Duck Swarm Algorithm-Based Clustering Technique

No Author Given

No Institute Given

Abstract. Data clustering is an unsupervised task that aims to subdivide a set of unlabeled data into a number of homogeneous groups, it is used in several scientific fields such as bioinformatics, social sciences, psychology, chemistry, materials science, medicine and healthcare. A central challenge to data clustering is verifying all possible solutions to find the best one, which is beyond our capacities for small values of instances and clusters, not to mention that most of clustering applications come with quite bigger parameters. Thus, an effective technique is needed that can be employed with usual and large datasets. This work presents an adaptation of a recent metaheuristic called Duck Swarm Algorithm (DSA) in order to tackle data clustering problem. The adapted version (DSAC) is compared to different well-known and recent algorithms and tested on several real clustering datasets to reveal its performance. Experimental results exposed the superiority of the proposed DSAC in finding optimal clusters.

Keywords: Duck Swarm Algorithm · data clustering · optimization algorithm.

1 Introduction

Data clustering is an unsupervised technique used in many fields such as machine learning, and data analysis. The main objective of data clustering is to find a certain division of a dataset, where each partition contains data points sharing similar characteristics or more closer to each other than data points from different divisions [28]. To find the global best (clustering) solution, one should pass by all possible clusterings, which callsfor NP-Hard problems. The clustering of 60 data points into two classes gives more than $5.76 * 10^{17}$ possible divisions. This whole number of divisions cannot be verified in a satisfactory amount of time, a machine with the capacity of verifying a million solution per second will take more than 9000 years to verify the half of all possible solutions. To solve such problem researchers use some techniques that are efficient and effective such metaheuristics.

Metaheuristics are techniques inspired from real life and natural behavior of components and living being. These techniques mimic the observed behaviors in nature in order to solve optimization problems. Metaheuristics do not search the whole space of solutions (all possibilities), however, they just verify

some parts of this space to find solutions near to the best one in a reasonable amount of time. Metaheuristics can be categorized into four categories [20, 32, 29]: Human-based, swarm intelligence-based, chemical and physical-based, and evolutionary-based methods. Genetic algorithms (GA), tabu search (TS), particle swarm optimization (PSO) are some of well-known earliest metaheuristics. GA is a well-known metaheuristic, it was applied to a significantly large number of problems including data clustering, natural language processing, software engineering, scheduling, image processing [6, 10]. For instance, in [17, 12, 26, 25, 30, 31], GA was used to optimize clustering results. In [30], the authors combined GA with k-means algorithm to tackle the clustering problem in order to address the challenges of identifying the optimal number of clusters. The technique was tested on twenty datasets and compared with other five metaheuristics, the comparisons were carried out through six different indices. The experimental results revealed the superiority of the proposed technique over algorithms compared with. Regarding TS, it was firstly applied to data clustering by Al-Sultan [4], the algorithm was compared to k-means and simulated annealing (SA) based clustering algorithm where the computational experiences indicate the advantage of the approach. In [5], the authors extended the previous approach to solve fuzzy clustering. PSO is a swarm intelligence-based algorithm, it imitates the social behavior of fish schooling and birds flocking. In [27, 21, 13], the authors applied PSO to data clustering. For instance, in [27], the authors proposed two data clustering approaches using PSO, the two approaches were tested on six datasets (two artificial and four real) and compared to k-means, both approaches provide better results than k-means.

During the last decade, countless researches were conducted tackling data clustering problem with the help of metaheuristics. For instance, authors in [24] applied grey wolf optimizer (GWO) to find the best cluster centroids. The algorithm was evaluated on eight benchmark clustering datasets and applied to gene expression. The experiments showed The efficiency of the algorithm and its ability to find better results than the algorithms compared with. In another research [34], authors develop an enhanced version of the GWO for data clustering (EGWAC) where it was tested on various datasets and compared to the main version and other algorithms. The results indicate the abbility of EGWAC in finding better results than GWO and other algorithms. Aljarah et al. [9] introduced a hybrid version of GWO with TS (GWOTS) in order to well search in the neighborhood of the best solution. Authors tested GWOTS on thirteen data clustering benchmarks, the assessments carried out through SSE, purity, and entropy revealed that GWOTS can find better results than the algorithms compared with including GWO and TS. Authors in [23], proposed a novel optimization technique based on water wave optimization (WWO) to tackle the clustering problem. The efficacy of the algorithm was tested on thirteen datasets using accuracy and F-score. The results revealed that WWO can produce better results than the algorithms compared to. In another research [35], authors used the rat swarm optimizer (RSO) to optimize the quality of clustering algorithms. The algorithm was tested on several datasets and compared to other algorithms.

