Chaotic Mountain Gazelle Optimizer (CMGO): A Robust Optimization Algorithm for K-Means Clustering of Diverse Data Types

Tanatip Watthaisong[1(✉)], Khamron Sunat[1], and Nipotepat Muangkote[2]

[1] Department of Computer Science, College of Computing, Khon Kaen University, Khon Kaen, Thailand
tanatip.w@kkumail.com, skhamron@kku.ac.th

[2] Department of Business Computer, Mahasarakham Business School, Mahasarakham University, Mahasarakham, Thailand

Abstract. Addressing challenges in data clustering for diverse data types, we introduce the Chaos Mountain Gazelle Optimizer (CMGO). This enhanced Mountain Gazelle Optimizer (MGO) is tailored for K-means clustering solutions. Noticing a skew in MGO's strategy distribution, we integrated a chaotic map into the Territorial Solitary Males strategy and omitted the Migration to Search for Food strategy. This adjustment increases exploration and curtails exploitation, improving CMGO's effectiveness in clustering complex datasets. We implemented the Gower distance technique to navigate K-means clustering's limitations with categorical and binary data. Tests on numeric, binary, categorical, and mixed data underscore the clustering's versatility. We evaluated CMGO against 14 algorithms on 28 UCI and OpenML datasets using the F-Measure metric and the tied rank test for statistical significance ranking. CMGO outperforms the original MGO and other tested algorithms in clustering pure numeric and categorical data, securing first place, and third for mixed data. Thus, CMGO emerges as a robust, efficient K-means optimizing method for complex, diverse datasets.

Keywords: Data clustering · K-Means Clustering · Mountain Gazelle Optimizer · Mixed-type data · Chaos map · Nature-Inspired Optimization

1 Introduction

Data science is vital in numerous industries, enabling informed decision-making through in-depth data analysis and interpretation. Through the application of techniques like machine learning, businesses can effectively use predictive analytics to anticipate future outcomes and meet customer needs [1]. Clustering algorithms automatically reveal data patterns and relationships. They analyze data to identify similarities, detect patterns, and group data points based on the characteristics and desired clustering techniques [2]. It is an unsupervised learning method that uncovers natural clusters in a dataset, facilitating data exploration and comprehension [3]. However, clustering faces challenges in

C. Anutariya and M. M. Bonsangue (Eds.): DSAI 2023, CCIS 1942, pp. 18–34, 2023.
https://doi.org/10.1007/978-981-99-7969-1_2

selecting suitable data representatives, handling diverse data, and dealing with distribution complexities. It's a computationally complex task in the class of NP-complete problems, aiming to minimize dissimilarity measures for identifying clusters in varied datasets [4]. Two fundamental approaches to data clustering include hierarchical clustering, which entails a tree-like division of data, and partition clustering. The objective in data clustering is to determine cluster centers (centroids) and improve the partitioning through iterative relocation, with examples of partition clustering algorithms like K-means [5].

Several nature-inspired optimization algorithms have gained attention in search clustering approaches. These algorithms aim to optimize an objective function considering the sum of intra-cluster distances to find centroids. Examples from the literature include the Gray Wolf Optimization (GWO) [6], the Jaya Algorithm (JAYA) [7], Chaotic League Championship Algorithm (KSCLCA) [8], Salp Swarm Algorithm (SSA) [9], Dandelion Optimizer (DO) [10], Leader Slime Mould Algorithm (LSMA) [11], Flow Direction Algorithm (FDA) [12], Artificial Gorilla Troops Optimizer (GTO) [13], Mountain Gazelle Optimizer (MGO) [14], Prairie Dog Optimization Algorithm (PDO1) [15], Chimp Optimization Algorithm (CHIMP) [16] and Opposition African Vultures Optimization Algorithm (OAVOA) [17].

The Mountain Gazelle Optimization (MGO) algorithm proposed [14], mimics the social behaviors of mountain gazelles and utilizes factors like male herds, maternity herds, territorial males, and migration for food exploration. While MGO excels in benchmark functions and engineering problems, its application to data clustering remains challenging. We propose a variation, Chaos Mountain Gazelle Optimizer (CMGO), which integrates a chaotic map into the distribution strategy to address this issue. Furthermore, the Migration to Search for Food strategy in MGO is unsuitable for clustering problems. To address this, we modify the strategy distribution by integrating a chaotic map into the Territorial Solitary Males strategy while excluding the Migration to search for Food strategy. In data clustering, K-means clustering is commonly used, but it encounters challenges when computing distances between objects and centroids, especially with categorical and binary data. To address this, the Gower distance technique is integrated into K-means clustering. According to [18] using Gower's similarity coefficients improved the accuracy of the K-means algorithm in experiments with various datasets. We selected a total of 28 real datasets from the UCI and OpenML repositories to assess the performance of the proposed algorithms on three data types: numerical, categorical, and mixed. The effectiveness of the algorithm was compared against 14 state-of-the-art approaches. The evaluation employed the F-Measure metric to assess performance, and statistical significance ranking was conducted using the tied rank test. Results revealed that CMGO exhibited lower intra-cluster distance and higher F-Measure values, outperforming both the original Mountain Gazelle Optimization (MGO) algorithm and other tested algorithms. Specifically, CMGO secured the first position in clustering numeric and categorical data, while ranking third for mixed data.

The remainder of the paper is organized as follows: Sect. 2 provides an overview of K-means clustering problems and the traditional MGO algorithm. Section 3 introduces the proposed method, CMGO. In Sect. 4, we discuss the performance evaluation experiments. Section 5 presents the discussion. Finally, Sect. 6 conclusion.

