Sādhanā (2018) 43:190
https://doi.org/10.1007/s12046-018-0962-3

Data clustering using *K*-Means based on Crow Search Algorithm

K LAKSHMI[1,*], N KARTHIKEYANI VISALAKSHI[2] and S SHANTHI[1]

[1]Department of Computer Applications, Kongu Engineering College, Perundurai, India
[2]Department of Computer Science, Government Arts and Science College, Kangeyam, India
e-mail: klakshmisanthosh@gmail.com; karthichitru@yahoo.co.in; shanthi.kongumca@gmail.com

MS received 10 January 2017; revised 14 November 2017; accepted 13 May 2018; published online 22 October 2018

Abstract. Cluster analysis is one of the popular data mining techniques and it is defined as the process of grouping similar data. *K*-Means is one of the clustering algorithms to cluster the numerical data. The features of *K*-Means clustering algorithm are easy to implement and it is efficient to handle large amounts of data. The major problem with *K*-Means is the selection of initial centroids. It selects the initial centroids randomly and it leads to a local optimum solution. Recently, nature-inspired optimization algorithms are combined with clustering algorithms to obtain the global optimum solution. Crow Search Algorithm (CSA) is a new population-based metaheuristic optimization algorithm. This algorithm is based on the intelligent behaviour of the crows. In this paper, CSA is combined with the *K*-Means clustering algorithm to obtain the global optimum solution. Experiments are conducted on benchmark datasets and the results are compared to those from various clustering algorithms and optimization-based clustering algorithms. Also the results are evaluated with internal, external and statistical experiments to prove the efficiency of the proposed algorithm.

Keywords. Data mining; cluster analysis; *K*-Means; Particle Swarm Optimization; Crow Search Optimization algorithm.

1. Introduction

Data mining techniques extract knowledge from large amount of data. These techniques include classification, clustering, association rules, etc. Cluster analysis is the unsupervised technique grouping the data without knowing the class labels. Clustering is applied in many application areas such as biology, security, business intelligence and web search [1]. Clustering can be divided into two categories: hard and soft clustering. In hard clustering, the same object can belong to only a single cluster. In soft clustering, the same object can belong to different clusters.

Clustering algorithms are classified into two categories: partitional and hierarchical. Partitional clustering algorithms form the clusters by partition of the data objects into groups. Hierarchical clustering algorithms form the clusters by the hierarchical decomposition of data objects. *K*-Means clustering algorithm is one of the partitional clustering algorithms and it is popular and most widely used due to its simplicity and efficiency. It chooses the initial centroid randomly from the data objects and uses the Euclidean distance to measure the distance between the data objects and its cluster centroid. *K*-Means algorithm gives a local optimum solution due to its selection of initial centroids.

A number of optimization algorithms are developed to provide the global optimum solution. Optimization algorithms are categorized into heuristic and metaheuristic. Heuristic means 'to find' or 'to discover by trial and error' and 'meta' means 'beyond' or 'higher level' [2]. Some of the nature-inspired metaheuristic optimization algorithms are Genetic Algorithm [3, 4], Ant Colony Optimization [5], Simulated Annealing (SA) [6], Particle Swarm Optimization [7, 8], Tabu Search [9, 10], Cat Swarm Optimization [11], Artificial Bee Colony [12–14], Cuckoo Search Algorithm [15, 16], Gravitational Search Algorithm [17], Firefly Algorithm [18], Bat Algorithm [19], Wolf Search Algorithm [20] and Krill Herd [21].

Crow Search Algorithm (CSA) is one of the population-based metaheuristic optimization algorithm and it was introduced by Alireza Askarzadeh [22]. This algorithm simulates the intelligent behaviour of crows. Crows are considered as one of the world's most intelligent birds. This algorithm is based on finding the hidden storage position of excess food. Finding food source hidden by another crow is not a easy task because if a crow finds anyone following it, it tries to fool the crow by moving to another position.

To overcome the *K*-Means local optimum problem, in this paper a new clustering algorithm by hybridized Crow Search Optimization and *K*-Means clustering algorithms called CSAK Means is proposed.

*For correspondence

The organization of this paper is as follows. Section 2 describes the related researches in the literature. Section 3 describes the *K*-Means clustering algorithm and the CSA is discussed in section 4. Section 5 describes the proposed CSAK Means clustering algorithm. The experimental analysis is discussed in section 6. Conclusion and future works are provided in section 7.

2. Related works

In this section, some of the optimization algorithms approaches for clustering problems and hybridization of optimization algorithms with *K*-Means are discussed.

Ant Colony Optimization approach for clustering problem is given in [23]. SA algorithm approach for clustering problem was proposed in [24]. Particle Swarm Optimization approach for clustering problem is given in [25]. Tabu Search algorithm approach for clustering problem was proposed in [26]. Artificial Bee Colony Optimization approach for clustering problem is given in [27, 28]. Cat Swarm Optimization algorithm for clustering was proposed in [29].

Genetic Algorithm combined with *K*-Means was developed in [30]. Hybrid clustering algorithm based on *K*-Means and ant colony algorithm was proposed in [31]. Cluster analysis with *K*-Means and SA was introduced in [32]. *K*-Means clustering algorithm based on Particle Swarm Optimization was proposed in [33, 34]. Tabu-Search-based *K*-Means was developed in [35]. Artificial Bee Colony based *K*-Means algorithm was proposed in [36]. Combination of Gravitational Search algorithm with *K*-Means was introduced in [37]. Firefly Algorithm combined with *K*-Means was proposed in [38]. Bat Algorithm combined with *K*-Means was proposed in [39]. Wolf Search Algorithm, Cuckoo Search, Bat Algorithm, Firefly Algorithm and Ant Colony Optimization algorithms integrated with *K*-Means are introduced in [40].

These algorithms try to solve the *K*-Means local optimum solution, but they suffer from low-quality results and low convergence speed, complicated operators, complex structure and parameter setting issues.

