Trends in Applied Sciences Research

ISSN 1819-3579

Academic
Journals Inc.

www.academicjournals.com

Trends in Applied Sciences Research 7 (2): 103-117, 2012
ISSN 1819-3579 / DOI: 10.3923/tasr.2012.103.117

Iterative Simulated Annealing for Medical Clustering Problems

R. Kittaneh, S. Abdullah and A. Abuhamdah
Data Mining and Optimisation Research Group, Center for Artificial Intelligence Technology, University Kebangsaan Malaysia, 43600 UKM, Bangi Selangor, Malaysia

Corresponding Author: R. Kittaneh, Data Mining and Optimisation Research Group, Center for Artificial Intelligence Technology, University Kebangsaan Malaysia, 43600 UKM, Bangi Selangor, Malaysia

ABSTRACT

Clustering problem is a type of classification under optimisation problems, which is considered as a critical area of Data Mining. Medical clustering problem is a type of unsupervised learning in data mining. This study has presented an enhancement of K-Means (i.e., Multi K-Means) and iterative simulated annealing algorithm for solving medical clustering problems. The aim of this study was to improve the K-Means algorithm for a better performance and produce an effective algorithm for partitioning N objects into K clusters. The structure of the Iterative Simulated Annealing (ISA) algorithm resembles a Simulated Annealing (SA) algorithm structure. The basic difference is that, in ISA the temperature is reinitialized for further improvement, whilst, in SA the temperature is initialized only once at the beginning of the search. Therefore, ISA has a better capability of escaping from a local optima compared to SA and attempts to enhance the trial solution by exploring different neighborhood structures to overcome the limitation of the SA and by the swap mechanism ISA is able to get further improve. Experimental results obtained by three way of calculating the minimal distance that have been tested on six benchmark medical datasets that are available in UCI Machine Learning Repository show that, ISA algorithm with more computational time (coded as IISA) is able to produce significantly good quality solutions and outperformed SA and ISA algorithms on all datasets.

Key words: Clustering, un-supervised learning, multi-K-Means, simulated annealing, iterative simulated annealing, temperature re-initialization

INTRODUCTION

Clustering is an important unsupervised data mining technique, which divides the input space into K regions based on some similarity/dissimilarity measures, where the K value might not be known priori (Saha *et al*., 2010). A cluster is referred as a group of similar objects (data) and dissimilar from other groups (in different clusters). This process is called clustering which is considered as a NP-hard problem (Barthelemy and Brucker, 2001). A cluster can be considered as a form of data compression. Recently, the problem with two clusters in a general Euclidean space is also considered as an NP-hard problem (Dasgupta and Freund, 2009; Mahajan *et al*., 2009).

Clustering aims to minimize and maximize the variance within-group and between-group, respectively which will result a number of heterogeneous groups with homogeneous contents. There are substantial differences between the groups, but the individuals within a single group are similar.

Clustering methods can be classified into five types (i.e., Density based, Grid based, Module based, Hierarchical and Partitioning methods). However, the existing algorithms for data clustering

can be classified either partitioning or hierarchical (Niknam and Amiri, 2010). Hierarchical clustering method deals in grouping the data patterns in a nested series of clusters (Jain, 2010). On the other hand, partitioning clustering method attempts to separate the dataset into a set of disjointed partitions using a density based partitioning criterion instead of hierarchical clusters bottom-up or top-down structuring (Niknam and Amiri, 2010).

There are many partitional clustering algorithms available in the literature, such as the K-Means algorithm (Kanungo *et al.*, 2002), the K-Medoids algorithm (Cao and Yng, 2010), the Fuzzy C-Means algorithm (Li *et al.*, 2009) and the K-Harmonic Means algorithm (Frackiewicz and Palus, 2008). In this study, the K-Means algorithm is selected since it is known for its simplicity and ability to deal with a huge amount (number of) data patterns since it is one of the crisp partitioning methods. However, K-Means depend on the initial center selection (state) and easily can converge to local minima (Selim and Ismail, 1984). Therefore, enhance K-Means algorithm is enhanced by proposing a Multi K-Means algorithm.

In the past few years, lot of algorithms has been applied on several domains to solve clustering problems, like Genetic Algorithm (Maulik and Bandyopadhyay, 2000; Lin and Wei, 2009; Al-Shboul and Myaeng, 2009), Tabu Search (Liu *et al.*, 2008) and Artificial Bee Colony (Zhang *et al.*, 2010).

