POLİTEKNİK DERGİSİ

JOURNAL of POLYTECHNIC

ISSN: 1302-0900 (PRINT), ISSN: 2147-9429 (ONLINE)
URL: http://dergipark.org.tr/politeknik

The adaptation of gray wolf optimizer to data clustering

Bozkurt optimizasyon yönteminin veri kümelemeye uyarlanması

Yazar(lar) (Author(s))*: Adem TEKEREK[1], Murat DÖRTERLER[2]*

ORCID[1]: 0000-0002-0880-7955

ORCID[2]: 0000-0003-1127-515X

Bu makaleye şu şekilde atıfta bulunabilirsiniz(To cite to this article): Tekerek A., Dörterler M., "The adaptation of gray wolf optimizer to data clustering", ***Politeknik Dergisi***, 25(4): 1761-1767, (2022).

Erişim linki (To link to this article): http://dergipark.org.tr/politeknik/archive

DOI: 10.2339/politeknik.778630

The Adaptation of Gray Wolf Optimizer to Data Clustering

Highlights

- *Gray Wolf Optimizer for clustering problem*
- *Meta-heuristic optimization for data mining*
- *Nature-inspired evolutionary algorithm*

Graphical Abstract

Gray Wolf Optimizer (GWO) is one of the nature-inspired evolutionary algorithm simulating the hunting of gray wolves. GWO has applied to solve several optimization issues in different fields. In this study GWO was examined in the case of data clustering. GWO was modified to get better clustering results and applied to well-known benchmark Iris, Wine, Glass, Cancer, Vowel, CMC datasets. The performance of GWO is compared the other K-means, PSO, GSA, BH and BB-BC algorithms used as clustering. The results show that GWO can be used for data clustering successfully.

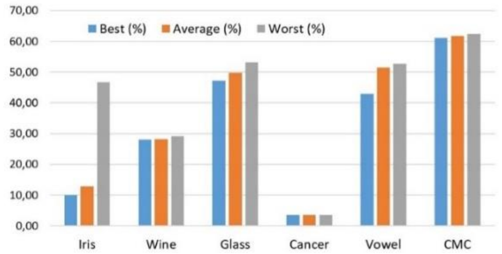

Figure The best error rates of GWO on the test datasets

Aim

The aim of this study is investigation of the capabilities of the Gray Wolf Optimization on the solving of clustering problems.

Design & Methodology

GWO is brought about to solve engineering optimization problems, thus the structure of the algorithm was adapted to solve clustering problems. Solutions are denoted as a vector made of floating point numbers.

Originality

In the study, it has been proved that GWO, one of the nature-inspired methods, can be used in the solution of data mining clustering problems.

Findings

GWO is suitable for applying to the data clustering problem successfully in spite of the few neglectable negative factors in result of intra cluster distances.

Conclusion

GWO is capable of finding out the best known solutions to the best-known solution in the literature. GWO tends to trap in local minimum solutions for complex datasets. Also, the performance of GWO gets lower as the length of coded solution increases. The optimizer can be benefited as a data cluster method in data science.

Declaration of Ethical Standards

The author(s) of this article declare that the materials and methods used in this study do not require ethical committee permission and/or legal-special permission.

The Adaptation of Gray Wolf Optimizer to Data Clustering

Araştırma Makalesi / Research Article

Adem TEKEREK[1*], Murat DÖRTERLER[1,2]

[1]Technology Faculty, Computer Engineering Department, Gazi University, Ankara, Turkiye
[2]Turkish Military Acamemy, Computer Engineering Department, National Defence University, Ankara, Turkiye

(Geliş/Received : 11.08.2020; Kabul/Accepted : 29.11.2020 ; Erken Görünüm/Early View :15.12.2020)

ABSTRACT

Data Clustering stands for a group of methods classifying patterns into groups and retrieving similarities or dissimilarities of a collection of objects. Clustering is used for pattern recognition, machine learning, etc. One of the approaches to clustering is optimization. The aim of the optimization is finding the best solution in the search space of a problem as much as possible. Many optimization methods were modified to solve clustering problems in literature. Gray Wolf Optimizer (GWO) is one of the nature-inspired meta-heuristic algorithms simulating the hunting of gray wolves. GWO has applied to solve several optimization issues in different fields. In this study, GWO was examined in the case of data clustering. GWO was modified to get better clustering results and applied to well-known benchmark data sets. The performance of GWO was compared to the other algorithms used as clustering. The results show that GWO can be used for data clustering successfully.

Keywords : Data clustering, meta-heuristic optimization, gray wolf optimizer, data mining.