2 Related Works

In this section, we will discuss the K-means algorithm for data clustering and introduce a traditional Mountain Gazelle Optimization (MGO) to enhance its capabilities.

2.1 K-means Clustering

The K-means clustering algorithm has received significant attention in the literature. Especially in nature-inspired optimization approaches, a large number of researchers employ optimization algorithms to search for cluster centers. These optimization algorithms aim to discover cluster centers by minimizing the objective function, which takes into account the sum of intra-cluster distances. The K-means algorithm partitions the dataset into K distinct clusters. The K-means algorithm operates through unsupervised learning. Based on the data points $X = [x_{1,}x_2x_3, \ldots x_N]$ and the positions of K cluster centroids $C = \{c_1, c_2, c_3, \ldots, c_k | \forall i = 1, \ldots, K : c_i \neq \varnothing \; and \; \forall i \neq j : c_i \cap c_j = \varnothing\}$. In clustering, each data point in set X is assigned to one of the K clusters in a manner that minimizes the objective fitness function. The sum of the squared Euclidean distance between data points x_N and the center of the cluster c_j is used as the objective function, as presented in Eq. (1).

$$f(k) = \sum_{k=1}^{K} \sum_{i=1}^{N_k} (x_i - c_k)^2, \tag{1}$$

where $k = 1, 2, ...K$ is the number of clusters, x_i, $i = 1, 2, ...N_k$ are the patterns in the k^{th} cluster, c_k is center of the k^{th} cluster. In this context, the cluster centers are depicted as:

$$c_k = \frac{1}{n_k} \sum_{i=1}^{n_k} x_i. \tag{2}$$

In this research, nature-inspired algorithms are employed for the purpose of identifying cluster centers within the dataset. The primary objective of the K-means algorithm is to determine optimal centers for each of the K clusters in a partitioned cluster.

2.2 Traditional Mountain Gazelle Optimization (MGO)

This section provides a brief explanation of the main inspiration behind the traditional MGO algorithm [14], followed by a description of the mathematical model.

Territorial Solitary Males
Male mountain gazelles establish solitary territories through intense territorial battles and competition for females, with adult males vigorously defending their boundaries.

$$TSM = male_{gazelle} - |(ri_1 \times BH - ri_2 \times X(t)) \times F| \times Cof_r \tag{3}$$

The $male_{gazelle}$ is the position vector of the best global solution, representing an adult male. ri_1 and ri_2 are random integer 1 or 2. The coefficient vector BH is the young male herd. Cof_r is randomly selected in each iteration.

$$BH = X_{ra} \times \lfloor r_1 \rfloor + M_{pr} \times \lfloor r_2 \rfloor, ra = \left(\left\lfloor \frac{N}{3} \right\rfloor \ldots N \right) \tag{4}$$

where X_{ra} is a random young male within the interval ra. M_{pr} is the average number of randomly selected search agents from a pool of N, based on the ceiling division of N by 3. N is the total number of gazelles. r_1 and r_2 are random values (0, 1].

$$F = N_1(D) \times exp\left(2 - Iter \times \left(\frac{2}{MaxIter}\right)\right) \tag{5}$$

N_1 is a randomly generated number from the standard distribution, exp is the exponential function, $MaxIter$ is the total iterations, and $Iter$ is the current iteration.

$$Cof_i = \begin{cases} (a+1) + r_3, \\ a \times N_2(D), \\ r_4(D), \\ N_3(D) \times N_4(D)^2 \times \cos((r_4 \times 2) \times N_3(D)), \end{cases} \tag{6}$$

r_3, r_4, and $rand$ are random numbers (0, 1). N_2, N_3 and N_4 are randomly generated numbers from a normal distribution, and cos represent the cosine function.

$$a = -1 + Iter \times \left(\frac{-1}{MaxIter}\right) \tag{7}$$

$MaxIter$ is the total number of iterations, while $Iter$ is the current iteration count.

Maternity Herds

Maternity herds facilitate robust male gazelle births, with active male participation in delivery and young males competing for dominance over females.

$$MH = \left(BH + Cof_{1,r}\right) + (ri_3 \times male_{gazelle} - ri_4 \times X_{rand}) \times Cof_r \tag{8}$$

ri_3 and ri_4 are random integers, either 1 or 2. The $male_{gazelle}$ is the global solution in the current iteration. X_{rand} is the position of a gazelle randomly selected from the entire population.

Bachelor Male Herds

Male gazelles establish territories and engage in intense battles for female possession, demonstrating dominance and control. This behavior is computed as follows:

$$BMH = (X(t) - D) + \left(ri_5 \times male_{gazelle} - ri_6 \times BH\right) \times Cof_r, \tag{9}$$

where $X(t)$ is the position vector of the gazelle in the current iteration. ri_5 and ri_6 are randomly selected integers, either 1 or 2.

$$D = \left(|X(t)| + \left|male_{gazelle}\right|\right) \times (2 \times r_6 - 1), \tag{10}$$

the parameter r_6 is a random value between 0 and 1.

Migration to Search for Food
The mathematical formulation representing the foraging and migratory behavior of mountain gazelles incorporates their ability to cover long distances and engage in migration, as well as their exceptional running speed and jumping abilities.

$$MSF = (ub - lb) \times r_7 + lb \tag{11}$$

ub and *lb* is upper and lower limits. r_7 is a randomly selected integer within the range of 0 and 1.

The mechanisms (TSM, MH, BMH, and MSF) are applied to all gazelles, generating new generations, and adding to the population. High-quality gazelles are preserved, while weak or old ones are removed, with the adult male gazelle considered the best among them.