In this study, a partitioning clustering method was used to solve medical clustering problems. The method aims to optimize the assignment of cluster members. A Simulated Annealing (SA) algorithm that was applied by Wang (2006) on the real world FTIR (Fourier Transform Infrared Micro-Spectroscopy) dataset is utilized and applied it using the same parameters by three ways of calculation the minimal distance over the six benchmark datasets (i.e., between objects, between centers and Combination of "between Centers and between Objects"). Then, enhance SA by proposing an iterative simulated annealing algorithm to overcome the limitation of SA. After that, the search process of ISA (coded as IISA) is prolonged with an aim to see the performance of ISA by giving more computational times.

PROBLEM DESCRIPTION

This study focuses on clustering problems using medical datasets where six well-known public domain benchmark datasets that are available in UCI machine learning repository (http://archive.ics.uci.edu/ml/index.html) are used to test the performance of the proposed approaches here by using three way of minimal distance calculation. The datasets are varied in terms of the number of records and attributes to show the different complexity of the tested data. All of the information about the diseases are taken from real infected patients and denoted for research purposes.

Medical clustering problems: Clustering is a multivariate analysis technique that has been espoused in medical diagnosis areas (Berkhin, 2002). Medical clustering problem opens a wide research opportunity especially in optimization area. It can be treated as a method to analyze the relation between the diagnosis and patient conditions data. The difficulty of dealing with the medical data is not only because of its complication and sensitivity, but also it comes with a huge number of records and a huge number of attributes. Therefore, a useful tool to be used in data clustering (where currently there is a substantial effort to investigate whether the initial diagnostic

of a certain diseases can be used as a diagnostic probe to identify the early stages of cancer or other diseases) is very important. This study is motivated by the investigation of a technique that can automatically explore and classify the infections of this disease.

Clustering aims to make the data patterns in shape to optimize an objective partitioning criterion, such as a similarity/dissimilarity function based on a distance, so that the objects within a cluster are "similar," whereas the objects of different clusters are "dissimilar" in terms of the attributes of the dataset. One of the most widely used in calculating the distance for the partitions (clusters) is the Euclidean Distance or Squared Error criteria, which performs very well when the clusters are isolated and compact (Jain and Dubes, 1988; Zhang *et al*., 2010).

Assuming a dataset $X = \{x_1, x_2,... x_n\}$ with n objects (data patterns) and need to be clustered it into K number of clusters, the Euclidean equation will be formed as in Eq. 1 (Wang, 2006), where, i and j are two of n-dimensional data objects:

$$d(i,j) = \sqrt{\sum_{i=1}^{n}\left(x_i - x_j\right)^2} \quad (1)$$

K-Means clustering is one of the early established algorithms in partition clustering (Kanungo *et al*., 2002). K-Means is considered as one of the best known clustering algorithms for its simplicity and its speed while dealing with a huge numbers of data patterns (Hong, 2006). K-Means is usually used to distribute the complex data into smaller partitions (clusters), where each cluster represents a particular area of the data, in order to represent and explain the total data simply (Kanungo *et al*., 2002). Figure 1 shows the pseudo code of K-Means procedure (Wang, 2006).

Even though, K-Means is a well-known and widely used in solving clustering problems, it still faces some disadvantages such as a number of clusters must be initially defined, a random selection of cluster centers, each run gives a completely different solution and it can easily get stuck to local minima i.e., it is sensitive to the starting centroids, so, as a solution for these problems, K-Means will be restarted 50 times (Hong, 2006) to get the most frequent solution for the tested dataset. Once the most common distribution for data is defined into the clusters, the K-Means is applied once again to check the best number determined of elements in each cluster. This process is repeated until no more changes (with respect to the objective function) in the cluster members.

Benchmark datasets: Generally, the number of clusters in a certain dataset is unknown in advance and the best clustering scheme which includes the number of clusters is not identified. So, by applying the clustering algorithm within the same range of commonly used number of clusters

(1) Arbitrarily select K points from the data set as initial centroids

(2) Repeat $v_j = \frac{1}{c_i}\sum_{i=1}^{c_i} x_{ij}$

(3) Form the new clusters by Assign /Reassign each point to its closest centroids

(4) Update the cluster center by re-computing the centroids using

(5) Until no centroids change

Fig. 1: Pseudo code for K-Means algorithm

is recommended in such a case. Since there is no definite answer for "what number of clusters is correct?", then the best thing to do is to use the problem specifications and to involve the human expert knowledge to decide the best number of clusters (Wang, 2006; Zhang *et al*., 2010). Datasets are summarized as follows:

Dataset 1: Wisconsin breast cancer database (Breast) with:

- Number of instances: 699
- Number of attributes: 10 (including the class attribute)
- Attributes type: Integers
- Number of clusters used in K-Means is 3 (i.e., 123, 240 and 363)