The results were carried out through six indices: error rate, homogeneity, completeness, v-measure, and purity. The results were promising and the approach proved remarkable efficacy in finding the most suitable cluster centroids and attaining better results compared to other algorithms. Deeb et al. [15], proposed an enhanced version of the black hole optimization algorithm (BH). This version proved its performance in data clustering, where it exposed an encouraging results.

The key contributions of this paper are:

- An adapted Duck swarm algorithm for data clustering was designed.
- A bunch of different clustering benchmark datasets are utilized to test the performance of the designed algorithm.
- The experiments indicated that the DSA technique can effectively handle data clustering problem and find optimized clusters.

The rest of this paper is organized as follows: a brief introduction to data clustering is conducted in section 2. Section 3 depicts the metaheuristic (Duck swarm algorithm), the original idea, the algorithm, and its parameters. In section 4, the adapted version of DSA for data clustering (DSAC) is explained. The experiments and results are presented and discussed in section 5. The final section (6) is conducted to the conclusion and future directions.

2 Background

Data clustering is the process of finding groups in a set of unlabeled data [22], in a such way where objects from different groups are less similar whereas objects from the same group are more similar or closer according to data characteristics. This similarity or distance can be measured by different functions such as Euclidean distance. This distance is among the well-utilized distance functions. It is defined for two data objects x, and y as:

$$dis_{Euc}\left(x,y\right)=\left(\sum_{i=1}^{F}(x_i-y_i)^2\right)^{\frac{1}{2}} \tag{1}$$

F here is the number of features, x_i is the value of the i^{th} feature of the data object x.

Another well-utilized distance function is its squared version:

$$d^2\left(x,y\right)=\sum_{i=1}^{F}(x_i-y_i)^2 \tag{2}$$

Generally, data clustering techniques can be subdivided into two categories, (i) partitional, and (ii) hierarchical methods. Hierarchical methods tend to create a hierarchy of clusters where in the lower stage each data point represents a cluster however in the upper stage all datapoints are considered as a single cluster. This

process can be done with two different techniques. The first start from the upper stage and goes to the bottom, this technique is referred as Divisive method. On the other hand, agglomerative method works reversely, starting from each data object as a single cluster and ending with the whole dataset as a cluster. At each stage in creating the hierarchy, a cluster is subdivided into two sub-clusters or two clusters are merged creating a bigger cluster. The partitional method in contrast, generates clusters without creating a hierarchy. Another categorization of data clustering methods is:(i) hard, and (ii) fuzzy clustering. In fuzzy clustering, a data object can be a member of more than one cluster with a variable indicating the ratio of its membership to each cluster. This work is based on partitional hard clustering. In hard clustering, each data should be a part of one and only one cluster, and a cluster should contain at least an object [16].

- $\forall i, j \in \{1, ..., k\}$ and $i \neq j, C_i \cap C_j = \emptyset$
- $\cup_{i=1}^{k} C_i = D$
- $\forall i \in \{1, ..., k\}, C_i \neq \emptyset$

There are several methods to evaluate clustering results, in general, they can be categorized into three categories: internal, external and relative criteria [18, 19]. Internal indices use information intrinsic to the data, they are used to measure the separation (clusters are distinguished and separated from each other) and compactness (data objects from the same cluster are closer to each other). There are significant numbers of internal indices such as Hartigan index, Davis and Buldin index, Dunn's index. External indices on the other hand, use external information, these external information may be the data classified by an expert or the actual classes of the dataset (ground truth). These measures asses the closeness of the clustering results to the external information. Rand Index, Entropy, Purity, V-measure are some examples of this category [8]. The relative criteria rely on comparing clustering results of the same algorithm yet with different parameter values [33, 19].

