International Journal of
Intelligent Systems and Applications in Engineering

ISSN:2147-6799 www.ijisae.org Original Research Paper

Grey Wolf Optimizer (GWO) Algorithm to Solve the Partitional Clustering Problem

Murat Karakoyun[1], Onur Inan[2], Ihtisam Akto[3]

Submitted: 21/08/2019 Accepted : 03/12/2019

Abstract: The clustering which is an unsupervised classification method is very important for data processing applications. The main purpose of the clustering is to separate the data samples into different groups by using the similarity (or dissimilarity) between data samples. There are many conventional and heuristic algorithms which are used for the clustering problem. Nevertheless, in last years, it is seen that many new techniques are proposed and improved to solve the clustering problem. In this paper, grey wolf optimization (GWO) algorithm which is modelled according to the social behavior of grey wolves is applied to partition the data samples by searching the optimal center of the clusters. The clustering performance of the GWO is compared with the performances of the three clustering algorithms: k-means, k-medoids and fuzzy c-means algorithms. The experiments show that the GWO algorithm has generally better results than the other clustering algorithms and can be alternatively applied on the clustering problem.

Keywords: *Data Clustering, Fuzzy C-means, Grey Wolf Optimization (GWO), K-means, K-medoids*

1. Introduction

Clustering which is a crucial task for data analysis is used to partition data into sub-sets according to the similarity or dissimilarity between data samples [1]. The main goal of the clustering techniques is to separate data samples into such groups which have maximum similarity within themselves. When any clustering method is applied on dataset and data samples are divided into groups, the variety between the clusters is wanted to be maximum [2], [3].

In literature, it is seen that there are many approaches used in the clustering problem, but clustering algorithms are generally categorized in two topics: hierarchical and non-hierarchical clustering. There is no requirement of cluster number when hierarchical clustering methods will be applied on clustering. The cluster number is determined after the clustering process. However, non-hierarchical clustering methods need cluster number for starting cluster process. In short, the aim of the non-hierarchical clustering is to divide N data samples into k clusters. In terms of time complexity, it is quadratic for hierarchical clustering methods, whereas it is about linear for non-hierarchical clustering methods [4-6].

Because of the importance of the clustering the researchers make an effort to improve new approaches on this field. As a result, there are many works about clustering problems by using different algorithms. Boushaki et al. proposed a new novel named as quantum chaotic cuckoo search (QCCS) to perform on data clustering by using real-life datasets [4]. Yang and Jiang improved a new approach to solve the initialization and automated model selection problems which are encountered by the Hidden markov model (HMM) based clustering [7]. Zhu and Xu proposed a new method called many objective fuzzy centroids clustering algorithm for categorical data by using reference point based genetic algorithm [8]. Karami and Zapata used k-means and particle swarm optimization (PSO) algorithms to improve a hybrid clustering algorithm to get the optimum number of the clusters and achieve good clustering results on this [9]. Wangchamhan et al. proposed new efficient algorithms on different data type clustering by using k-means and chaotic league championship algorithm [10]. Nidheesh et al. proposed a new clustering method which is density based of k-means algorithm to select initial centroids and so, achieve good results for clustering problem [11].

In this work, the GWO algorithm is applied on ten data sets which are taken from UCI Machine Learning Repository [12] for non-hierarchical (partitional) clustering. The clustering performance of the GWO algorithm is compared with the performances of the other clustering algorithms: k-means, k-medoids and fuzzy c-means. In the rest of this work: In chapter II, we review the clustering problem. In chapter III, we describe the algorithms used for clustering. In chapter IV, we show the experimental results. Finally, in chapter V, we analyse the experimental results as conclusion.

2. The Clustering Problem

The clustering is an important process to divide a set of data samples into sub groups in respect to similarity or dissimilarity between data samples. After the clustering process, it is expected that the samples which have similar characteristics are in same cluster and the dissimilar samples in different groups. In short, the main aim of the clustering is to create the homogenous data groups [2], [4], [13].