Dataset 2: Lung cancer database (Lung) with:

- Number of instances: 32
- Number of attributes: 56
- Attributes type: Integers
- Number of clusters used in K-Means: 3 (i.e., 9, 13 and 10)

Dataset 3: BUPA, liver disorders database (BUPA) with:

- Number of instances: 345
- Number of attributes: 7
- Attributes type: Integer, categorical and real
- Number of clusters used in K-Means: 2 (i.e., 145 and 200)

Dataset 4: Pima indian diabetes database (Diabetes) with:

- Number of instances: 768
- Number of attributes: 8
- Attributes type: Integer and real
- Number of clusters used in K-Means: 2 (i.e., 500 and 268)

Dataset 5: Haberman's survival database (HS) with:

- Number of instances: 306
- Number of attributes: 4
- Attributes type: Integer
- Number of clusters used in K-Means: 2 (i.e., 225 and 81)

Dataset 6: Thyroid gland data disease database (Thyroid) with:

- Number of instances: 215
- Number of attributes: 21

- Attributes type: Categorical and real
- Number of clusters used in K-Means: 3 (i.e., 150, 30 and 35)

Cluster quality calculation: In cluster analysis, it is very important issue to evaluate the clustering results quality that is produced by a certain measure. Those measures can be used to compare solutions obtained from different algorithms and also can be used to guide some optimization search processes to find the best partitioning procedure which fits the underlying dataset (Halkidi *et al*., 2001).

In this study, three functions are used to measure the clusters and their centers i.e.,

- Between objects
- Between centers

These measurement on the cluster objects and/or changes on the cluster centers have the direct effect to the cluster quality. The two types of functions are portrayed as below:

Between objects: This method depends on the calculation on the distance (cluster quality) between each data pattern. A method for clustering in which samples are added in if they are "close" to at least one sample in the candidate cluster is produced. This idea was inspired from K-Nearest Neighbor classifiers. The K-Nearest Neighbor method was firstly introduced in early 1950s as a supervised learning method for classification. K-Nearest Neighbor depends on "learning by analogy", which is a comparison between some discovered patterns and undiscovered patterns and each data pattern is represented with a point in a multidimensional space. When given a dataset, the K-Nearest Neighbor method searches the pattern space for the K training patterns that are closest to the unknown patterns. These K patterns are the K "nearest neighbors" of the unknown patterns. "Closeness" is defined in term of distance, such as Euclidean distance. The Euclidean distance is calculated between two points or patterns. In this part, there are distinct data items, so need to collect sets of data items that make sense to be together in one cluster.

Between centers: This method depends on calculating the distance (cluster quality) using the sum of distance between each data pattern and the cluster center that it belongs to.

INITIAL SOLUTION GENERATION: MULTI K- MEANS

K-Means clustering is one of the early established algorithms in partition clustering (Kanungo *et al*., 2002). The K-Means algorithm takes K as an input parameter and partitions the set of n objects into K clusters so that, intra-cluster similarity is high and the inter-cluster similarity is low. However, this standard K-Means algorithm has many disadvantages that can be summarized as follows:

- K (number of clusters) must be initially defined
- Centers are randomly selected
- Each run (of the method) will give a completely different solution
- Tends to get stuck to local minima i.e., it is sensitive to the starting centroids

Thus, as a solution for these problems, especially the starting centroids that affects which local minima the algorithm will converge to, as well as the rate of convergence between the points, then K-Means is executed M times and get the average clustering results (Hong, 2006).

In Multi K-Means, once the average cluster is determined after the K-Means is restarted for 50 times as recommended (Davidson and Satyanarayana, 2003), the final result of the clusters distribution will be executed under the K-Means algorithm using a "move reallocating" criterion aiming to find the best object-cluster assignments. So, once the execution is done, then each cluster of the K clusters has its final initial known number of objects.

THE ALGORITHMS

In this study, two (i.e., N1 and N2) neighborhood structures are employed within the Iterative Simulated Annealing algorithm (ISA), which is driven and based on the original Simulated Annealing algorithm (SA) that was proposed by Wang (2006) for solving medical data clustering problems.

Neighborhood structures: In solving clustering problems, one of the widely used neighborhood structures in improving the cluster quality is a swap mechanism, which is simply select a random pattern from one cluster and exchange it with another random pattern from another cluster and check if the cluster quality has improved by this operation or not.