3. *K*-Means Clustering Algorithm

K-Means is the most widely used and easy to implement clustering algorithm. It partitions the data objects into predefined K number of groups based on the data objects that are closest to the centroid. The main objective of *K*-Means clustering is to minimize total intra-cluster distance, or the squared error function. The squared error function is calculated using Eq. (1):

$$\sum_{j=1}^{K}\sum_{i=1}^{N} \| x_i^{(j)} - c_j \|^2 . \tag{1}$$

A dataset consists of N number of objects X_i, $i = 1, 2, \ldots, N$ with D number of features D_j, $j = 1, 2, \ldots, D$.

The *K*-Means clustering algorithm is described as follows:

i Input the number of clusters K.
ii Randomly select the K initial centroids $c_j, j = 1, 2, \ldots, K$ from the data objects.
iii Find the distance between each K-cluster centroid and the data objects using the formula

$$dis(x_i, c_j) = \sqrt{\sum_{j=1}^{d}(x_i - c_j)^2}. \tag{2}$$

iv Find the minimum distance and assign the data objects to clusters.
v Update the centroids using Eq. (3), i.e., calculate the mean of all data objects assigned to the cluster:

$$c_j = \frac{1}{N_j}\sum_{x_i \in s_j} x_i. \tag{3}$$

The *K*-Means algorithm is terminated when one of the following conditions is satisfied: (i) the average change in the centroids, (ii) the maximum number of iterations is reached and (iii) no change in the clustership of objects.

The main features of *K*-Means clustering are the following: (i) simple and easy to implement and (ii) can handle large amount of data objects efficiently. The main issues are the following: (i) needs the number clusters in advance, (ii) handles numeric data only and (iii) produces local optimum solutions.

4. CSA

The principles of CSA are the following: (i) crows live in the form of groups, (ii) remember the position of food hiding locations, (iii) follow each other for stealing food and (iv) protect their food source.

The number of crows, i.e., flock size, is P in D-dimensional environment and the position of the crow at iteration time i in the search space is specified as $X_{i,iter}$, $i = 1, 2, \ldots, N$; $iter = 1, 2, \ldots, itermax$; $itermax$ is the maximum number iterations. Each crow has a memory m to remember the position of the hiding place. At each iteration, the position of hiding place for crow i is specified by $m_{i,iter}$ and it shows the best position obtained so far. Metaheuristic algorithms should provide a good balance between diversification and intensification. In CSA, these

two are controlled by the Awareness Probability (AP) parameter.

The CSA is described as follows:

1. Initialize the parameters, number of flocks P, maximum number of iterations *itermax*, Flight Length FL and Awareness Probability AP.
2. Initialize the position of crows randomly in PD-dimensional search space.
3. Initialize the memory of the crows with position of crows.
4. Evaluate the position of the crows.
5. While *iter*<*maxiter*

   (a) for all crows

      i. randomly choose any one of the crows to follow (for example v);
      ii. if crow v does not know that crow μ is following it, new position of v is obtained using Eq. (4); if crow v does know that crow μ is following it, new position of v is obtained randomly:

$$\begin{cases} x^{i,it} + r_i \times FL^{i,it} \times (m^{j,it} - x^{i,it}) & r_j \geqslant AP^{j,it} \\ \text{a random position} & \text{otherwise} \end{cases} \tag{4}$$

      iii. check the feasibility of the new position; if the new position of crow is feasible, its position is updated; otherwise, the crow stays in the current position;
      iv. evaluate the new position of the crows using Eq. (1);
      v. update the memory of the crows using Eq. (5):

$$\begin{cases} x^{i,it+1} + f(x^{i,it+1}) \text{ is better than } f(m^{m^i,it}) \\ m^{i,it} \text{ otherwise} \end{cases} \tag{5}$$

6. End of while.

5. Proposed algorithm

The K-Means clustering algorithm is easy to implement and efficiently handles large datasets. The main drawback is that it produces local optimum solutions. To obtain the global optimum solution, K-Means is combined with global optimization algorithms. CSA is the metaheuristic global optimization algorithm and combined with K-Means to obtain the global optimum solution. In this section, CSA combined with K-Means algorithm is proposed.

The proposed CSAK Means algorithm is described as follows:

1. Input the values of number of clusters K, flock size N, maximum number of iterations *maxiter*, flight length FL and awareness probability AP.
2. Initialize the position of crows N and memory of crows M.
3. Generate the matrix of size $K*D$ with random numbers (number of features in the dataset). The maximum range of random numbers is the total number of instances in the data objects.
4. Encode the random numbers with the data objects. Each row specifies the K cluster centres for clustering algorithm. For example, if $K = 3$, $D = 4$, a single row looks as shown in figure 1.
5. Initialize the memory of the crows with the values of the positions of the crows because initially crows hid their foods at their initial positions.
6. Evaluate the fitness of initial position of crows using Eq. (1).
7. Initialize the fitness of memory of the crows with the fitness position of the crows.
8. Update the position of crows:

   (a) while iteration $\leq$ maxiter

      i. for all crows

         A. choose any one of the crows to follow randomly (for example μ);
         B. if crow μ does not know that crow v is following it, new position of μ is obtained using Eq. (4);

      C. if crow μ does know that crow v is following it, new position of μ is obtained randomly;
      D. check the feasibility of the new position; if the new position of crow is feasible, its position is updated; otherwise, the crow stays in the current position;

      ii. end of while;

   (b) evaluate the fitness of new position of crows using Eq. (1);

   (c) update the memory of the crows using Eq. (5).

9. Calculate the Euclidean distance from each data to best obtained solution centroid from CSA.

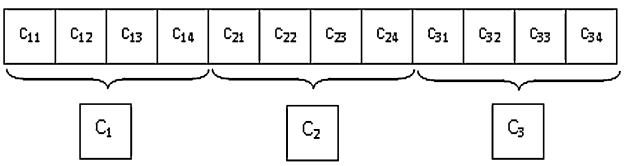

Figure 1. Encoding.

6. Experimental results

6.1 *Datasets*

To evaluate the performance of proposed CSAK Means algorithm, six benchmark datasets, Iris, Wine, Glass, Breast Cancer, Contraceptive Method Choice (CMC) and Haberman's Survival, are used. For each dataset the number of instances and number of classes are specified in table 1. These datasets are collected from UCI machine repository [41].