Bozkurt Optimizasyon Yönteminin Veri Kümelemeye Uyarlanması

ÖZ

Veri Kümeleme, veri desenlerini gruplar halinde sınıflandıran ve bir nesne benzerliklerini veya farklılıklarını ayrıştıran bir yöntemlerdir. Kümeleme, örüntü tanıma, makine öğrenimi vb. için kullanılır. Veri Kümelemeye yönelik yaklaşımlardan biri de optimizasyondur. Optimizasyonun amacı, bir problemin arama alanında mümkün olan en iyi çözümün bulunmasıdır. Literatürdeki kümeleme problemlerini çözmek için birçok optimizasyon yöntemi uyarlanmıştır. Bozkurt Optimizasyonu (BO), boz kurtların avlanmasını simüle eden doğadan ilham alan sezgi ötesi algoritmalardan biridir. BO, farklı alanlardaki çeşitli optimizasyon sorunlarına başarılı çözüm üretmektedir. Bu çalışmada BO, veri kümeleme için incelenmiştir. BO, daha iyi kümeleme sonuçları elde etmek için değiştirilerek, iyi bilinen veri kümelerine kıyaslama amacıyla uygulanmıştır. BO'nun performansı, kümeleme olarak kullanılan diğer algoritmalarla karşılaştırılmıştır. Sonuçlar, BO'nun veri kümeleme için başarıyla kullanılabileceğini göstermektedir.

Anahtar Kelimeler: Veri kümeleme, meta-sezgisel optimizasyon, bozkurt kurt optimizasyonu, veri madenciliği.

1. INTRODUCTION

Data clustering, grouping of a set of data, is one of the most significant methods for data analytics. It executes a process to separate the data according to the similarities and dissimilarities. [1-3]. Clustering has been applied to problems in a variety of areas, including exploratory data mining [4], image processing [5,6], disease diagnostic [7], astronomy [8], genetic [9] and, mathematical programming [10], etc.

Data clustering approaches can be grouped into two types. The first one named supervised technique uses an external trainer indicating the target class to which a data vector should belong. The other one named unsupervised clustering does not have a trainer. Data vectors are grouped by distance from each other in unsupervised clustering. The distance is utilized to figure out similarities between data objects in this technique. The clustering is defined as giving *N* objects and assigning every object to one of *K* clusters. It is aimed to minimize the result of squared Euclidean distances between every data object and the centroid of the cluster that belongs to all allocated data object:

$$F(O,Z) = \sum_{i=1}^{N}\sum_{j=1}^{K} W_{ij}\left\|O_i - Z_j\right\|^2 \qquad (1)$$

Where $\|O_i - Z_j\|$ is the Euclidean distance between the cluster center Z and a data object O_i. W_{ij} indicates whether O_i is assigned to cluster *j* or not. If the object is assigned, the value takes *1*, otherwise *0*. W_{ij} can take values in the interval between *[0, 1]* in fuzzy clustering [11].

In order to solve the clustering problems, many heuristic approaches have been implemented. Data clustering algorithms are mostly divided as hierarchical structure and partitioned techniques [2,3,12]. For example, K-means is a famed clustering algorithm due to its performance and simplicity [2,3]. Furthermore, Black Hole (BH) algorithm [11], tabu search optimization [13],

*Sorumlu Yazar (Corresponding Author)
e-posta : atekerek@gazi.edu.tr

genetic algorithm optimization [14,15,16,17], ant colony optimization [18,19,20], honey bee optimization [21], Particle Swarm Optimization (PSO) [22,23,24], bee colony algorithm [25], Gravitational Search Algorithm (GSA) [26, 27], a binary search algorithm [28] and Big Bang Big Crunch (BB-BC) algorithm [29] were used to solve data clustering problems. In computer science, many optimization algorithms have been developed by inspiring with living creatures in nature to find out the optimum solution among all feasible solutions. The nature-inspired optimization algorithms are being used in numerous research areas such as computer science [30,31], data mining [32, 33], industry [34], agriculture [35], medicine [36], economy [37], and engineering [38].

GWO is a comparably novel nature-inspired optimization approach. It is applied to many optimization issues in different areas successfully. The GWO was developed based on the gray wolf behavior that mimics the leadership hierarchy and hunting mechanism in wildlife [39]. GWO is a meta-heuristic optimizer developed to solve the restricted continuous optimization problems such as engineering design problems. GWO and its modifications were applied to various problems in different fields successfully. In this study, GWO has been applied to clustering problems for the first time. Authors investigated the capability of GWO on solving clustering problems which are out of the target scope of GWO.

In this study, GWO was modified to solve clustering issues and applied to Wine [40], Iris [41], Wisconsin Breast Cancer (WBC) [42], Vowel [43], Glass [44] and Contraceptive Method Choice (CMC) [45] well-known data sets in literature. The clustering performance of the GWO on these datasets were compared with K-means, PSO, GSA, BH and BB-BC. PSO was inspired from behaviors of the swarms such as bird or fish swarms training in nature [46]. The GSA was developed based on the notion of mass interactions and the law of gravity [47]. The BB–BC optimizer is based on one of the theories of the evolution of the universe. It is composed of the BB–BC phases [48]. BH is inspired by the black hole phenomenon. According to the experimental studies, the GWO algorithm can be applied to data clustering issues successfully. In the study, Matlab was used as an application development environment for the cluster analysis.

The organization of the study is as follows: In Section 2, explanation of the GWO is detailed. In Section 3, proposed GWO and its adaptation for cluster applications is introduced. The experimental results of the optimizers applied to the benchmark problems are given in Section 4. In section 5, the conclusion of the study is presented.