Mainly, a simple random selection for the patterns is used in the research with taking in consideration that the random pattern will be from different clusters, since the cluster quality will not be affected by the swap if it is done within the cluster itself. For an aim of having a better quality of the clusters, a set of five random selections are done in a sequence formatting between the clusters. For example, if there are three clusters, then five random swaps will be suggested between cluster one and cluster two, cluster two and cluster three and cluster one and cluster three and then the best improvement of the quality achieved will be selected to be the next swap operation to be processed.

Two neighborhood structures are used in this work, which is adopted from (Wang, 2006) (coded as N1 and N2). The description of the employed neighborhood structures are given as follows:

- **N1:** Randomly selects one pattern from each cluster to swap their data. For example, if there are two clusters (i.e., cluster 1 and cluster 2), randomly select the first pattern (R1) from cluster 1 and the second pattern (R2) from cluster 2. Swap R1 data with R2 data. Another example, if there are three clusters (i.e., cluster 1, cluster 2 and cluster 3), randomly select the first pattern (R1) from cluster 1, the second pattern (R2) from cluster 2, then swap R1 data with R2 data, in which consider as solution-1 (S1). Then, repeat the same process (between cluster 1 and cluster 3 to obtain solution-2 (S2) and between cluster 2 and cluster 3 to obtain solution-3 (S3). After that, the best quality (minimal distance) solution between S1, S2 and S3, will be selected as a candidate solution (working solution)
- **N2:** Randomly select two different patterns from the same cluster and swap their data. For example, if there are two clusters (i.e., cluster 1 and cluster 2), randomly select the first pattern (R1) and the second pattern (R2) from cluster1, then swap R1 data with R2 data to obtain

solution-1 (S1). Repeat the same process for cluster 2 to obtain solution-2 (S2) and the best solution between S1 and S2 will be selected as a candidate solution (working solution)

Iterative simulated annealing: Simulated Annealing (SA) was introduced by Metropolis *et al.* (1953) where the original idea of SA comes from heating and cooling process. Kirkpatrick (1984) proposed the first SA for combinatorial optimization problems and the idea is to use a simulated annealing to search for feasible solutions by accepting the new (worse) solution based on a certain probability P. The probability is calculated by $P = e^{-\delta/Temp}$, where Temp is the temperature parameter that decreases during the search according to some cooling schedule, δ is the difference between the quality of the candidate solution ($S_{working}$) and current solution (S_{source}) as in Eq. 2.

$$\delta = f(S_{working}) - f(S_{source}) \quad (2)$$

In this study, the original simulated annealing is applied which is based on (Wang, 2006), where, the algorithm starts with a given K-Means partitions. The notations used in this work are listed as follow:

- S_o : Initial solution
- $F(S_o)$: Quality of S_o
- T_0 : Initial temperature
- T_f : Final temperature
- Temp : Current temperature, where it is initialized to T_0
- a : Decreasing rate
- $S_{Arrange}$: Best solution
- $F(S_{Arrange})$: The quality of $S_{Arrange}$
- S_{source} : The current solution
- $F(S_{source})$: The quality of S_{source}
- $S_{working}$: The candidate solution
- $F(S_{working})$: The quality of $S_{working}$.

The same parameters as those employed by Selim and Al-Sultan (1991) are imposed in this work, where the initial temperature T_0 is equal to 10 and the final temperature T_f is 0. At the beginning of the search, Temp is set to be T_0 and at every iteration the temperature Temp is decreased by α, where α is equal to 0.7. The pseudo code for the SA to solve medical data clustering problems is shown in Fig. 2.

Figure 2 shows that, the algorithm starts by initializing the required parameters as in Step-1 by setting the temperature (Temp) equal to the initial temperature (T_o) and define the decreasing temperature rate (α). Note that the initial solution (S_o) is generated using K-Means.

In the improvement phase (Step-2), basically the initial solution is iteratively be improved by employing the simulated annealing until the stopping condition is met. In Step 2.1, neighborhood structures N1 and N2 are applied to generate candidate solutions (in this case, five candidate solutions are generated) and the best candidate solution is selected as $S_{working}$. There are two cases that need to take into account i.e.,