Iris: This dataset contains 150 samples of iris flower with 3 different species. The species include Setosa, Versicolour and Virginica. For each species there are 50 observations. The attributes in each species are sepal length, sepal width, petal length and petal width.

Wine: This dataset contains the chemical analysis of wines grown in the same region but derived from three different cultivars. There are 13 quantities found in each of the three types of wines.

Glass: This dataset contains the types of glass motivated by criminological investigation. At the scene of the crime, the glass left can be used as evidence, if it is correctly identified. There are 10 quantities found in each of the six types of glass.

Wisconsin Breast Cancer: This dataset contains the samples to identify the type of breast cancer. It is identified using 9 quantities found in each of the two types of breast cancer.

CMC: This dataset contains the samples of married women who were either not pregnant or did not know at the time of interview. The problem is to predict the current contraceptive method choice (no use, long-term methods or short-term methods) of a woman based on her demographic and socio-economic characteristics. There are 9 quantities found in each of the three types of choices.

Habermans Survival: This dataset contains the cases from a study that was conducted between 1958 and 1970 at the University of Chicago's Billings Hospital on the survival of patients who had undergone surgery for breast cancer. There are 3 quantities found in each of the two types of status.

6.2 *Measures*

The performance of CSAK Means is evaluated with internal and external measures. The internal measure used is Silhouette and the external measures used are Purity, Normalized Mutual Information, Rand Index and FMeasure. Also the convergence time and time taken for each iteration are compared for the algorithms. ANOVA and statistical tests for significance are also performed for all algorithms.

Table 1. Dataset details.

Dataset	No. of instances	No. of classes
Iris	150	3
Wine	178	3
Glass	214	6
Cancer	683	2
CMC	1473	3
Survival	306	2

6.2a *Purity:* Purity is the external evaluation measure to measure the quality of clustering algorithm. It is calculated as the count of all correct predictions divided by the total count of the data objects. It is calculated using Eq. (6):

$$purity(X, Y) = \frac{1}{N}\sum_{i=1}^{K} max_j|c_i \cap t_j|. \tag{6}$$

N is the total number of objects, K is the number of clusters, c_i is the cluster in C and t_j is maximum count for cluster c_i.

6.2b *Normalized Mutual Information:* Normalized Mutual Information is an external measure to validate the quality of clustering. It is the information theoretic measure on how well the predicted clusters and the actual clusters predict the normalized amount of information inherent from these two. It is calculated using Eq. (7):

$$NMI(X, Y) = \frac{2I(X, Y)}{[H(X) + H(Y)}. \tag{7}$$

X is the actual class label, Y is the label predicted by the algorithm, H is the entropy and $I(X;Y)$ is the mutual information between X and Y.

6.2c *Rand Index:* Rand Index is an external measure to find the similarity between actual labels and predicted labels. This measure has a value between 0 and 1, 0 indicating that the two data clusters do not agree on any pair of points and 1 indicating that the data clusters are exactly the same. Rand Index is calculated using Eq. (8):

$$RandIndex = \frac{TP + TN}{TP + FP + FN + TN}. \tag{8}$$

TP means True Positive; it is the count of similar objects in the same cluster. TN means True Negative; it is count of dissimilar objects in different clusters. FP means False Positive; it is the count of dissimilar objects in the same cluster. FN means False Negative; it is the count of similar objects in different clusters.

6.2d *FMeasure:* FMeasure is the external measure to obtain the accuracy of the clustering results. It is the harmonic mean of precision and recall. FMeasure can be computed using formula (9):

$$FMeasure = 2 \times \frac{precision \times recall}{precision + recall}. \tag{9}$$

Precision is calculated as the number of correct positive predictions divided by the total number of positive predictions. The best precision is 1, whereas the worst is 0. Precision is

Table 2. Algorithm-specific parameters.

Criteria	Iterations	Particles	Parameters
K-Means	20	N/A	K
K-Means++	20	N/A	K
Genetic *K*-Means	100	15	$\text{MP} = 0.05, K$
PSOK Means	100	15	$w = 0.79$, $c1 = 1.49$, $c2 = 1.49$, K
CSAK Means	100	15	$\text{fl} = 2, \text{AP} = 0.1, K$

Table 3. Fitness, measures and computation time values of Iris Dataset.

	Criteria	*K*-Means	*K*-Means++	Genetic *K*-Means	PSOK Means	CSAK Means
Fitness values	Best	97.33	97.33	97.22	97.19	97.12
	Worst	254.75	131.10	124.51	128.47	100.88
	Mean	115.01	103.41	100.78	101.33	97.74
	Std. dev.	47.35	11.68	9.59	10.97	1.27
Measures	Silhouette	0.7098	0.7251	0.7289	0.7167	0.7349
	Purity	32.00	44.00	44.67	88.67	89.33
	NMI	0.7582	0.7582	0.7419	0.7419	0.7582
	Rand Index	0.5462	0.6267	0.6311	0.9244	0.9289
	FMeasure	0.2857	0.4330	0.4410	0.8922	0.9002
Computation time	Conv. time	0.0725	0.0398	0.1000	0.1423	0.1504
	Iter. time	0.0105	0.0103	0.0031	0.0457	0.0470

calculated as true positive divided by the sum of false positive and true positive. It is calculated using Eq. (10):

$$precision = \frac{TP}{TP + FP}. \tag{10}$$

Recall is calculated as the number of correct positive predictions divided by the total number of positives. The best sensitivity is 1.0, whereas the worst is 0.0. It is calculated using Eq. (11):

$$recall = \frac{TP}{TP + FN}. \tag{11}$$

6.2e *Silhouette:* The silhouette is an internal measure that measures how similar an object is to its own cluster compared with other clusters. This measure combines both the cohesion and separation. It is calculated using Eq. (12):

$$sil(i) = \frac{b_i - a_i}{max(a_i, b_i)} \tag{12}$$

where a_i is the average dissimilarity of i with respect to all other objects within the same cluster and b_i is the average dissimilarity of i with respect to all other objects in other clusters.