2. GWO FOR CLUSTERING

The GWO, is a population based meta-heuristic method and inspired by the communal life of gray wolves [39]. The gray wolves have a strict hierarchical structure in their population. A wolf in a pack is assigned to one of the four ranks named as Alpha, Beta, Delta and Omega from top to bottom, respectively. The wolf with the rank of alpha is the leader of the pack and takes decisions and gives orders to the others. Beta, who is the deputy of the Alpha, both advises the alpha and organizes the pack. The hierarchical position of the Deltas is between Betas and Omegas. The wolves, with rank of delta take the role of scout, sentinel, elder, hunter, and caretaker. Omega, which is the lowest rank in the pack, corresponds to the rank of the wolves excepting for the top three ranks. Though omegas are considered as trivial, their absence leads to major problems in pack for daily routines [40-50].

GWO is especially inspired by the hunting strategies of the gray wolves shaped by the hierarchical contexture. The hunting is performed in three stages below:

- Follow, approach and catch the prey,
- Hunt, surround and harassment until prey is motionless,
- Attacking towards prey,

Mirjali et al. [39] brought about the GWO algorithm and figured out the mathematical model of the hunting strategies of the grey wolves. They applied the algorithm to well known engineering optimization problems successfully. GWO is typically a population based meta-heuristic method. Individuals correspond to the wolves while population corresponds to the pack, the individuals with the top three fitness values are considered to be the Alpha (α), Beta (β) and Delta (δ), consecutively. The other individuals in the pack are assumed to be Omega (ω). Moreover, the prey stands for the optimum solution and hunting area corresponds to the search space.

The wolves in pack surround the prey during the hunting. Their locations are updated on each iteration according to Equations (2) and (3).

$$\vec{\mathrm{D}} = |\vec{\mathrm{C}}.\overrightarrow{X_{\mathrm{P}}}(\mathrm{t}) - \vec{\mathrm{X}}(t) \quad (2)$$

$$\vec{\mathrm{X}}(\mathrm{t}+1) = \overrightarrow{X_{\mathrm{P}}}(\mathrm{t}) - \vec{\mathrm{A}} \quad (2)$$

In the Equations (2) and (3), $\vec{\mathrm{D}}$ represents the distance vector between the prey and the wolves. t stands for the current iteration, $\vec{\mathrm{A}}$ and $\vec{\mathrm{C}}$ the coefficient vectors, $\overrightarrow{X_{\mathrm{P}}}$ the location vector of the prey, and $\vec{\mathrm{X}}$ indicate the locations of the individuals. $\vec{\mathrm{A}}$ and $\vec{\mathrm{C}}$ are calculated as the Equations (4) and (5):

$$\vec{\mathrm{A}} = 2\vec{\mathrm{a}}.\overrightarrow{\mathrm{r_1}} - \vec{\mathrm{a}} \quad (4)$$

$$\mathrm{a} = 2 - \mathrm{t} * 2/\mathrm{T} \quad (5)$$

$$\vec{\mathrm{C}} = 2.\overrightarrow{\mathrm{r_2}} \quad (6)$$

Where, the vectors $\overrightarrow{\mathrm{r_1}}, \overrightarrow{\mathrm{r_2}}$ are changed in the range of *[0,1]* randomly at each iteration. T is the number of iterations and $\vec{\mathrm{a}}$ decreases from *2* to *0* linearly during the iterations (4). $\vec{\mathrm{A}}$ represents the moving of individual and take value between *[-1,1]*. They move away from the prey in case of $|A| > 1|$ and closing in case of $|A| < 1$. $\vec{\mathrm{C}}$ represents the weight of the location of the prey in the calculation of the $\vec{D}$ in Equation (2).

The location of the optimum solution is unknown in unphysical and multi-dimensional search space in comparison with real life. Therefore, it is utilized from the closest solutions in the population (Figure 1). Equation (3) is rearranged in terms of the locations of Alpha, Beta and Delta. Therefore, the locations of individuals are updated as per the Equation (13) at each iteration.

$$\overrightarrow{D_\alpha} = |\overrightarrow{C_1}.\overrightarrow{X_\alpha} - \vec{X}| \quad (7)$$

$$\overrightarrow{D_\beta} = |\overrightarrow{C_2}.\overrightarrow{X_\beta} - \vec{X}| \quad (8)$$

$$\overrightarrow{D_\delta} = |\overrightarrow{C_3}.\overrightarrow{X_\delta} - \vec{X}| \quad (9)$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1}.(\overrightarrow{D_\alpha}) \quad (10)$$

$$\overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_2}.(\overrightarrow{D_\beta}) \quad (11)$$

$$\overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_3}.(\overrightarrow{D_\delta}) \quad (12)$$

$$\vec{X}(t+1) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3} \quad (13)$$

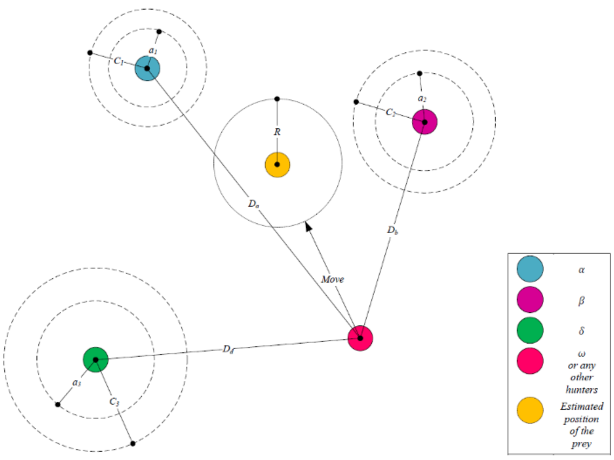

Figure 1. Position shift of a gray wolf for 2D search space in GWO [39]

Algorithm 1 GWO Algorithm[39].