Procedure simulated annealing algorithm
Step-1: Initialization Phase
Determine K-mean solution (initial solution) S_o and
$f(S_o)$;
$S_{Arrange} = S_o$; $f(S_{Arrange}) = f(S_o)$;
$S_{source} = S_o$; $f(S_{source}) = f(S_o)$;
Set initial temperature T_o;
Set final temperature T_f;
Set Temp = T_o;
Set decreasing temperature rate as a, where a = 0.7;
Step-2: Improvement (Iterative) Phase
repeat (while termination condition is not satisfied)
Step-2.1: Selecting candidate solution $S_{working}$
Generate candidate solutions by applying all
neighborhood structures (N1 and N2) and
the best solution consider as candidate
solution ($S_{working}$);
Step-2.2: Accepting Solution
if $f(S_{working}) < f(S_{Arranget})$
$S_{Arrange} := S_{working}$; $f(S_{arrange}) = f(S_{working})$;
$S_{source} = S_{working}$; $f(S_{source}) = f(S_{working})$;
else
$d = f(S_{working}) - f(S_{source})$
Generate a random number called RN between 0 and 1;
if $RN = e^{-d/Temp}$
$S_{source} = S_{working}$;
Temp = Temp-Temp*a;
end if
until Temp < T_f (termination condition is met)
Step-3: Termination phase
Return the best solution found $S_{Arrange}$

Fig. 2: Pseudo code for simulated annealing algorithm for medical clustering problems

Case 1: Better solution: If $f(S_{working})$ is better than $f(S_{Arrange})$, then $S_{working}$ is accepted as a current solution ($S_{source} \leftarrow S_{working}$) and the best solution is updated ($S_{Arrange} \leftarrow S_{working}$) as shown in Step-2.2. The *Temp* will be decreased by the value α (i.e., Temp = Temp-Temp*a).

Case 2: Worse solution: If $f(S_{working})$ is less than $f(S_{Arrange})$, then the different between the quality of $S_{working}$ and S_{source} is calculated. A random number [0.1], RN, is generated. If the probability (i.e., $e^{-\delta/Temp}$, where $\delta = f(S_{working}) - f(S_{source})$) is less than or equal to RN) then $S_{working}$ is accepted and the current solution is updated ($S_{source} \leftarrow S_{working}$). Otherwise, $S_{working}$ will be rejected. Again, the Temp will be updated by a (i.e., Temp = Temp-Temp*a).

The process will continue until the termination condition is met (Temp< T_f) and return the best solution found so far $S_{Arranget}$ (Step-3).

However, there are two drawbacks in SA over medical data clustering problems by Wang (2006) i.e., (i) the number of recommended iterations is 600 iterations, in which there is no guarantee to find good solutions for some datasets. Therefore, the Iterative Simulated Annealing (ISA) is introduced to overcome these drawbacks as shown in Fig. 3. ISA structure resembles SA structure, but the basic difference is in term of the stopping condition. In ISA, when the temperature (Temp) is less or equal to the final temperature, it will be reinitialized equal to the initial temperature (T_0). This process of reinitializing the temperature will be repeated in ISA, until no improvement obtained on the best solution ($S_{Arranget}$). Further, experiment is also carried out on ISA by prolonging the search (coded as IISA) where the number of iteration is set to 100000 (Table 1) with an aim to examine whether the algorithm is still able to work well given extra computational time.

Table 1: Parameters setting used in simulated annealing

Parameter	Description	Value
T_o	Initial temperature	0
T_f	Final temperature	10
α	Cooling rate	0.7
N.iters	Number of iterations	100000

Procedure Iterative Simulated Annealing Algorithm
Step-1: Initialization Phase
Determine K-mean solution (initial solution) S_o
and $f(S_o)$;
$S_{Arrange} = S_o$; $f(S_{arrange}) = f(S_o)$;
$S_{source} = S_o$; $f(S_{source}) = f(S_o)$;
$S_{ISA} = S_o$; $f(S_{ISA}) = f(S_o)$;
Set initial temperature T_o;
Set final temperature T_f;
Set Temp = T_o;
Set decreasing temperature rate as a, where a = 0.7;

Step-2: Improvement (Iterative) Phase
repeat (while termination condition is not satisfied)
Step-2.1: Selecting candidate solution $S_{working}$
Generate candidate solutions by applying all
neighborhood structures (N1 and N2) and the best
solution consider as candidate solution $S_{working}$);

Step-2.2: Accepting Solution
if $f(S_{working}) < f(S_{Arrange})$
$S_{Arrange} := S_{working}$; $f(S_{arrange}) = f(S_{working})$;
$S_{source} = S_{working}$; $f(S_{source}) = f(S_{working})$;
$S_{ISA} = S_{working}$; $f(S_{ISA}) = f(S_{working})$; // ISA
else
$\delta = f(S_{working}) - f(S_{source})$
Generate a random number called RN between 0 and 1;
if $RN = e^{-d/Temp}$
$S_{source} = S_{working}$;
end if
Temp = Temp-Temp*a;
end if
if Temp <= T_f and $f(S_{ISA}) <> (S_{Arrange})$
Temp = T_o;
until Temp < T_f (termination condition is met)