The main goal of the partitional clustering approaches is to separate N data samples into k groups (clusters). To achieve this

[1] *Computer Eng., Necmettin Erbakan Univ, Konya – 42140, TURKEY*
ORCID ID : 0000-0002-0677-9313
[2] *Computer Eng., Necmettin Erbakan Univ, Konya – 42140, TURKEY*
ORCID ID : 0000-0003-4573-7025
[3] *Computer Eng., Necmettin Erbakan Univ, Konya – 42140, TURKEY*
ORCID ID : 0000-0002-1616-833X
** Corresponding Author Email: oinan@erbakan.edu.tr*

purpose, the algorithms perform to find the best centroids which represent the clusters. The obtained centroids should ensure that similar samples are collected in same cluster, as dissimilar samples must be in different clusters. There are many approaches, like Manhattan distance [14], Minkowski distance [15], Euclidean distance [16] etc., to measure the distance between two data vector [3],[4]. In this work, the sum of squared Euclidean (SSE) distance between each data sample and the cluster center which data sample belongs is used as objective function to be minimized by algorithms.

3. The Reflective Process

The Three clustering algorithms (k-means, k-medoids and fuzzy c-means) and one optimization algorithm (GWO) are used for partitional clustering on data sets.

3.1. K-Means Clustering Algorithm

K-means is one of the most popular partitional clustering algorithms. The algorithm is centroid based and each cluster is represented by a centroid point (or vector). Like other partitional clustering algorithms, the aim of the k-means algorithm is to find best k centers for N data samples. The clusters are generated according to the similarities between data samples, which calculated by a distance metric. K-means is a sharp clustering algorithm so that each data sample can belong to only one cluster. K-means can be seen as a minimization algorithm which performs to minimize the sum of distance between cluster centers and the data samples assigned to centers. Equation (1) shows the objective function which wanted to be minimized by the k-means algorithm to obtain the best cluster centers [17-19].

$$\boldsymbol{SSE} = \sum_{i=1}^{k}\sum_{x \partial C_i} d\left(x, C_i\right) \qquad (1)$$

where C_i is the set of k centers, x is the data sample which assigned to the C_i cluster and $d(x, C_i)$ is the Euclidean distance between x and its center C_i. Equation (1) gives the sum squared Euclidean distance between the all data samples and their relative cluster centers. The value of each center (C_i) is evaluated by (2) given below [3], [20-23].

$$\boldsymbol{C_i} = \frac{1}{N_i}\sum_{j=1}^{N} w_{ji} x_j \qquad (2)$$

where N_i shows the number of data samples belong to ith cluster and w_{ji} is the membership degree of x_j for ith cluster. If x_j is a member of the ith cluster, w_{ji} will be 1, otherwise it will be 0.

Velmurugan summarizes k-means clustering algorithm in the following four steps [24]:

1. Randomly, initialize k cluster centers within boundaries.
2. For each data sample, match the data samples with the nearest center.
3. After assignment process, update the center of the clusters.
4. Continue to apply steps 2 and 3 until there is no more change for centers or iteration number reaches maximum.

K-means clustering algorithm generally starts with a randomly initializing of the centers. The success of the algorithm depends on the random positions of the cluster centers. So, the algorithm which is initiated in a bad random location may experience sticking problem to the local minimum. After the initialization of centers, each data sample is matched with the nearest center. All cluster centers are updated with the new samples of clusters as next step. The assigning of data samples to a center and the updating of the centers are done in a loop until there is no any change for location of the centers or it reaches the maximum iteration number [19], [24].

3.2. K-Medoids Clustering Algorithm

K-medoids clustering algorithm is also a partitional clustering algorithm to separate the N data samples into k clusters. The main purpose of the k-medoids algorithm is to choose best medoids as cluster centers. A medoid can be defined as the data sample of a cluster which average dissimilarity to all data samples in cluster is minimum. The medoid which represents a center of cluster has to be chosen among data samples in data set. K-means and k-medoids algorithms are similar in general structure. The main difference between two algorithms is the selection of the cluster centers. According to k-medoids algorithm, each cluster center must be a data sample from data set. However, any location in boundaries can be a cluster center for k-means algorithm [19], [25], [26].

In this work, the Partitioning Around Medoids (PAM) approach which is proposed in [27] is used as a method of k-medoids algorithm. Although the PAM method is an efficient approach, it takes a considerable amount of time in its working structure and is not very useful for big data sets. The PAM method can be summarized in the following steps [17], [28]:

1. Randomly, select k medoids as cluster center from data set.
2. Match the all data samples with the nearest cluster center.
3. For each medoid (m) sample and non-medoid (o) sample which related with m; swap m and o, and now o is a potential medoid. The cost of the objective function given by (1) is calculated with new potential medoid set. And the optimal medoids-cost options are selected.
4. Apply steps 2 and 3 until no more change for set of medoids or maximum iteration number is reached.