3 Proposed method Chaotic Mountain Gazelle Optimizer

3.1 Motivation

The Mountain Gazelle Optimizer (MGO) algorithm draws inspiration from the social structure of wild mountain gazelles. While MGO demonstrates strong search capabilities in benchmark functions and engineering problems [14], its application to NP-complete real-world problems like data clustering remains challenging. To address this, we enhance MGO by incorporating a chaotic map into the Territorial Solitary Males strategy and excluding the Migration to Search for Food strategy. Additionally, we introduce the Gower distance technique to overcome challenges in computing distances for categorical and binary data in K-means clustering.

Chaotic Territorial Solitary Males Strategy
In our proposed CMGO algorithm, the Territorial Solitary Males strategy is enhanced by incorporating a chaotic map. The updated mathematical expression for the territory of adult male $TSMC^{t+1}$ is given by the following equation.

$$TSMC^{t+1} = male_{gazelle} - \left| \left(\left(\frac{C^{t+1}}{ri_1} \right) \times BH - ri_2 \times X(t) \right) \times F \right| \times Cof_r, \tag{12}$$

$male_{gazelle}$ is the position vector of the best global solution. ri_1 and ri_2 are random integers, either 1 or 2. The coefficient vector *BH* corresponds to the young male herds from the original MGO algorithm. *F* and Cof_r Similar to the original MGO.

The Chaotic Parameter
The parameters ri_1 and ri_2 serve as controls for updating the territory of the adult male $TSMC^{t+1}$ in our CMGO algorithm. The parameter ri_1 is a random integer, taking a value of either 1 or 2, and directly in fluences the search solution. If ri_1 is 1, the coefficient vector *BH* remains unchanged. However, when incorporating a chaotic map

into the computation of ri_1, the coefficient vector BH undergoes changes throughout the entire evolution process. Previous studies have demonstrated the seamless and effective integration of a chaotic map with the biogeography-based optimization (BBO) algorithm [19]. In our proposed CMGO algorithm, we introduce the use of the Piecewise map within the Territorial Solitary Males strategy.

The iterative form of the Piecewise map is defined as:

$$C^{t+1} = \begin{cases} \frac{C^t}{P}, & 0 \leq C^t < P \\ \frac{C^t - P}{0.5 - P}, & P \leq C^t < 0.5 \\ \frac{1 - P - C^t}{0.5 - P}, & 0.5 \leq C^t < 1 - P \\ \frac{1 - C^t}{P}, & 1 - P \leq C^t < 1 \end{cases}, P = 0.4 \tag{13}$$

where the parameter P is set to 0.4. The visualization of the Piecewise map is depicted in Fig. 1.

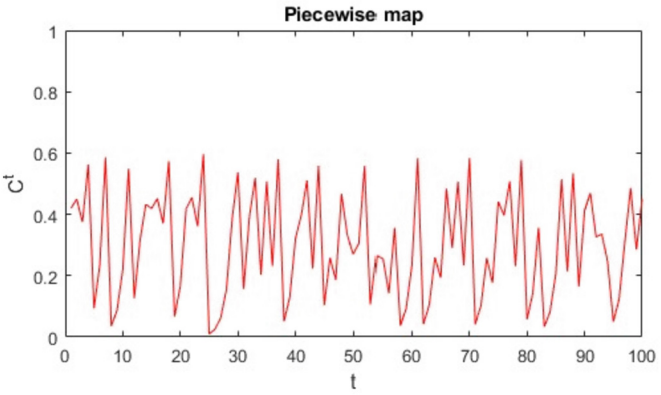

Fig. 1. The behavior of the Piecewise maps employed in our CMGO algorithm.

The Gower Similarity Coefficient

To improve the performance of K-means clustering when dealing with categorical and binary data, a similarity measure, such as the Gower coefficient [20] or the Gower distance technique, is used instead of the squared Euclidean distance to calculate the dissimilarity measure $D_{Gow}(X_n, C_j)$ during the clustering process. The Gower distance (D_{Gow}), is employed in this context. The computation of the Gower distance is as follows:

$$D_{Gow}(X_n, C_j) = \frac{\sum_{k=1}^{N_k} S_{njk}\delta_{njk}w_k}{\sum_{k=1}^{N_k} \delta_{njk}w_k} \tag{14}$$

In the case of binary and categorical attributes, $S_{njk} = 0$ if $X_{nk} = C_{jk}$ otherwise $S_{njk} = 1$. For continuous attributes, $S_{njk} = |X_{nk} - C_{jk}|/(max_l X_{lk} - min_l X_{lk})$, where

l run for all non-missing values for the attribute k. If we can compare X_n and C_j for the attribute k then $\delta_{njk} = 1$, zero otherwise. w_k is the weight for the attribute k. For simplicity, we will set $w_k = 1$. N_k the total number of species recorded across both units.

3.2 The Main Process of Proposed CMGO Algorithm

The Chaos Mountain Gazelle Optimizer (CMGO) algorithm is developed to tackle challenges in data clustering for various data types. This enhanced version of the Mountain Gazelle Optimizer (MGO) is specifically designed for K-means clustering solutions.

The relationship between the CMGO optimizer and K-means clustering can be explained as follows: We utilize the CMGO to optimize the cluster centers in K-means clustering. First, it initializes the cluster centers with random positions and then proceeds to perform the K-means algorithm from each of these random positions. Secondly, during the evolutionary process, the CMGO iteratively updates the position of the optimal cluster center. The process continues until it reaches the desired optimal position (the best cluster center). Lastly, all positions are assigned to the cluster centers, resulting in the output of the clustering results. The pseudo-code of the CMGO algorithm is also shown in Algorithm 1.