6.2f *ANOVA*: "Analysis of Variance" is a statistical test and it determines whether there is any statistically significant difference between the means of two or more groups. A one-way ANOVA is used to find out whether the means of groups are significantly different from one another or each group is relatively the same.

The one-way ANOVA table has six columns: (i) source of variability, (ii) sum of squares (ss) of each source, (iii) degrees of freedom (df) of each source, (iv) mean square (MS) for each source, (v) F-statistic, the ratio of the MSs and (vi) probability, the corresponding p-value of F.

6.3 *Results*

The algorithms are implemented using Matlab R2012a on an Intel i5 of 2.30 GHz with 4 GB RAM. The *K*-Means, *K*-Means++, Genetic *K*-Means, PSOK Means and CSAK Means algorithms are executed in 10 distinct runs with parameters specified in table 2. The values for the Particle Swarm Optimization algorithm are suggested in [42]. The values for the CSA are suggested in [22].

The fitness values of *K*-Means, *K*-Means++, Genetic *K*-Means, PSOK Means and CSAK Means for all datasets are shown in tables 3–8. The ANOVA statistical test results are shown in tables 9–14. Figures 2–7 show a comparison of convergence behaviour of the datasets for all algorithms. The boxplot for the silhouette of fitness values is shown in figures 8–13.

6.4 *Discussion*

Table 3 shows the results of fitness, measures and computation time values of Iris Dataset. For the Iris Dataset, the CSAK Means provides the best solution and the standard deviation is also smaller than those of other algorithms. The

Table 4. Fitness, measures and computation time values of Wine Dataset.

	Criteria	*K*-Means	*K*-Means++	Genetic *K*-Means	PSOK Means	CSAK Means
Fitness values	Best	16529.85	18436.95	16487.30	18123.03	16346.58
	Worst	25209.18	22220.27	20036.32	19010.48	16555.68
	Mean	17377.78	19061.42	17192.79	18246.17	16516.30
	Std. dev.	2490.08	1209.04	741219.81	291.35	78.14
Measures	Silhouette	0.7097	0.7319	0.7046	0.7160	0.7316
	Purity	35.39	38.20	47.75	50.56	70.22
	NMI	0.4288	0.4241	0.4288	0.4140	0.4288
	Rand Index	0.5693	0.5880	0.6517	0.6704	0.8015
	FMeasure	0.3098	0.3754	0.5049	0.4238	0.7096
Computation time	Conv. time	0.0991	0.0200	0.1494	0.2174	0.1929
	Iter. time	0.0134	0.0140	0.0035	0.0565	0.0527

Table 5. Fitness, measures and computation time values of Glass Dataset.

	Criteria	*K*-Means	*K*-Means++	Genetic *K*-Means	PSOK Means	CSAK Means
Fitness values	Best	229.0584	215.7317	255.5493	218.9672	219.5148
	Worst	343.2669	244.2439	555.8802	277.4907	253.7976
	Mean	246.8569	219.2352	316.5006	229.1003	224.6620
	Std. dev.	37.7219	8.5506	88.3677	16.6631	10.2062
Measures	Silhouette	0.4993	0.6172	0.6954	0.4577	0.5221
	Purity	33.64	36.92	37.85	40.65	44.39
	NMI	0.3902	0.4293	0.4127	0.4270	0.4375
	Rand Index	0.7788	0.7897	0.7928	0.8022	0.8146
	FMeasure	0.2203	0.3145	0.3333	0.2829	0.4973
Computation time	Conv. time	0.1824	0.1796	0.2966	0.4919	0.4482
	Iter. time	0.0256	0.0250	0.0038	0.1191	0.1263

Table 6. Fitness, measures and computation time values of Cancer Dataset.

	Criteria	*K*-Means	*K*-Means++	Genetic *K*-Means	PSOK Means	CSAK Means
Fitness values	Best	2978.66	2988.43	2988.43	2988.43	2982.54
	Worst	4990.39	3493.54	5248.62	3584.56	3205.89
	Mean	3271.34	3003.11	3071.94	3006.44	2993.07
	Std. dev.	660.86	85.35	389.60	100.69	37.05
Measures	Silhouette	0.6970	0.7550	0.7428	0.7535	0.7555
	Purity	96.19	96.05	96.05	96.05	96.19
	NMI	0.7546	0.7478	0.7478	0.7478	0.7546
	Rand Index	0.9619	0.9605	0.9605	0.9605	0.9619
	FMeasure	0.9580	0.9564	0.9564	0.9564	0.9580
Computation time	Conv. time	0.4438	2.2741	2.5487	2.6082	3.0861
	Iter. time	0.0607	0.0506	0.0053	0.1588	0.1645

internal and external index solutions of CSAK Means are better than those of other algorithms. The convergence time and time for each iteration for CSAK Means are higher than those of other algorithms.

Table 4 shows the fitness, measures and computation time values of of fitness values of Wine Dataset. For the Wine Dataset, the CSAK Means provides the best solution and the standard deviation is also smaller than those of other algorithms. The internal and external index solutions of CSAK Means are better than those of other algorithms. The convergence time and time for each iteration for CSAK Means are lower and higher, respectively, than those of other algorithms except PSOK Means.

Table 5 shows the fitness, measures and computation time values of of fitness values of Glass Dataset. For the Glass Dataset, *K*-Means++ provides the best solution. The internal measure silhouette for Genetic *K*-Means is better than those of other algorithms. The external measure index

Table 7. Fitness, measures and computation time values of CMC Dataset.

	Criteria	*K*-Means	*K*-Means++	Genetic *K*-Means	PSOK Means	CSAK Means
Fitness values	Best	5542.18	5541.16	5545.3334	5542.18	5544.88
	Worst	7983.1	8035.95	6540.7506	6982.62	5746.41
	Mean	5745.35	5587.27	5558.9578	5562.16	5548.08
	Std. dev.	550.47	299.67	115.6987	167.42	23.37
Measures	Silhouette	0.6192	0.6231	0.6478	0.6367	0.6481
	Purity	38.43	39.0360	40.12	39.71	40.12
	NMI	0.0331	0.0325	0.0318	3.3089	0.6008
	Rand Index	0.5895	0.5936	0.6008	0.5981	0.4161
	FMeasure	0.3736	0.4033	0.4161	0.4096	0.0318
Computation time	Conv. time	1.5124	16.5272	18.1365	19.2167	19.2082
	Iter. time	1.5124	0.2142	0.0055	0.4813	0.4912

Table 8. Fitness, measures and computation time values of Survival Dataset.

