions are generated by a completely arbitrary mechanism, resulting in a subisotropic display. The following is the S-like family:

$$X_i^d(t+1) = \begin{cases} 0 & \text{if} \quad T(V_i^d(t+1)) \preceq k_1 \\ 1 & \text{if} \quad k_1 \prec T(V_i^d(t+1) \preceq k_2 \\ X_{best}^d & \text{otherwise} \end{cases} \tag{3.26}$$

V-shape family

$$X_i^d(t+1) = \begin{cases} X_{best}^d(t) & \text{if} \quad T(V_i^d(t+1)) \leq k_1 \\ X_i^d(t) & \text{if} \quad k_1 < T(V_i^d(t+1)) \leq k_2 \\ 1 - X_i^d(t) & \text{otherwise} \end{cases} \tag{3.27}$$

Both k_1 and k_2 can be calculated through Eqs. (3.28) and (3.29) [20]:

$$k_1 = \frac{1}{3} rand, \tag{3.28}$$

$$k_2 = \frac{1}{3} + \frac{1}{3} rand, \tag{3.29}$$

rand is a random in $[0, 1]$, X_i^d and V_i^d denote the position and velocity, $X_{best}$is the best atom position, and t is current iteration. From Eqs. (3.24), and (3.25) [21]. We see that atomic positions can vary randomly according to the rand scale. The example is applied using Eq. (3.25). When the rand is low, the atom can maintain its location even if $T(V)$ is low. Additionally, the atom's position shifts when the rand's value declines. From the foregoing, it can be observed that atom's position is changed using a fully random technique, which results in mediocre performance.

4. The Proposed Approach

The proposed hybrid algorithm combines binary atom search algorithm (BASA) with Fuzzy c-means (FCM) clustering to improve clustering performance through optimized feature selection BASA first generates a binary population, each feature representing a subset. For each subgroup, FCM clustering is used, and cluster quality is evaluated using metrics such as the Silhouette Score or the Davies-Bouldin Index. BASA then repeatedly modifies the feature subsets to increase the clustering quality, and selects the most appropriate features. Once BASA converges to the optimal subset, the final FCM clustering is performed with reduced dimensionality, resulting in well-defined, accurate clusters with improved computational efficiency. The following steps illustrate the working of the proposed method.

- **Step 1:** Start with a number of atoms, where each atom represents a possible solution (a binary vector of feature selection options). Each binary vector indicates either a specific choice (1) or no choice (0).
- **Step 2:** For each atom in the BASA population, determine the subset of features indicated by its binary vector. Use FCM clustering on a data set with this subset of attributes, grouping the data into C clusters.
- **Step 3:** Calculate the fitness of each atom (solution) based on the clustering quality metric. Using the Silhouette Score, this fitness function helps BASA estimate which subsets of features result in well-defined clusters.
- **Step 4:** BASA updates the binary vector of each atom in a way that simulates atomic motions and interactions, searching the feature space by selecting or excluding features. This iteration attempts to optimize the fitness function so that the best subsets are selected.
- **Step 5:** Once BASA has converged, the best atoms (solutions) in the population represent the best subsets. Use this best subset for the final FCM clustering.

- **Step 6:** Use FCM clustering with optimal feature subsets and evaluate clustering performance using metrics such as Silhouette Score, cluster cohesion, or separation to assess the effectiveness of the feature selection.

5. Results and Discussion

The UCI Machine Learning Repository provides five real-world data sets that are used in the experiments. Table 1 lists the various sizes, dimensions, and cluster counts of these data sets.

Table 1. Description of the datasets

Data	N	Dim
Data1 (zoo)	101	17
Data2 (biodeg)	1055	41
Data3 (Glass)	214	9
Data4 (Wine)	178	13
Data5 (Hill-valley)	606	100

The effectiveness of the developed algorithms is assessed using the Silhouette score. In order to assess how tight a cluster is, the silhouette clustering validation technique maximizes the distance between clusters and minimizes the distance between points within clusters. The quality of clustering results can be assessed by calculating the mean distance of a sample to each other point in the same cluster and subtracting the mean distance of the sample to each other point in the nearest cluster.The sample is well matched to its own cluster if the silhouette score is high; if it is low or negative, there may be an excessive number of For the point i, the silhouette width $s(i)$ is defined:

$$s(i) = \frac{b(i) - a(i)}{\max[b(i), a(i)]}, \tag{5.1}$$

This is defined as follows: $a(i)$ is the average distance between(i) and every other data point in the same cluster c_i, and $b(i)$ is the average distance between (i) and every other data point in the opposite cluster c_i.

$$b(i) = \min_{I \neq J} \frac{1}{|c_J|} \sum_{j \in J} d(i, j), \tag{5.2}$$

where $d(i, j)$ is the distance between (i) and (j).