Algorithm 1 Pseudo-code of K-mean clustering based on CMGO

1: Input the population size N and maximum number of iterations T
2: Initialize cluster centers on a random population using $X_i(i = 1, 2, \ldots, N)$
3: **For** i = 1 to N Do
4: Perform the K-means algorithm for each Gazelle formation, or row, of X_i
5: **End for**
6: Evaluate the cluster center (Gazelle's fitness) on X_i by Eq. (1)
7: Generate the Chaos sequence C^{t+1}
8: **While** (stopping condition is not met)
9: **For** (each Gazelle (X_i)) **Do**
10: Calculate $TSMC^{t+1}$ using Eq. (12)
11: Calculate MH using Eq. (8)
12: Calculate BMH using Eq. (9)
13: Calculate the fitness values of $TSMC^{t+1}$, MH, BMH, Then add to habitat
14: **End for**
15: Sort the entire population in ascending order
16: Update $best_{gazelle}$
17: save the N Best Gazelle in Max number of population
18: **End while**
19: Return $X_{bestGazelle}$, best Fitness (the best cluster center)

---

4 Experimental Results and Analysis

The experiments were conducted using MATLAB R2022a 64-bit on a desktop computer with an AMD Ryzen 9 5950X 16-Core Processor (3.40 GHz), 32.00 GB RAM, SSD M.2 500 GB, and Microsoft Windows 11 Professional 64-bit operating system.

4.1 Performance Evaluation

The CMGO algorithm was evaluated against competing algorithms using UCI and OpenML datasets, employing the F-Measure metric and tied rank test for statistical significance rankings. The F-Measure, which integrates both precision and recall, served as the metric for evaluating performance, can be calculated by a confusion matrix as follows:

$$F - Measure(x) = \frac{2 \times precision \times recall}{precision + recall} \times 100, \tag{17}$$

where, precision and recall are calculated using the following equations based on a confusion matrix:

$$precision = \frac{TP}{TP + FP}, \tag{15}$$

$$recall = \frac{TP}{TP + FP}, \tag{16}$$

where, TP represents true positives, FP corresponds to false positives, and FN signifies false negatives. Higher precision values indicated superior algorithm performance, while greater recall captured more true positives, thereby indicating improved performance in correctly identifying positive instances. The evaluation of the F-Measure occurs upon termination of the optimization algorithm. A higher F-Measure leads to increased clustering accuracy. In [8] stressed the importance of a low objective fitness value for accurate cluster formation, guiding our adoption of the F-measure metric to evaluate and achieve precise clusters.

The Benchmark Dataset for Clustering

We partitioned the benchmark dataset into three distinct groups: numerical datasets, categorical datasets, and mixed-data type datasets. We utilized a total of 28 well-known datasets taken from the UCI [21] and OpenML [22] repositories. The numerical dataset consisted of: Iris, Glass, Breast-Cancer-Wisconsin, Wine, Thyroid, Synthetic-Control-Charts, Ionosphere, Sonar, Diabetes, Ecoli, and Banknote-Authentication. The categorical dataset included: Balance Scale, Hayes-Roth, Monks, SPECT-Heart, asnd Nursery. The mixed-data type dataset encompassed the following datasets: Acute-Inflammations, Analcatdata-Seropositive, Churn, Cloud, Fruitfly, Haberman, Newton-Hema, Sleuth-Case2002, Socmob, Tae, Heart-Disease, and ACA. By incorporating diverse datasets representing different data types, we aimed to comprehensively evaluate the performance of our algorithm across various scenarios. The characteristics of the three datasets are presented in Table 1.

4.2 Experimental Results

To verify the proposed CMGO, we compared it against 14 algorithms on 28 UCI and OpenML datasets. The algorithms used for comparison included: Opposition African Vultures Optimization Algorithm (OAVOA) [17], Salp Swarm Algorithm (SSA) [9],

Table 1. The characteristics of the three datasets.

Numeric	Categorical	Mixed-type dataset
1. Iris (N=150, D=4, k=3) 2. Glass (N=214, D=9, k=6) 3. Breast-Cancer-Wisconsin (N=699, D=9, k=2) 4. Wine (N=1178, D=13, k=3) 5. Thyroid (N=215, D=5, k=3) 6. Synthetic-Control-Chart (N=600, D=60, k=6) 7. Ionosphere (N=351, D=34, k=2) 8. Sonar (N=208, D=60, k=2) 9. Diabetes (N=768, D=8, k=2) 10. Ecoli (N=336, D=7, k=4) 11. Banknote-Authentication (N=1370, D=4, k=2)	1. Balance-Scale (N=625, D=4, k=3) 2. Hayes-Roth (N=160, D=4, k=3) 3. Monks (N=432, D=6, k=2) 4. SPECT-Heart (N=267, D=22, k=2) 5. Nursery (N=12,960, D=8, k=2)	1. Acute-Inflammations (N=120, D=6(c=5, η=1), k=2) 2. Analcatdata-Seropositive (N=132, D=3(c=1, η=2), k=2) 3. Churn (N=5000, D=20(c=4, η=16), k=2) 4. Cloud (N=108, D=7(c=1, η=6), k=2) 5. Fruitfly (N=125, D=4(c=2, η=2), k=2) 6. Haberman (N=306, D=3(c=1, η=2), k=2) 7. Newton-Hema (N=140, D=3(c=1, η=2), k=2) 8. Sleuth-Case2002 (N=147, D=6(c=4, η=2), k=2) 9. Socmob (N=1156, D=5(c=4, η=1), k=2) 10. Tae (N=151, D=5(c=2, η=3), k=2) 11. Heart-Disease (N=1025, D=13(c=8, η=5), k=2) 12. ACA (N=690, D=14(c=8, η=6), k=2)