Step-3: Termination phase
Return the best solution found $S_{arrange}$

Fig. 3: Pseudo code for iterative simulated annealing algorithm to solve the medical data clustering problems

RESULTS AND DISCUSSION

In this study, the algorithm is ran 20 times across 6 datasets. The algorithms are programmed in Java language and are tested on a PC with an Intel dual core 1800 MHZ, 2GB RAM. In the analysis part, the terms used are as follows:

- N-NI : Number of not improved iterations
- N-I : Number of improved iterations
- Std : Standard Deviation

The parameters setting used in these experiments are shown in Table 1, where it shows that, SA algorithm employing three parameters i.e., T_0, T_f and (α), which was defined by Metropolis *et al.* (1953) and Kirkpatrick (1984). ISA employs the same parameters as SA, whereas, IISA adds N.iters parameter as this algorithm try to prolong the search until no improvement is obtained.

Table 2 shows the results comparison between SA, ISA and IISA algorithms based on the minimal distance calculation (i.e., between objects). The "Avg" represents the average results out of 20 runs. The best results are presented in bold. The results in Table 2 show that ISA algorithm outperforms the SA in all the cases. However, in SA the number of recommended iterations is 600 iterations (Wang, 2006), in which there is no guaranty to find good solutions for some datasets. Whereas, ISA which is an extension of SA is able to find better solution by reinitializing the temperature, that represents exploration mechanism. By prolonging the search, IISA performs better than SA and ISA and is able to further improve the quality of the results obtained by ISA.

The observation (Table 2) indicates that, IISA algorithm using N1 and N2 neighborhood structures better than ISA and SA algorithms using N1 and N2 based on best minimal distance and average score.

Table 3 shows a further analysis on IISA using N1 and N2 neighborhood structures over all tested datasets. For example, the best results for BUPA dataset is 5771.59 that is obtained within

Table 2: Results between Objects obtained from SA, ISA and IISA algorithms using N1 and N2 neighborhood structures on six datasets

		20 Runs-Minimal distance calculated as between objects					
		SA		ISA		IISA	
Dataset	Initial length by K-Means	Best	Avg.	Best	Avg.	Best	Avg.
Breast	6379.694	3457	3561.8	2649	2762.15	2338	2430.05
Thyroid	3178.714	1730.25	1820.99	1518.73	1641.76	1228.8	1375.23
BUPA	17258.715	9842.82	10387.47	6049.72	6316.17	5771.59	6028.67
Lung	182.577	162.93	165.27	161.12	164.77	158.98	161.14
HS	2463.972	1710.05	1793.58	1070.54	1133.94	987.77	1049.03
Diabetes	100880.390	58191.71	60084.85	25431.7	26734.4	24920.39	26038.86

Table 3: Results analysis "between Objects" for IISA algorithm using N1 and N2 neighborhood structures on six datasets

Dataset description	Breast	Thyroid	BUPA	Lung	HS	Diabetes
Best	2338	1228.8	5771.59	158.98	987.77	24920.39
Iterations for best	95288	100593	98596	29377	99707	100078
Time	00:52:08	00:07:13	00:03:22	00:33:23	00:01:01	00:06:08
N-I	99430	100166	532	18846	402	97478
N-NI	431	62	99030	81716	99518	2210
Average	2430.05	1375.23	6028.67	161.14	1049.03	26038.86
Result range	2338	1228.8	5771.59	158.98	987.77	24920.39
	2531	1477.49	6253.33	162.54	1121.82	27679.15
Iterations range	57918	25103	92169	5435	93450	98945
	100145	100593	99998	97761	100080	100132
Time range	00:52:07	00:07:12	00:02:28	00:21:46	00:00:58	00:06:03
	00:52:43	00:07:35	00:03:26	00:39:40	00:01:03	00:06:22
Std	52.17	70.08	140.29	0.86	32.65	796.96