3 Duck swarm algorithm

3.1 Inspiration

Duck swarm algorithm (DSA) is a novel swarm intelligence optimization algorithm, it is inspired from ducks food foraging behaviors. Ducks are amphibious (terrestrial and aquatic) animals, the three main species of ducks are water ducks, diving ducks, and roosting ducks [36]. The well-known ducks belong to water ducks, they can be considered also as a bird.

3.2 Mathematical model and Duck swarm algorithm

There are three main processes to be modeled: ducks position after queuing (initialization of population), searching for food (exploration), and foraging in groups (exploitation). These processes are modeled as follows.

Initialization of population The population initialization is generated with random values inside the upper bound (the greatest value) and the lower bound (the lowest value) of the search space as follows:

$$X_i = L_b + (U_b - L_b) * o \tag{3}$$

X_i represents a duck (solution), U_b and L_b are, respectively, the upper and lower bounds. o is a random number between (0,1).

Exploration Ducks starts by following each other (queuing), and after getting to a place with more food, each duck spreads out to find something to eat, this behavior is modeled as follows:

$$X_i^{t+1} = \begin{cases} X_i^t + \mu.X_i^t.sign(r - 0.5) & \text{if } P > rand \\ X_i^t + CF_1.(X_{leader}^t - X_i^t) + CF_2.\left(X_j^t - X_i^t\right) & \text{else} \end{cases} \tag{4}$$

$sign(r-0.5)$ is a parameter that can take the values 1 or -1. μ is the global search parameter,P is the conversion probability, CF_2 and CF_1 coefficients represent respectively competition and cooperation between ducks defined by Eq.7. X_{leader}^t is the best solution so far. X_j^t is an agent (duck) around X_i^t in the t^{th} iteration. μ is calculated by:

$$\mu = K.(1 - t/t_{max}) \tag{5}$$

where:

$$K = \sin 2.rand + 1$$

Exploitation Ducks are satisfied when sufficient food is available, which represents the exploitation phase. This process is defined by:

$$X_i^{t+1} = \begin{cases} X_i^t + \mu.(X_{leader}^t - X_i^t) & \text{if } f(X_i^t) > f(X_i^{t+1}) \\ X_i^t + KF_1.(X_{leader}^t - X_i^t) + KF_2.\left(X_k^t - X_j^t\right) & \text{else} \end{cases} \tag{6}$$

Where μ is the same parameter defined in the exploration phase (Eq.5). KF_2 and KF_1 coefficients represent respectively the competition and cooperation between ducks in the exploitation phase which is defined by Eq.7. X_k^t and X_j^t are two distinguish agents around X_i^t. CF_1, CF_2, KF_1, and KF_2 are defined by:

$$CF_i or KF_i = \frac{1}{FP}.rand(0,1)|(i = 1,2) \tag{7}$$

where FP is a constant set to 0.618.

4 Duck swarm-based clustering algorithm

In this algorithm, DSA is used to find optimized clusters' centers. Thus, the solution is the cluster centers and each duck can be represented as:

$$D_i = \{C_1, C_2, ..., C_k\}$$

or, by replacing clusters' centers by their values as:

$$D_i = \{(o_{11}, o_{12}, ..., o_{1d},), (o_{21}, o_{22}, ..., o_{2d}), ..., (o_{k1}, o_{k2}, ..., o_{kd},)\}$$

where D_i is a solution, C_j is the center of the j^{th} cluster, o_{jl} is the feature number l of the cluster number j. k, and d are, respectively, the number of clusters and features.

The first step on duck swarm-based clustering algorithm (DSAC) is the initialization of population , where each duck from the population is initialized by k random points from the dataset which represent the clusters' centers. Then, the fitness of each duck is calculated after clustering the data using one iteration of k-means where the initial centers are the cluster centers of each duck, and the best one is taken as the leader X_{leader}.