* N = Number of data, D = Number of Dimension(c = categorical, η = numerical), k = Number of Center

Artificial Gorilla Troops Optimizer (GTO) [13], Jaya Algorithm (JAYA) [7], Dandelion Optimizer (DO) [10], Gray Wolf Optimization (GWO) [6], modified particle swarm optimization (MPSO) [23], Leader Slime Mould Algorithm (LSMA) [11], Flow Direction Algorithm (FDA) [12], Mountain Gazelle Optimizer (MGO) [14], Prairie Dog Optimization Algorithm (PDO1) [15], Time-varying Acceleration Coefficients Particle Swarm Optimization algorithm (TACPSO) [23], Chimp Optimization Algorithm [16], and Chaotic League Championship Algorithm (KSCLCA) [8].

Table 2, the analysis encompasses numeric, categorical, and mixed data types. The average tied rank of 15 algorithms is determined based on the F-Measure. The average tied rank (Avg. tied rank) displayed in Table 2 represents the mean scores of tied rank scores for the F-Measure of each algorithm when implemented on datasets belonging to their respective data types. For numeric data, CMGO achieves the top rank with an average score (avg.sc) of 4.27, outperforming the original MGO ranking which holds the 6 positions with an avg.sc of 6.55. When considering Categorical data, the algorithm ranking first obtains an avg.sc of 4.40, demonstrating superior performance compared to the original MGO ranking at the 8 positions with an avg.sc of 7.70. In the case of Mixed data, the algorithm secures rank 3 with an avg.sc of 6.46, surpassing the original MGO ranking at the 6 positions with an avg.sc of 6.88.

Table 3 presents a thorough evaluation of data clustering performance, comparing the MGO and the proposed CMGO algorithms across three groups of datasets. The findings consistently indicated that CMGO outperformed the original MGO algorithm across the datasets, with a ratio of 18:9 in favor of CMGO. The performance measures utilized for evaluation were the F-Measure and tied rank.

Table 2. Compares the tied rank of 15 algorithms across three datatypes.

Methods	Tied rank of average F-measure					
	Numeric		Categorical		Mixed data type	
	Avg. tied rank	Rank	Avg. tied rank	Rank	Avg. tied rank	Rank
OAVOA	6.86	7	8.80	10	7.38	7
SSA	9.18	10	7.20	7	8.21	8
GTO	4.82	3	6.60	5	9.92	13
JAYA	4.32	2	9.40	11	8.88	10
DO	7.91	9	8.70	9	6.38	2
GWO	9.55	12	10.20	13	9.54	12
MPSO	12.73	14	9.80	12	10.33	14
LSMA	9.50	11	5.60	2	8.29	9
FDA	5.91	4	6.40	4	6.54	4
MGO	6.55	6	7.70	8	6.88	6
PDO1	11.18	13	7.00	6	11.08	15
TACPSO	7.73	8	6.10	3	6.58	5
CHIMP	13.36	15	10.60	14	9.04	11
KSCLCA	6.14	5	11.50	15	**4.50**	**1**
Proposed CMGO	**4.27**	**1**	**4.40**	**1**	6.46	3

Table 4 presents the average ranking, tied rank, and average tied rank for all 3 types of information based on the F-Measure. The rankings are associated with a set of 15 algorithms and all datasets. The algorithms are ranked as follows: CMGO, FDA, KSCLCA, MGO, TACPSO, JAYA, GTO, DO, OAVOA, LSMA, SSA, GWO, PDO1, CHIMP, and MPSO, respectively. CMGO achieved the first rank. In contrast, MGO obtained a rank of 4. These results highlight the capability of the proposed CMGO approach to enhance the performance of the original MGO algorithm for all datasets.

Table 3. The comparative of the tied rank between the MGO and proposed CMGO algorithms.

Types	Datasets	F-Measure		Rank	
		MGO	Proposed CMGO	MGO	Proposed CMGO
Numerical datasets	Iris	85.81717	**89.80941**	2	**1**
	Glass	**43.67836**	38.16387	**1**	2
	Breast-Cancer-Wisconsin	**95.12551**	95.11379	**1**	2
	Wine	72.46922	**72.74876**	2	**1**
	Thyroid	67.37993	**72.45515**	2	**1**
	Synthetic-Control-Chart	42.79285	**43.47290**	2	**1**
	Ionosphere	**68.18605**	68.13166	**1**	2
	Sonar	**53.93852**	49.75967	**1**	2
	Diabetes	59.02891	**59.55515**	2	**1**
	Ecoli	72.21830	**74.65103**	2	**1**
	Banknote-Authentication	59.07605	**59.30723**	2	**1**
Categorical datasets	Balance-Scale	50.10709	**50.11917**	2	**1**
	Hayes-Roth	36.78197	**37.66778**	2	**1**
	Monks	49.69362	**49.71152**	2	**1**
	SPECT-Heart	62.98250	**63.10270**	2	**1**
	Nursery	38.50658	**40.71493**	2	**1**
Mixed-data type datasets	Acute-Inflammations	**77.41939**	74.90621	**1**	2
	Analcatdata-Seropositive	**82.81519**	82.79930	**1**	2
	Churn	57.12235	**59.17942**	2	**1**
	Cloud	**56.49374**	**56.49374**	**1**	**1**
	Fruitfly	52.90640	**52.94304**	2	**1**
	Haberman	50.53389	**51.79722**	2	**1**
	Newton-Hema	**63.48013**	63.43048	**1**	2
	Sleuth-Case2002	56.90424	**59.32102**	2	**1**
	Socmob	85.28709	**86.42502**	2	**1**
	Tae	62.99309	**63.16615**	2	**1**
	Heart-Disease	**75.16155**	66.17032	**1**	2
	ACA	**67.83375**	56.90297	**1**	2

Table 4. The comparative analysis of the tied rank of 15 algorithms, utilizing the F-Measure metric computed for all 28 datasets.

