Journal of Universal Computer Science, vol. 30, no. 11 (2024), 1529-1568
submitted: 22/9/2023, accepted: 13/3/2024, appeared: 28/10/2024 CC BY-ND 4.0

Multi-Search Strategy-based Improved Water Flow Optimizer Algorithm for Cluster Analysis

Prateek Thakral
(Jaypee University of Information Technology, Solan, Himachal Pradesh, India
https://orcid.org/0000-0002-5090-3983, 18.prateek@gmail.com)

Yugal Kumar
(School of Technology Management and Engineering, NMIMS, Chandigarh, India
Jaypee University of Information Technology, Solan, Himachal Pradesh, India
https://orcid.org/0000-0003-3451-4897 , yugalkumar.14@gmail.com)

Abstract: Clustering is a popular technique that has proven its capability in diverse fields like data analytics, business intelligence, social mining, image recognition, document clustering and bioinformatics. This technique determines the valuable information from a pre-defined set of data and groups similar information into the same cluster. In literature, many algorithms have been presented based on several clustering approaches. It is observed that partitional clustering algorithms are widely popular due to there simplicity and easy implementation such as k-means, k-medoids, and k-harmonic means. However, traditional algorithms suffer from several limitations like being trapped in local optima and depending on the initial solution quality. Heuristic algorithms are also proposed to alleviate the problems of traditional clustering algorithms. But, sometimes, these algorithms also get stuck in local minima and exhibit a lack of balance between search mechanisms. Hence, this work presents an improved water flow optimizer (IWFO) algorithm for cluster analysis that can address the issues of traditional and heuristic algorithms. In the proposed IWFO algorithm, the initial solution is generated based on the logistic chaotic map instead of random initialization, and in turn, an optimal quality solution is generated. The search mechanism of the WFO algorithm is enhanced based on an improved search mechanism which is the combination of non-linear functions and the previous best solution. Further, the local optima issue is alleviated using a multi-start search mechanism. The efficacy of the proposed IWFO algorithm is evaluated using twelve benchmark clustering datasets and results are compared with seventeen clustering algorithms. The simulation results are assessed using intra-cluster distance (intra), standard deviation (SD), rank, accuracy rate (AR) and detection rate (DR) parameters. Further, a statistical test is also conducted to validate the efficacy of the proposed IWFO algorithm. The results showed that proposed IWFO algorithms obtain superior quality results on most of the datasets.

Keywords: Cluster Analysis, Meta-Heuristic, Evolutionary Algorithm, Water Flow Optimizer
Categories: H.2.8, H.3.3, G.1.6, I.2.8
DOI:10.3897/jucs.113087

1 Introduction

Clustering is a well-known data analysis method that can be used to arrange the data into different clusters. Similar data are placed in the same cluster, while dissimilar data are put in another cluster. The dissimilarity between data is calculated using the distance function [Reddy, 18]. In literature, clustering can be used in a variety of fields including

text mining, social network analysis, data exploration, medical science, finance, and multimedia data [Gan, 20] [Zhao, 05]. Additionally, the subcategories of clustering are (i) hard clustering and (ii) soft clustering. In hard clustering, the data can be assigned to a single cluster, while, in soft clustering, data can belong to several clusters depending on probability function value [Aggarwal, 18]. The main issue associated with clustering is the quality of partition. Based on the optimal partition (i.e. cluster centroids), a dataset is divided into numerous clusters, and the clustering techniques are utilized to calculate optimal centroids for obtaining optimal partitioning (clusters). These partitions are validated based on internal, external, and relative cluster validation measures [Brock, 08]. These validation methods have defined the quality of the clustering algorithm. The internal validation is based on cluster creation, such as compactness, separation, and connectivity. The closeness between the data within a cluster can be used to define compactness. If a cluster displays the bare minimum of compactness, it is sufficient. The distance between two or more clusters is used to calculate separation, and it can be on the extreme side. The identical cluster data is used to describe the connectivity is described. External validation is described by comparing the clustering result against the class labels that are mentioned in the dataset. It includes various performance indicators like purity, rand index, entropy, etc. The relative validation evaluates the clustering structure through various user-defined parameter values that are provided by the algorithms [Halkidi, 01].

In literature, different clustering algorithms based on diverse methodologies have been presented in the literature for addressing clustering problems and also considered the various viewpoints for solving these problems [Kaur, 22] [Sahoo, 14] [Kumar, 16] [Saxena, 17] [Ozbakir, 17]. Further, the clustering techniques are divided into five categories such as partitional, hierarchical, grid, density, and model-based [Patel, 16]. Additionally, each technique has several advantages and disadvantages. Recently, the research community has focused on grouping uncertain and high-dimensional data [Xu, 11] [Jiang, 11]. Despite this, current developments in clustering can be described as fuzzy, evolutionary, meta-heuristic, and multimedia clustering [Mukhopadhyay, 15] [Sevillano, 14]. A few clustering studies also presented novel distance functions to improve the results of clustering. Some studies also focus on validation metrics to evaluate the effectiveness of clustering algorithms[Saxena, 17]. However, no one approach can more effectively handle the clustering problem with a wide range of datasets. Every algorithm has several advantages and disadvantages.According to a thorough literature review, it is identified that partitional clustering can be used more frequently than other clustering techniques, like the hierarchical, grid, and model-based clustering techniques due to being less time-consuming [Ezugwu, 22]. Partitional clustering determines a distinct set of clusters and allocates the data to a particular cluster based on distance measures. Additionally, prior knowledge of clusters (K) is the prerequisite for this clustering. K-Means is one of the popular partitional clustering algorithms[MacQueen, 67], and it is important to anticipate the number of clusters.In turn, the clustering process can become more extensive and even more difficult to compute the optimal partitioning for the given dataset. Further, it is noticed that clustering results do not converge on the global optima[Jain, 99]. Therefore, to solve the aforementioned problem, either cluster information should be provided beforehand, or the number of clusters should be calculated automatically to achieve optimal partitioning. Several meta-heuristic algorithms have been reported for handling

partitional clustering in the literature. Few of these algorithms are summarized as PSO[Cura, 12] [Jordehi, 15], MOA [Kushwaha, 18], CSSA [Kumar, 14], BH [Hatamlou, 13], BA [Kaur, 22], ABC [Karaboga, 07], ACOA [Dorigo, 06], BB-BC [Erol, 06], BBO[Ergezer, 09], etc. These algorithms have balanced search capabilities and produce prominent clustering results. However, these algorithms have some other drawbacks such as population diversity, trade-offs, convergence rate, and sometimes being trapped in local optima [Osmani, 22]. The aforementioned weaknesses of meta-heuristic algorithms can be got rid of with the assistance of additional meta-heuristic algorithms. To achieve superior clustering results, the weak point of one meta-heuristic algorithm is substituted by the strong point of another meta-heuristic algorithm. For example, the slow convergence rate of the PSO algorithm is handled by hybridizing the PSO with the k-harmonic mean [Bouyer, 16]. Similarly, chaotic maps are incorporated into PSO to accelerate convergence speed [Chuang, 11]. To increase the performance of k-means and limit the effect of initial centroids on final clustering results, the k-mean and PSO algorithms are combined [Liu, 18]. ABC trade-off problem is handled via a self-adaptive system [Du, 19]. The BB-BC local optima problem is solved by integrating chaotic maps [Singh, 19]. Based on approximation functions, these methods also provide near-to-optimal solutions for clustering issues. However, the No Free Lunch theorem states that no single clustering approach can be used to solve all clustering problems as well as applicable to all datasets. Hence, there is a scope to develop a new clustering algorithm that can generate a more optimal solution for clustering problems and is also applicable to a wide range of datasets. As a result, a new algorithm for obtaining optimal solutions and solving large-scale clustering problems can be devised. Recently, a new meta-heuristic algorithm, named water flow optimizer (WFO) is presented to handle the variety of constrained and unconstrained optimization problems [Luo, 21], [Thakral, 24], [Macedo, 22]. The hydraulic processes of water particles inspired this method, which describes the flow of water from highland to lowland. Further, the laminar and turbulent flows are taken into account to devise the stochastic search operators for the optimization process. Froman optimization viewpoint, the water flow can be classified into two types such as either to maximize or minimize an objective function that can be designed to solve the problems. Second, an iterative process can be used to find the best solution and convergence behaviour of an algorithm. Hence, this work aims to examine the efficacy of the WFO algorithm for solving clustering problems. However, before implementing the WFO in the clustering field, several modifications are integrated into the WFO algorithm to make it more effective and generate optimal clustering results.

1.1 Motivation and Objectives of the Work

The purpose of this research work is to investigate the effectiveness of the WFO algorithm in solving clustering problems. Clustering is an optimization problem with constraints that computes the grouping of similar objects. These groups are identified based on the optimal centroids. Hence, the goal of the WFO algorithm is to find optimal centroids for grouping data objects into clusters, and these clusters are compact. However, before implementing the WFO, numerous modifications are made to improve its performance. However, WFO has better optimization capability, but this algorithm also suffers from several shortcomings [Said, 21][Cheng, 23]. These shortcomings are

expressed as - (i) a random distribution function is utilized to seed the population of WFO in a pseudo-chaotic manner, hence, the process of finding the optimal solution is sometimes blind. (ii)Further, the laminar flow (p_l) operator and turbulent flow (pe) operator consist of constant values, which cannot adequately balance the exploration and exploitation ability of the algorithm. (iii)It is also observed that the laminar flow phase includes a parallel one-way search strategy that may result in search holes, and in turn, WFO may fall into local optima. The aforementioned shortcomings of the WFO are handled through the logistic chaotic map, an improved balance mechanism and a multi-strategy search method. The main objectives of the work are highlighted below.

- The random distribution function is utilized to generate the initial population of the WFO algorithm in search space. This distribution function cannot generate a uniform population. Hence, a logistic chaotic map is utilized to generate the population in a more systematic manner instead of random.
- The exploration and exploitation ability of the WFO algorithm cannot be well balanced due to constant values of the laminar flow (pl) operator and turbulent flow (pe) operator. The constant values of the laminar flow (pl) operator and turbulent flow (pe) operator are substituted by dynamic values that can be changed in each iteration. Hence, an improved balance mechanism is designed to improve the exploration and exploitation ability of the WFO algorithm.
- The two-way strategy is adopted in the laminar flow phase of the WFO algorithm to search the candidate solutions and in turn, the algorithm may stick in local optima. This issue is getting rid of by using a multi-strategy search process.
- The aforementioned improvements are integrated into the WFO algorithm to make it more effective and generate the optimal solutions. The proposed algorithm is implemented to solve the clustering problems and this algorithm aims to compute the optimal centroid for partitioning of the dataset.
- The efficacy of the proposed WFO algorithm is evaluated using several well-known clustering datasets downloaded from the UCI repository based on SSE, AR and DR rates. The simulation results are compared with conventional as well as meta-heuristic algorithms. The findings stated that the proposed WFO algorithm obtains superior clustering results compared to other algorithms.

Section 2 summarizes the current state of artworks on partitional clustering. Section 3 describes the fundamentalsof water flow optimizer. The proposed WFO-based clustering technique is demonstrated in Section 4. Section 5 discusses the findings of the proposed WFO algorithm. Section 6 summarizes the outcomes of this investigation.

2 Literature Review: Optimization Techniques for Clustering

The latest works on partitional clustering are discussed in this section.

A variable neighbourhood strategy-based firefly algorithm (VNS-FA) is presented for effective data clustering [Behadili, 22]. It is observed that the firefly algorithm (FA) converges on premature solutions due to a lack of exploitation capability. Further, variable neighbourhood strategy (VNS) is incorporated into FA to resolve the aforementioned issues. The efficacy of the proposed VNS-FA is evaluated using eight well-known clustering datasets. The results are assessed using intra-cluster distance,

internal CH metric, entropy and F-measure parameters. The results stated that the proposed VNS-FA method obtains superior results with most of the datasets.

A cat-based meta-heuristic algorithm is reported for addressing the partitional clustering [Singh, 22]. Before implementing this algorithm, several modifications are proposed to the cat algorithm in terms of trade off mechanism between local and global searches, diversity issues and premature convergence. In turn, an improved search mechanism, accelerated velocity equation and neighborhood mechanism are introduced for handling the aforementioned issues. The efficiency of the cat algorithm is examined over eight clustering datasets based on intra-cluster distance and f-measure parameters. It is analyzed that the proposed cat algorithm has a minimum intra-cluster distance with most of the datasets while obtaining a better f-measure rate than other algorithms.

Kushwaha et al. [Kushwaha, 22] presented an electromagnetic field optimization (EFO) method for resolving the issues of the k-mean algorithm. The k-mean algorithm suffers from poor selection of initial centroid and in turn traps in local optima. To overcome this issue of the k-mean algorithm, the EFO algorithm is utilized for generating the optimal initial centroid for the k-mean algorithm. It is also reported that the EFO algorithm may not stuck in local optima due to attraction and repulsion mechanisms. Several well-known datasets are considered for assessing the efficacy of the proposed clustering algorithm based on normalized mutual information (NMI), rand index (RI), and purity. The results showed that the proposed algorithm gets far better clustering results than the same class of algorithms.

To perform the clustering in massive data, Hashemi et al. [Hashemi, 22] designed an updated PSO method. The proposed method utilizes the multi-start pattern reduction mechanism to decrease the calculation time for clustering. The clustering time is reduced using a reduction operator while the multi-start operator is utilized for ensuring population diversity and local minima. The six clustering datasets are considered to evaluate the performance of the proposed method based on accuracy and execution time. The results confirmed that better clustering results are obtained by the proposed PSO method.

To overcome the effect of outliers in clustering, Kuo et al. [Kuo, 21] presented the FPCOM fuzzy c-means and fuzzy c-ordered means algorithm. Additionally, the parameters and initial centroids are set using the SCA, called SCA-FPCOM. The performance of the proposed SCA-FPCOM is examined over ten clustering datasets taken from the UCI repository based on the Silhouette coefficient and rand index measures. The results demonstrated that SCA-FPCOM outperforms compared to other algorithms. It is seen that the majority of clustering algorithms use single-featured datasets, distance, density, and characteristics to validate the clusters. However, the inclusion of semantic data may violate the concept of generated clusters. However, evolutionary and statistical operators with multidisciplinary methodologies can generate meaningful clusters.