	Criteria	*K*-Means	*K*-Means++	Genetic *K*-Means	PSOK Means	CSAK Means
Fitness values	Best	2629.0500	2626.7598	3196.5920	2626.4104	2580.4919
	Worst	5574.2265	3805.5485	3425.4870	3112.1183	2626.8107
	Mean	2975.4275	2710.7807	3224.5040	2658.0138	2623.8713
	Std. dev.	925.9800	294.2829	60.9140	121.1427	11.5678
Measures	Silhouette	0.5240	0.5594	0.5536	0.5578	0.5660
	Purity	50.0000	52.2876	75.82	51.96	52.29
	NMI	0.0012	0.0001	0.0785	0.0177	0.0001
	Rand Index	0.5000	0.5229	0.7582	0.5196	0.5229
	FMeasure	0.4806	0.5051	0.6493	0.4925	0.5051
Computation time	Conv. time	0.1291	0.2723	0.3522	0.4316	0.4324
	Iter. time	0.0175	0.0181	0.0039	0.0768	0.0744

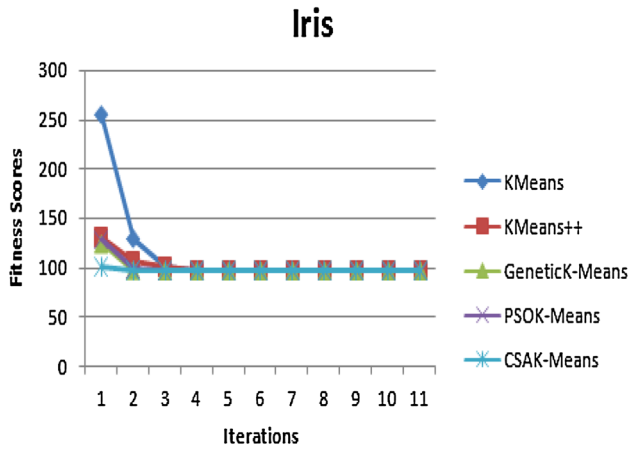

Figure 2. Fitness values of Iris Dataset.

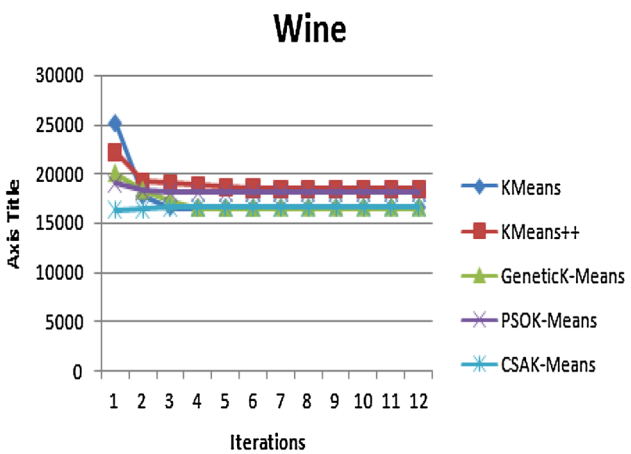

Figure 3. Fitness values of Wine Dataset.

solutions of CSAK Means are better than those of other algorithms. The convergence time and time for each iteration for CSAK Means are lower and higher, respectively, than those of other algorithms.

Table 6 shows the fitness, measures and computation time values of Cancer Dataset. For the Cancer Dataset, CSAK Means provides the best solution. The internal and external measure index solutions of CSAK Means are better than those of other algorithms. The convergence time and time for each iteration for CSAK Means are lower and higher, respectively, than those of other algorithms.

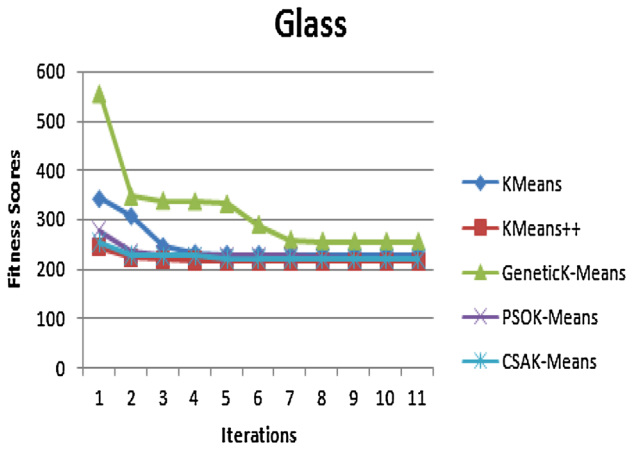

Figure 4. Fitness values of Glass Dataset.