Types	Datasets	Methods														
		OAVOA	SSA	GTO	JAYA	DO	GWO	MPSO	LSMA	FDA	MGO	PDO1	TACPSO	CHIMP	KSCLCA	Proposed
	Iris	12.0	6.0	8.0	**2.5**	10.0	10.0	7.0	10.0	**2.5**	13.0	14.0	**2.5**	15.0	**2.5**	5.0
Numerical datasets	Glass	4.0	13.0	5.0	3.0	7.0	11.0	12.0	9.0	6.0	2.0	15.0	8.0	14.0	**1.0**	10.0
	Breast-Cancer-Wisconsin	4.0	3.0	8.5	6.0	6.0	12.0	15.0	10.0	11.0	2.0	**1.0**	13.0	14.0	8.5	6.0
	Wine	6.0	9.0	7.0	5.0	3.0	4.0	13.0	11.0	10.0	8.0	15.0	12.0	14.0	**1.0**	2.0
	Thyroid	3.0	14.0	5.0	4.0	6.0	7.0	12.0	10.0	2.0	9.0	15.0	8.0	11.0	13.0	**1.0**
	Synthetic-Control-Chart	6.0	7.0	**1.0**	8.0	14.0	15.0	13.0	11.0	5.0	4.0	9.0	10.0	12.0	3.0	2.0
	Ionosphere	6.5	9.0	**1.5**	3.0	11.0	12.0	15.0	10.0	6.5	**1.5**	14.0	8.0	13.0	5.0	4.0
	Sonar	5.0	7.0	2.0	3.0	9.0	13.0	15.0	11.0	**1.0**	4.0	14.0	10.0	12.0	6.0	8.0
	Diabetes	10.0	10.0	7.0	10.0	10.0	2.0	14.0	4.0	10.0	5.5	15.0	3.0	13.0	5.5	**1.0**
	Ecoli	7.0	10.0	4.0	2.0	6.0	13.0	14.0	11.0	**1.0**	8.0	9.0	3.0	15.0	12.0	5.0
	Banknote-Authentication	12.0	13.0	4.0	**1.0**	5.0	6.0	10.0	7.5	10.0	15.0	2.0	7.5	14.0	10.0	3.0
Categorical datasets	Balance-Scale	14.0	6.0	7.0	4.0	10.0	8.0	12.0	11.0	3.0	2.0	9.0	5.0	13.0	15.0	**1.0**
	Hayes-Roth	13.0	6.0	**1.0**	7.0	12.0	9.0	4.0	5.0	8.0	15.0	3.0	10.0	2.0	14.0	11.0
	Monks	**1.0**	9.0	11.0	13.0	6.5	10.0	12.0	2.0	4.0	6.5	14.0	6.5	15.0	6.5	3.0
	SPECT-Heart	5.0	11.0	6.0	13.0	3.0	10.0	12.0	9.0	14.0	2.0	4.0	7.0	8.0	15.0	**1.0**
	Nursery	11.0	4.0	8.0	10.0	12.0	14.0	9.0	**1.0**	3.0	13.0	5.0	2.0	15.0	7.0	6.0
Mixed-data type datasets	Acute-Inflammations	5.0	**1.0**	9.0	15.0	12.0	13.0	11.0	8.0	6.0	2.0	3.0	10.0	14.0	7.0	4.0
	Analcatdata-Seropositive	11.0	5.0	14.0	12.0	3.0	6.0	13.0	2.0	9.0	7.0	15.0	4.0	10.0	**1.0**	8.0
	Churn	3.0	6.0	9.0	2.0	13.0	7.0	12.0	10.0	4.0	14.0	**1.0**	8.0	5.0	15.0	11.0
	Cloud	8.5	8.5	**1.0**	8.5	8.5	8.5	2.0	8.5	8.5	8.5	15.0	8.5	8.5	8.5	8.5

(*continued*)

Table 4. (*continued*)

Types	Datasets	Methods														
		OAVOA	SSA	GTO	JAYA	DO	GWO	MPSO	LSMA	FDA	MGO	PDO1	TACPSO	CHIMP	KSCLCA	Proposed
	Fruitfly	4.0	13.0	8.0	9.0	7.0	12.0	10.0	11.0	3.0	6.0	15.0	**1.0**	14.0	2.0	5.0
	Haberman	14.0	7.0	10.0	5.0	8.0	9.0	**1.0**	3.0	4.0	15.0	2.0	11.0	6.0	13.0	12.0
	Newton-Hema	6.0	5.0	10.0	9.0	4.0	15.0	11.0	13.0	7.0	**1.5**	12.0	8.0	14.0	**1.5**	3.0
	Sleuth-Case2002	5.0	11.0	10.0	12.0	6.0	9.0	14.0	13.0	7.0	8.0	15.0	3.0	**1.0**	2.0	4.0
	Socmob	10.0	3.0	13.0	12.0	8.0	9.0	15.0	4.0	7.0	6.0	14.0	5.0	11.0	**1.0**	2.0
	Tae	15.0	9.0	13.0	11.0	3.0	3.0	12.0	3.0	10.0	7.5	14.0	7.5	6.0	**1.0**	5.0
	Heart-Disease	3.0	15.0	10.0	6.0	2.0	14.0	12.0	11.0	7.0	4.0	13.0	5.0	9.0	**1.0**	8.0
	ACA	4.0	15.0	12.0	5.0	2.0	9.0	11.0	13.0	6.0	3.0	14.0	8.0	10.0	**1.0**	7.0
Average Rank		7.43	8.41	7.32	7.18	7.39	9.66	11.18	8.29	6.27	6.89	10.39	6.95	11.02	6.39	**5.23**
		9	11	7	6	8	12	15	10	2	4	13	5	14	3	**1**