The Prior information of the cluster is the main prerequisite for partitional clustering, so the research community is paying close attention to automatic partitional clustering. The aforementioned problem of partitional clustering was also taken into account by Zhu et al. [Zhu, 22] who presented the AC-DPHS dynamic parameter-based HS algorithm for automatic data clustering. In the proposed AC-DPHS algorithm, the parameter is modified dynamically instead of static. The efficiency of the proposed AC-DPHS is assessed using five datasets. The results are evaluated using ARI, FM, and

PBM parameters. It is stated that the proposed algorithm obtains superior results than other algorithms being compared.

A clustering algorithm based on enhanced gradient (IGE) is developed for handling partitional clustering problems [Kuo, 20]. The proposed algorithm comprises two objective functions instead of focusing on a single objective function. The dominating and non-dominating set of solutions is generated based on the Pareto optimality. The performance of the IGE algorithm is assessed using several clustering datasets. The simulation results are evaluated using SSE, SSB, and accuracy parameters and compared with PSO, GA, ABC, and DE algorithms. It is claimed that IGE obtains more promising clustering results than other clustering algorithms.

Duan et al. combined the gradient-based approach with the elephant herding optimization algorithm, known as GBEHO [Duan, 22], for efficient cluster analysis. Several heuristics are designed for selecting the initial centroids. Further, the trade-off issue of EHO is resolved based on the Gaussian chaos map. Moreover, wandering and variation operators are used to update the population. Several artificial and real-life datasets are considered for evaluating the efficacy of the proposed GDEHO algorithm based on accuracy, detection rate, specificity, and f-measure parameters. It is reported that GBEHO provides more consistent results in contrast to other algorithms.

A hybrid k-prototype clustering method based on enhanced SCA [Kuo, 22] is developed by Kuo and Wang. SCA algorithm is utilized to compute the optimal weight of attributes as well as initial attribute selection. Furthermore, the k-prototype method incorporates different mutation strategies, like Gaussian, Cauchy, levy, and single-point to produce better clustering outcomes. The ten datasets are chosen from the UCI repository to examine the efficacy of the k-prototype algorithm based on accuracy and Cohen kappa. The findings showed that the proposed k-prototype algorithm produces better clustering results than other algorithms.

A meta-heuristic clustering approach based on the behaviour of micro-bats [Kaur, 22] is proposed by Kaur and Kumar. Several modifications are designed to resolve the issues related to the bat algorithm like convergence rate, local optima, and trade-off issues. The elitist process handles the slow convergence rate, the initialization issue is handled by a collaborative approach. The effectiveness of the proposed bat-based metaheuristic algorithm is evaluated using several healthcare and non-healthcare datasets utilizing well-known performance metrics like intra-cluster distance, standard deviation, accuracy, and rand index. The results of the proposed approach are compared to conventional clustering algorithms and it is claimed that the proposed approach achieves a better accuracy rate than conventional algorithms.

A learning automata-based hybrid MPA and JS algorithm is presented for effectively handling the data clustering problem [Barshandeh, 22]. The MPA and learning automata are utilized for enhancing proficiency and reducing the complexity of the JS algorithm. MPA is applied to memorize the best solution achieved so far while learning automata is used to improve the learning strategy of the JS algorithm. The efficiency of the hybrid MPA-JS algorithm is evaluated using ten clustering datasets based on SSE, average execution time and CHC. The results are compared with several state of art meta-heuristic algorithms and it is found that the proposed hybrid MPA-JS algorithm gets better results than other algorithms.

Ezgi combined the LaF and DE algorithms to discover the optimal centroids for data clustering problems [Zorarpacı, 23]. The weak exploitation process of the LaF algorithm is improved using the DE/best/1 mutation operator. The performance of the

proposed LaF-DE algorithm is evaluated using twelve clustering datasets based on the SSE and accuracy parameters. The results showed that the proposed LaF-DE provides better clustering results with eight datasets out of twelve. It is also noticed that the proposed algorithm also obtains satisfactory clustering results with the rest of the datasets compared to most of the algorithms being compared.

Duan et al. handled the automatic clustering problem in high dimensional data by an improved affinity propagation based on an optimization algorithm [Duan, 23]. Initially, the dimensionality of data is reduced using the t-distributed stochastic neighbour method. Further, an improved equilibrium optimizer is utilized for optimizing the preference selection. The local search and convergence efficiency are enhanced using the crisscross strategy. Finally, the performance of the above-mentioned combination is assessed using seven high-dimensional datasets and results are compared with well-known four clustering algorithms based on NMI and RI. It is stated that the performance of the improved affinity propagation is significantly improved using the above-mentioned improvements.

Singh et al. presented an enhanced whale optimization algorithm (EWOA) [Singh, 23] for effectively handling the clustering problems. The whale optimization algorithm (WOA) suffers from local optima, convergence rate and trade-off issues. The trade-off issue of WOA is addressed through searching behaviour water wave optimization algorithms. The local optima and convergence issues are resolved through the neighbourhood mechanism and tabu search algorithm. The eight benchmark datasets are considered to examine the performance of the EWOA based on average intra-cluster distance and f-measure. The results showed that superior clustering results are obtained by the EWOA with most of the datasets.

To achieve better computational time, Suryanarayana et al. [Suryanarayana, 22] developed a dynamic k-mode clustering algorithm for effective cluster analysis. The PSO algorithm is adopted to compute the optimal centroid for the k-Mode algorithm. Further, a frequency-based technique is also utilized for updating the modes and decreasing the cost function for clustering tasks. The efficacy of the dynamic k-mode algorithm is evaluated using six well-known clustering datasets using accuracy, f-measure and NMI parameters. The results demonstrated that the proposed dynamic k-mode algorithm obtains more accurate clustering results than others.

3 Water Flow Optimizer

Recently, a new meta-heuristic algorithm, called WFO based on the water flow theory has been developed for solving global optimization problems [Luo, 21]. This algorithm is inspired by the shapes of water flow which is described through laminar and turbulent flows. Hence, the WFO algorithm also comprises two different evolutionary operators (laminar and turbulent) as stated above. In turn, the WFO algorithm imitates the hydraulic phenomenon of water particles i.e. flowing from highland to lowland by using two operators. The optimization process of the WFO is designed by using the laminar and turbulent operators. The optimization problems are described as either minimization or maximization problems. So, an objective function can be defined either in terms of minimization or maximization and this objective function is being solved by the optimization steps of the WFO algorithm. As WFO is inspired by the flow of water from highland to lowland, hence there exists a similar pattern among water flow

and searching of the solutions in optimization problems. So in WFO, a water particle can act as a possible solution, the position of the water particle corresponds to the solution value, and the objective function is described in terms of the potential energy of the water particle. The description of the laminar and turbulent operators is mentioned below.

3.1 Laminar Operator

The working of this operator is inspired by a laminar flow of water and this flow specified that water particles should be moved in parallel straight lines as mentioned in Figure 1. Further, the particle velocities can differ depending on the viscosity of water. The particles that are far from obstacles or walls, can move faster than those closer to obstacles and walls. Mathematically, this behaviour of water flow is demonstrated using equation 1, called the laminar operator.

$$y_t(i) = x_t(i) + R \times \vec{V} \quad \forall t = \{1,2,3,4, \ldots . m\} \tag{1}$$

In equation 1, $y_t(i)$described as the moving position of t^{th}water particle after i^{th} iteration, $x_t(i)$ corresponds to the position of t^{th}water particle in i^{th} iteration, Ris a random number in the range of$[0, 1]$, called water coefficient and $\vec{V}$ defines a vector value corresponding to a common motional direction (dimension) and m is the total number of water particles. The direction of common motional using equation 2.

$$\vec{V} = x_e(i) - x_o(i) \text{ such that } \left(e \neq f, \left(f(x_e(i)) \leq f(x_o(i))\right)\right) \tag{2}$$

In equation 2, $x_e(i) \leq x_o(i)$describes the potential energy of water particles that can be used to select the best water particle in the i^{th} iteration. If e^{th} particle energy is lower than o^{th} particle energy, then the best particle is$x_e(i)$, otherwise$x_o(i)$.

3.2 Turbulent Operator

The working of this operator is inspired by the turbulent flow of water and this can be described as a contingent pushing of water particles. In turn, an inconsistent motion can disrupt the bonding of water particles, and produce the fast flow of water to destabilize the obstruction and cause a local oscillation. In turn, an amplitude is generated for oscillation, and the amplitude can grow with time. A shearing force is produced due to the aforementioned process, known as torque, which is responsible for swirling water particles.If the dimension of the problem to be solved can be described through a layer of water, then the dimension transformation of problems can be viewed as an irregular motion of water particles in turbulent flow. Mathematically, it can be understood as randomly selecting the dimension and position of particles during oscillation. Finally, the behaviour of the turbulent operator is described through equation 3.

$$y_t(i) = \left(x_t^d(i),\ p,\ x_t^{d+1}(i), p,\ x_t^{d+2}(i),\ p,\ x_t^{d+3}(i), \ldots \ldots p,\ x_t^{d+n}(i)\right) \text{ where } d = \{1,2,3, \ldots . n\} \tag{3}$$

In equation 3, 'p' corresponds to the jostling operator, and it can be described through equation 4.

$$p = \begin{cases} \gamma\left(x_t^d(i),\ x_j^d(i)\right), \text{if } r < pe \\ \vartheta\left(x_t^d(i),\ x_j^{d+1}(i)\right), \quad \text{otherwise} \end{cases} \tag{4}$$

In equation 4, j describes the index of a randomly chosen particle (e) such as $j \in \{1, 2, 3, \ldots . s\}andi \neq j$.'d+1' defines a dimension that can be chosen randomly such that$d + 1 \neq d$. P_{ed}can be defined as an eddying parameter and r is a random number in the range of [0, 1]. Further, it is observed that the eddy shape is similar to the Archimedean spiral and it can be expressed using the equation 5.

$$\gamma\left(x_t^d(i),\ x_j^d(i)\right) = x_t^d(i) + \varphi \times \theta \times \cos(\theta) \tag{5}$$

In equation 5, Θ is a random value in the range of $[-\pi, \pi]$ and φcorresponds to the shearing force between t^{th} and e^{th} particles. The shearing force (φ) between t^{th} and e^{th} particles is determined using equation 6.

$$\varphi = \left|x_t^d(i),\ x_j^d(i)\right| \tag{6}$$

Further, the general behavior of water particles is described through a transformation function and this function is summarized in equation 7.

$$\vartheta\left(x_t^d(i),\ x_j^{d+1}(i)\right) = \left(ub^d - lb^d\right)\frac{x_j^{d+1}(i) - lb^{d+1}}{ub^{d+1} - lb^{d+1}} \tag{7}$$

In equation 7, ub^d denotes the upper bound in d^{th} dimension, lb^d denotes the lower bound of the d^{th} dimension. $x_j^{d+1}(i)$ denotes the j^{th} index of e^{th} particle in dimension (d+1).

3.3 Evolutionary Rule

This subsection discusses that the water flow can be either laminar or turbulent. The distinction between laminar and turbulent flows can be made with the help of the Reynolds number. It is stated that a threshold function is defined to determine the flow of water. If the flow of water is less than the threshold, then it can be considered as laminar, otherwise turbulent. Further, laminar flow probability can be denoted through(P_{la}), whereas, turbulent flow probability is defined using$\left(1 - (P_{la})\right)$. Moreover, laminar probability can be described as a control parameter and it is a random number in the range of $[0, 1]$. The flow of water is determined using

P_{la}.However, it is also discussed that water can be flowed from highland to low land, in turn, the position of water particles may change. Hence, this behaviour of water can be characterized using the evolutionary rule. As par this rule, the moving position of the water particles can be changed according totheir potential energy, if it is less than the threshold, a new moving position is calculated for the water particles otherwise, there is no change in the current position. This behaviour is illustrated using the equation 8.

$$X_t(i+1) = \begin{cases} y_t(i), & \text{if } f(y_t(i)) \leq f(x_t(i)) \\ x_t(i), & \text{otherwise} \end{cases} \quad (8)$$

4 Proposed Improved Water Flow Optimizer (IWFO)

This section presents the improved water flow optimizer algorithm for solving the hard clustering problems. It is observed that several limitations are associated with the water flow algorithm such as (i) random distribution function is utilized to generate the initial population, (ii) adequately less balance between exploration and exploitation mechanisms, and (iii) due to one-way strategy adopted in laminar flow phase, WFO may be stuck in local optima. Firstly, these aforementioned limitations of the WFO algorithm are handled through a logistic chaotic map-based function for generating the initial population, the balance between exploration and exploitation mechanisms is enhanced using improved solution search equations, and local optima issuesare handled through the multi-strategy search process. Subsection 4.1 discusses the aforementioned improvements for the WFO algorithm in detail.

4.1 Novelty of the Work

This subsection presents the novelty of the work. The novelty of the work is described as (i) the chaotic map-based distribution function is designed to generate the initial population for the WFO algorithm, (ii) the nonlinear sigmoid and tanh functions are considered for better tradeoff among exploration and exploitation mechanisms, and (iii) a neighbourhood search mechanism is developed to overcome the local optima issue. The proposed improvements are discussed in subsections 4.1.1-4.1.3.