Create an initial population $X_i = (i=1, 2, \ldots, n)$
Initialize the coefficients a, A, and C
Calculate the fitness values of each search agent
X_α = the best individual
X_β = the second individual
X_δ = the third individual
***while** (t < Max number of iterations)*
  ***for** each individual*
    Update the position of the current individual by equation (12)
  end for
  Update the coefficients a, A, and C
  Calculate the fitness of all individuals
  Update X_α, X_β, and X_δ
  t=t+ 1
end while
return X_α

GWO is brought about to solve engineering optimization problems, thus the structure of the algorithm was adapted to solve clustering problems. Structure of solutions was coded according to the clustering problems in question at the initial stage. Solutions are denoted as a vector made of floating point numbers. The vectors consist of the centers of the clusters; $Z = \{Z_1, Z_2, \ldots, Z_j\}$ if j is the number of clusters. For each $j=1,\ldots,j$, the Z_j, is also a vector denoting values of the center of a cluster; $Z_j = \{z_{j1}, z_{j2}, \ldots, z_{jf}\}$ where f is the number of features for the problem handled. Thus, the length of a solution equals to $j * f$ (Figure 2). Though each value in a structure stands for a design variable in original GWO; each value corresponds to a feature value of a center of clusters.

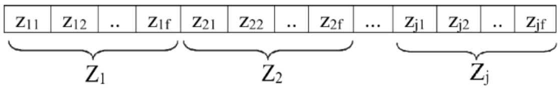

Figure 2. The structure of a solution.

Another modification was made to get better results for clustering problems. The search area shrinks during the iterations depending on the value of a. The value of $\vec{a}$ decreases according to Equation (14) rather than the Equation (5). It was aimed with this modification that GWO can converge to optimum solution faster and make much more search iteration around the best solution. Thus, the algorithm can get better results by the sensitive searching. The chart of the modified version of the "*a*" value during the iterations is given in (figure 3.a and figure 3.b).

$$a = -log((t+4)/T)/2 \quad (14)$$

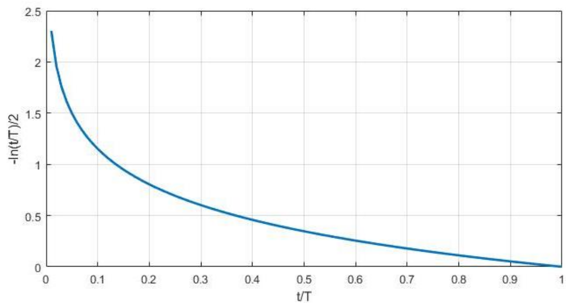

Figure 3.a. The chart of the modified version of the "*a*" value during the iterations

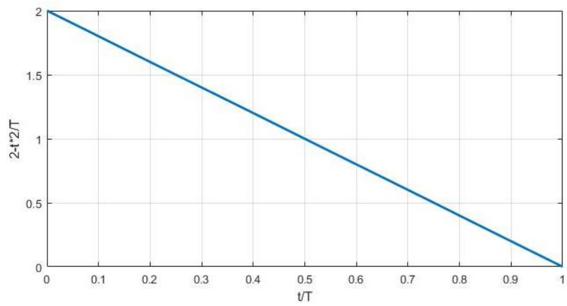

Figure 3.b. The chart of the original version of the "*a*" value during the iterations

3. EXPERIMENTAL STUDIES

GWO was applied to the six datasets benefited in the similar studies in literature frequently to evaluate the performance of GWO in clustering issues. The datasets have different levels of complexity. The datasets are available, as named Wine, Iris, WBC, Vowel, Glass and CMC, in the public repository of the Machine Learning Database. The features of the datasets are presented in Table 1. Evaluation studies were conducted relying on two metrics, these are the result of intra-cluster distances as an internal quality and error rate (ER) as an external quality. The proposed GWO algorithm was simulated 50 times on the test datasets in the evaluation process.

In the first stage, the performance evaluation was conducted in terms of the result of intra-cluster distances as an internal quality measure. The metric is computed by summing up the distance between each data object and the centroid of its cluster corresponding, as defined in Equation (1). Also, Equation (2) is used as the fitness function of the proposed GWO. Therefore, the best solution is regarded as the one with the smallest value of the result of intra-cluster distances.

The proposed GWO was applied to the data sets in 200, 500, 1000, 2000 and 5000 iterations to evaluate the sum of intra-cluster distance performance. As given results in Table 2, the performance is getting better while the number of iterations is increasing. This result is caused due to the fact that GWO searched the search space more comprehensively with little intervals of the "*a*" value in Equation (14).