The next step is the solutions optimization, this step is repeated a number of times (T_{max}) defined by the user. The value of parameters μ, CF_i, and KF_i are updated respectively by Eqs.(5,7). Then, in the exploration phase, each duck updates its position with the help of Eq.(4). If there is any duck out of the search range, it will be adapted. The data is then clustered by each duck and the fitness is calculated. Furthermore, The position of X_{leader} position is updated if there is a better solution. After that, in the exploitation phase, each duck updates its position using Eq.(6). For another time, ducks' positions are checked to confirm that each one is in the search range, additionally the data is clustered by the ducks and the new fitness of each solution is calculated. After that, the algorithm updates the position of X_{leader} if there are better solutions. The position that will be held in the next iteration of each ducks is the one with the best fitness value among the two positions that are calculated in the exploration and exploitation phases. The algorithm repeats this process (the solutions optimization), a certain number of times defined by the user.

Finally, the algorithm clusters the data using X_{leader} centers and returns it with the clustered data. Algorithm 1 depicts the proposed DSAC.

The fitness function used in DSAC is the sum of intra-cluster distances, it is calculated as follows:

$$f = \sum_{i=1}^{k} \sum_{x \in C_i} d^2(x, c_i) \tag{8}$$

Where k is the number of clusters, and c_i is the center of the cluster C_i. d^2 is the squared Euclidean distance (Eq. 2).

5 Performance

The main goal of this work is to apply DSA to data clustering and assess its performance through real-world clustering benchmark datasets in terms of the error rate index. For this assessment, five real datasets from UCI repository [11] were used. Table 1 presents the characteristics of the utilized datasets.

Algorithm 1 DSAC

Require: max number of iterations T, population size, number of clusters, and the dataset to be clustered
Ensure: The best clusters centers, its fitness value, and the clustered data
Initialize parameters $P, FP, \mu, CF_1, CF_2, KF_1, KF_2$
Initialize ducks position
Cluster data by each duck
Calculate fitness value f of ducks and select the best one X_{leader}
while $i \leq T$ **do**
  Update parameter values of $\mu, CF_1, CF_2, KF_1, KF_2$ using Eq.(5, 7)
  Update ducks positions using eq.(4) ▷ Exploration phase
  Cluster data by each duck
  Calculate fitness value f of ducks
  if there is better solutions than X_{leader} **then**
    Update the best duck position
  end if
  Update the new ducks positions using Eq.(6) ▷ Exploitation phase
  Cluster data by the new positions
  Calculate the new fitness value f_{new} of the swarm
  for each duck
  if $f_{new} < f$ **then**
    Update its position and fitness value
  end if
  end for
  if there is better solutions than X_{leader} **then**
    Update the best duck position
  end if
end while
Cluster the dataset using X_{leader}

Table 1. Used datasets.

Datasets	Number of instances	Number of features	Number of classes
Iris	150	4	3
Wine	178	13	3
Cancer	683	10	2
CMC	1473	9	3
Glass	214	9	6

Experimental Setup. The experiments were performed on a laptop with an Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz, 12.0 GB of RAM DDR4, using matlab R2021a under windows 10 Home (64 bits).

Since the results of other algorithms were taken directly from [3, 1] Parameters of algorithms compared with can be found in [3, 1]. Parameters of DSAC are the same as for H-HHO, max number of iterations is set to (1000). FP is

a constant set to 0.61 and P to 0.5 . Results are collected over 15 independent runs. The results of the experiments are depicted in Fig. (1,2) and Tables (2,3).