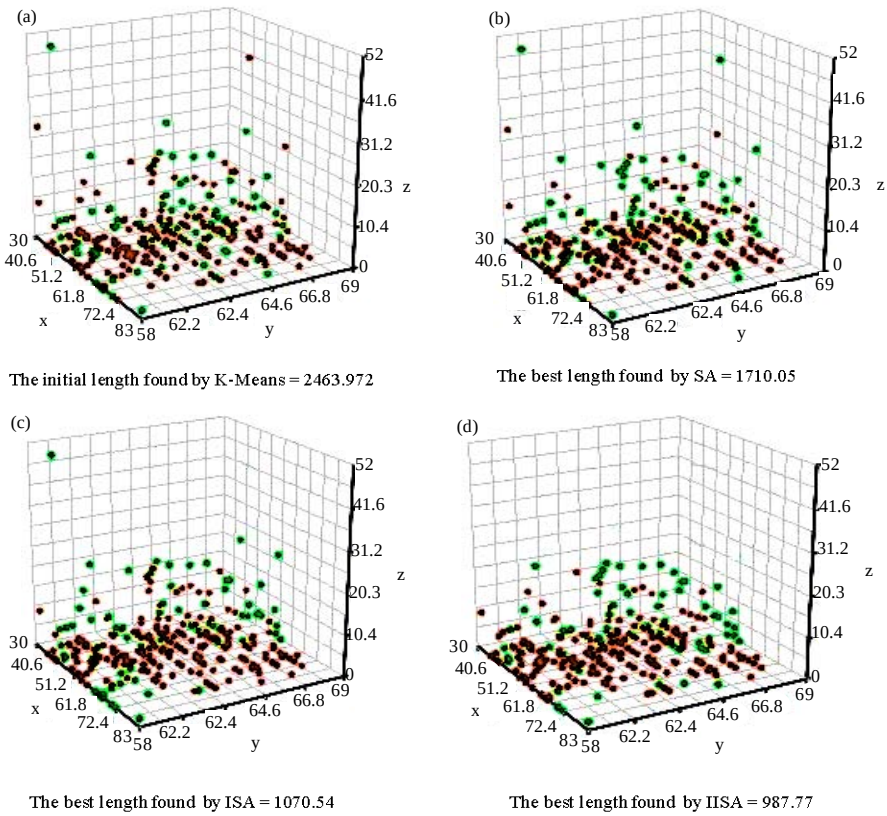

Fig. 4(a-d): Scatter graph for K-Means, SA, ISA, IISA algorithms over HS dataset for minimal distance calculation between objects using N1 and N2 neighborhood structures

3 min and 22 sec under 98596 iterations. Meanwhile, the range for minimum and maximum results is in between 5771.59 and 6253.33. In most of the cases, the results are obtained between 2 min and 28 sec to 3 min and 26 sec that are considered acceptable.

Figure 4 shows a 3D scatter graph for K-Means, SA, ISA and IISA algorithms using N1 and N2 neighborhood structures over HS dataset. Two clusters are represented by the two colors. Figure 4a shows that, the two clusters (colors) are mixed with initial minimal distance obtained by K-Means is 2463.972. Whereas, Fig. 4d shows that, IISA obtained better improvement in terms of the minimal distance than SA and ISA algorithms which is equal to 987.77.

Table 4 shows the comparison between SA, ISA and IISA algorithms using N1 and N2 neighborhood structures based on the minimal distance calculation (i.e. between centers). Again, the best results are presented in bold.

Table 4 shows that, the best results and average score using N1 and N2 neighborhood structures obtained indicate that, ISA algorithm outperformed SA algorithm in all datasets, except in one dataset (i.e., Lung dataset); whereas, they equally obtained the same best results; but for the average score, ISA is better than SA algorithm. Table 4 also shows that, IISA algorithm outperformed ISA algorithm in almost all datasets, except in two datasets (i.e., Thyroid and HS

datasets); whereas, they equally obtained the same best results. It is believed because of the temperature re-initialization process and more computational time given to the algorithm that helps to better explore the search space, later further The observation (Table 4) indicates that, IISA algorithm is better than ISA and SA algorithms based on best minimal distance and average score. Table 5 shows a further analysis on IISA over all datasets. For example in Table 5, Lung dataset the best result obtained by IISA algorithm is 152.37 in 18 min and 23 sec and 20505 iterations. Whereas, all the results was obtained between 11 min and 49 sec to 19 min and 17 sec. Meanwhile, the result range obtained between 152.37 and 155.34.

Figure 5 shows a 3D scatter graph for K-Means, SA, ISA, IISA algorithms using N1 and N2 neighborhood structures over H.S dataset. Two clusters are represented by two colors. Figure 5a shows that, the two clusters (colors) are mixed with initial minimal distance obtained by K-Means is 3626.530. Whereas, Fig. 5d shows that, there is an improvement in terms of the minimal distance which is equal to 2721.36 after employing the IISA algorithm. Meanwhile, ISA obtains same result as IISA and by referring to Fig. 5c, it can be seen that the dispersion of the colors in IISA is less (more concentrated) shows it is better clustered than ISA (which is more scattered).