3.3. Fuzzy C-Means Clustering Algorithm

Fuzzy c-means algorithm which is developed by Dunn [29] and improved by Bezdek [30] is a partitional clustering method to divide a set of data samples into sub-groups. According to traditional clustering approaches, each data sample belongs to only one cluster; whereas in fuzzy clustering, each sample can belong to two or more clusters by having a degree of membership for each cluster. For each data sample, the sum of the membership degree for all clusters is equal to 1. The fuzzy c-means algorithm is very similar to k-means algorithm. The aim of the fuzzy c-means algorithm is to minimize the function given by (3) [31],[32].

$$\boldsymbol{J_m} = \sum_{i=1}^{N}\sum_{j=1}^{k} u_{ij}^{m} \quad \left\| x_i - C_j \right\|^2 \qquad (3)$$

where N is the number of data samples in data set, k is the number of clusters, x_i is the ith data sample, m is a number bigger than 1, u_{ij} is the membership degree of the ith sample in jth cluster and C_j is the center of the jth cluster. Respectively, (4) and (5) show that how u_{ij} and C_j are updated in each step of the iteration.

$$u_{ij} = \frac{1}{\sum_{n=1}^{k} \left(\frac{\left\| x_i - C_j \right\|^2}{\left\| x_i - C_n \right\|^2} \right)^{2/m-1}} \quad (4)$$

$$C_j = \frac{\sum_{i=1}^{N} u_{ij}^m \cdot x_i}{\sum_{i=1}^{N} u_{ij}^m} \quad (5)$$

The fuzzy c-means clustering algorithm generally can be summarized in the following steps [24]:

Randomly, initialize the u membership degree matrix by using the constraints given below:

$0 \le u_{ij} \le 1$ and $\sum_{i=1}^{k} u_{ij} = 1$ for each cluster.

Calculate the center of the each cluster according to the memberships by using (5).

Update the membership degree matrix by using (4).

Repeat steps 2 and 3 until stopping criteria is provided.

3.4. Grey Wolf Optimizer Algorithm

The GWO [33] algorithm which is based on social behavior of grey wolves is proposed by Mirjalili et al. in 2014. The hunting behavior of wolves and the social hierarchy between wolves are modeled mathematically to design the GWO algorithm. The modeling of the algorithm basically consists of four steps: social hierarchy, encircling prey, hunting and attacking prey [34-36].

Social Hierarchy: There are four types of wolves such as alpha (α), beta (β), delta (δ) and omega (ω) in social hierarchy of mathematically model of algorithm. According to the algorithm, the best three positions of the population are represented by alpha, beta and delta wolves respectively. The rest of the wolves are accepted to be omega. The hunting organization is guided by alpha, beta and delta. And the wolves which assumed as omega follow these three leader wolves [33],[36].

Encircling Prey: The grey wolves surround the victim during the hunting. Equation (6) shows the updated position of each member in population during encircling [33].

$$\vec{X}' = \vec{X}_p - \vec{A}.\vec{D} \quad (6)$$

$$\vec{D} = \left| \vec{C}.\vec{X}_p - \vec{X} \right| \quad (7)$$

where $\vec{A}$ and $\vec{C}$ are the coefficient vectors which calculated by (8) and (9) respectively, $\vec{X}_p$ is the position of the victim, and $\vec{X}$ is the current position of the gray wolf.

$$\vec{A} = 2.\vec{a}\ .\vec{r}_1 - \vec{a} \quad (8)$$

$$\vec{C} = 2.\vec{r}_2 \quad (9)$$

where $\vec{a}$ is a number which linearly decreases from 2 to 0 depends on iteration number, and $\vec{r}_1, \vec{r}_2$ are vectors that randomly generated between 0 and 1.