4.1.1 Chaotic Map-Based Distribution Function

It is stated that the initial population for the WFO algorithm is computed through a random function from the search space. In the literature, it is mentioned that the random function cannot explore the entire search space uniformly. In WFO, equation 9 is utilized for computing the initial population form the search space.

$$x = lb + rand(N, D) \cdot [ub - lb] \quad (9)$$

In equation 9, x describes the initial position of water particles, lb denotes the lower bound of the search space, ub denotes the upper bound of the search space. N denotes the number of water particles and D represents the dimension of the search space. Let us assume that the number of water particles (N) is three, dimension (D) is 2 and suppose the lower bound of the search space is defined as (1, 9) and the upper bound

for the search space is defined as (40, 60). The $rand(\cdot)$the function generates the three numbers in the range of [0,1] in random order and the position of the water particles are described using (2.3928, 43.6155), (4.1160, 47.6447), and (5.4257, 46.8998). It is observed that the population cannot be determined uniformly throughout the search space. In literature, it is mentioned that chaotic maps can explore the search space more systematically than random functions. Hence, in this work, a logistic chaotic map is considered to generate the initial position of water particles more effectively instead of a random number. Finally, the initial position of water particles is computed using equation 10.

$$x = lb + c_k(N, D) \cdot [ub - lb] \tag{10}$$

In equation 10, x describes the initial position of water particles, lb denotes the lower bound of the search space, ub denotes the upper bound of the search space. N denotes the number of water particles and D represents the dimension of the search space, c_k describes the k[th] logistic chaotic map such that $k == N$. The equation 11 is used to compute the logistic chaotic map.

$$c_{k+1} = a \times c_k(1 - c_k), \qquad a = 4,\ c_0 = 0.6 \tag{11}$$

In equation 11, a is a user-defined parameter whose value is set to 4, k denotes the index number, c_k denotes the kth logistic chaotic map, and c_{k+1} denotes the (k+1)[th] logistic chaotic map, and the value of c_0 is set to 0.6.

4.1.2 Balance between Exploration and Exploitation Mechanisms

The exploration and exploitation mechanisms are the essential part of the meta-heuristic algorithms. Exploration corresponds to the global search, while exploitation corresponds to the local search. To obtain effective solutions, both search mechanisms could be balanced. If local search is weak, then the search space cannot be fully explored. Other side, the global search is responsible for exploiting the solution that can be found during the local search. This search also determines the capability of a local solution such that it can act as either a global solution or not. Moreover, the global search is also responsible for guiding the search towards global optima. Hence, this search mechanism also considers the direction of the previous best solution. This work considers the nonlinear sigmoid and tanh functions to make a better tradeoff between the exploration and exploitation mechanisms. The laminar operator (pl) and turbulent operator (pe) of the WFO algorithm are described as mentioned below.

$$pl = (c_1 - c_2) \times \left(\frac{1}{1 + e^{-t_{curr}/t_{max}}}\right) \tag{12}$$

$$pe = c_2 \times \left(\frac{2}{1 + e^{-2t_{curr}/t_{max}}} - 1\right) + c_1 \tag{13}$$

In equations 12-13, c_1 and c_2 describe as two cognitive parameters and the values of these parameters are 0.5 and 0.3. t_{curr} denotes the current iteration and t_{max} denotes

the maximum iteration. If, $pl > rand$, the position of the water particle (y) is computed using equation 14.

$$y_t(i) = \begin{cases} x_t(i) + r \times \vec{d}, & \text{if } rand < pl \\ x_t(i) + dt.(x_{best} - x_t(i)), & \text{otherwise} \end{cases} \quad (14)$$

In equation 14, $x_t(i)$ corresponds to the current position of the water particle in ith iteration, x_{best} denotes the best position of the water particle, r is the random number in between [0, 1], $\vec{d}$ denotes the direction of water particles which is computed using 15.dt is a decrement operator inspired by the whale optimization algorithm. It can be used to explore the global search solution with a small step size and it is described in equation 16.

$$\vec{d} = x_t(i) - x_s(i) \; such \; that \; \left(t \neq s, f(x_t(i)) \leq f(x_s(i))\right) \quad (15)$$

In equation 15, $x_t(i)$ and $x_s(i)$ represent the t^{th} and s^{th} water particles in i^{th} iteration, $f(x_t(i))$ and $f(x_s(i))$ describe the potential energy of t^{th} and s^{th} water particles. Further, the potential energy of the particle (x_t) is computed using the equation 17

$$dt = \frac{1}{1 - cos(\pi l)} \quad (16)$$

In equation 16, l is a random number in the range of [0.6, 1]. The aforementioned procedure is integrated into the proposed WFO algorithm to improve the exploration and exploitation mechanisms, especially in laminar operator.

$$f(x_t(i)) = \sum_{i}^{n} \frac{SSE(x_t(i))}{\sum_{t=1}^{n} SSE(x_t(i))} \quad (17)$$

In equation,$f(x_t(i))$ indicates the potential energy of the water particle (x_t) in the ith iteration, and $SSE(x_t(i))$ indicates the sum of the squared error of the t^{th} water particle (cluster centroid). So, the fitness of the particle is described as the sum of the squared error of the given particle divided by the sum of the squared error of all particles.

4.1.3 Multi-Strategy Search Process

This subsection discusses the multi-strategy search process to get rid-off one way search strategy and in turn local optima issue of the WFO algorithm can effectively handle this strategy. The neighbourhood search mechanism can be successfully applied to get rid of local optima problems and improve the solution quality [Abreu, 22] [Li, 19]. This work considers the three strategies to overcome the one-way search strategy

of the laminar phase and local optima. These strategies are chaotic local search strategy, mutation strategy and crossover strategy. These strategies are explained below.

Chaotic local search strategy: In literature, it is reported that local search is widely adopted for exploring the search space for optimal solutions and also computing the optimal local solution. It is also seen that local search can converge rapidly and also trap in local optima. In this work, a chaotic local search strategy is proposed to explore the search space and determine the optimal local solution. The process of the chaotic local search strategy is mentioned below.

$$x_t'(i) = (1 - \epsilon) \times x_t(i) + \tau_c \tag{18}$$

In equation 18, $x_t'(i)$ represents the new position of the water particle, $x_t(i)$ denotes the current position of the water particle that can be responsible for local optima, τ_c is a chaotic variable which is computed using the equation 19.

$$\tau_c = lb^d + c_k(ub^d - lb^d) \times g_{best}^d \;\; where\ d \in (1,2,3,..,D) \tag{19}$$

In equation 19, lb^d and ub^d denote the lower and upper constraints in the d[th] dimension, g_{best}^d denotes the best position of water particles, c_k denotes the logistic chaotic map that can be computed using equation 11.

Mutation-Based Strategy: This strategy considers a multipoint mutation operator for generating the mutant position of water particles to alleviate the local optima issue. This strategy considers the previously global best positions of water particles in a random order such as $x_e, x_f, x_g\ where, e \neq f \neq g$. The behaviour of the mutation strategy is expressed using equation 20.

$$x_t'(i) = x_e(i) + \; c_k \times \left(x_f(i) - x_g(i)\right) \tag{20}$$

In equation 20, c_k denotes the logistic chaotic map computed using equation 11 and it is also utilized to control the chaos of mutation parameter.

Crossover-Based Strategy: This strategy is inspired by the functioning of the crossover operator. In this strategy, two water particles are chosen randomly from the list of gbest particles. The new position of the water particle is generated based on the multi-point crossover. The behaviour of this strategy is depicted using equation 21.

$$x_t'(i) = \begin{cases} x_t(i) + c_k\left(x_p - x_q\right) \;\; if, rand < \left(\mathrm{f}(\mathrm{x}_p)||\mathrm{f}(\mathrm{x}_q)\right) \\ x_t(i) + c_k \times \left(1 - \dfrac{t_{curr}}{t_{max}}\right) \qquad\qquad otherwise \end{cases} \tag{21}$$

In equation 21, $x_t'(i)$ denotes the new position of water particle, $x_t(i)$ denotes the previous position of the water particle, $x_p, and\ x_q$ are two randomly chosen water particles from the list of the gbest particles, $\mathrm{f}(\mathrm{x}_p)$ and $\mathrm{f}(\mathrm{x}_q)$describe the potential energy of p[th] and q[th] particles, c_k denotes the logistic chaotic map which is computed using equation 11, t_{curr} denotes the current iteration and t_{max} denotes the maximum number of iterations.

The aforementioned improvements are integrated into the WFO algorithm to make it more effective and also to overcome the shortcomings associated with it. Figure 1 illustrates the main components of the proposed IWFO algorithm.

4.2 Proposed IWFO Algorithm for Cluster Analysis

This subsection discusses the proposed IWFO algorithm for cluster analysis. The IWFO algorithm aims to compute the optimal partitioning for the given dataset. The optimal partitioning is achieved by using optimal centroids, so the main activity of the IWFO algorithm is to compute optimal centroids for the optimal partitioning of the given dataset. The algorithmic steps of the IWFO algorithm are listed in Algorithm 1. The implementing behaviour of the proposed IWFO algorithm is divided into six major steps such as (i) Initialization, (ii) Objective Function and Data Allocation, (iii) Laminar Flow, (iv) Turbulent Flow, (v) Evolving and updation. The description of these major steps of WFO is discussed below.

(i) Initialization: In this, user-defined parameters of the WGO algorithm need to be defined. The various parameters of the IWFO algorithm are the number of particles (number of clusters (K)), dimension (D), laminar probability (pl), eddying probability (pe), lb, ub, and maximum number of iterations. Further, the initial position of the water particles can be chosen using equation 9. It is specified that the initial position of water particles represents the initial centroids for the cluster analysis.

(ii) Objective function and Data Allocation: In this step, an objective function in terms of distance measure is defined for allocating the data to clusters. This work considers the Euclidean distance as an objective function and data is allocated to clusters based on minimum Euclidean distance. The mathematical description of the objective function is mentioned using equation 22.

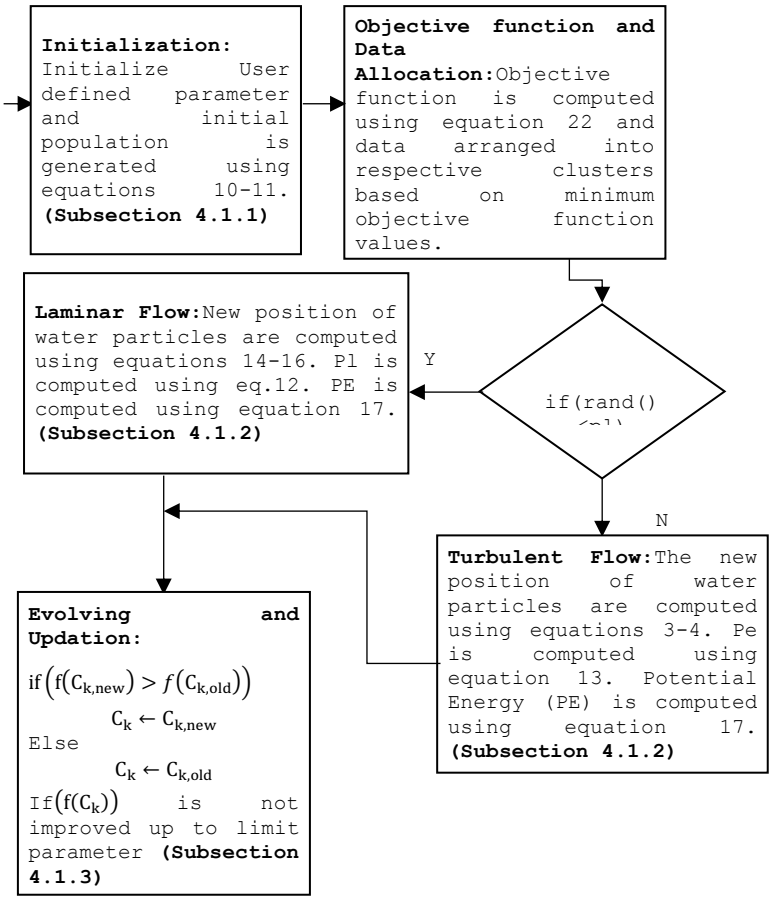

Figure 1: Illustrates the main components of the proposed IWFO algorithm

$$D(x_i, c_j) = \sqrt{\sum_{m=1}^{d} (x_{im}, c_{jm})^2} \tag{22}$$

In equation 22, $D(x_i, c_j)$ denotes the Euclidean distance among data (x_i) data and cluster centroids(c_j), d indicates the dimension of data, and it can be expressed as m= 1,2,3, …. d. The similarity among data is determined based on Euclidean distance. In turn, it is responsible for partitioning the dataset based on the distance measure.

(iii)Laminar Flow: After allocating the data to respective clusters based on the minimum value of Euclidean distance. The next step is to update the position of water particles or in other words to update the cluster centroids. So, in this direction, the

initial laminar probability(pl) is checked to a $rand()$ function. If(rand () $< P_{la}$), the flow of water can be described as laminar flow, and then the position of water particles is computed using equation 14. The potential energy of the particle is computed using equation 17.

(iv)Turbulent Flow: Otherwise, the flow is turbulent, and the position of the water particle is computed using equation 3, in turn, the position of the current particle is mutated with a generated vector p. Further,if(r $< pe$), the turbulent probability is higher than a random number (r), then, the current position of the water particle is mutated with the randomly chosen dimension of the other water particle that can be selected in random order. Finally, a new position of the water particle is generated using 4 and further, the potential energy of the newly generated particle is computed using equation 17.

(v)Evolving and Updation: This step corresponds to finding the updated position of water particles i.e. cluster centroids. The potential energy of new particles is compared with potential energy of previous particles. If $\left(f\left(x_{t,new}(i)\right) > f\left(x_{t,old}(i)\right)\right)$ is higher than the previous, then the new centroid is given as $x_t(i+1) \leftarrow x_{t,new}(i)$, otherwise $x_t(i+1) \leftarrow x_{t,old}(i)$. Further, a limit operator is also utilized to alleviate the local optima issue. The value of the limit operator is set to 5. If the potential energy of the water particle is not significantly improved in five consecutive iterations, then a multi-strategy search is invoked and the new position is computed using a multi-strategy search procedure. The termination criteria are set to the maximum number of iterations. It termination criteria are not met, then steps (ii)-(v) are repeated. Otherwise, obtain the optimal position of water particles (optimal cluster centroids).