5 Discussion

In this section, we compare the exploitation and exploration abilities of the original MGO algorithm and the proposed CMGO algorithm. The evaluation employed strategies such as TSM, MH, BMH, and MSF, revealing differences between the two algorithms (Fig. 2). The Socmob dataset, representing mixed data types, was used to compare their performance. Results indicated that in the original MGO algorithm, there was an initial emphasis on exploitation (as observed in the TSM strategy on the red line), which gradually increased until reaching the final iteration. While its exploration capability (as seen in the MH strategy on the green line) isn't heavily emphasized in the initial stages, it also gradually diminishes until reaching the final iteration. On the contrary, the CMGO algorithm intentionally reduced exploitation (as observed in the TSM strategy on the red line) to prevent premature convergence. However, it experienced a slight reduction that continued until the final iteration. To achieve a more effective balance between exploration and exploitation, the focus on the exploration capability (MH strategy on the green line) begins with less emphasis in the initial stages, increases rapidly during the intermediate stages, and then remains almost constant until the final iteration. The same behavioral curves can be observed for both BMH and MSF strategies. Note that the MSF strategy was removed from the CMGO algorithm due to the negligible changes observed throughout the evaluation process.

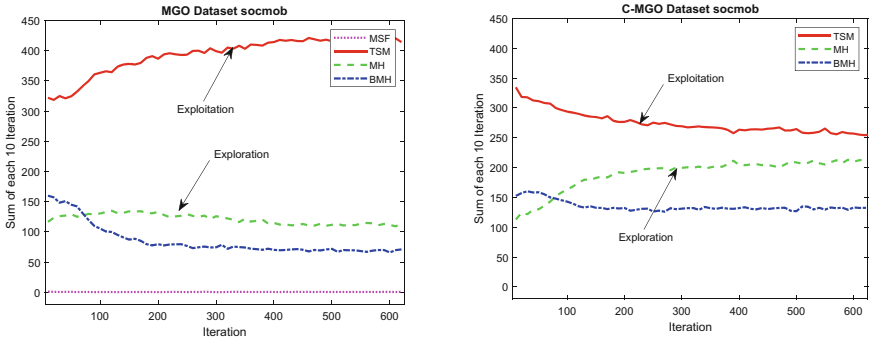

Fig. 2. The behaviors of the 4 strategies of the MGO and CMGO for Mixed data type.

It is worth noting, as shown in Table 2, that the mixed data type in all datasets exhibits only two classes (cluster center) and generally has a relatively small number of dimensions, while the other two data types, numerical and categorical, differ. It can be assumed that finding a solution for the problem of mixed data types with fewer classes is not challenging. Some algorithms with a high degree of exploitation ability, such as the KSCLCA algorithm, perform exceptionally well on this problem, ranking first. In contrast, the proposed CMGO algorithm dropped to third place in the context of mixed data types. It can be inferred that the CMGO algorithm aims to enhance the balance between exploration and exploitation abilities by reducing exploitation and increasing exploration, albeit to a lesser degree than the KSCLCA algorithm. Nevertheless, the

CMGO's performance ranks it among the top three algorithms, securing the third position, with a slight variation from the second-ranked DO algorithm. In summary, it can be inferred that our proposed CMGO algorithm demonstrates its effectiveness particularly when dealing with problems having more than two classes. Further exhaustive investigations will be pursued in future work.

6 Conclusion

We anticipate that the findings and techniques presented in this study will prove valuable to individuals and researchers who have a keen interest in advancing the field of data clustering. The CMGO algorithm, along with the integration of the Gower distance technique, offers novel insights and solutions for addressing challenges in clustering diverse data types. By introducing the CMGO algorithm, we have expanded the capabilities of the traditional MGO for K-means clustering. The incorporation of a chaotic map into the Territorial Solitary Males strategy and the exclusion of the Migration to Search for Food strategy have enhanced CMGO's exploration and exploitation abilities. This adjustment allows CMGO to effectively handle complex datasets by striking a balance between thorough exploration and efficient exploitation of the solution space. Furthermore, our utilization of the Gower distance technique has overcome the limitations of K-means clustering when dealing with categorical and binary data. This technique has enabled CMGO to accurately compute distances between objects and cluster centers, ensuring reliable clustering results across a wide range of data types. We believe that the comprehensive evaluation of CMGO against 14 other state-of-the-art algorithms using 28 diverse datasets adds significant value to the field. The use of the F-Measure metric and the tied rank test for statistical significance ranking provides robust and reliable measures of CMGO's performance. The results clearly demonstrate CMGO's superiority over the original MGO and other tested algorithms, particularly in clustering pure numeric and categorical data.

In summary, we are confident that the insights and innovations presented in this study will inspire further developments in the field of data clustering. The CMGO algorithm, along with the integration of the Gower distance technique, offers a promising avenue for researchers and practitioners to tackle the challenges posed by diverse datasets. We hope that our contributions will serve as a foundation for future advancements in the field and encourage further exploration and experimentation in this area of study.

Acknowledgments. The authors extend their gratitude to Tanachapong Wangchamhan, Ph.D. (tanachapong.wa@ksu.ac.th) for generously sharing his intellectual resources throughout the duration of this project.