Table 2 demonstrates that BAS-FCM outperforms FCM in terms of clustering accuracy for all datasets, as indicated by the Silhoutee value. This indicates that BAS-FCM is more effective for feature selection clustering tasks compared to FCM. Additionally, the algorithm is suitable for high-dimensional data sets.

Table 3 shows that the BAS-FCM algorithm yields the best feature selection across all datasets. This implies that BAS-FCM is able to create clusters that are near the expected clusters on average. Additionally, BAS-FCM outshines FCM in terms of computational complexity.

Table 2. The Silhouette value results for BAS-FCM and FCM clustering accuracy

Datasets	BAS-FCM	FCM
Data1 (zoo)	0.8809	0.5960
Data2 (biodeg)	0.9292	0.6413
Data3 (Glass)	0.8283	0.4352
Data4 (Wine)	0.7566	0.5773
Data5 (Hill-valley)	0.8914	0.8902

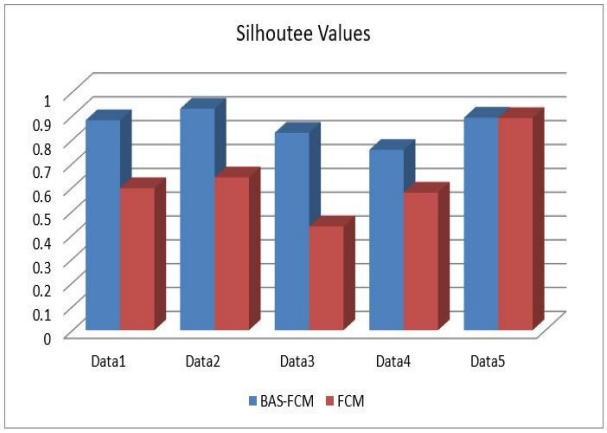

Figure 1. Comparison between BAS-FCM and FCM in Silhouette value results

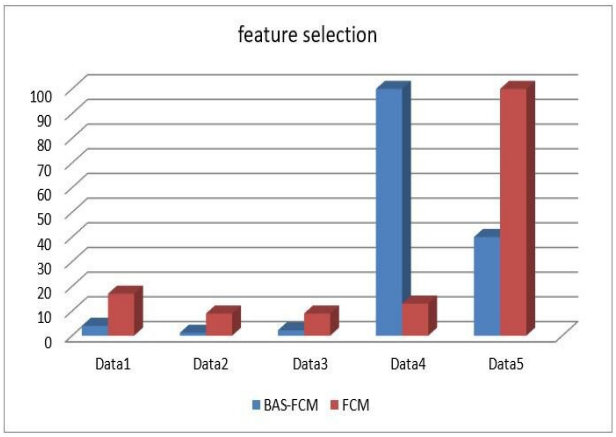

Figure 2. Comparison between BAS-FCM and FCM in average feature selection

6. Conclusion

This study presents a proposed method for improving the performance of unsupervised data clustering by combining the Binary Atomic Search Algorithm (BASA) and the Fuzzy C-means (FCM). Through a comprehensive study on a set of five different datasets shown in Tables (2,3) and Figures (1-2), the results of the proposed method, BASA-FCM, showed superior performance over the default Fuzzy C-means, through performance indicators, such as silhouette coefficient and feature selection efficiency.

Table 3. Shows how the FCM and BAS-FCM algorithms compare in terms of feature selection.

Datasets	BAS-FCM	FCM
Data1 (zoo)	4	17
Data2 (biodeg)	1.2	9
Data3 (Glass)	2.2	9
Data4 (Wine)	2.4	13
Data5 (Hill-valley)	40	100

Data Availability: Data are available upon request from the corresponding author.
Author contributions: All authors approved the manuscript and take full responsibility for its content.
Conflict of Interest Statement: The authors declare no conflicts of interest related to this work.