Hunting: As mentioned before, alpha, beta and delta wolves have best positions in population. So, they have better knowledge about a potential position for prey. Therefore, the members of the population use the position of the leader wolves to update their current position to achieve a better position. The following formulas given in (10), (11) and (12) have been developed to provide a mathematical representation of this action [33],[36].

$$\begin{aligned} \vec{D}_{\alpha} &= \left| \vec{C}_1.\vec{X}_{\alpha} - \vec{X} \right| \\ \vec{D}_{\beta} &= \left| \vec{C}_2.\vec{X}_{\beta} - \vec{X} \right| \\ \vec{D}_{\delta} &= \left| \vec{C}_3.\vec{X}_{\delta} - \vec{X} \right| \end{aligned} \quad (10)$$

$$\begin{aligned} \vec{X}_1 &= \vec{X}_{\alpha} - \vec{A}_1.\vec{D}_{\alpha} \\ \vec{X}_2 &= \vec{X}_{\beta} - \vec{A}_2.\vec{D}_{\beta} \\ \vec{X}_3 &= \vec{X}_{\delta} - \vec{A}_3.\vec{D}_{\delta} \end{aligned} \quad (11)$$

$$\vec{X}' = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \quad (12)$$

Attacking Prey: The last step of the grey wolves is to attack the victim to finish hunting. The attacking phase is the exploitation process. The $\vec{a}$ value decreases from 2 to 0 for each iteration step while the algorithm is working. So that, value of $\vec{A}$ changes depends on value of $\vec{a}$. The value of $\vec{A}$ is randomly generated in the gap $[-2\vec{a}, 2\vec{a}]$. When $|\vec{A}| < 1$, the wolves are forced to attack the victim. On the other hand, if $|\vec{A}| > 1$, this forces the wolves to diverge from the victim to explore [33],[35],[36].

Flow diagram of the GWO algorithm, while applying on clustering problem, is shown in Fig. (1). The algorithm starts with settings of the parameters. Then first population is generated with random cluster centers in boundaries. For each member of the population, according to the cluster centers, the fitness value is calculated. The best three members are assigned as *alpha*, *beta* and *delta* respectively. After that a loop is created to update the positions (which present the cluster centers) of the members in population. According to the new positions; *alpha*, *beta* and *delta* are updated at each step of the loop. When the criterion, which finishes the loop, is provided then the value of the *alpha* member is sent as the result of the algorithm.

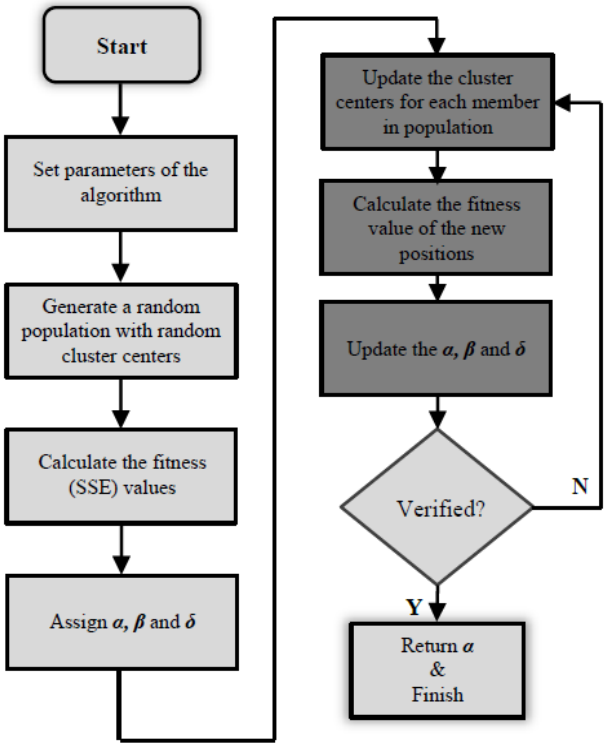

Fig 1. Images showing diagram of adaptation of GWO algorithm to clustering problem

4. Experimental Results

Ten data sets which have different attribute size and cluster number are used to compare the performances of the algorithms on partitional clustering problem. Table I gives the information about data sets which used in this work.

The algorithms that used in this work have been performed on an Intel® Core™ i5-2400 CPU @ 3.1 GHz processor, 4GB RAM and Windows 7 (64-bit) Professional operating system.

K-means, k-medoids and fuzzy c-means algorithms have only maximum iteration number as parameter. Nevertheless, the GWO algorithm has a set of parameters. The only mutual parameter is maximum iteration number for algorithms and its value is 1000. The population size is 100, $\vec{a}$ starts as 2 and linearly decreases to 0, and $\vec{r}_1, \vec{r}_2$ is randomly generated in the gap [0, 1].