Algorithm 1: IWFO Algorithm for cluster analysis **Input: Dataset and Cluster(K)** **Output: Optimal Centroids**
/*Initialization, Phase*/
1. Upload the dataset and set the user-defined parameters such as the number of clusters$\langle k_i \in (i = 1, 2, \ldots, K)\rangle$, population of water particles (K), dimension (D), laminar probability (pl), eddying probability(pe), lb, ub, C_1, C_2 and maximum number of iterations (i_{max}).
2. Choose the initial population of the water particles(C_k) using the equation 9.
3. While ($i_{curr} \leq i_{max}$), do following
4. Compute the objective function (Euclidean distance) using the equation 22 and arrange the data (X_l) into different clusters (C_k)based on the minimum value of Euclidean distance.
5. Compute the potential energy (E) for each water particle(C_k) using the equation 17.

6.	Compute the laminar probability (pl) and turbulent operator (pe) using the equations 12-13.
	/*Start the Laminar Flow Phase */
7.	If $(\text{rand}() < pl)$
8.	Determine the laminar flow direction based on equation 15.
9.	Compute the new position of water particles using the equation 14. Check boundary constraints, if these are not satisfied, generate the new position using equation 7.
10.	Compute the potential energy $\left(f(C_k(i))\right)$ of water particles $(\mathrm{C_k})$using the equation 17.
	/*Start the Turbulent Flow Phase */
11.	Determine the trial position of water particles$(\mathrm{C_k})$ in random order.
12.	Randomly chosen the jostling operator (p) using the equation 4 such that $k \neq j$.
13.	Compute the new position of water particles using the equation 3. Check boundary constraints, if these are not satisfied, generate the new position using equation 7.
14.	Compute the potential energy $\left(f(C_k(i))\right)$ of water particles $(\mathrm{C_k})$using the equation 17.
	/*Start Evolving and updation */
15.	For each water particle $(\mathrm{C_k})$, do following
16.	If $\left(f(C_{k,new}) > f(C_{k,old})\right)$
17.	$f(C_k) \leftarrow f(C_{k,new})$
18.	$C_k \leftarrow C_{k,new}$
19.	Else
20.	$C_k \leftarrow C_{k,old}$
21.	Compute the global best position(g_{best}) and store in the list in increasing order of potential energy.
22.	If$\left(f(C_k)\right)$ is not improved up to limit parameter *local optima issue*/
23.	Call Multi-Strategy Search Process

```
24. If(f(C_k) < 0.4)
25. Invoke chaotic local search strategy using
    equation 18.
26. Else If(0.4 ≤ f(C_k) < 0.7)
27. Invoke mutation-based strategy using equation
    20.
28. Else
29. Invoke crossover-based strategy using equation
    21.
30. End if
31. If maximum iteration is not reached, repeat
    the steps 4-30.
32. Otherwise, obtain optimal centroids (C_k).
```

5 Experimental Results

This section presents the simulation result of the proposed IWFO algorithm and other algorithms. The efficacy of the IWFO algorithm is examined using twelve standard clustering datasets. These datasets are downloaded from the UCI repository and a description of these datasets is mentioned in Table 1. The simulation results of the proposed IWFO algorithm are compared with a wide variety of meta-heuristic algorithms. These algorithms are described as K-means [Chowdhury, 21], PSO [Cura, 12], ACO [Jiang, 10], ABC [Kumar, 17], DE [Kwedlo, 11], GA [Maulik, 00], BB-BC [Bijari, 18], BAT [Senthilnath, 16], VS [Dogan, 15], MBOA[Wang, 19], WOA [Singh, 23], ICSO [Kumar, 18], TLBO [Kumar, 19], CS [Boushaki, 18], GSA [Han, 17], LION [Chander, 18], and GWO [Aljarah, 20]. The simulation results are evaluated using intra, SD, rank, AR and DR parameters. The results indicate an average of thirty separate runs. The intra-parameter can be defined as the sum of the distance between data objects and respective cluster centres. This parameter indicates the closeness between data objects and cluster centres. Standard deviation (SD) is also computed for validating the intra-results. Further, AR denotes the accuracy rate while DR denotes the detection rate. The maximum iteration is set to 100. The parameter settings of the other algorithms except the proposed IWFO are recommended the same as reported in the corresponding literature. The parameters setting of the proposed IWFO are given as population size (P) and can be defined in terms of clusters (K), laminar probability $(P_{la}) \in (0.2, 0.5)$, eddying probability$(P_e) \in (0.5, 0.9)$, limit operator $(limt_{op}) = 5$, maximum iteration (max_iter) is set to 100. The experiment is conducted on a MATLAB environment using a corei5-based processor with 32 GB of RAM.

5.1 Simulation Results

This subsection discusses the simulation results of the proposed IWFO and other clustering algorithms using intra, SD, rank, AR and DR parameters. Table 2 presents

the simulation results of the proposed IWFO and other standard existing clustering algorithms based on intra, SD and rank parameters. The simulation results of the proposed algorithm are compared withK-means, PSO, ACO, ABC, DE, GA, BB-BC, and BAT algorithms. A variety of datasets are considered to evaluate the efficacy of the proposed IWFO and other clustering algorithms.

Sr. No.	Data sets	Cluster(K)	Dimension(D)	Instance(N)
1	Iris	3	4	150
2	Glass	6	9	214
3	Wine	3	13	178
4	Ionosphere	2	34	351
5	Control	6	60	600
6	Vowel	6	3	871
7	Balance	3	4	625
8	Crude Oil	3	5	511
9	CMC	3	9	1,473
10	LD	2	6	416
11	WBC	2	9	699
12	Thyroid	3	5	215

Table 1: Description of the datasets used in this work

5.1.1 Using Intra Cluster Distance Parameter

- **Existing Clustering Algorithm**

This subsection discusses the results of the proposed IWFO algorithm using intra-parameter. It is seen that the proposed IWFO algorithm obtains the minimum value of intra-parameter with most of the datasets. The proposed algorithm achieves minimum intra with iris (9.21E+01), glass (2.03E+02), ionosphere(9.04E+02), vowel(1.56E+05), balance(5.78E+04), CMC(5.64E+03), cancer(1.23E+03), and thyroid(1.27E+03) except wine, control, crude oil and LD datasets. For the wine dataset, the DE algorithm obtains the minimum intra-value (1.58E+04) while the proposed IWFO algorithm obtains the second minimum intra-value (1.59E+04) among all algorithms. It is also seen that the BB-BC algorithm gets minimum intra values (2.38E+04 and 3.13E+03) for the control and LD dataset, while the proposed IWFOalgorithm achieves (2.41E+04 and 2.85E+03)minimum intra values for both datasets. For the crude oil dataset, the ACO algorithm achieves minimum intra value (2.47E+02), whereas the proposed IWFO algorithm obtains (2.63E+02) value for intra parameter.SD is also an important parameter that can verify the performance of the proposed IWFO algorithm in successive iterations. By analyzing the SD parameter, it is revealed that the proposed IWFO algorithm obtains a minimum SD rate for most datasets compared to other algorithms. The proposed IWFO algorithm gets minimum SD rate with wine

(4.34E+01), ionosphere (1.53E+01), vowel (1.41E+01), balance (3.34E+02), crude oil (1.08E+01), CMC (2.14E+01), LD (1.27E+01), cancer (1.46E+02), and thyroid (1.12E+01). For the iris dataset, the ACO algorithm obtains a minimum SD rate (1.31E+00), while the proposed IWFO algorithm obtains a (3.20E+00) SD rate. For the glass dataset, the GA algorithm gets a minimum SD rate (4.14E+00) while the algorithm achieves (2.41E+04 and 2.85E+03)minimum intra values for both datasets. For the crude oil dataset, the ACO algorithm achieves minimum intra value (2.47E+02), whereas the proposed IWFO algorithm obtains (2.63E+02) value for intra parameter. SD is also an important parameter that can verify the performance of the proposed IWFO algorithm in successive iterations. By analyzing the SD parameter, it is revealed that the proposed IWFO algorithm obtains a minimum SD rate for most datasets compared to other algorithms. The proposed IWFO algorithm gets minimum SD rate with wine (4.34E+01), ionosphere (1.53E+01), vowel (1.41E+01), balance (3.34E+02), crude oil (1.08E+01), CMC (2.14E+01), LD (1.27E+01), cancer (1.46E+02), and thyroid (1.12E+01). For the iris dataset, the ACO algorithm obtains a minimum SD rate (1.31E+00), while the proposed IWFO algorithm obtains a (3.20E+00) SD rate. For the glass dataset, the GA algorithm gets a minimum SD rate (4.14E+00) while the proposed IWFO algorithm achieves a (9.98E+00) SD rate. For the control dataset, the BB-BC algorithm achieves a minimum SD rate (1.09E+02), while the proposed algorithm obtains a (1.42E+02) SD rate. It is analyzed that the proposed algorithm achieves comparable SD results with the aforementioned datasets. Further, a rank parameter is also used to describe the ranking of each algorithm and this parameter is devised based on the minimum intra parameter. By analyzing the rank parameter, it is revealed that the proposed IWFO algorithm obtains the first rank with eight datasets (iris, glass, ionosphere, vowel, balance, CMC, cancer, thyroid). For wine, control and LD datasets, the proposed algorithm obtains the second rank. For the crude oil dataset, the proposed algorithm gets third rank. Hence, it is stated that the proposed algorithm obtains superior results with most of the datasets.

Dataset	Measure	Existing Well-known Clustering Algorithms								
		K-means	PSO	ACO	ABC	DE	GA	BB-BC	BAT	IWFO
Iris	Intra	9.22 E+01	9.86 E+01	1.01 E+02	1.08 E+02	1.21 E+02	1.25 E+02	9.68 E+01	1.15 E+02	**9.21 E+01**
	SD	2.67 E+01	4.67 E-01	1.31 E+00	6.63 E+00	5.23 E+00	1.46 E+01	4.22 E+00	3.76 E+01	3.20 E+00
	Rank	2	4	5	6	8	9	3	7	1
Glass	Intra	3.79 E+02	2.76 E+02	2.19 E+02	3.29 E+02	3.62 E+02	2.82 E+02	6.64 E+02	3.75 E+02	**2.03 E+02**
	SD	7.05 E+01	1.86 E+01	3.36 E+00	1.14 E+01	1.21 E+01	4.14 E+00	6.89 E+01	1.03 E+01	9.98 E+00
	Rank	8	3	2	5	6	4	9	7	1
Wine	Intra	1.81 E+04	1.64 E+04	1.62 E+04	1.69 E+04	**1.58 E+04**	1.65 E+04	1.67 E+04	1.71 E+04	1.59 E+04
	SD	9.06 E+02	8.55 E+01	3.69 E+01	4.74 E+02	5.60 E+01	7.84 E+01	6.29 E+01	5.66 E+01	4.34 E+01
	Rank	9	4	3	7	1	5	6	8	2

Ionosphere	Intra	2.42 E+03	1.01 E+03	9.38 E+02	1.11 E+03	1.13 E+03	1.00 E+03	1.07 E+03	1.33 E+03	**9.04 E+02**
	SD	4.55 E+02	3.34 E+02	4.48 E+02	2.61 E+02	3.17 E+02	4.13 E+02	2.99 E+02	2.34 E+02	1.53 E+01
	Rank	9	3	2	6	7	4	5	8	1
Control	Intra	1.01 E+06	4.18 E+04	2.39 E+04	5.12 E+04	5.23 E+04	4.62 E+04	**2.38 E+04**	2.68 E+04	2.41 E+04
	SD	5.05 E+03	1.02 E+03	1.71 E+02	1.32 E+03	9.16 E+02	1.58 E+03	1.09 E+02	1.78 E+02	1.42 E+02
	Rank	9	5	3	7	8	6	1	4	2
Vowel	Intra	1.60 E+05	1.58 E+05	1.89 E+05	1.70 E+05	1.81 E+05	1.59 E+05	1.94 E+05	1.96 E+05	**1.56 E+05**
	SD	4.52 E+03	2.88 E+03	2.58 E+03	4.64 E+03	2.86 E+03	3.11 E+03	2.44 E+04	3.98 E+03	1.41 E+01
	Rank	4	2	6	5	7	3	8	9	1
Balance	Intra	1.20 E+05	6.20 E+04	5.94 E+04	6.61 E+04	6.78 E+04	6.91 E+04	5.96 E+04	6.02 E+04	**5.78 E+04**
	SD	9.28 E+03	4.01 E+03	7.56 E+02	6.79 E+02	5.25 E+03	5.62 E+03	3.72 E+02	8.26 E+02	3.34 E+02
	Rank	9	8	2	5	6	7	3	4	1
Crude oil	Intra	2.91 E+02	2.86 E+02	**2.47 E+02**	2.81 E+02	3.69 E+02	2.83 E+02	2.61 E+02	2.89 E+02	2.63 E+02
	SD	2.63 E+01	1.14 E+01	4.71 E+01	1.09 E+01	2.33 E+01	8.14 E+00	1.17 E+02	1.76 E+01	1.08 E+01
	Rank	9	7	1	5	4	6	2	8	3
CMC	Intra	5.59 E+03	5.85 E+03	5.83 E+03	5.94 E+03	5.95 E+03	5.76 E+03	5.71 E+03	5.79 E+03	**5.64 E+03**
	SD	4.68 E+01	4.89 E+01	1.23 E+02	1.31 E+02	8.69 E+01	5.04 E+01	2.86 E+01	3.67 E+01	2.14 E+01
	Rank	2	7	6	8	9	4	3	5	1
LD	Intra	1.17 E+04	3.24 E+03	3.24 E+03	9.85 E+03	1.15 E+04	2.54 E+03	**1.13 E+03**	1.74 E+03	1.23 E+03
	SD	6.68 E+02	2.88 E+01	1.64 E+01	8.20 E+02	2.07 E+03	4.18 E+01	2.41 E+01	1.52 E+01	1.27 E+01
	Rank	9	4	5	7	8	6	1	3	2
WBC	Intra	1.93 E+04	4.26 E+03	3.37 E+03	3.50 E+03	3.73 E+03	3.00 E+03	2.96 E+03	3.06 E+03	**2.85 E+03**
	SD	5.14 E-12	2.08 E+02	4.17 E+01	2.12 E+02	1.84 E+02	2.25 E+02	5.57 E+02	1.98 E+02	1.46 E+02
	Rank	9	8	5	6	7	4	2	3	1
Thyroid	Intra	2.39 E+03	1.11 E+04	1.99 E+03	1.98 E+03	2.96 E+02	1.22 E+04	1.94 E+03	1.39 E+03	**1.27 E+03**
	SD	2.46 E+02	2.71 E+01	3.09 E+01	2.23 E+02	2.06 E+01	3.26 E+01	1.95 E+02	2.28 E+01	1.12 E+01
	Rank	7	8	6	5	3	9	4	2	1