Table 1. Specifications of the benchmark datasets.

Datasets	Features	Clusters	Length of solution	Data objects
Iris	4	3	12	150 (50,50,50)
Wine	13	3	36	178 (59, 71,48)
Glass	9	6	54	214 (70, 76,17, 13, 9,29)
Cancer	9	2	18	683 (444,239)
Vowel	3	6	18	871 (72, 89,172,151,207,180)
CMC	9	3	27	1473 (629,334,510)

Table 2. The best values of GWO in the sum of intra-cluster distances for different iterations.

	Iterations				
Dataset	200	500	1000	2000	5000
Iris	96.65642	96.65562	96.65553	96.65549	96.65549855
Wine	16,306.14	16,301.05	16,301.20	16,299.54	16,299.71
Glass	284.0556	275.2466	254.5954	242,6367	239,1630
Cancer	2,964.388	2,964.387	2,964.387	2,964.387	2,964.38697
Vowel	148,985.9	148,968.8	148,968.8	148,967.3	148,967.27
CMC	5,550.572	5,545.597	5,536.514	5,534.756	5,533.6491

ER is the rate of the data objects assigned to a wrong cluster to all data objects. The value is figured out by the Equation (15). Statistical evaluation of ER performance values is presented in Figure 4 and Table 3. According to the results in Figure 4, there are different results between the best and the worst values except the cancer dataset. This situation affects the values of the standard deviation and the average adversely (Table 3). Nevertheless, the low standard deviation values indicate the stability of the GWO on the data clustering in ER.

$$ER = \frac{number\ of\ wrong\ assigned\ objects}{total\ number\ of\ objects\ within\ dataset} \times 100 \quad (15)$$

Table 3. The standard deviation rates of GWO on the benchmark datasets

Iris	Wine	Glass	Cancer	Vowel	CMC
0.1005	0.0019	0.0159	0	0.0168	0.0034

The efficiency of the GWO is also compared to known algorithms applied to the same datasets in the literature, such as PSO [39], K-means [3], GSA [26] and the BB–BC algorithm [29]. The comparison results are presented in Table 4. GWO is capable of clustering successfully as well as other algorithms. Furthermore, GWO can find out the best known solutions for some datasets. Yet, the local minimum issue is also seen within performance values of GWO. Thus, relative performance loss is occurring in terms of standard deviation and average values.

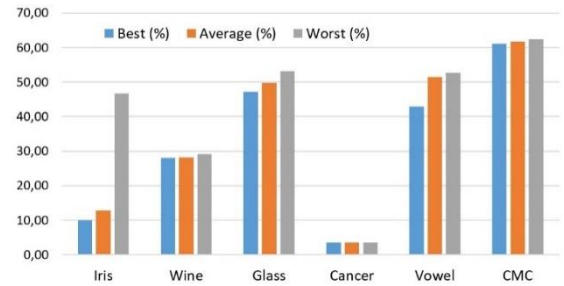

Figure 4. The best error rates of GWO on the test datasets

Another outcome is that the solution length of the handed dataset is related to the clustering performance of GWO. While the length of the solution is increased, the performance of the GWO is decreasing.

Experimental results show that GWO is suitable for applying to the data clustering problem successfully in spite of the few neglectable negative factors in the result of intra cluster distances. Though the result of intra cluster distances is one of the performance metrics, ER is a more important indicator to evaluate the performance of the method. ER shows the rate of the instances, assigned to the wrong class. So, ER is related to the aim of data clustering directly. So it is indicated that GWO is suitable for data clustering.

Table 4. The result of intra cluster distances scores of the optimizers for the six datasets.

Datasets	Criteria	K-means*	PSO*	GSA*	BB-BC*	BH*	GWO
Iris	Best	97.32592	96.87935	96.68794	96.67648	96.65589	**96.65549**
	Average	105.72902	98.14236	96.73105	96.76537	96.65681	98.58327
	Worst	128.40420	99.76952	97.42865	97.42865	96.66306	120.7324
	Std	12.38759	0.84207	0.20456	0.20456	0.00173	6.597108
Wine	Best	16,555.67942	16,304.48576	16,298.67356	16,298.67356	16,293.41995	16,299.71
	Average	16,963.04499	16,316.27450	16,303.41207	16,303.41207	16,294.31763	16,308.29
	Worst	23,755.04949	16,342.78109	16,310.11354	16,310.11354	16,300.22613	16,365.49
	Std	1180.69420	12.60275	2.66198	2.66198	1.65127	9,470148
Glass	Best	215.67753	223.90546	223.89410	223.89410	210.51549	239.1630
	Average	227.97785	230.49328	231.23058	231.23058	211.49860	276.4556
	Worst	260.83849	246.08915	243.20883	243.20883	213.95689	314.4163
	Std	14.13889	4.79320	4.65013	4.65013	1.18230	17.52596
Cancer	Best	2986.96134	2974.48092	2964.38753	2964.38753	2964.38878	**2,964.38697**
	Average	3032.24781	2981.78653	2964.38798	2964.38798	2964.39539	2,964.387
	Worst	5216.08949	3053.49132	2964.38902	2964.38902	2964.45074	2,964.387
	Std	315.14560	10.43651	0.00048	0.00048	0.00921	1.983873
Vowel	Best	149,394.80398	152,461.56473	149,038.51683	149,038.51683	148,985.61373	**148,967.27**
	Average	153,660.80712	153,218.23418	151,010.03392	151,010.03392	149,848.18144	149,011.93
	Worst	168,474.26593	158,987.08231	153,090.44077	153,090.44077	153,058.98663	153,053.6
	Std	4123.04203	2945.23167	1859.32353	1859.32353	1306.95375	5.905301
CMC	Best	5542.18214	5539.17452	5534.09483	5534.09483	5532.88323	5,533.6491
	Average	5543.42344	5547.89320	5574.75174	5574.75174	5533.63122	5,642.834
	Worst	5545.33338	5561.65492	5644.70264	5644.70264	5534.77738	5,890.324
	Std	1.52384	7.35617	39.43494	39.43494	0.59940	8.217255