Table 1. The properties of the data sets [2]

Data Set	Number of Clusters	Number of Attributes	Number of Data
Balance	3	4	625
Cancer	2	30	569
Cancer-Int	2	9	699
Credit	2	14	690
Dermatology	6	34	366
E. Coli	5	7	327
Glass	6	9	214
Iris	3	4	150
Thyroid	3	5	215
Wine	3	13	178

The SSE value given in (1) is used as objective function for all algorithms. The aim of the algorithms is to perform to find best cluster centers which minimize the SSE value. Algorithms worked 30 times and the generated results are presented in Table II, where B is best, W is worst, A is the average and S is the standard deviation result of the 30 times of working. For each data set, the best average result which generated by any algorithm is marked as bold. According to the results given in Table II, the GWO algorithm generated better solutions than the other algorithms for six (balance, cancer-int, credit, iris, thyroid and wine) data sets. For cancer data set, GWO and k-means have same result which is better than the solution of the other two algorithms. K-means algorithm generated better solutions on the rest of the data sets (dermatology, e. coli, and glass). K-medoids and fuzzy c-means clustering algorithms have no better results on any data set against the other algorithms.

In Table III, the average value of objective function of 30 times working for algorithms and according to these results, the ranking of algorithms on each data set are given with the average rank of algorithms. The average rank shows that the GWO algorithm has better ranking against other clustering algorithms. The GWO algorithm has best average rank with 1.3; and k-means, fuzzy c-means and k-medoids have 1.9, 3.3 and 3.4 average ranks, respectively.

Table 2. The results of the 30 times for algorithms

Data sets		K-means	K-medoids	F.C-means	GWO
Balance	B	1423.8514	1672.4587	1722.2446	1423.8205
	W	1433.0977	1822.7225	1722.2446	1423.8213
	A	1425.7619	1716.0266	1722.2446	1423.8209
	S	2.05E+00	3.44E+01	1.16E-12	1.95E-04
Cancer	B	1.34E+154	1.79E+308	7.63E+156	1.34E+154
	W	1.34E+154	1.79E+308	7.63E+156	1.34E+154
	A	1.34E+154	1.79E+308	7.63E+156	1.34E+154
	S	0.00E+00	0.00E+00	0.00E+00	0.00E+00
Cancer-Int	B	2986.9613	3311.5612	3286.1132	2964.3876
	W	2988.4278	4755.7844	3286.1132	2964.3906
	A	2988.0856	3755.4299	3286.1132	2964.3888
	S	6.31E-01	4.51E+02	3.68E-13	7.96E-04
Credit	B	748491.65	562404.23	759180.47	556743.89
	W	808744.44	670524.54	759180.47	557162.85
	A	796620.05	596216.51	759180.47	556797.08
	S	2.33E+04	3.46E+04	6.49E-11	1.00E+02
Dermatology	B	2021.0276	2792.4665	5196.3797	2221.5083
	W	2298.0339	3275.9505	5196.3797	2439.2836
	A	2088.3342	2978.1435	5196.3797	2342.1929
	S	6.82E+01	1.19E+02	2.50E-11	4.49E+01
E. Coli	B	66.0246	127.784	108.4402	65.6273
	W	70.3212	182.2736	108.4402	74.9942
	A	68.105	152.0576	108.4402	71.5338
	S	1.10E+00	1.56E+01	9.05E-11	2.67E+00
Glass	B	213.4205	303.9722	400.9817	275.7853
	W	266.5812	424.9596	404.2808	436.4331
	A	240.9204	338.7686	402.0121	313.4421
	S	1.43E+01	2.54E+01	1.49E+00	2.74E+01
Iris	B	97.3259	182.0554	106.3591	96.6567
	W	122.2789	253.9361	106.3591	120.8957
	A	102.328	210.6296	106.3591	98.4404
	S	1.01E+01	2.09E+01	8.29E-14	6.11E+00
Thyroid	B	1988.0143	2076.709	2812.49	1868.262
	W	2019.3404	2390.4087	2812.49	1940.0841
	A	2009.7801	2194.24	2812.49	1905.7626
	S	9.09E+00	7.96E+01	3.90E-11	1.99E+01
Wine	B	16555.68	16901.08	17128.45	16305.47
	W	18436.95	26491.45	17128.45	16336.63
	A	17788.93	20476.13	17128.45	16316.89
	S	8.90E+02	2.68E+03	6.26E-12	8.49E+00