Table 2: Simulation results of the proposed IWFO and standard existing clustering algorithms using intra, SD and rank parameters

- **Recently Developed Metaheuristic Clustering Algorithm**

Further, the performance of the proposed IWFO algorithm is also compared with the recently developed meta-heuristic algorithms. These algorithms are VS, MBOA, WOA, ICSO, TLBO, CS, GSA, LION and GWO. The results are evaluated using intra, SD and rank parameters. The intra-parameter is utilized to compute the compactness among the data objects within clusters. Table 3 presents the simulation results of the proposed IWFO and other algorithms based on the intra, SD and rank parameters. It is noticed that the proposed IWFO algorithm gets the minimum value of intra-parameter with most of the datasets except vowel, balance, crude oil and thyroid datasets. The proposed IWFO algorithm obtains minimum intra values with iris (9.21E+01), wine (1.59E+04), glass (2.03E+02), ionosphere(9.04E+02), balance (5.78E+04), CMC(5.64E+03), cancer(1.23E+03), and LD (2.85E+03) except wine, control, crude oil and LD datasets. For vowel and thyroid datasets, the ICSO algorithm obtains minimum intra values (1.55E+05 and 9.90E+02) while the proposed IWFO algorithm obtains (1.56E+05 and 1.27E+03) intra values. It is also seen that the TLBO algorithm gets minimum intra values (5.36E+04 and 2.59E+02) for balance and crude oil datasets, while the proposed IWFO algorithm achieves (5.78E+04 and 2.63E+02)intra values for both datasets. SD is also a significant parameter that can verify the performance of the proposed IWFO algorithm in successive iterations. By analyzing the SD parameter, it is revealed that the proposed IWFO algorithm obtains a minimum SD rate for most of the datasets except iris, wine, control and balance compared to other algorithms. The proposed IWFO algorithm gets a minimum SD rate with glass (9.98E+00), ionosphere (1.53E+01), vowel (1.41E+01), crude oil (1.08E+01), CMC (2.14E+01), LD (1.27E+01), cancer (1.46E+02), and thyroid (1.12E+01). For the iris dataset, the WOA algorithm obtains a minimum SD rate (1.06E+00), while the proposed IWFO algorithm obtains a (3.20E+00) SD rate. For the wine dataset, the TLBO algorithm gets a minimum SD rate (2.94E+01) while the proposed IWFO algorithm achieves a (4.34E+01) SD rate. For control and balance datasets, the VS algorithm achieves minimum SD rates (1.17E+02 and 1.04E+02), while the proposed algorithm obtains (1.42E+02 and 3.34E+02) SD rates. For the crude oil dataset, the ABC algorithm gets a minimum SD rate (1.09E+01), while the proposed IWFO algorithm obtains a (3.80E+01) SD rate. It is analyzed that the proposed algorithm achieves comparable SD results with the aforementioned datasets.Further, a rank parameter is also used to describe the ranking of each algorithm and this parameter is devised based on the minimum intra parameter. By analyzing the rank parameter, it is revealed that the proposed IWFO algorithm obtains the first rank with most of the datasets except vowel, CMC, balance, crude oil and thyroid datasets. The proposed algorithm obtains the second rank with vowel, crude oil and thyroid datasets. For CMC and balance datasets, the proposed algorithm obtains the third rank among all algorithms being compared. Hence, it is stated that the proposed algorithm obtains better clustering results with most of the datasets based on intra, SD and rank parameters.

Dataset	Measure	Recent Clustering Algorithms									
		VS	MBO	WO	ICSO	TLBO	CS	GSA	LION	GWO	IWFO
Iris	Intra	9.89E+01	9.83E+01	9.84E+01	9.57E+01	9.69E+01	9.64E+01	9.79E+01	9.76E+01	9.98E+01	**9.21E+01**

	SD	1.21E+00	1.24E+00	1.06E+00	1.92E+00	2.88E+00	3.25E+00	3.19E+00	4.03E+00	3.12E+00	3.20E+00
	Rank	9	7	8	2	4	3	6	5	10	1
Glass	Intra	2.34E+02	2.31E+02	2.18E+02	2.26E+02	2.37E+02	2.41E+02	2.39E+02	2.38E+02	2.36E+02	**2.03E+02**
	SD	1.01E+01	1.27E+01	1.37E+01	1.10E+01	10.9E+00	1.12E+01	9.87E+00	1.07E+01	1.41E+01	9.98E+00
	Rank	5	4	2	3	7	10	9	8	6	1
Wine	Intra	1.83E+04	1.91E+04	1.84E+04	1.69E+04	1.68E+04	1.65E+04	1.70E+04	1.65E+04	1.66E+04	**1.59E+04**
	SD	6.90E+01	7.07E+01	4.83E+01	3.26E+01	2.94E+01	2.64E+01	2.74E+01	2.66E+01	1.94E+01	4.34E+01
	Rank	8	10	9	6	5	2	7	3	4	1
Ionosphere	Intra	2.25E+03	2.93E+03	2.45E+03	1.45E+03	1.11E+03	1.23E+03	2.83E+03	1.42E+03	1.58E+03	**9.04E+02**
	SD	3.67E+01	4.39E+01	3.69E+01	3.62E+01	2.35E+01	3.46E+01	4.04E+01	3.38E+01	2.98E+01	1.53E+01
	Rank	7	10	8	5	2	3	9	4	6	1
Control	Intra	3.15E+04	3.04E+04	2.99E+04	2.51E+04	2.55E+04	3.03E+04	3.13E+04	2.75E+04	2.63E+04	**2.41E+04**
	SD	1.17E+02	9.68E+02	8.47E+01	7.08E+01	4.83E+01	6.18E+01	5.37E+01	4.47E+01	6.65E+01	1.42E+02
	Rank	10	8	6	2	3	7	9	5	4	1
Vowel	Intra	1.59E+05	1.62E+05	1.58E+05	**1.55E+05**	1.57E+05	1.59E+05	1.60E+05	1.59E+05	1.61E+05	1.56E+05
	SD	4.52E+02	2.94E+02	2.54E+02	1.77E+02	1.78E+02	3.42E+01	3.87E+01	1.74E+01	2.09E+01	1.41E+01
	Rank	7	10	4	1	3	5	8	6	9	2
Balance	Intra	5.84E+04	5.96E+04	6.06E+04	5.39E+04	**5.36E+04**	5.79E+04	5.81E+04	5.86E+04	5.83E+04	5.78E+04
	SD	1.04E+02	1.81E+02	1.83E+02	2.17E+02	1.29E+02	2.19E+02	3.02E+02	2.98E+02	1.78E+02	3.34E+02
	Rank	7	9	10	2	1	4	5	8	6	3
Crude oil	Intra	2.81E+02	2.86E+02	2.83E+02	2.64E+02	**2.59E+02**	2.74E+02	2.71E+02	2.69E+02	2.68E+02	2.63E+02
	SD	1.54E+01	2.08E+01	1.62E+01	1.48E+01	1.37E+01	1.24E+01	1.75E+01	1.39E+01	1.50E+01	1.08E+01
	Rank	8	10	9	3	1	7	6	5	4	2
CMC	Intra	5.76E+03	**5.21E+03**	5.69E+03	5.32E+03	5.65E+03	5.85E+03	5.67E+03	5.70E+03	5.73E+03	5.64E+03
	SD	6.42E+01	5.60E+01	6.98E+00	7.45E+00	6.38E+00	2.35E+02	6.32E+01	9.47E+01	8.35E+01	2.14E+01
	Rank	9	1	6	2	4	10	5	7	8	3
LD	Intra	1.52E+03	1.32E+03	1.91E+03	1.41E+03	1.50E+03	1.39E+03	1.73E+03	1.33E+03	1.36E+03	**1.23E+03**
	SD	6.32E+01	6.52E+01	5.91E+01	2.95E+01	2.39E+01	5.62E+01	4.79E+01	2.68E+01	2.15E+01	1.27E+01
	Rank	8	2	10	6	7	5	0	3	4	1
WBC	Intra	4.09E+03	3.61E+03	3.76E+03	3.12E+03	2.99E+03	3.72E+03	2.92E+03	2.89E+03	3.13E+03	**2.85E+03**
	SD	4.33E+01	2.87E+01	2.26E+01	1.81E+01	2.35E+01	1.73E+01	8.36E+00	7.67E+00	1.65E+01	1.46E+02
	Rank	10	7	9	5	4	8	3	2	6	1
Thyroid	Intra	1.96E+03	2.16E+03	1.67E+03	**9.90E+02**	1.68E+03	1.43E+03	1.86E+03	1.54E+03	1.39E+03	1.27E+03
	SD	1.81E+01	1.66E+01	2.09E+01	1.16E+01	1.48E+01	2.39E+01	1.90E+01	2.46E+01	1.73E+01	1.12E+01

	Rank	9	10	6	1	7	4	8	5	3	2

Table 3: Simulation results of the proposed IWFO and recent clustering algorithms based on intra, SD and rank parameters

5.1.2 Using AR and DR Parameters

- **Existing Clustering Algorithm**

The efficacy of the proposed IWFO algorithm is also assessed using AR and DR parameters. The simulation results of the proposed IWFO and other standard existing algorithms are reported in Table 4. It is seen that the proposed IWFO algorithm gets superior clustering results based on AR parameter compared to KM, PSO, ACO, ABC, DE, GA, BB-BC and BAT algorithms. The proposed IWFO algorithm obtains better AR results with most of the datasets such as iris (0.961), glass (0.717), wine (0.873), ionosphere (0.789), control (0.801), vowel (0.878), balance (0.913), crude oil (0.796), CMC (0.613), LD (0.685), cancer (0.836), and thyroid (0.738). DR is also described as one of the potential parameters for analyzing the performance of the clustering algorithms. By analyzing the DR parameter, it is stated that the proposed

Dataset	Parameter	KM	PSO	ACO	ABC	DE	GA	BB-BC	BAT	IWFO
Iris	AR	0.673	0.833	0.789	0.887	0.842	0.741	0.868	0.904	**0.961**
	DR	0.696	0.857	0.794	0.892	0.868	0.773	0.882	0.907	**0.968**
Glass	AR	0.519	0.537	0.374	0.489	0.481	0.490	0.555	0.484	**0.717**
	DR	0.538	0.572	0.384	0.509	0.492	0.511	0.586	0.504	**0.743**
Wine	AR	0.739	0.711	0.746	0.773	0.741	0.729	0.766	0.787	**0.873**
	DR	0.752	0.737	0.781	0.793	0.762	0.749	0.794	0.813	**0.896**
Ionosphere	AR	0.712	0.648	0.607	0.644	0.630	0.601	0.626	0.621	**0.789**
	DR	0.728	0.647	0.612	0.664	0.653	0.618	0.634	0.632	**0.804**
Control	AR	0.597	0.412	0.395	0.356	0.393	0.467	0.394	0.668	**0.801**
	DR	0.614	0.452	0.437	0.396	0.417	0.492	0.421	0.695	**0.819**
Vowel	AR	0.763	0.753	0.775	0.796	0.698	0.745	0.813	0.832	**0.878**
	DR	0.846	0.795	0.806	0.832	0.747	0.79	0.856	0.859	**0.904**
Balance	AR	0.850	0.898	0.743	0.767	0.750	0.780	0.797	0.868	**0.913**
	DR	0.863	0.904	0.772	0.783	0.776	0.824	0.835	0.879	**0.932**
Crude oil	AR	0.655	0.765	0.591	0.568	0.665	0.632	0.636	0.632	**0.796**
	DR	0.684	0.773	0.645	0.587	0.681	0.654	0.649	0.654	**0.823**
CMC	AR	0.357	0.514	0.369	0.416	0.437	0.403	0.447	0.426	**0.613**

	DR	0.455	0.598	0.465	0.489	0.466	0.435	0.512	0.487	**0.646**
LD	AR	0.522	0.541	0.529	0.499	0.520	0.493	0.502	0.531	**0.685**
	DR	0.635	0.587	0.564	0.549	0.648	0.590	0.613	0.654	**0.746**
Cancer	AR	0.698	0.721	0.749	0.786	0.654	0.691	0.670	0.792	**0.836**
	DR	0.751	0.748	0.773	0.824	0.678	0.715	0.698	0.836	**0.868**
Thyroid	AR	0.638	0.689	0.649	0.644	0.658	0.632	0.639	0.658	**0.738**
	DR	0.661	0.692	0.654	0.661	0.697	0.643	0.674	0.703	**0.786**

Table 4: Simulation results of the proposed IWFO and other existing standard clustering algorithms using AR and DR parameters

IWFO algorithm gets superior results with most of datasets such as iris (0.968), glass (0.743), wine (0.896), ionosphere (0.804), control (0.819), vowel (0.904), balance (0.932), crude oil (0.823), CMC (0.646), LD (0.746), cancer (0.868), and thyroid (0.786). Hence, it is stated that the proposed IWFO algorithm obtains better AR and DR results than similar class of algorithms.