References

1. Provost, F., Fawcett, T.: Data science and its relationship to big data and data-driven decision making. Big Data **1**(1), 51–59 (2013). https://doi.org/10.1089/big.2013.1508
2. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd edn. Springer, New York

3. Bahmani-Firouzi, B., Shasadeghi, M., Niknam, T.: A new hybrid algorithm based on PSO, SA, and K-means for cluster analysis. Int. J. Innov. Comput. Inform. Control **6**, 3177–3192 (2010)
4. Niknam, T., Amiri, B.: An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis. Appl. Soft Comput. J. **10**(1), 183–197 (2010). https://doi.org/10.1016/j.asoc.2009.07.001
5. Krishnasamy, G., Kulkarni, A.J., Paramesran, R.: A hybrid approach for data clustering based on modified cohort intelligence and K-means. Expert Syst. Appl. **41**(13), 6009–6016 (2014). https://doi.org/10.1016/j.eswa.2014.03.021
6. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey wolf optimizer. Adv. Eng. Softw. **69**, 46–61 (2014). https://doi.org/10.1016/j.advengsoft.2013.12.007
7. Venkata Rao, R.: Jaya: a simple and new optimization algorithm for solving constrained and unconstrained optimization problems. Int. J. Indust. Eng. Comput. **7**(1), 19–34 (2016). https://doi.org/10.5267/j.ijiec.2015.8.004
8. Wangchamhan, T., Chiewchanwattana, S., Sunat, K.: Efficient algorithms based on the k-means and Chaotic League Championship Algorithm for numeric, categorical, and mixed-type data clustering. Expert Syst. Appl. **90**, 146–167 (2017). https://doi.org/10.1016/j.eswa.2017.08.004
9. Mirjalili, S., Gandomi, A.H., Mirjalili, S.Z., Saremi, S., Faris, H., Mirjalili, S.M.: Salp Swarm Algorithm: a bio-inspired optimizer for engineering design problems. Adv. Eng. Softw. **114**, 163–191 (2017). https://doi.org/10.1016/j.advengsoft.2017.07.002
10. Zhao, S., Zhang, T., Ma, S., Chen, M.: Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications. Eng. Appl. Artif. Intell. **114** (2022). https://doi.org/10.1016/j.engappai.2022.105075
11. Naik, M.K., Panda, R., Abraham, A.: Normalized square difference based multilevel thresholding technique for multispectral images using leader slime mould algorithm. J. King Saud Univ. Comput. Inform. Sci. **34**(7), 4524–4536 (2022). https://doi.org/10.1016/j.jksuci.2020.10.030
12. Karami, H., Anaraki, M.V., Farzin, S., Mirjalili, S.: Flow Direction Algorithm (FDA): a novel optimization approach for solving optimization problems. Comput. Ind. Eng. **156** (2021). https://doi.org/10.1016/j.cie.2021.107224
13. Abdollahzadeh, B., Gharehchopogh, F.S., Mirjalili, S.: Artificial gorilla troops optimizer: a new nature-inspired metaheuristic algorithm for global optimization problems. Int. J. Intell. Syst. **36**(10), 5887–5958 (2021). https://doi.org/10.1002/int.22535
14. Abdollahzadeh, B., Gharehchopogh, F.S., Khodadadi, N., Mirjalili, S.: Mountain gazelle optimizer: a new nature-inspired metaheuristic algorithm for global optimization problems. Adv. Eng. Software **174**, 103282 (2022). https://doi.org/10.1016/j.advengsoft.2022.103282
15. Ezugwu, A.E., Agushaka, J.O., Abualigah, L., Mirjalili, S., Gandomi, A.H.: Prairie dog optimization algorithm. Neural Comput. Appl. **34**(22), 20017–20065 (2022). https://doi.org/10.1007/s00521-022-07530-9
16. Khishe, M., Mosavi, M.R.: Chimp optimization algorithm. Expert Syst. Appl. **149**, 113338 (2020). https://doi.org/10.1016/j.eswa.2020.113338
17. Jena, B., Naik, M.K., Panda, R., Abraham, A.: A novel minimum generalized cross entropy-based multilevel segmentation technique for the brain MRI/dermoscopic images. Comput. Biol. Med. **151**, 106214 (2022). https://doi.org/10.1016/j.compbiomed.2022.106214
18. Ben Ali, B., Massmoudi, Y.: K-means clustering based on gower similarity coefficient: a comparative study. In: 2013 5th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO), pp. 1–5 (2013). https://doi.org/10.1109/ICMSAO.2013.6552669
19. Saremi, S., Mirjalili, S., Lewis, A.: Biogeography-based optimisation with chaos. Neural Comput. Appl. **25**(5), 1077–1097 (2014). https://doi.org/10.1007/s00521-014-1597-x

20. Gower, J.C.: A general coefficient of similarity and some of its properties. Biometrics **27**(4), 857–871 (1971). https://doi.org/10.2307/2528823
21. Markelle, K., Rachel, L., Kolby, N.: UCI Dataset. The UCI Machine Learning Repository. https://archive.ics.uci.edu. Accessed 24 Jun 2023
22. Vanschoren, J., van Rijn, J.N., Bischl, B., Torgo, L.: OpenML: networked science in machine learning. SIGKDD Explor. **15**(2), 49–60 (2013). https://doi.org/10.1145/2641190.2641198
23. Mirjalili, S., Lewis, A., Sadiq, A.S.: Autonomous particles groups for particle swarm optimization. Arab. J. Sci. Eng. **39**(6), 4683–4697 (2014). https://doi.org/10.1007/s13369-014-1156-x