* These values were obtained from [11]

The best centroid values obtained through GWO on the benchmark datasets are shown from Table 5 to Table10. The best centroid values by GWO are given to confirm the result of intra-cluster distances in Table 4. The best values given in Table 4 can be figured out by matching the data objects to the closest centroids in Table 5-10 corresponding to each dataset.

Table 5. The best centroid values by the GWO on Iris

Centroid 1	Centroid 2	Centroid 3
6.73334398	5.01215680	5.93429679
3.06782007	3.40311599	2.79781223
5.63005805	1.47165067	4.41790502
2.10675929	0.23590453	1.41722526

Table 6. The best centroids values by the GWO on Cancer

Centroid 1	Centroid 2
2.88928946	7.11712971
1.12779310	6.64109914
1.20064472	6.62547440
1.16413571	5.61431300
1.99338427	5.24077130
1.12120833	8.10099069
2.00545249	6.07815154
1.10130729	6.02183011
1.03163940	2.32573144

Table 7. The best centroid values by the GWO on Wine

Centroid 1	Centroid 2	Centroid 3
12.80658372	13.75843441	12.54180067
1.96508144	3.37192665	3.06934434
2.34493014	2.72629556	1.55715722
19.49626401	16.89807027	21.30135468
98.93607532	105.25348871	92.51565716
1.35523503	1.89796866	1.38748617
1.71259028	1.74577570	0.62248771
0.19946529	0.55089370	0.17843652
1.50235910	2.03359464	0.73796214
5.48311739	4.61103946	4.14280891
0.56750425	1.47572125	0.51701115
2.99125610	2.29850551	1.67319456
686.97619901	1137.35875577	463.62273635

Table 8. The best centroid values by the GWO on CMC

Centroid 1	Centroid 2	Centroid 3
24.41838394	33.49378043	43.63946211
3.04268594	3.13382643	3.00003085
3.51307004	3.55270109	3.45329002
1.79184917	3.64505059	4.58234918
0.92762447	0.79115542	0.78125578
0.79696551	0.65314008	0.72949883
2.30230870	2.10101650	1.82408931
2.97220900	3.28699977	3.43221127
0.04529515	0.00000000	0.22780963

Table 9. The best centroid values by the GWO on Glass

Centroid 1	Centroid 2	Centroid 3	Centroid 4	Centroid 5	Centroid 6
1.52282348	1.52368061	1.51778986	1.51569913	1.52687543	1.53066913
13.11785923	14.67676849	13.18392736	13.82924814	13.76948684	11.80741490
3.53090954	0.03918885	0.15427764	3.17324394	1.16623307	1.66755480
1.36036679	2.16393150	1.27414275	0.43529372	1.47752216	0.84549628
72.81365330	73.25622821	72.94790833	71.81929700	71.54959457	71.89547215
0.48727383	0.06002588	1.25315407	0.44588649	1.64233565	0.26554644
8.39808721	8.74141930	11.40378743	9.69922309	5.88760101	14.95504240
0.16335778	0.83393072	0.08103364	0.12688376	0.78324327	0.91370209
0.03829633	0.00000000	0.10766524	0.40252224	0.23447034	0.09689924

Table 10. The best centroid values by the GWO on Vowel

Centroid 1	Centroid 2	Centroid 3	Centroid 4	Centroid 5	Centroid 6
407.96094162	623.86795143	357.48259176	439.26145682	375.54852609	506.91553722
1018.0765515	1309.6438279	2291.3751102	987.67131324	2149.3836590	1839.6873301
2317.8152581	2333.4010569	2977.4118058	2665.42488447	2678.42068007	2556.19805340

4. CONCLUSION

The capabilities of the GWO on the solving of clustering problems are investigated in this study. Thus the GWO can be applied to clustering problems. The optimizer is capable of finding out the best known solutions or the closest solutions to the best-known solution in the literature. On the other hand, GWO needs a few impartments for better performance in clustering. GWO tends to trap in local minimum solutions for complex datasets. Also, the performance of GWO gets lower as the code length of solutions increases. The optimizer can be benefited as a data cluster method by the researchers and the analyzers in data science. In future studies, improvements to overcome the local minimum and the length of solution issues of the GWO can be made.