- **Recently Developed Metaheuristic Clustering Algorithm**

Moreover, the performance of the proposed IWFO algorithm is also compared with several recent clustering algorithms reported in the literature. The simulation results of the proposed IWFO and other algorithms based on AR and DR parameters are presented in Table 5. It is revealed that the proposed IWFO algorithm obtains better AR results with most of the datasets except wine, balance, control and cancer. The proposed algorithm achieves a higher accuracy rate with iris (0.961), glass (0.717), wine (0.873),ionosphere (0.789), control (0.801), vowel (0.878), balance (0.913),crude oil (0.796), CMC (0.613), LD (0.685), cancer (0.836), and thyroid (0.738).For the wine dataset, the LION algorithm obtains a higher AR rate (0.878) among all algorithms, while the proposed IWFO gets (0.873) AR rate. It is also noticed that the GWO algorithm obtains a higher AR rate (0.814) for the control dataset, whereas, the proposed algorithm obtains a (0.801) AR rate. For the balance dataset, the CS algorithm achieves a superior AR rate (0.917), while the proposed algorithm gets (0.913) AR results. Furthermore, the TLBO algorithm achieves a better AR rate (0.921) for the cancer dataset, while the proposed algorithm obtains (0.836) AR rates. It is also analyzed that the proposed algorithm gets comparable AR results for the aforementioned datasets. By analyzing the DR parameter, it is said that the proposed IWFO algorithm obtains better DR results with most of datasets except wine, balance, control and cancer. The proposed algorithm achieves higher DR results with iris (0.968), glass (0.743), wine (0.896), ionosphere (0.804), control

Dataset	Measure	Recent Clustering Algorithms									
		VS	MBO	WO	ICSO	TLBO	CS	GSA	LION	GWO	IWFO
Iris	AR	0.941	0.954	0.946	0.914	0.912	0.885	0.783	0.852	0.852	**0.961**
	DR	0.956	0.959	0.951	0.932	0.928	0.893	0.774	0.812	0.868	**0.968**
Glass	AR	0.589	0.593	0.684	0.691	0.695	0.689	0.664	0.681	0.678	**0.717**
	DR	0.602	0.625	0.693	0.726	0.742	0.726	0.684	0.713	0.709	**0.743**
Wine	AR	0.697	0.7031	0.684	0.732	0.725	0.803	0.79	**0.878**	0.792	0.873
	DR	0.737	0.717	0.703	0.754	0.767	0.824	0.818	**0.908**	0.837	0.896
Ionosphere	AR	0.633	0.646	0.611	0.687	0.694	0.735	0.752	0.769	0.746	**0.789**
	DR	0.649	0.683	0.629	0.725	0.737	0.746	0.762	0.779	0.753	**0.804**
Control	AR	0.578	0.603	0.620	0.713	0.728	0.698	0.674	0.738	**0.814**	0.801
	DR	0.629	0.634	0.673	0.747	0.734	0.722	0.694	0.762	**0.833**	0.819
Vowel	AR	0.640	0.569	0.587	0.653	0.649	0.835	0.847	0.851	0.815	**0.878**
	DR	0.678	0.581	0.618	0.682	0.652	0.866	0.858	0.897	0.842	**0.904**
Balance	AR	0.713	0.720	0.693	0.786	0.810	**0.917**	0.849	0.855	0.793	0.913
	DR	0.732	0.756	0.728	0.816	0.856	**0.938**	0.887	0.896	0.845	0.932
Crude oil	AR	0.623	0.652	0.635	0.746	0.734	0.704	0.761	0.805	0.748	**0.796**
	DR	0.647	0.663	0.658	0.779	0.752	0.713	0.792	0.834	0.768	**0.823**
CMC	AR	0.411	0.442	0.424	0.468	0.465	0.571	0.547	0.534	0.516	**0.613**
	DR	0.456	0.482	0.443	0.501	0.483	0.614	0.582	0.558	0.548	**0.646**
LD	AR	0.509	0.507	0.514	0.530	0.532	0.659	0.631	0.662	0.615	**0.685**
	DR	0.568	0.549	0.583	0.571	0.609	0.689	0.654	0.713	0.648	**0.746**
Cancer	AR	0.792	0.853	0.834	0.918	0.921	0.820	0.728	0.789	0.824	**0.836**
	DR	0.826	0.884	0.876	0.946	0.952	0.841	0.786	0.841	0.872	**0.868**
Thyroid	AR	0.604	0.594	0.621	0.682	0.714	0.691	0.678	0.724	0.706	**0.738**
	DR	0.647	0.628	0.655	0.697	0.748	0.748	0.729	0.759	0.739	**0.786**

Table 5: Simulation results of proposed IWFO and recent clustering algorithms using AR and DR parameters

(0.819), vowel (0.904), balance (0.932), crude oil (0.823), CMC (0.646), LD (0.746), cancer (0.868), and thyroid (0.786). For the wine dataset, the LION algorithm obtains a higher DR rate (0.908) among all algorithms, while the proposed IWFO gets a (0.896) DR rate. It is also noticed that the GWO algorithm obtains a higher DR rate (0.833) for the control dataset, whereas, the proposed algorithm obtains a (0.819) DR rate. For the balance dataset, the CS algorithm achieves a superior DR rate (0.938), while the proposed algorithm gets (0.932) DR results. Furthermore, the TLBO algorithm

achieves a better DR rate (0.952) for the cancer dataset, while the proposed algorithm obtains (0.868) DR rates. Finally, it is stated that the proposed IWFO algorithm gets superior clustering with most datasets in terms of intra, SD, rank, AR and DR parameters.

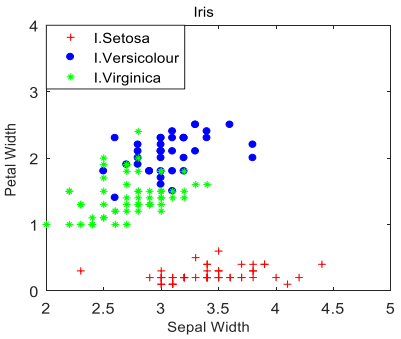

Fig. 2(a): Iris Dataset

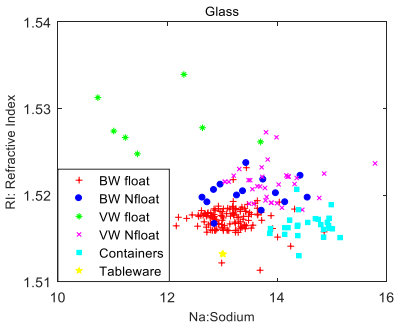

Fig. 2(b): Glass Dataset

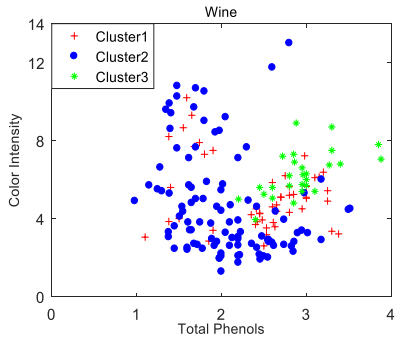

Fig. 2(c): Wine Dataset

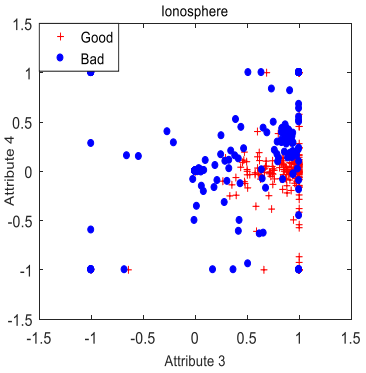

Fig. 2(d): Ionosphere Dataset

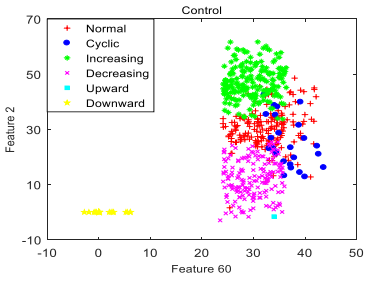

Fig. 2(e): Control DatasetFig.

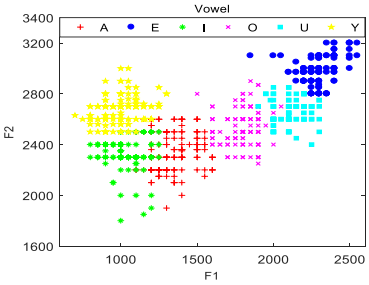

Fig. 2(f): Vowel Dataset

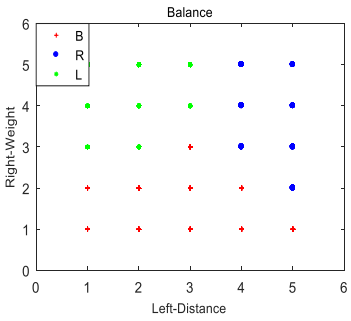

Fig. 2(g): Balance Dataset

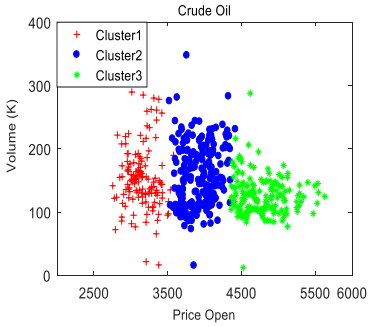

Fig. 2(h): Crude oil Dataset

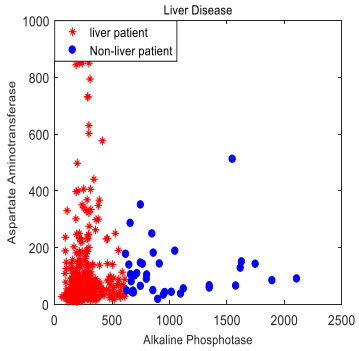

Fig. 2(i): LD Dataset

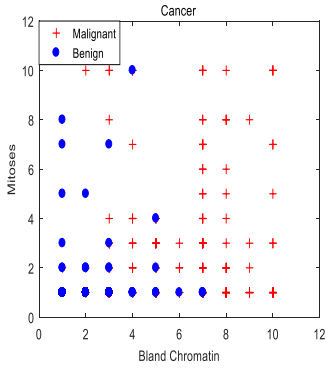

Fig. 2(j): Cancer Dataset

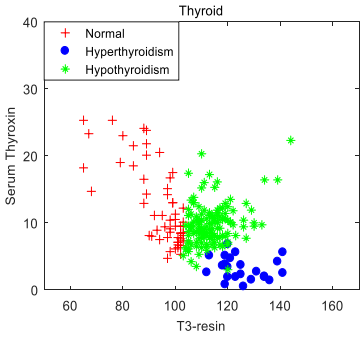

Fig. 2(k): ThyroidDataset

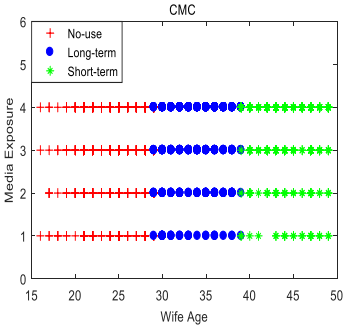

Fig. 2(l): CMC Dataset

Figure 2 (a-l): Demonstrates the different clusters of data using the IWFO algorithm

5.1.3 Cluster Analysis Using the IWFO algorithm

This subsection illustrates the efficiency of the IWFO algorithm based on the cluster analysis using different datasets. Figure 2 (a-l) illustrates the grouping of the data into different clusters based on the proposed IWFO clustering algorithm. This work considers the well-known clustering datasets such as iris, glass, wine, ionosphere, control, vowel, balance, crude oil, CMC, LD, cancer, and thyroid. The clusters of the iris dataset are depicted in Fig.2 (a) based on the petal width and sepal width attributes. The proposed algorithm divides the data into three clusters i.e. I.Setosa, I.Versicolour, and I.Virginica. It is seen that the data belongs to I.Setosa is linearly separable from the rest of the clusters, but the data belongs to I.Versicolour, and I.Virginica clusters are non-linear separable. But, the proposed algorithm effectively allocates the data to respective clusters. Fig. 2(b) depicts the clusters in the glass dataset. The glass dataset is divided into six clusters (BW float, BW Nfloat, VW float, VW Nfloat, Containers, and Tableware) based on the proposed IWFO algorithm. It is observed that the data allocatedto the VM Float cluster is linearly separable from other clusters, while data belonging to the rest of the clusters are nonlinear. It is also stated that the proposed algorithm effectively allocates the data to different clusters. Figure 2(c) illustrates the data allocated to different clusters using the wine dataset. The wine dataset consists of three clusters i.e. cluster1, cluster2 and cluster3. The clusters are identified using colour intensity and total phenol attributes. It is also noticed that data are non-linear. Fig. 2(d) depicts the clusters of the ionosphere dataset using attribute3 and attribute4. It is also reported that the ionosphere dataset comprises thirty-four attributes. The results showed that this dataset is non-linearly separable, while the proposed IWFO algorithm obtains good results in terms of data allocation to respective clusters. Fig. 2(e) demonstrates the clusters in the control dataset using feature2 and feature60. The control dataset consists of sixty attributes which makes it a non-linearly separable dataset. The proposed IWFO algorithm divides the balance dataset into six clusters. The results stated that data belonging to the downward cluster is linearly separable from the rest of the clusters, but it is quite complicated to allocate the data rest of the clusters. The clusters of the vowel dataset based on F1 and F2 attributes are presented in Fig. 2(f). The proposed algorithm divides the vowel dataset into six clusters. The clusters of the balance dataset are depicted in Fig. 2(g). The data is divided into three clusters i.e. L, B, and R using left distance and right distance attributes. It is observed that clusters are linearly separable. Fig. 2(h) illustrates the clusters of crude oil dataset using the price open and volume attributes. The data is divided into three clusters and it is analyzed that it is difficult to allocate the data to appropriate clusters. Several medical datasets are also considered to evaluate the efficacy of the proposed IWFO algorithm and the task of the proposed algorithm is to compute the accurate clusters in medical datasets. Fig. 2(i) demonstrates the clusters in liver disease (LD) datasets. The clusters are identified using the alkaline phosphatase and aspartate aminotransferase attributes. It is seen that the proposed algorithm accurately allocates the data into two clusters i.e. live patient and non-liver patient. Fig. 2(j) illustrates the results of the cancer dataset using bland chromatin and mitoses attributes. The proposed algorithm divides the dataset into two clusters i.e. Malignant and Benign. It is analyzed that data is not linearly separable. The clusters of the thyroid dataset are presented in Fig. 2(k) using T3-resin and serum thyroxin. The proposed IWFO algorithm divides the data into three clusters i.e. Normal, Hyperthyroidism, and Hypothyroidism. It is revealed that normal and hyperthyroidism

clusters are non-linear while hypothyroidism clusters are linear. The clusters of the CMC dataset are presented in Fig. 2(l) using wife age and media exposure attributes. The proposed algorithm divides the data into three clusters and it is stated that data is linearly separable. Hence, it is summarized that the proposed algorithm allocates the data to appropriate clusters more accurately.