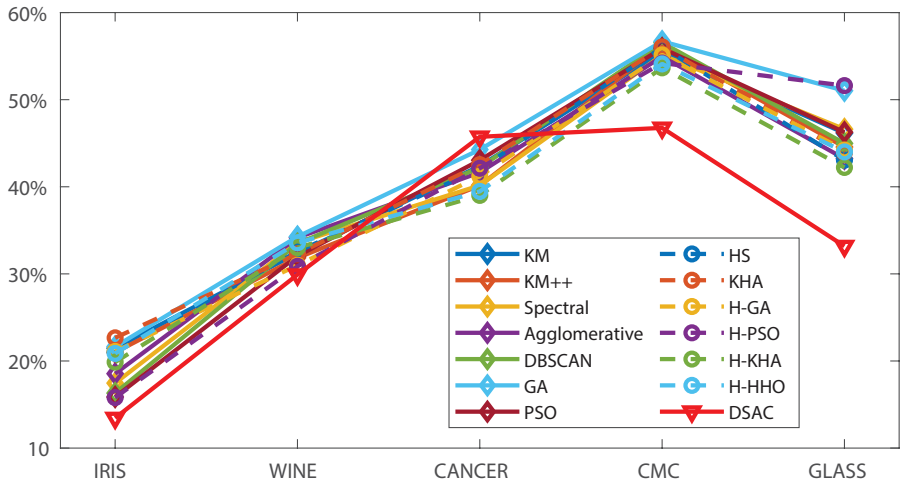

Fig. 1. Visual comparison of error rate results.

5.1 Discussion

In this comparison, DSAC was tested on five clustering benchmark datasets and compared to thirteen well-known and recent optimization algorithms namely: GA [26], PSO, K-means (KM), K-means++ (KM++), Spectral, Agglomerative [14], DBSCAN, Harmony Search (HS) [7], Krill Herd Algorithm (KHA) [2], Hybrid GA (H-GA), Hybrid PSO (H-PSO), H-KHA [3] and H-HHO [1]. The results were assessed through the error rate index. The proposed technique demonstrates its ability to find optimized clusters compared to other algorithms.

Table 2 showed that DSAC gave the best error rate results in all datasets but one (Cancer), where it unexpectedly showed the worst value and H-KHA took the first place followed by H-HHO, KM++, spectral, H-GA, agglomerative, HS, H-PSO, DBSCAN, KM, KHA, PSO, GA, and DSAC. Regarding the rest of datasets, DSAC outperformed all other algorithms showing the best results. In Iris, DSAC gave the least error rate value followed by H-PSO, PSO, DBSCAN, spectral, agglomerative, H-KHA, H-HHO, KM++, HS, H-GA, KM, GA, and finally KHA. The ranks for Wine dataset are as follows: the best result were showed by DSAC then, H-PSO, H-GA, KM++, PSO, KHA, KM, HS, H-KHA, DBSCAN, H-HHO, spectral, agglomerative, and GA. As expected in CMC, DSAC revealed the best error rate result, and after a significant gap comes H-KHA, then H-HHO, H-PSO, agglomerative, spectral, H-GA, KM, PSO, HS,

Table 2. Error Rate results.

	Criterion	Iris	Wine	Cancer	CMC	Glass	Rank
K-means	MEAN	21.467	32.388	42.388	55.470	46.154	
	BEST	10.660	29.775	39.865	54.660	42.262	12
	WORST	56.667	43.820	45.970	56.667	46.215	
KM++	MEAN	20.983	31.841	40.145	56.258	44.566	
	BEST	10.101	30.546	39.500	52.003	45.123	07
	WORST	54.274	43.534	44.965	57.001	45.250	
Spectral	MEAN	17.458	33.585	40.154	55.120	46.614	
	BEST	10.547	29.189	38.111	53.541	38.541	09
	WORST	55.541	43.137	44.685	54.044	51.991	
Agglomerative	MEAN	18.544	34.154	41.645	54.944	43.222	
	BEST	9.874	30.665	39.148	52.391	32.001	06
	WORST	48.397	42.688	46.699	57.487	52.140	
DBSCAN	MEAN	16.311	33.487	42.199	56.544	44.984	
	BEST	9.987	30.140	39.654	54.280	33.717	11
	WORST	43.111	42.009	44.021	56.654	51.123	
GA	MEAN	21.652	34.270	44.270	56.697	51.028	
	BEST	10.666	29.310	39.510	54.656	42.991	14
	WORST	43.333	47.753	47.753	57.296	56.075	
PSO	MEAN	15.867	32.051	43.051	55.899	46.262	
	BEST	10.667	29.775	40.775	54.101	43.925	10
	WORST	43.447	44.449	45.455	56.486	52.804	
HS	MEAN	21.054	32.568	42.054	56.001	43.054	
	BEST	10.509	29.865	40.111	55.430	41.162	08
	WORST	44.286	44.467	45.640	57.906	46.255	
KHA	MEAN	22.658	32.303	42.543	56.056	43.925	
	BEST	9.430	29.213	39.256	53.936	38.318	13
	WORST	42.548	47.191	47.191	56.999	50.476	
H-GA	MEAN	21.100	30.989	41.214	55.142	44.219	
	BEST	9.765	29.654	40.254	53.124	35.249	05
	WORST	44.667	44.001	46.214	56.214	51.985	
H-PSO	MEAN	15.800	30.871	42.125	54.204	51.617	
	BEST	9.666	29.775	39.775	53.201	41.589	03
	WORST	44.333	43.888	46.758	55.333	56.075	
H-KHA	MEAN	19.866	33.000	**39.012**	53.656	42.219	
	BEST	9.000	29.650	38.670	52.213	32.242	02
	WORST	43.333	42.134	44.154	54.333	51.420	
H-HHO	MEAN	20.866	33.564	39.470	54.109	44.002	
	BEST	9.332	29.653	39.119	53.165	34.242	03
	WORST	43.333	43.584	45.365	55.693	51.445	
DSAC	MEAN	**13.448**	**29.911**	45.737	**46.762**	**33.186**	
	BEST	11.409	26.179	45.737	46.025	30.929	**01**
	WORST	29.852	49.920	45.737	53.387	34.996	