DECLARATION OF ETHICAL STANDARDS

The author(s) of this article declare that the materials and methods used in this study do not require ethical committee permission and/or legal-special permission.

AUTHORS' CONTRIBUTIONS

Adem TEKEREK: All procedures for the article were carried out with the equal contribution of the authors.

Murat DÖRTERLER: All procedures for the article were carried out with the equal contribution of the authors.

CONFLICT OF INTEREST

There is no conflict of interest in this study.

REFERENCES

[1] Barbakh, W., Wu, Y., Fyfe, C., "Review of clustering algorithms", Non-Standard Parameter Adaptation for Exploratory Data Analysis, Springer, ***Berlin Heidelberg***, 7–28, (2009).

[2] Han, J., Kamber, M., "Data Mining: Concepts and Techniques", ***Academic Press,*** (2006).

[3] Jain, A.K., "Data clustering: 50 years beyond K-means", ***Pattern Recognition Letters*** 31: 651–666, (2010).

[4] Evangelou, I. E., Hadjimitsis, D. G., Lazakidou, A. A., Clayton, C., "Data Mining and Knowledge Discovery in Complex Image Data using Artificial Neural Networks", ***Workshop on Complex Reasoning an Geographical Data***, Cyprus, (2001).

[5] Andrews, H. C., "Introduction to Mathematical Techniques in Pattern Recognition", ***John Wiley & Sons***, New York, (1972).

[6] Topaloglu, N., "Revised: Finger print classification based on gray-level fuzzy clustering co-occurrence matrix", ***Energy Education Science and Technology Part A: Energy Science and Research***, 31(3): 1307-1316, (2013).

[7] Sakar, B. E., Isenkul, M. E., Sakar, C. O., Sertbas, A., Gurgen, F., Delil, S., Kursun, O., "Collection and analysis of a Parkinson speech dataset with multiple types of sound recordings". ***IEEE Journal of Biomedical and Health Informatics***, 17(4): 828-834, (2013).

[8] Mo, H. J., & White, S. D., "An analytic model for the spatial clustering of dark matter haloes", ***Monthly Notices of the Royal Astronomical Society***, 282(2): 347-361, (1996).

[9] Yeung, K. Y., Haynor, D. R., Ruzzo, W. L., "Validating clustering for gene expression data", ***Bioinformatics***, 17(4): 309-318, (2001).

[10] Rao, M. R., "Cluster Analysis and Mathematical Programming", ***Journal of the American Statistical Association,*** 22: 622-626, (1971).

[11] Hatamlou, A., "Black hole: A new heuristic optimization approach for data clustering", ***Information sciences***, 222: 175-184, (2013).

[12] Jain , A.K., Murty, M.N., Flynn, P.J., "Data clustering: a review", ***Computing Surveys, ACM***, 264–323, (1999).

[13] Liu, Y., Yi, Z., Wu, H., Ye, M., Chen, K., "A tabu search approach for the minimum sum-of-squares clustering problem", ***Information Sciences***, 178: 2680–2704, (2008).

[14] Liu, R., Jiao, L., Zhang, X., Li, Y., "Gene transposon based clone selection algorithm for automatic clustering", ***Information Sciences***, 204: 1–22, (2012).

[15] Maulik, U., Bandyopadhyay, S., "Genetic algorithm-based clustering technique", ***Pattern Recognition***, 33: 1455–1465, (2000).

[16] Maulik , U., Bandyopadhyay, S., "Fuzzy partitioning using a real-coded variable-length genetic algorithm for

pixel classification", ***IEEE Transactions on Geoscience and Remote Sensing***, 41, 1075–1081, (2003).

[17] Murthy , C.A., Chowdhury, N., "In search of optimal clusters using genetic algorithms", ***Pattern Recognition Letters,*** 17: 825–832, (1996).

[18] A. Ghosh, A. Halder, M. Kothari, S. Ghosh, Aggregation pheromone density based data clustering, ***Information Sciences*** 178 (2008) 2816–2831.

[19] Niknam , T., Amiri, B., "An efficient hybrid approach based on PSO, ACO and K-means for cluster analysis", ***Applied Soft Computing***, 10: 183–197, (2010).

[20] Zhang, L., Cao, Q., "A novel ant-based clustering algorithm using the kernel method", ***Information Sciences***, 181: 4658-4672, (2010).

[21] Fathian, M., Amiri, B., Maroosi, A., "Application of honey-bee mating optimization algorithm on clustering", ***Applied Mathematics and Computation,*** 190: 1502–1513, (2007).

[22] Ahmadi , A., Karray, F., Kamel, M.S., "Model order selection for multiple cooperative swarms clustering using stability analysis", ***Information Sciences***, 182: 169–183, (2012).

[23] Izakian, H., Abraham, A., "Fuzzy C-means and fuzzy swarm for fuzzy clustering problem", ***Expert Systems with Applications,*** 38: 1835–1838, (2011).