References

1. Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. (2017). Feature selection: A data perspective. ACM Computing Surveys (CSUR), 50(6), 1–45.
2. Kahya, M. A., Altamir, S. A., & Algamal, Z. Y. (2020). Improving whale optimization algorithm for feature selection with a time-varying transfer function. Numerical Algebra, Control and Optimization, 11(1), 87–98
3. El-Sherpieny, E. A., Almetwally, E. M., Muse, A. H., & Hussam, E. (2023). Data analysis for COVID-19 deaths using a novel statistical model: Simulation and fuzzy application. PLOS ONE, 18(4), e0283618.
4. Ivanova, M., Petkova, P., & Petkov, N. (2021). Machine learning and fuzzy logic in electronics: Applying intelligence in practice. Electronics, 10(22), 2878.
5. Riad, F. H., Alruwaili, B., Almetwally, E. M., & Hussam, E. (2022). Fuzzy Reliability Analysis of the COVID-19 Mortality Rate Using a New Modified Kies Kumaraswamy Model. Journal of Mathematics, 2022(1), 3427521.
6. Hussein, T. F., Hasan, F. M., Saleem, H. D., & Qasim, O. S. (2025). Feature subset selection based on differential evolution algorithm and K-mean clustering. Mathematical Modelling of Engineering Problems, 12(2), 494–500.
7. Morariu, C., Morariu, O., Răileanu, S., & Borangiu, T. (2020). Machine learning for predictive scheduling and resource allocation in large scale manufacturing systems. Computers in Industry, 120, 103244.
8. Al-Kababchee, S. G. M., Algamal, Z. Y., & Qasim, O. S. (2023). Enhancement of K-means clustering in big data based on equilibrium optimizer algorithm. Journal of Intelligent Systems, 32(1), 20220230.
9. Thamer, M. K., Algamal, Z. Y., & Zine, R. (2023). Enhancement of kernel clustering based on pigeon optimization algorithm. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 31(S1), 121–133.

10. Saeys, Y., Inza, I., & Larranaga, P. (2007). A review of feature selection techniques in bioinformatics. Bioinformatics, 23(19), 2507–2517.
11. Rahman, M. M., Usman, O. L., Muniyandi, R. C., Sahran, S., Mohamed, S., & Razak, R. A. (2020). A review of machine learning methods of feature selection and classification for autism spectrum disorder. Brain Sciences, 10(12), 949.
12. Hussain, I., Sinaga, K. P., & Yang, M. S. (2023). Unsupervised multiview fuzzy c-means clustering algorithm. Electronics, 12(21), 4467.
13. Sharma, A., Sharma, A., Averbukh, M., Jately, V., Rajput, S., Azzopardi, B., & Lim, W. H. (2023). Performance investigation of state-of-the-art metaheuristic techniques for parameter extraction of solar cells/module. Scientific Reports, 13(1), 11134.
14. Müser, M. H., Sukhomlinov, S. V., & Pastewka, L. (2023). Interatomic potentials: Achievements and challenges. Advances in Physics: X, 8(1), 2093129.
15. Alturki, F. A., Omotoso, H. O., Al-Shamma'a, A. A., Farh, H. M., & Alsharabi, K. (2020). Novel manta rays foraging optimization algorithm based optimal control for grid-connected PV energy system. IEEE Access, 8, 187276–187290.
16. Hasan, M. K., Budhati, A. K., Solaiman, R., Islam, S., Pandey, B., Abbas, H. S., & Saeed, M. M. (2024). An improved binary spider wasp optimization algorithm for intrusion detection for industrial Internet of Things. IEEE Open Journal of the Communications Society,6, 2926–2944.
17. Sarkar, D., Bali, R., & Ghosh, T. (2018). Hands-on transfer learning with Python: Implement advanced deep learning and neural network models using TensorFlow and Keras. Packt Publishing Ltd.
18. Al-Kababchee, S. G. M., Qasim, O. S., & Algamal, Z. Y. (2021, May). Improving penalized regression-based clustering model in big data. In Journal of Physics: Conference Series (Vol. 1897, No. 1, p. 012036). IOP Publishing.
19. Bera, S., Saha, A., Mondal, S., Biswas, A., Mallick, S., Chatterjee, R., & Roy, S. (2022). Review of defect engineering in perovskites for photovoltaic application. Materials Advances, 3(13), 5234-5247.
20. Tayduganov, A., Kou, E., & Le Yaouanc, A. (2012). Strong decays of K1 resonances. Physical Review D, 85(7), 074011.
21. Hasan, F. M., Hussein, T. F., Saleem, H. D., & Qasim, O. S. (2024). Enhanced unsupervised feature selection method using crow search algorithm and Calinski-Harabasz. International Journal of Computational Methods and Experimental Measurements, 12(2), 185–190.