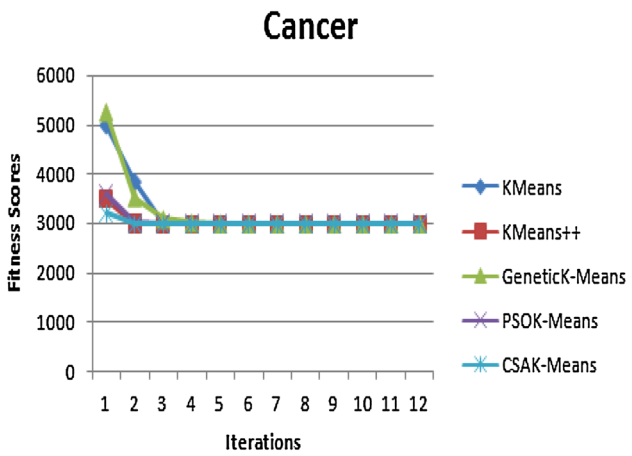

Figure 5. Fitness values of Cancer Dataset

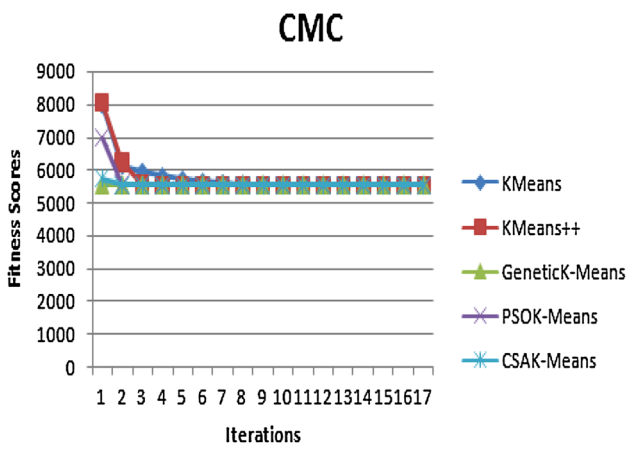

Figure 6. Fitness values of CMC Dataset.

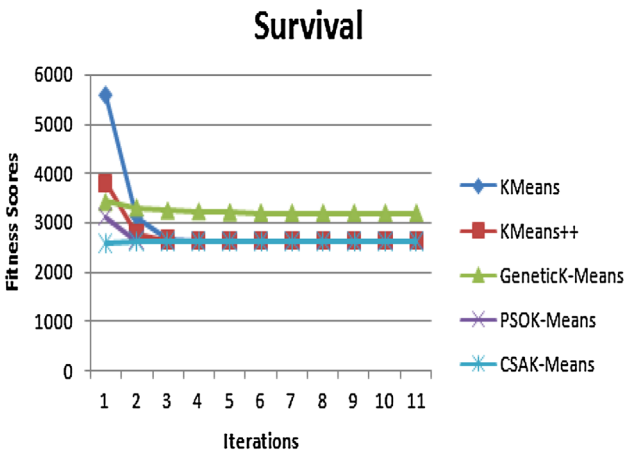

Figure 7. Fitness values of Survival Dataset.

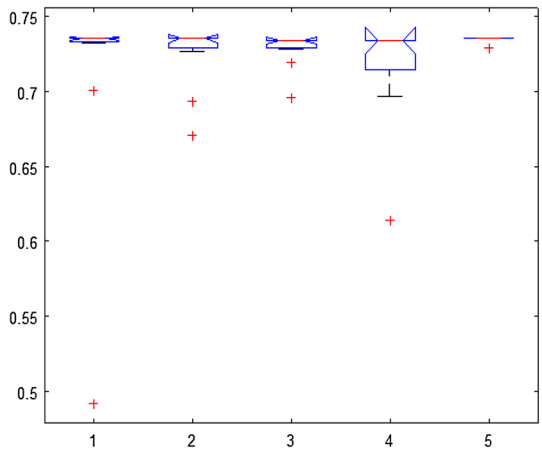

Figure 8. Boxplot view of Iris Dataset.

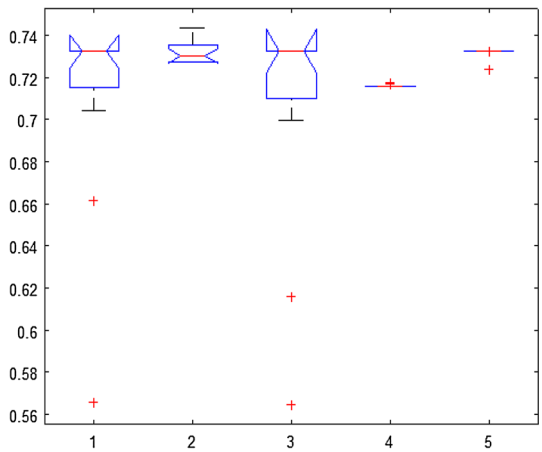

Figure 9. Boxplot view of Wine Dataset.

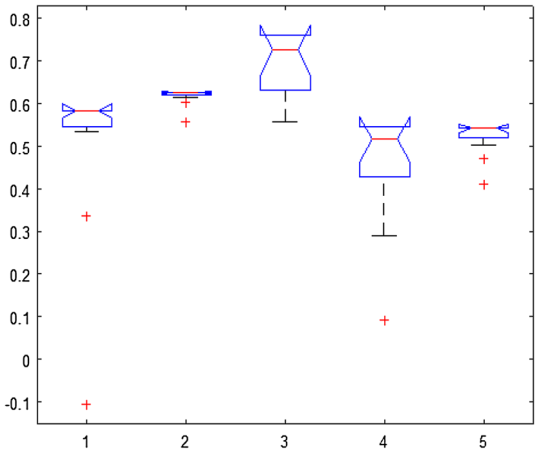

Figure 10. Boxplot view of Glass Dataset.

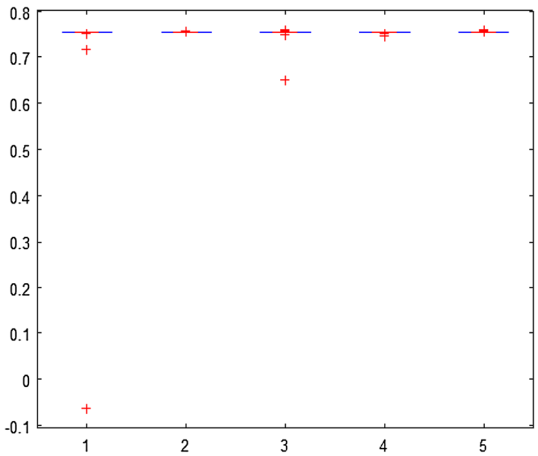

Figure 11. Boxplot for Cancer Dataset.