Table 3. The average value of objective function of 30 times and the rank depends on these results

	K-means	K-medoids	F.C-means	GWO
Balance	1425.76192	1716.02663	1722.24464	1423.82091
Cancer	1.34E+1541	1.79E+3084	7.63E+1563	1.34E+1541
Cancer-Int	2988.085622	3755.429944	3286.113233	2964.388881
Credit	796620.0554	596216.5122	759180.4733	556797.0881
Dermatology	2088.334211	2978.143533	5196.379744	2342.192922
E. Coli	68.1051	152.05764	108.44023	71.53382
Glass	240.92041	338.76863	402.01214	313.44212
Iris	102.3282	210.62964	106.35913	98.44041
Thyroid	2009.780122	2194.243	2812.494	1905.762611
Wine	17788.9333	20476.1344	17128.4522	16316.8911
Avg. Rank	1.9	3.4	3.3	**1.3**

5. Conclusion

In this study, the GWO algorithm which is based on behavior of grey wolves is applied on partitional clustering problem. To measure the performance of the GWO on clustering problem, its performance is compared with three standard clustering algorithms: k-means, k-medoids and fuzzy c-means. As a result, the GWO algorithm generally generated better solutions than the other clustering algorithms. Therefore, the GWO can be proposed as an alternative algorithm to use on clustering problems.

References

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

[2] M. Karakoyun and A. Babalik, "Data Clustering with Shuffled Leaping Frog Algorithm (SFLA) for Classification," in 2015 Int'l Conference on Intelligent Computing, Electronics Systems and Information Technology (ICESIT-15), Kuala Lumpur (Malaysia), 25-26 Aug 2015.

[3] D. Karaboga and C. Ozturk, "A novel clustering approach: Artificial Bee Colony (ABC) algorithm," Applied Soft Computing, vol. 11, pp. 652-657, 2011.

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

[5] W. A. Barbakh, Y. Wu, and C. Fyfe, "Review of clustering algorithms," in Non-Standard Parameter Adaptation for Exploratory Data Analysis, ed: Springer, 2009, pp. 7-28.

[6] H. Frigui and R. Krishnapuram, "A robust competitive clustering algorithm with applications in computer vision," Ieee transactions on pattern analysis and machine intelligence, vol. 21, pp. 450-465, 1999.

[7] Y. Yang and J. Jiang, "Bi-weighted ensemble via HMM-based approaches for temporal data clustering," Pattern Recognition, vol. 76, pp. 391-403, 2018.

[8] S. Zhu and L. Xu, "Many-objective fuzzy centroids clustering algorithm for categorical data," Expert Systems with Applications, vol. 96, pp. 230-248, 2018.

[9] A. Karami and M. Guerrero-Zapata, "A fuzzy anomaly detection system based on hybrid pso-kmeans algorithm in content-centric networks," Neurocomputing, vol. 149, pp. 1253-1269, 2015.

[10] T. Wangchamhan, S. Chiewchanwattana, and K. Sunat, "Efficient algorithms based on the k-means and Chaotic League Championship Algorithm for numeric, categorical, and mixed-type data clustering," Expert Systems with Applications, vol. 90, pp. 146-167, 2017.

[11] N. Nidheesh, K. A. Nazeer, and P. Ameer, "An enhanced deterministic K-Means clustering algorithm for cancer subtype prediction from gene expression data," Computers in biology and medicine, vol. 91, pp. 213-221, 2017.

[12] (Access Date: 25 Dec, 2017). UCI Machine Learning Repository. Available: https://archive.ics.uci.edu/ml/datasets.html

[13] J. F. Hair, W. C. Black, B. J. Babin, R. E. Anderson, and R. L. Tatham, Multivariate data analysis vol. 5: Prentice hall Upper Saddle River, NJ, 1998.

[14] S. N. Neyman, B. Sitohang, and S. Sutisna, "Reversible fragile watermarking based on difference expansion using manhattan distances for 2d vector map," Procedia Technology, vol. 11, pp. 614-620, 2013.