5.2 Statistical Test Results

This section analyses the statistical results of the proposed IWFO and other clustering algorithms. It is stated that the effectiveness of the new clustering algorithm can be validated by using simulation results as well as statistical tests. Hence, a Friedman statistical test is used to validate the efficacy of the proposed IWFO in the clustering field. This test statistically validates the existence of the proposed IWFO algorithm. A newly designed algorithm should outperform other algorithms both statistically and experimentally. Before using the Friedman statistical test, two hypotheses are developed: (i) H_0 represents no significant difference between the proposed IWFO and other clustering algorithms, called null hypothesis, and (ii) H_1 represents a significant difference between the performance of IWFO and other algorithms. The Friedman test utilizes the Friedman statistics to obtain the average rank of each algorithm. The average ranking of each algorithm based on intra-parameter using all datasets is mentioned in Table 6. It is analyzed that the proposed IWFO algorithm obtains a higher rank i.e. 2.25 among all algorithms being compared. The ICSO algorithm achieves a second higher rank i.e. 5.04 and the KM algorithm gets a lower rank i.e. 13.9 among all algorithms. Hence, it is revealed that there is a significant difference between the ranking of the proposed IWFO and other algorithms.

Algorithm	Rank	Algorithm	Rank	Algorithm	Rank
KM	13.9	BB-BC	8.79	TLBO	5.63
PSO	11.9	BAT	11.9	CS	8.1
ACO	8.58	VS	11.2	GSA	9.3
ABC	13.3	MBOA	10.75	LION	7
DE	13.1	WOA	10.4	GWO	8.25
GA	11.7	ICSO	5.04	IWFO	2.25

Table 6: Average ranking of the proposed IWFO and other algorithms based on intra parameter using different datasets

Further, the statistical results of the Friedman test are reported in Table 7. It is observed that Friedman's statistics is 70.3671. The critical value can be defined as $X(0.01, 17)$ which is described through the significance level 0.01 and degree of freedom (DF) 17 and the critical value for Friedman test is 27.5871. Further, the p-value computed for this test is 1.864E-8. Hence, it is stated that the p-value is far less than critical values, in turn, the hypothesis (H_0) is rejected at the significance level of 0.01. The rejection

of the null hypothesis stated that a substantial difference occurrs between the performance of the proposed IWFO and other algorithms.

Statistical Value	p-Value	DF	Critical Value	Hypothesis
70.3671	1.864E-8	17	27.5871	Rejected

Table 7: Friedman test statistical results using intra-parameter

6 Conclusion

In this work, an improved WFO algorithm is presented for effectively handling the clustering problems. The proposed algorithm is inspired based on the water flow. Before implementing the WFO algorithm in the clustering field, several amendments are incorporated into WFO to make it more robust and effective for dealing the clustering problems. These amendments are summarized as the initial population of the WFO algorithm is computed through a logistic chaotic map-based function instead of random initialization. The balance between exploration and exploitation processes is maintained by nonlinear sigmoid and tanh functions. Further, a multi-search strategy is integrated to avoid the local optima issue. The performance of the proposed IWFO algorithm is evaluated using twelve benchmark clustering datasets. The simulation results of the proposed IWFO algorithm are compared with seventeen meta-heuristic algorithms based on intra, rank, SD, AR and DR parameters. The simulation results indicated that the proposed IWFO algorithm obtains superior intra and SD results with most of the datasets compared to other algorithms being compared. By analyzing AR and DR parameters, it is stated that the proposed IWFO algorithm has superior AR and DR rates with most of the datasets in comparison to other algorithms. Further, the efficacy of the proposed algorithm is also validated using the Friedman statistical test and this test also confirmed that the proposed algorithm is different than others. Finally, it is concluded that the proposed algorithm achieves better clustering results due to the optimal partition of the datasets. Hence, it is said that IWFO is one of efficient and robust algorithms in the field of data clustering. In future, the proposed IWFO will be explored for obtaining spherical-shaped clusters and multi-objective clustering. The efficacy of WFO will be examined in some other fields like feature extraction and reduction, image processing, classification, constrained optimization problems and outlier detection.

Acknowledgements

Authors have not received any funding to carry out this work.

Appendix-1

Toy example

This subsection explains the working of the proposed IWFO algorithm with the help of the toy example. The dataset is summarized into Table 8. The datasets consists of twelve data objects, four attributes and three clusters. The working of the proposed IWFO algorithm is described using six major components.

Dim_1	Dim_2	Dim_3	Dim_4
5.1	3.5	1.4	0.2
4.9	3	1.4	0.2
4.7	3.2	1.3	0.2
4.6	3.1	1.5	0.2
5	3.6	1.4	0.2
5.4	3.9	1.7	0.4
4.6	3.4	1.4	0.3
5	3.4	1.5	0.2
4.4	2.9	1.4	0.2
4.9	3.1	1.5	0.1
5.4	3.7	1.5	0.2
4.8	3.4	1.6	0.2

Table 8: Toy example for clustering

1. **Initialization:** This step corresponds to initialize the user defined parameters and compute the initial population (water particles) for IWFO algorithm. The water particles are represented through cluster centroids. The initial position of water particles are computed using equation 9 for toy dataset and it is summarized into Table 9.

4.8348	3.3348	1.4739	0.2304
4.7642	3.2642	1.4457	0.2092
4.6980	3.198	1.4192	0.1894

Table 9: Initial position of water particles (initial cluster centroids)

The UB and LB for the toy example are (5.4, 3.9, 1.7, 0.4) and (4.4, 2.9, 1.3, 0.1) respectively. The value of $C_1 and C_2$ are 0.3 and 0.5. The maximum iteration is set to 10 for toy example.

2. Objective Function and Data Allocation: In this step, the objective function is computed and it is described using the sum of squared error. The objective function is computed using each water particles and each data objects. Table 10 illustrates the objective function value of each water particles and data objects. WP denotes the water particles. Further, the data is allocated to respective clusters based on minimum value of the objective function. Table 11 depicts the arrangement of the data objects into different clusters using the initial position of the water particles.

WP_1	WP_2	WP_3
0.3225	0.4130	0.5032
0.3503	0.3007	0.2837
0.2598	0.1719	0.1197
0.3344	0.2387	0.1608
0.3225	0.4130	0.5032
0.8478	0.9537	1.0530
0.2640	0.2361	0.2510
0.1821	0.2776	0.3723
0.6201	0.5172	0.4221
0.2776	0.2455	0.2548
0.6741	0.7728	0.8668
0.1493	0.2089	0.2898

Table 10: Illustrates the value of the objective function for each water particles

Dim_1	Dim_4	Dim_3	Dim_4	Clusters
5.1	3.5	1.4	0.2	1
4.9	3	1.4	0.2	3
4.7	3.2	1.3	0.2	3
4.6	3.1	1.5	0.2	3
5	3.6	1.4	0.2	1
5.4	3.9	1.7	0.4	1
4.6	3.4	1.4	0.3	2
5	3.4	1.5	0.2	1
4.4	2.9	1.4	0.2	3
4.9	3.1	1.5	0.1	2

5.4	3.7	1.5	0.2	1
4.8	3.4	1.6	0.2	1

Table 11: Depicts the allocation of data objects into different clusters

After allocation of data objects into different clusters, goodness (fitness) of the each water particle is computed. The fitness is computed using the equation 17. The fitness of each water particle is (WP_1, 0.3963), (WP_2, 0.3319), (WP_3, 0.2716). Further, the global best position of water particle is chosen based on the maximum value of potential energy. Hence, the global best position of water particle is (4.8348, 3.3348, 1.4739, 0.2304). In this step, laminar and turbulent probability are also computed using equations 12-13. The laminar probability is 0.11, while turbulent probability is 0.15 for 1st iteration.

3. Laminar Flow: The direction of the flow is decided by comparing the laminar probability with a random function such as if $(\text{rand}\ () < pl)$. The rand() function gives the value 0.081. The laminar probability is higher than rand() function. So, the direction of the flow is laminar. Now, the new position of water particles is computed using equation 14. But prior to this, the flow direction and potential energy should be computed and these are computed using equations 15-16. The flow direction ($\vec{\text{d}}$) of each water particle is mentioned in Table 12 below. The potential energy of each water particle is (WP_1, 0.6835), (WP_2, 0.5180) and (WP_3, 0.7261).

WP_1	0.07	0.07	0.03	0.02
WP_2	0.07	0.07	0.03	0.02
WP_3	0.14	0.14	0.05	0.04

Table 12: Flow of direction ($\vec{d}$) for each water particle

Again, the laminar probability is compared with rand() function. If, the laminar probability is higher than rand( ) function, then new position of water particles is generated using $x_t(i) + \text{r} \times \vec{\text{d}}$, otherwise the new position is generated using $\text{x}_t(i) + dt.(x_{best} - \text{x}_t(i))$. It is noticed that rand() function returns 0.098 which is less than laminar probability (0.11). Hence, the new position of water is presented into Table 13 and the potential energy of newly generated water particles are (WP_1, 0.3615), (WP_2, 0.3457), and (WP_3, 0.2927).

5.0552	3.6452	1.6821	0.1865
4.2813	3.1383	1.5130	0.2315
4.6135	3.3275	1.4035	0.2913

Table 13: New position of the water particles

4. Turbulent Flow: If laminar probability is less than the random function i.e. rand(), then turbulent flow phase is invoked. But, in first iteration, this condition is false and the flow of water is not turbulent. This phase is also used to update the old position of

the water particles and the equation 4 is used to compute the same. Further, potential energy (fitness) of each particle is computed using equation 17 in successive iterations water flow is either turbulent or laminar based on the condition.

5. Evolving and updation: This step compares the potential energy of newly generated water particles and old water particles. The potential energy of old water particles are (WP_1, 0.3963), (WP_2, 0.3319), (WP_3, 0.2716), while the potential energy of new water particles are (WPN_1, 0.3615), (WPN_2, 0.3457), and (WPN_3, 0.2927). By comparing the potential energy of each water particles, the new position of the water particles are decided which is listed in the Table 14.

4.8348	3.3348	1.4739	0.2304
4.2831	3.1383	1.513	0.2315
4.6135	3.3275	1.4035	0.2913

Table 14: New position of the water particles in evolving and updation phase

If the potential energy of water particles are not improved up to limit operator in successive iteration. It is assumed that algorithm sticks in local optima and this problem is handled through multi strategy search procedure. This search procedure is invoke after five consecutive iteration, if there is no a significant improvement in potential energy of water particles. The size of the toy example is small and number of iteration is also set to 10. So, this method is not invoked for toy example. The final position of water particles after 10^{th} iteration is mentioned in Table 15.

4.6833	3.1167	1.4167	0.2011
5.4010	3.8003	1.6120	0.3021
4.9750	3.4750	1.4750	0.2310

Table 15: Final position (cluster centroids) of the water particles

References

[Abreu, 22] de Abreu, L.R., Araújo, K.A.G., de Athayde Prata, B., Nagano, M.S. and Moccellin, J.V., 2022. A new variable neighbourhood search with a constraint programming search strategy for the open shop scheduling problem with operation repetitions. Engineering Optimization, 54(9), pp.1563-1582.

[Aggarwal, 18] Aggarwal, C.C., 2018. An introduction to cluster analysis. In Data clustering (pp. 1-28). Chapman and Hall/CRC.

[Barshandeh, 22] Barshandeh, S., Dana, R. and Eskandarian, P., 2022. A learning automata-based hybrid MPA and JS algorithm for numerical optimization problems and its application on data clustering. Knowledge-Based Systems, 236, p.107682.

[Behadili, 22] Al-Behadili, H.N.K., 2022. Improved firefly algorithm with variable neighborhood search for data clustering. Baghdad Science Journal, 19(2), pp.0409-0409.

[Bijari, 18] Bijari, K., Zare, H., Veisi, H. and Bobarshad, H., 2018. Memory-enriched big bang–big crunch optimization algorithm for data clustering. Neural Computing and Applications, 29, pp.111-121.

[Boushaki, 18] Boushaki, S.I., Kamel, N. and Bendjeghaba, O., 2018. A new quantum chaotic cuckoo search algorithm for data clustering. Expert Systems with Applications, 96, pp.358-372.

[Bouyer, 16] Bouyer, A., 2016. An optimized k-harmonic means algorithm combined with modified particle swarm optimization and Cuckoo Search algorithm. Foundations of Computing and Decision Sciences, 41(2), pp.99-121.

[Brock, 08] Brock, G., Pihur, V., Datta, S. and Datta, S., 2008. clValid: An R package for cluster validation. Journal of statistical Software, 25, pp.1-22.

[Chander, 18] Chander, S., Vijaya, P. and Dhyani, P., 2018. Multi kernel and dynamic fractional lion optimization algorithm for data clustering. Alexandria engineering journal, 57(1), pp.267-276.

[Cheng, 23] Cheng, M.M., Zhang, J., Wang, D.G., Tan, W. and Yang, J., 2023. A Localization Algorithm Based on Improved Water Flow Optimizer and Max-Similarity Path for 3D Heterogeneous Wireless Sensor Networks. IEEE Sensors Journal.