Table 3. Ranks of algorithms.

	Iris	Wine	Cancer	CMC	Glass	Sum
k-means	12	7	10	8	10	47
km++	9	4	3	12	8	36
Spectral	5	12	4	6	12	39
Agglomerative	6	13	6	5	4	34
DBSCAN	4	10	9	13	9	45
GA	13	14	13	14	13	67
PSO	3	5	12	9	11	40
HS	10	8	7	10	3	38
KHA	14	6	11	11	5	47
H-GA	11	3	5	7	7	33
H-PSO	2	2	8	4	14	30
H-KHA	7	9	**1**	2	2	21
H-HHO	8	11	2	3	6	30
RSOC	**1**	**1**	14	**1**	**1**	**18**

KHA, KM++, DBSCAN, and GA. For Glass, DSAC ranked the first and after another significant gap comes H-KHA followed by HS, agglomerative, KHA, H-HHO, H-GA, KM++, DBSCAN, KM, PSO, spectral, GA, and unexpectedly, HPSO.

To recapitulate, Fig. (1,2) and Tables (2,3) indicate that DSAC ranked the first showing the best results in all datasets but Cancer, where it went to H-KHA. The unique handle of the two mechanisms exploration and exploitation at the same time during each step indicated the power of this method from the aforementioned results. H-KHA, which takes the second place in CMC, and Glass and the second global place. After H-KHA, H-HHO and H-PSO shared the third place, then come, H-GA, agglomerative, KM++, HS, spectral, PSO, DBSCAN, KM and KHA shared the twelfth place, and finally GA.

6 Conclusion

Prior to this work, the practical and performance of DSAC were validated through a well-used index (error rate), and the experiments revealed the ability of DSAC to find optimized clusters. This research contributes to the idea of tackling data clustering problem with metaheuristics. Even with the aforementioned results, this technique still falling on premature convergence as it is clear by examining the differences between the best, worst and mean values.