[24] Kuo, R.J., Syu, Y.J., Chen, Z.-Y., Tien, F.C., "Integration of particle swarm optimization and genetic algorithm for dynamic clustering", ***Information Sciences***, 195: 124–140, (2012).

[25] Karaboga, D., Ozturk, C., "A novel clustering approach: artificial bee colony (ABC) algorithm", ***Applied Soft Computing,*** 11: 652–657, (2011).

[26] Hatamlou, A., Abdullah, S., Nezamabadi-Pour, H., "Application of gravitational search algorithm on data clustering", ***In International conference on rough sets and knowledge technology***, Springer, Berlin, Heidelberg, 337-346, (2011).

[27] Hatamlou, A., Abdullah, S., Nezamabadi-Pour, H. "A combined approach for clustering based on K-means and gravitational search algorithms", ***Swarm and Evolutionary Computation***, 6: 47-52, (2012).

[28] Hatamlou, A., "In search of optimal centroids on data clustering using a binary search algorithm", ***Pattern Recognition Letters,*** 33: 1756–1760, (2012).

[29] Hatamlou, A., Abdullah, S., Hatamlou, M., "Data clustering using big bang–big crunch algorithm", ***Communications in Computer and Information Science***, 383–388, (2011).

[30] El-Abd, M., "Performance assessment of foraging algorithms vs. evolutionary algorithms", ***Information Sciences***, 182: 243–263, (2012).

[31] Ghosh, S., Das, S., Roy, S., Minhazul Islam, S.K., Suganthan, P.N., "A differential covariance matrix adaptation evolutionary algorithm for real parameter optimization", ***Information Sciences***, 182: 199–219, (2012).

[32] Rana, S., Jasola , S., Kumar, R., "A review on particle swarm optimization algorithms and their applications to data clustering", ***Artificial Intelligence Review***, 35: 211–222, (2011).

[33] Yeh, W. C., "Novel swarm optimization for mining classification rules on thyroid gland data", ***Information Sciences***, 197: 65–76, (2012).

[34] Fox, B., Xiang, W., Lee, H., "Industrial applications of the ant colony optimization algorithm", ***The International Journal of Advanced Manufacturing Technology*** 31: 805–814, (2007).

[35] Cisty, M., "Application of the harmony search optimization in irrigation", ***In Recent Advances in Harmony Search Algorithm***, Springer, Berlin, Heidelberg, 123-134, (2010).

[36] Christmas, J., Keedwell, E., Frayling, T. M., & Perry, J. R., "Ant colony optimisation to identify genetic variant association with type 2 diabetes", ***Information Sciences***, 181(9): 1609-1622, (2011).

[37] Zhang, Y., Gong, D. W., & Ding, Z., "A bare-bones multi-objective particle swarm optimization algorithm for environmental/economic dispatch", ***Information sciences***, 192: 213-227, (2012).

[38] Atila, U., Dörterler, M., Durgut, R., Sahin, İ., "A comprehensive investigation into the performance of optimization methods in spur gear design", ***Engineering Optimization***, 1-16, (2019).

[39] Mirjalili , S., Mirjalili, S. M., Lewis A., "Grey wolf optimizer", ***Advances in Engineering Software***, 69: 46–61, (2014).

[40] ***Wine Data Set***, https://archive.ics.uci.edu/ml/datasets/wine, Access Time : 19.11.2020

[41] ***Iris Data Set,*** https://archive.ics.uci.edu/ml/datasets/iris , Access Time : 19.11.2020

[42] ***Breast Cancer Wisconsin (Original) Data Set***, https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(original) , Access Time : 19.11.2020

[43] ***Connectionist Bench (Vowel Recognition - Deterding Data) Data Set,*** https://archive.ics.uci.edu/ml/datasets/Connectionist+Bench+(Vowel+Recognition+-+Deterding+Data) , Access Time : 19.11.2020

[44] ***Glass Identification Data Set,*** https://archive.ics.uci.edu/ml/datasets/glass+identification , Access Time : 19.11.2020

[45] "***Contraceptive Method Choice Data Set",*** https://archive.ics.uci.edu/ml/datasets/Contraceptive+Method+Choice , Access Time : 19.11.2020

[46] Kennedy , J., Eberhart, R., "Particle swarm optimization", in: ***Proceedings of IEEE International Conference on Neural Networks***, 1944: 1942–1948, (1995).

[47] Rashedi, E., Nezamabadi-pour, H., Saryazdi, S., "GSA: a gravitational search algorithm", ***Information Sciences*** 179: 2232–2248, (2009).

[48] Erol, O.K., Eksin, I., "A new optimization method: big bang–big crunch", ***Advances in Engineering Software*** 37: 106–111, (2006)

[49] Mech, L. D., "Alpha status, dominance, and division of labor in wolf packs", ***Canadian Journal of Zoology-Revue Canadienne De Zoologie***, 77: 1196–1203, (1999).

[50] Muro, C., Escobedo, R., Spector, L., Coppinger, R., "Wolf-pack (Canis lupus) hunting strategies emerge from simple rules in computational simulations", ***Behav Process***, 88: 192-197, (2011)