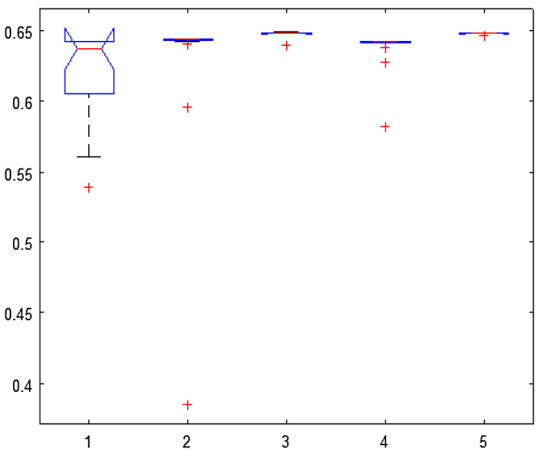

Figure 12. Boxplot view of CMC Dataset.

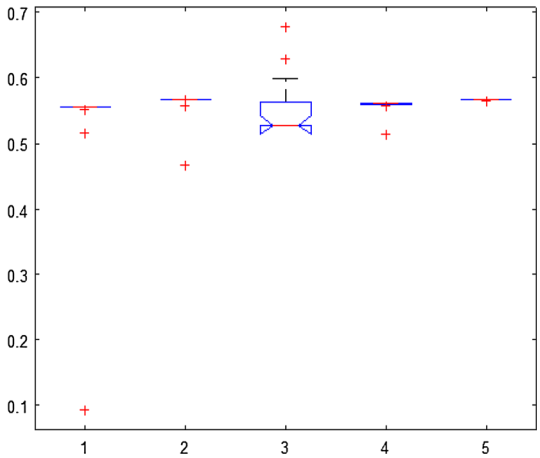

Figure 13. Boxplot view of Survival Dataset.

Table 7 shows the fitness, measures and computation time values of fitness values of CMC Dataset. For the CMC Dataset, *K*-Means++ provides the best solution but the worst, average and standard deviation of CSAK Means are better than those of other algorithms. The internal measure silhouette for CSAK Means is better than those of other algorithms. The external measure index solutions of CSAK Means and Genetic *K*-Means are the same. These values are better than those of other algorithms. The convergence time for CSAK Means is higher than those of other algorithms except PSOK Means. The time taken for each iteration is higher than those of all algorithms.

Table 8 shows the fitness, measures and computation time values of fitness values of Survival Dataset. For the Survival dataset, CSAK Means provides the best solution. The internal and external measure values of CSAK Means are better than those of other algorithms. The convergence time and time taken for each iteration for CSAK Means are higher than those of other algorithms.

Tables 9–14 show the results of ANOVA test results. The reason behind the ANOVA test is to test if there is any significance between the accuracies of the algorithms. The null hypothesis for an ANOVA is no significant differences among the groups and the alternative hypothesis is there is significant difference among the groups. Here, in all cases where Prob>*F*, the null hypothesis is rejected and alternative hypothesis is accepted; this implies that accuracies of all algorithms are not equal.

7. Conclusion and future work

In this paper, hybridized CSA and *K*-Means clustering algorithm is proposed and this new algorithm is called CSAK Means. The results of proposed algorithm are compared to those of *K*-Means, *K*-Means++, Genetic *K*-Means and PSOK Means algorithms. To evaluate the

Table 9. ANOVA test results of Iris Dataset.

Source	SS	df	MS	*F*	Prob > *F*
Columns	0.0043178	4	0.0010795	0.74684	0.56476
Error	0.072268	50	0.0014454		
Total	0.076586	54			

Table 10. ANOVA test results of Wine Dataset.

Source	SS	df	MS	*F*	Prob>*F*
Columns	0.0075128	4	0.0018782	1.6807	0.16759
Error	0.061462	55	0.0011175		
Total	0.068974	59			

Table 11. ANOVA test results of Glass Dataset.

Source	SS	df	MS	*F*	Prob>*F*
Columns	0.44614	4	0.11154	8.2101	2.8888e–05
Error	0.74718	55	0.013585		
Total	1.1933	59			

Table 12. ANOVA test tesults of Cancer Dataset.

Source	SS	df	MS	*F*	Prob>*F*
Columns	0.03828	4	0.00957	1.0604	0.38267
Error	0.63173	70	0.0090247		
Total	0.67001	74			

Table 13. ANOVA test results of CMC Dataset.

Source	SS	df	MS	*F*	Prob>*F*
Columns	0.010964	4	0.002741	2.339	0.063528
Error	0.082031	70	0.0011719		
Total	0.092995	74			

Table 14. ANOVA test results of Survival Dataset.

Source	SS	df	MS	*F*	Prob>*F*
Columns	0.017124	4	0.0042811	1.3249	0.26851
Error	0.24234	75	0.0032312		
Total	0.25946	79			

CSAK Means algorithm, fitness function used here is Mean Square Error Criterion. Afore-mentioned experimental results show that CSA outperforms the *K*-Means, *K*-Means++, Genetic *K*-Means and PSOK Means algorithms. In Genetic Algorithm, three operators, namely selection, crossover and mutation, need to be applied. PSO needs four parameters, namely inertia weight, individual learning factor, social learning factor and maximum velocity. CSA

needs the two parameters *AP* and *FL*. Each optimization algorithm has its own parameters and it is tedious to fix the optimum values for each parameter. In future, this is extended to dynamically determine the number of clusters.