[15] M. Luo and B. Liu, "Robustness of interval-valued fuzzy inference triple I algorithms based on normalized Minkowski distance," Journal of Logical and Algebraic Methods in Programming, vol. 86, pp. 298-307, 2017.

[16] D. P. Mesquita, J. P. Gomes, A. H. S. Junior, and J. S. Nobre, "Euclidean distance estimation in incomplete datasets," Neurocomputing, vol. 248, pp. 11-18, 2017.

[17] P. Arora and S. Varshney, "Analysis of K-means and K-medoids algorithm for big data," Procedia Computer Science, vol. 78, pp. 507-512, 2016.

[18] M. Capó, A. Pérez, and J. A. Lozano, "An efficient approximation to the K-means clustering for massive data," Knowledge-Based Systems, vol. 117, pp. 56-69, 2017.

[19] M. Karakoyun, A. Saglam, N. A. Baykan, and A. A. Altun, "Non-locally color image segmentation for remote sensing images in different color spaces by using data-clustering methods," in 5th International Conference on Advanced Technology & Sciences (ICAT'17), Istanbul (Turkey), 9-12 May, 2017.

[20] J. Xiao, Y. Yan, J. Zhang, and Y. Tang, "A quantum-inspired genetic algorithm for k-means clustering," Expert Systems with Applications, vol. 37, pp. 4966-4973, 2010.

[21] M. A. Rahman and M. Z. Islam, "A hybrid clustering technique combining a novel genetic algorithm with K-Means," Knowledge-Based Systems, vol. 71, pp. 345-365, 2014.

[22] M. I. Malinen, R. Mariescu-Istodor, and P. Fränti, "K-means*: Clustering by gradual data transformation," Pattern Recognition, vol. 47, pp. 3376-3386, 2014.

[23] Y. Marinakis, M. Marinaki, M. Doumpos, N. Matsatsinis, and C. Zopounidis, "A hybrid stochastic genetic–GRASP algorithm for clustering analysis," Operational Research, vol. 8, pp. 33-46, 2008.

[24] T. Velmurugan, "Performance based analysis between k-Means and Fuzzy C-Means clustering algorithms for connection oriented telecommunication data," Applied Soft Computing, vol. 19, pp. 134-146, 2014.

[25] H.-S. Park and C.-H. Jun, "A simple and fast algorithm for K-medoids clustering," Expert systems with applications, vol. 36, pp. 3336-3341, 2009.

[26] L. Kaufman and P. Rousseeuw, "Statistical Data Analysis Based on the L1-Norm and Related Methods," ed: North Holland: Amsterdam, 1987.

[27] L. Kaufman and P. J. Rousseeuw, "Partitioning around medoids (program pam)," Finding groups in data: an introduction to cluster analysis, pp. 68-125, 1990.

[28] A. Khatami, S. Mirghasemi, A. Khosravi, C. P. Lim, and S. Nahavandi, "A new PSO-based approach to fire flame detection using K-Medoids clustering," Expert Systems with Applications, vol.

68, pp. 69-80, 2017.
[29] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," 1973.
[30] J. C. Bezdek, "Objective Function Clustering," in Pattern recognition with fuzzy objective function algorithms, ed: Springer, 1981, pp. 43-93.
[31] J. Jędrzejowicz and P. Jędrzejowicz, "Distance-based online classifiers," Expert Systems with Applications, vol. 60, pp. 249-257, 2016.
[32] X. Qiu, Y. Qiu, G. Feng, and P. Li, "A sparse fuzzy c-means algorithm based on sparse clustering framework," Neurocomputing, vol. 157, pp. 290-295, 2015.
[33] [S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," Advances in engineering software, vol. 69, pp. 46-61, 2014.
[34] [A. K. M. Khairuzzaman and S. Chaudhury, "Multilevel thresholding using grey wolf optimizer for image segmentation," Expert Systems with Applications, vol. 86, pp. 64-76, 2017.
[35] [L. Rodríguez, O. Castillo, J. Soria, P. Melin, F. Valdez, C. I. Gonzalez, et al., "A fuzzy hierarchical operator in the grey wolf optimizer algorithm," Applied Soft Computing, vol. 57, pp. 315-328, 2017.
[36] N. Singh and S. Singh, "A novel hybrid GWO-SCA approach for optimization problems," Engineering Science and Technology, an International Journal, 2017.