[Chowdhury, 21] Chowdhury, K., Chaudhuri, D. and Pal, A.K., 2021. An entropy-based initialization method of K-means clustering on the optimalnumber of clusters. Neural Computing and Applications, 33, pp.6965-6982.

[Chuang, 11] Chuang, L.Y., Hsiao, C.J. and Yang, C.H., 2011. Chaotic particle swarm optimization for data clustering. Expert systems with Applications, 38(12), pp.14555-14563.

[Cura, 12] Cura, T., 2012. A particle swarm optimization approach to clustering. Expert Systems with Applications, 39(1), pp.1582-1588.

[Cura, 12] Cura, T., 2012. A particle swarm optimization approach to clustering. Expert Systems with Applications, 39(1), pp.1582-1588.

[Dogan, 15] Doğan, B. and Ölmez, T., 2015. A new metaheuristic for numerical function optimization: Vortex Search algorithm. Information sciences, 293, pp.125-145.

[Dorigo, 06] Dorigo, M., Birattari, M. and Stutzle, T., 2006. Ant colony optimization. IEEE computational intelligence magazine, 1(4), pp.28-39.

[Du, 19] Du, Z., Han, D. and Li, K.C., 2019. Improving the performance of feature selection and data clustering with novel global search and elite-guided artificial bee colony algorithm. The Journal of Supercomputing, 75, pp.5189-5226.

[Duan, 22] Duan, Y., Liu, C., Li, S., Guo, X. and Yang, C., 2022. Gradient-based elephant herding optimization for cluster analysis. Applied Intelligence, 52(10), pp.11606-11637.

[Duan, 23] Duan, Y., Liu, C., Li, S., Guo, X. and Yang, C., 2023. An automatic affinity propagation clustering based on improved equilibrium optimizer and t-SNE for high-dimensional data. Information Sciences, 623, pp.434-454.

[Ergezer, 09] Ergezer, M., Simon, D. and Du, D., 2009, October. Oppositional biogeography-based optimization. In 2009 IEEE international conference on systems, man and cybernetics (pp. 1009-1014). IEEE.

[Erol, 06] Erol, O.K. and Eksin, I., 2006. A new optimization method: big bang–big crunch. Advances in engineering software, 37(2), pp.106-111.

[Ezugwu, 22] Ezugwu, A.E., Ikotun, A.M., Oyelade, O.O., Abualigah, L., Agushaka, J.O., Eke, C.I. and Akinyelu, A.A., 2022. A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects. Engineering Applications of Artificial Intelligence, 110, p.104743.

[Gan, 20] Gan, G., Ma, C. and Wu, J., 2020. Data clustering: theory, algorithms, and applications. Society for Industrial and Applied Mathematics.

[Halkidi, 01] Halkidi, M., Batistakis, Y. and Vazirgiannis, M., 2001. On clustering validation techniques. Journal of intelligent information systems, 17, pp.107-145.

[Han, 17] Han, X., Quan, L., Xiong, X., Almeter, M., Xiang, J. and Lan, Y., 2017. A novel data clustering algorithm based on modified gravitational search algorithm. Engineering Applications of Artificial Intelligence, 61, pp.1-7.

[Hashemi, 22] Hashemi, S.E., Tavana, M. and Bakhshi, M., 2022. A New Particle Swarm Optimization Algorithm for Optimizing Big Data Clustering. SN Computer Science, 3(4), 311.

[Hatamlou, 13] Hatamlou, A., 2013. Black hole: A new heuristic optimization approach for data clustering. Information sciences, 222, pp.175-184.

[Jain, 99] Jain, A.K., Murty, M.N. and Flynn, P.J., 1999. Data clustering: a review. ACM computing surveys (CSUR), 31(3), pp.264-323.

[Jiang, 10] Jiang, H., Yi, S., Li, J., Yang, F. and Hu, X., 2010. Ant clustering algorithm with K-harmonic means clustering. Expert Systems with Applications, 37(12), pp.8679-8684.

[Jiang, 11] Jiang, B., Pei, J., Tao, Y. and Lin, X., 2011. Clustering uncertain data based on probability distribution similarity. IEEE Transactions on Knowledge and Data Engineering, 25(4), pp.751-763.

[Jordehi, 15] Jordehi, A.R., 2015. Enhanced leader PSO (ELPSO): a new PSO variant for solving global optimisation problems. Applied Soft Computing, 26, pp.401-417.

[Karaboga, 05] Karaboga, D., 2005. An idea based on honey bee swarm for numerical optimization (Vol. 200, pp. 1-10). Technical report-tr06, Erciyes university, engineering faculty, computer engineering department.

[Karaboga, 07] Karaboga, D. and Basturk, B., 2007. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of global optimization, 39, pp.459-471.

[Kaur, 22] Kaur, A. and Kumar, Y., 2022. A multi-objective vibrating particle system algorithm for data clustering. Pattern Analysis and Applications, 25(1), pp.209-239.

[Kaur, 22] Kaur, A. and Kumar, Y., 2022. A new metaheuristic algorithm based on water wave optimization for data clustering. Evolutionary Intelligence, 15(1), pp.759-783.

[Kaur, 22] Kaur, A. and Kumar, Y., 2022. Neighborhood search based improved bat algorithm for data clustering. Applied Intelligence, 52(9), pp.10541-10575.

[Kumar, 14] Kumar, Y. and Sahoo, G., 2014. A charged system search approach for data clustering. Progress in Artificial Intelligence, 2(2-3), pp.153-166.

[Kumar, 16] Kumar, Y., Gupta, S., Kumar, D. and Sahoo, G., 2016. A clustering approach based on charged particles. Optimization Algorithms-Methods and Applications, pp.245-263.

[Kumar, 17] Kumar, Y. and Sahoo, G., 2017. A two-step artificial bee colony algorithm for clustering. Neural Computing and Applications, 28, pp.537-551.

[Kumar, 18] Kumar, Y. and Singh, P.K., 2018. Improved cat swarm optimization algorithm for solving global optimization problems and its application to clustering. Applied Intelligence, 48, pp.2681-2697.

[Kumar, 19] Kumar, Y. and Singh, P.K., 2019. A chaotic teaching learning based optimization algorithm for clustering problems. Applied Intelligence, 49(3), pp.1036-1062.

[Kumar, 21] Kumar, Y. and Kaur, A., 2021. Variants of bat algorithm for solving partitional clustering problems. Engineering with Computers, pp.1-27.

[Kuo, 20] Kuo, R.J. and Zulvia, F.E., 2020. Multi-objective cluster analysis using a gradient evolution algorithm. Soft Computing, 24(15), pp.11545-11559.

[Kuo, 21] Kuo, R.J., Zheng, Y.R. and Nguyen, T.P.Q., 2021. Metaheuristic-based possibilistic fuzzy k-modes algorithms for categorical data clustering. Information Sciences, 557, pp.1-15.

[Kuo, 22] Kuo, T. and Wang, K.J., 2022. A hybrid k-prototypes clustering approach with improved sine-cosine algorithm for mixed-data classification. Computers & Industrial Engineering, 169, p.108164.

[Kushwaha, 18] Kushwaha, N., Pant, M., Kant, S. and Jain, V.K., 2018. Magnetic optimization algorithm for data clustering. Pattern Recognition Letters, 115, pp.59-65.

[Kushwaha, 22] Kushwaha, N., Pant, M. and Sharma, S., 2022. Electromagnetic optimizationbased clustering algorithm. Expert Systems, 39(7), p.e12491.

[Kwedlo, 11] Kwedlo, W., 2011. A clustering method combining differential evolution with the K-means algorithm. Pattern Recognition Letters, 32(12), pp.1613-1621.

[Li, 19] Li, W., Zhang, Y., Sun, Y., Wang, W., Li, M., Zhang, W. and Lin, X., 2019. Approximate nearest neighbor search on high dimensional data—experiments, analyses, and improvement. IEEE Transactions on Knowledge and Data Engineering, 32(8), pp.1475-1488.

[Liu, 18] Liu, F., Sun, Y., Wang, G.G. and Wu, T., 2018. An artificial bee colony algorithm based on dynamic penalty and Lévy flight for constrained optimization problems. Arabian Journal for Science and Engineering, 43, pp.7189-7208.

[Luo, 21] Luo, K., 2021. Water flow optimizer: a nature-inspired evolutionary algorithm for global optimization. IEEE Transactions on Cybernetics, 52(8), pp.7753-7764.

[Macedo, 22] de Matos Macêdo, F.J. and da Rocha Neto, A.R., 2022, November. A Binary Water Flow Optimizer Applied to Feature Selection. In International Conference on Intelligent Data Engineering and Automated Learning (pp. 94-103). Cham: Springer International Publishing.

[MacQueen, 67] MacQueen, J., 1967, June. Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability (Vol. 1, No. 14, pp. 281-297).

[Maulik, 00] Maulik, U. and Bandyopadhyay, S., 2000. Genetic algorithm-based clustering technique. Pattern recognition, 33(9), pp.1455-1465.

[Mukhopadhyay, 15] Mukhopadhyay, A., Maulik, U. and Bandyopadhyay, S., 2015. A survey of multiobjective evolutionary clustering. ACM Computing Surveys (CSUR), 47(4), pp.1-46.

[Osmani, 22] Osmani, A., Mohasefi, J.B. and Gharehchopogh, F.S., 2022. Sentiment classification using two effective optimization methods derived from the artificial bee colony optimization and imperialist competitive algorithm. The Computer Journal, 65(1), pp.18-66.

[Ozbakir, 17] Özbakır, L. and Turna, F., 2017. Clustering performance comparison of new generation meta-heuristic algorithms. Knowledge-Based Systems, 130, pp.1-16.

[Patel, 16] Patel, K.A. and Thakral, P., 2016, April. The best clustering algorithms in data mining. In 2016 International Conference on Communication and Signal Processing (ICCSP) (pp. 2042-2046). IEEE.

[Reddy, 18] Reddy, C.K., 2018. Data clustering: algorithms and applications. Chapman and Hall/CRC.

[Sahoo, 14] Sahoo, A.J. and Kumar, Y., 2014. Modified teacher learning based optimization method for data clustering. In Advances in Signal Processing and Intelligent Recognition Systems (pp. 429-437). Springer International Publishing.

[Said, 21] Said, M., Shaheen, A.M., Ginidi, A.R., El-Sehiemy, R.A., Mahmoud, K., Lehtonen, M. and Darwish, M.M., 2021. Estimating parameters of photovoltaic models using accurate turbulent flow of water optimizer. Processes, 9(4), p.627.

[Saxena, 17] Saxena, A., Prasad, M., Gupta, A., Bharill, N., Patel, O.P., Tiwari, A., Er, M.J., Ding, W. and Lin, C.T., 2017. A review of clustering techniques and developments. Neurocomputing, 267, pp.664-681.

[Saxena, 17] Saxena, A., Prasad, M., Gupta, A., Bharill, N., Patel, O.P., Tiwari, A., Er, M.J., Ding, W. and Lin, C.T., 2017. A review of clustering techniques and developments. Neurocomputing, 267, pp.664-681.

[Senthilnath, 16] Senthilnath, J., Kulkarni, S., Benediktsson, J.A. and Yang, X.S., 2016. A novel approach for multispectral satellite image classification based on the bat algorithm. IEEE Geoscience and remote sensing letters, 13(4), pp.599-603.

[Sevillano, 14] Sevillano, X. and Alías, F., 2014. A one-shot domain-independent robust multimedia clustering methodology based on hybrid multimodal fusion. Multimedia tools and applications, 73, pp.1507-1543.

[Singh, 19] Singh, H. and Kumar, Y., 2019. Cellular automata based model for e-healthcare data analysis. International Journal of Information System Modeling and Design (IJISMD), 10(3), pp.1-18.

[Singh, 22] Singh, H. and Kumar, Y., 2022. An Enhanced Version of Cat Swarm Optimization Algorithm for Cluster Analysis. International Journal of Applied Metaheuristic Computing (IJAMC), 13(1), pp.1-25.

[Singh, 23] Singh, H., Rai, V., Kumar, N., Dadheech, P., Kotecha, K., Selvachandran, G. and Abraham, A., 2023. An enhanced whale optimization algorithm for clustering. Multimedia Tools and Applications, 82(3), pp.4599-4618.

[Suryanarayana, 22] Suryanarayana, G., Prakash K, L.N.C., Mahesh, P.S. and Bhaskar, T., 2022. Novel dynamic k-modes clustering of categorical and non categorical dataset with optimized genetic algorithm based feature selection. Multimedia Tools and Applications, 81(17), pp.24399-24418.

[Thakral, 24] Thakral, P., & Kumar, Y. (2024). An Improved Water Flow Optimizer for Data Clustering. SN Computer Science, 5(6), 715.

[Wang, 19] Wang, G.G., Deb, S. and Cui, Z., 2019. Monarch butterfly optimization. Neural computing and applications, 31, pp.1995-2014.

[Xu, 11] Xu, R. and Wunsch II, D.C., 2011. BARTMAP: A viable structure for biclustering. Neural Networks, 24(7), pp.709-716.

[Zhao, 05] Zhao, Y. and Karypis, G., 2005. Data clustering in life sciences. Molecular biotechnology, 31, pp.55-80.

[Zhao, 14] Zhao, M., Tang, H., Guo, J. and Sun, Y., 2014. Data clustering using particle swarm optimization. In Future Information Technology: FutureTech 2014 (pp. 607-612). Springer Berlin Heidelberg.

[Zhu, 22] Zhu, Q., Tang, X. and Elahi, A., 2022. Automatic clustering based on dynamic parameters harmony search optimization algorithm. Pattern Analysis and Applications, 25(4), pp.693-709.

[Zorarpacı, 23] Zorarpacı, E., 2023. Data clustering using leaders and followers optimization and differential evolution. Applied Soft Computing, 132, p.109838.