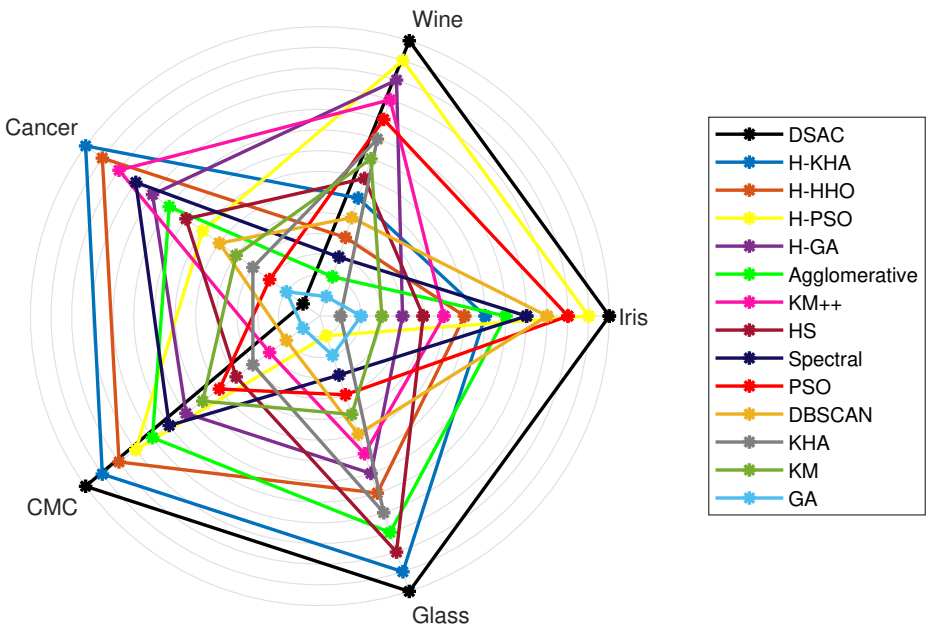

Fig. 2. Visual comparison of algorithms' ranks.

As future directions, this work will be extended and optimized by proposing an enhanced version of this idea or hybridizing this technique with other optimization algorithms trying to avoid premature convergence.

Acknowledgements Please place your acknowledgments at the end of the paper, preceded by an unnumbered run-in heading (i.e. 3rd-level heading).

References

1. Abualigah, L., Abd Elaziz, M., Shehab, M., Ahmad Alomari, O., Alshinwan, M., Alabool, H., Al-Arabiat, D.A.: Hybrid harris hawks optimization with differential evolution for data clustering. In: Metaheuristics in Machine Learning: Theory and Applications, pp. 267–299. Springer (2021)
2. Abualigah, L.M., Khader, A.T., AlBetar, M.A., Hanandeh, E.S.: A new hybridization strategy for krill herd algorithm and harmony search algorithm applied to improve the data clustering. In: First EAI international conference on computer science and engineering. pp. 54–63 (2017)
3. Abualigah, L.M., Khader, A.T., Hanandeh, E.S., Gandomi, A.H.: A novel hybridization strategy for krill herd algorithm applied to clustering techniques. Applied Soft Computing **60**, 423–435 (2017)
4. Al-Sultan, K.S.: A tabu search approach to the clustering problem. Pattern recognition **28**(9), 1443–1451 (1995)