References

[1] Han J, Pei J and Kamber M 2011 *Data mining: concepts and techniques*. Elsevier, United States

[2] Yang X S 2008 *Introduction to computational mathematics*. World Scientific, Singapore

[3] Holland J H 1975 *Adaption in natural and artificial systems*. Ann Arbor, MI: The University of Michigan Press

[4] Goldberg D 1989 *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley, United States

[5] Dorigo M 1992 *Optimization, learning and natural algorithms*. PhD Thesis, Politecnico di Milano

[6] Brooks S P and Morgan B J 1995 Optimization using simulated annealing. *The Statistician* 44(2): 241–257

[7] Eberhart R and Kennedy J 1995 A new optimizer using particle swarm theory. In: *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, MHS'95, pp. 39–43, IEEE

[8] Kennedy J and Eberhart R 1995 Particle swarm optimization. In: *Proceedings of the IEEE International Conference on Neural Networks*, Perth, WA, vol. 4, pp. 1942–1948

[9] Glover F and Laguna M 1997 *Tabu search*. Boston: Kluwer

[10] Holland J H 1975 *Adaptation in natural and artificial systems: an introductory analysis with application to biology, control, and artificial intelligence*. Ann Arbor, MI: University of Michigan Press, pp. 439–444

[11] Chu S C, Tsai P W and Pan J S 2006 Cat swarm optimization. In: *Proceedings of the Pacific Rim International Conference on Artificial Intelligence*. Berlin–Heidelberg: Springer, pp. 854–858

[12] Basturk B and Karaboga D 2006 An artificial bee colony (ABC) algorithm for numeric function optimization. In: *Proceedings of the IEEE Swarm Intelligence Symposium*, Indianapolis, Indiana, USA

[13] Karaboga D and Basturk B 2007 A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization* 39(3): 459–471

[14] Karaboga D and Basturk B 2007 Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems. In: *Proceedings of the International Fuzzy Systems Association World Congress*. Berlin–Heidelberg: Springer, pp. 789–798

[15] Yang X S and Deb S 2009 Cuckoo search via Levy flights. In: *Proceedings of the World Congress on Nature and Biologically Inspired Computing*, NaBIC 2009, IEEE, pp. 210–214

[16] Yang X S and Deb S 2014 Cuckoo search: recent advances and applications. *Neural Computing and Applications* 24(1): 169–174

[17] Rashedi E, Nezamabadi-Pour H and Saryazdi S 2009 GSA: a gravitational search algorithm. *Information Sciences* 179(13): 2232–2248

[18] Yang X S 2010 Firefly algorithm, Levy flights and global optimization. In: *Proceedings of Research and Development in Intelligent Systems XXVI*. London: Springer, pp. 209–218

[19] Yang X S 2010 A new metaheuristic bat-inspired algorithm. In: *Proceedings of Nature Inspired Cooperative Strategies for Optimization*, NICSO 2010. Berlin–Heidelberg: Springer, pp. 65–74

[20] Tang R, Fong S, Yang X S and Deb S 2012 Wolf search algorithm with ephemeral memory. In: *Proceedings of the Seventh International Conference on Digital Information Management* (ICDIM), IEEE, pp. 165–172

[21] Gandomi A H and Alavi A H 2012 Krill herd: a new bio-inspired optimization algorithm. *Communications in Nonlinear Science and Numerical Simulation* 17(12): 4831–4845

[22] Askarzadeh A 2016 A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. *Computers and Structures* 169: 1–12

[23] Shelokar P S, Jayaraman V K and Kulkarni B D 2004 An ant colony approach for clustering. *Analytica Chimica Acta* 509(2): 187–195

[24] Selim S Z and Alsultan K 1991 A simulated annealing (SA) algorithm for the clustering problem. *Pattern Recognition* 24(10): 1003–1008

[25] Chen C Y and Ye F 2004 Particle swarm optimization algorithm and its application to clustering analysis. In: *Proceedings of the IEEE International Conference on Networking, Sensing and Control*, IEEE, vol. 2, pp. 789–794

[26] Al-Sultan K S 1995 A tabu search approach to the clustering problem. *Pattern Recognition* 28(9): pp.1443–1451

[27] Zhang C, Ouyang D and Ning J 2010 An artificial bee colony approach for clustering. *Expert Systems with Applications* 37(7): 4761–4767

[28] Karaboga D and Ozturk C 2011 A novel clustering approach: Artificial Bee Colony (ABC) algorithm. *Applied Soft Computing* 11(1): 652–657

[29] Santosa B and Ningrum M K 2009 Cat swarm optimization for clustering. In: *Proceedings of the International Conference on Soft Computing and Pattern Recognition*, SOCPAR'09, IEEE, pp. 54–59

[30] Krishna K and Murty M N 1999 Genetic *K*-means algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 29(3): 433–439

[31] Lu J and Hu R 2013 A new hybrid clustering algorithm based on *K*-means and ant colony algorithm. In: *Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering* (ICCSEE 2013)

[32] Sun LX, Xu F, Liang Y Z, Xie Y L and Yu R Q 1994 Cluster analysis by the *K*-means algorithm and simulated annealing. *Chemometrics and Intelligent Laboratory Systems* 25(1): 51–60

[33] Van der Merwe D W and Engelbrecht A P 2003 Data clustering using particle swarm optimization. In: *Proceedings of the 2003 Congress on Evolutionary Computation*, CEC'03, IEEE, vol. 1, pp. 215–220

[34] Ahmadyfard A and Modares H 2008 Combining PSO and *k*-means to enhance data clustering. In: *Proceedings of the International Symposium on Telecommunications*, IEEE, pp. 688–691

[35] Liu Y, Liu Y, Wang L and Chen K 2005 A hybrid tabu search based clustering algorithm. In: *Proceedings of the International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. Berlin–Heidelberg: Springer, pp. 186–192

[36] Armano G and Farmani M R 2014 Clustering analysis with combination of artificial bee colony algorithm and *k*-means technique. *International Journal of Computer Theory and Engineering* 6(2): 141

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

[38] Hassanzadeh T and Meybodi M R 2012 A new hybrid approach for data clustering using firefly algorithm and *K*-means. In: *Proceedings of the CSI International Symposium on Artificial Intelligence and Signal Processing* (AISP), IEEE, pp. 007–011

[39] Komarasamy G and Wahi A 2012 An optimized *K*-means clustering technique using bat algorithm. *European Journal of Scientific Research* 84(2): 263–273

[40] Tang R, Fong S, Yang, X S and Deb S 2012 Integrating nature-inspired optimization algorithms to *K*-means clustering. In: *Proceedings of the Seventh International Conference on Digital Information Management* (ICDIM), IEEE, pp. 116–123

[41] Asuncion A and Newman D 2007 *UCI machine learning repository*

[42] Van den Bergh F 2002 *An analysis of particle swarm optimizers*. PhD Thesis, Department of Computer Science, University of Pretoria, Pretoria, South Africa