5. Al-Sultan, K.S., Fedjki, C.A.: A tabu search-based algorithm for the fuzzy clustering problem. Pattern Recognition **30**(12), 2023–2030 (1997)
6. Alhijawi, B., Awajan, A.: Genetic algorithms: Theory, genetic operators, solutions, and applications. Evolutionary Intelligence pp. 1–12 (2023)
7. Alia, O.M., Al-Betar, M.A., Mandava, R., Khader, A.T.: Data clustering using harmony search algorithm. In: Swarm, Evolutionary, and Memetic Computing: Second International Conference, SEMCCO 2011, Visakhapatnam, Andhra Pradesh, India, December 19-21, 2011, Proceedings, Part II 2. pp. 79–88. Springer (2011)
8. Aljarah, I., Habib, M., Nujoom, R., Faris, H., Mirjalili, S.: A comprehensive review of evaluation and fitness measures for evolutionary data clustering. Evolutionary Data Clustering: Algorithms and Applications pp. 23–71 (2021)
9. Aljarah, I., Mafarja, M., Heidari, A.A., Faris, H., Mirjalili, S.: Clustering analysis using a novel locality-informed grey wolf-inspired clustering approach. Knowledge and Information Systems **62**, 507–539 (2020)
10. Bagirov, A.M., Karmitsa, N., Taheri, S.: Partitional clustering via nonsmooth optimization. Cham, Switzerland: Springer Nature (2020)
11. Blake, C.L.: Uci repository of machine learning databases. http://www. ics. uci. edu/˜ mlearn/MLRepository. html (1998)
12. Cowgill, M.C., Harvey, R.J., Watson, L.T.: A genetic algorithm approach to cluster analysis. Computers & Mathematics with Applications **37**(7), 99–108 (1999)
13. Cura, T.: A particle swarm optimization approach to clustering. Expert Systems with Applications **39**(1), 1582–1588 (2012)
14. Davidson, I., Ravi, S.: Agglomerative hierarchical clustering with constraints: Theoretical and empirical results. In: European Conference on Principles of Data Mining and Knowledge Discovery. pp. 59–70. Springer (2005)
15. Deeb, H., Sarangi, A., Mishra, D., Sarangi, S.K.: Improved black hole optimization algorithm for data clustering. Journal of King Saud University-Computer and Information Sciences **34**(8), 5020–5029 (2022)
16. Ezugwu, A.E., Ikotun, A.M., Oyelade, O.O., Abualigah, L., Agushaka, J.O., Eke, C.I., Akinyelu, A.A.: 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**, 104743 (2022)
17. Falkenauer, E.: Genetic algorithms and grouping problems. John Wiley & Sons, Inc. (1998)
18. Halkidi, M., Batistakis, Y., Vazirgiannis, M.: Cluster validity methods: part i. ACM Sigmod Record **31**(2), 40–45 (2002)
19. Halkidi, M., Batistakis, Y., Vazirgiannis, M.: Clustering validity checking methods: Part ii. ACM Sigmod Record **31**(3), 19–27 (2002)
20. Houssein, E.H., Mahdy, M.A., Shebl, D., Mohamed, W.M.: A survey of metaheuristic algorithms for solving optimization problems. In: Metaheuristics in machine learning: theory and applications, pp. 515–543. Springer (2021)
21. Kao, Y.T., Zahara, E., Kao, I.W.: A hybridized approach to data clustering. Expert Systems with Applications **34**(3), 1754–1762 (2008)
22. Kaufman, L., Rousseeuw, P.J.: Finding groups in data: an introduction to cluster analysis. John Wiley & Sons (2009)
23. Kaur, A., Kumar, Y.: A new metaheuristic algorithm based on water wave optimization for data clustering. Evolutionary Intelligence **15**(1), 759–783 (2022)
24. Kumar, V., Chhabra, J.K., Kumar, D.: Grey wolf algorithm-based clustering technique. Journal of Intelligent Systems **26**(1), 153–168 (2017)
25. Liu, Y., Wu, X., Shen, Y.: Automatic clustering using genetic algorithms. Applied mathematics and computation **218**(4), 1267–1279 (2011)

26. Maulik, U., Bandyopadhyay, S.: Genetic algorithm-based clustering technique. Pattern recognition **33**(9), 1455–1465 (2000)
27. Van der Merwe, D., Engelbrecht, A.P.: Data clustering using particle swarm optimization. In: The 2003 Congress on Evolutionary Computation, 2003. CEC'03. vol. 1, pp. 215–220. IEEE (2003)
28. Mirkin, B.: Clustering for data mining: a data recovery approach. Chapman and Hall/CRC (2005)
29. Naik, A., Satapathy, S.C.: Past present future: a new human-based algorithm for stochastic optimization. Soft Computing **25**, 12915–12976 (2021)
30. Rahman, M.A., Islam, M.Z.: A hybrid clustering technique combining a novel genetic algorithm with k-means. Knowledge-Based Systems **71**, 345–365 (2014)
31. Soundarya, V., Kanimozhi, U., Manjula, D.: Recommendation system for criminal behavioral analysis on social network using genetic weighted k-means clustering. J. Comput. **12**(3), 212–220 (2017)
32. SS, V.C., HS, A.: Nature inspired meta heuristic algorithms for optimization problems. Computing **104**(2), 251–269 (2022)
33. Theodoridis, S., Koutroumbas, K.: Pattern recognition. Elsevier (2006)
34. Zebiri, I., Zeghida, D., Mohamed, R.: Enhanced Grey Wolf Optimizer for Data Clustering, pp. 147–159 (03 2023). https://doi.org/10.1007/978-3-031-28540-0_12
35. Zebiri, I., Zeghida, D., Redjimi, M.: Rat swarm optimizer for data clustering. Jordanian Journal of Computers and Information Technology **8**(3) (2022)
36. Zhang, M., Wen, G.: Duck swarm algorithm: theory, numerical optimization, and applications. Cluster Computing **27**(5), 6441–6469 (2024)