Table 4: Results analysis "Between Centers" for SA, ISA and IISA algorithms using N1 and N2 neighborhood structures on six datasets

		20 Runs-Minimal distance calculated as between objects					
		SA		ISA		IISA	
Dataset	Initial length by K-Means	Best	Avg.	Best	Avg.	Best	Avg.
Breast	5360.710	3045	3904.7	2795	3304.25	2778	3104.4
Thyroid	2459.620	2049.07	2061.25	2039.89	2054.37	2039.89	2054.09
BUPA	22646.89	10921.36	11816.61	10502.81	11043.82	10498.9	10855.71
Lung	168.520	152.52	154.30	152.52	154.3	152.37	154.28
HS	3626.530	2912.28	3017.7	2721.36	2769.54	2721.36	2722
Diabetes	102398.583	56031.25	63083.42	49342.91	55376.06	48909.2	54751.22

Table 5: Results analysis "Between Centers" for IISA algorithm using N1 and N2 neighborhood structures on six datasets

Dataset /Result	Breast	Thyroid	BUPA	Lung	HS	Diabetes
Best	2778	2039.89	10498.9	152.37	2721.36	48909.2
Iterations for best	63550	80949	1774	20505	87345	97601
Time	01:58:09	00:07:30	00:03:05	00:18:23	00:01:15	00:13:02
N-I	100158	96658	91486	98977	96375	89414
N-NI	120	3844	8255	1603	3480	10072
Average	3104.4	2054.09	10855.71	154.28	2722	54751.22
Result range	2778	2039.89	10498.9	152.37	2721.36	48909.2
	3480	2055.67	11460.03	155.34	2722.54	58993.91
Iterations range	8536	2192	17745	34	19194	57331
	88544	98708	96185	97526	99094	100103
Time range	01:41:29	00:07:22	00:03:04	00:11:49	0:01:03	00:10:13
	02:01:39	00:08:10	00:03:11	00:19:17	00:01:20	00:13:02
Std	304.30	4.85	354.91	1.3	0.6	4889.73

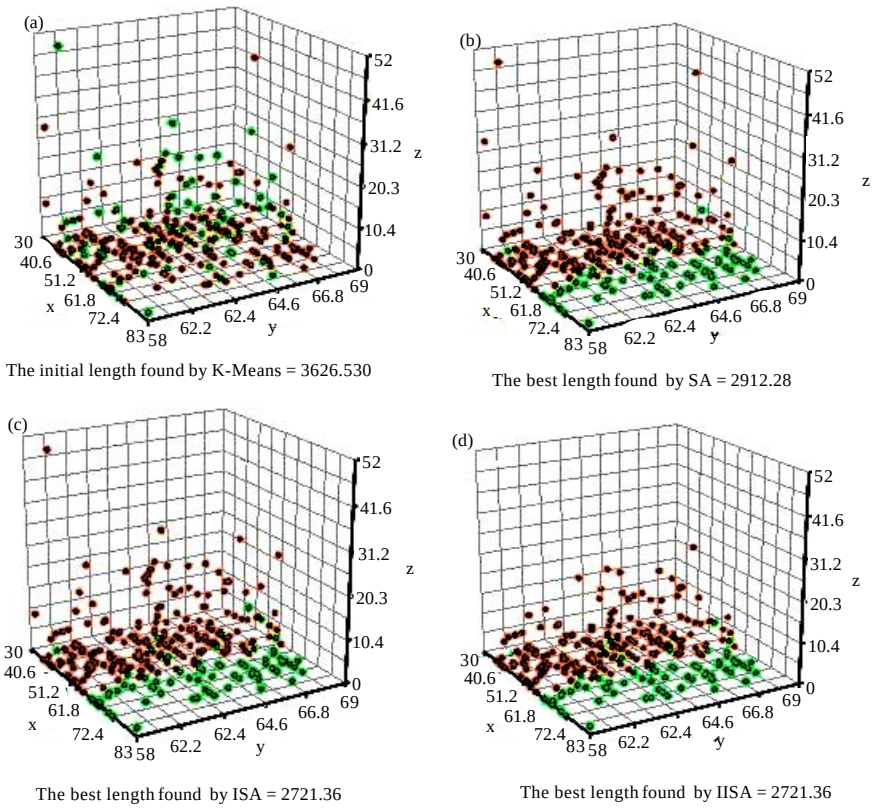

Fig. 5(a-d): Scatter graph for K-Means, SA, ISA, IISA algorithms over HS dataset for minimal distance calculation between centers using N1 and N2 neighborhood structures

CONCLUSION

This study propose an iterative simulated annealing algorithm, then prolong the search for further improvement, after that the performances of the proposed algorithms are compared to the basic simulated annealing based on the minimal distance that is calculated based on (i) between objects and (ii) between centers. The algorithms has been implemented and tested on six well known real datasets. The algorithms in the comparison are three variations of simulated annealing algorithms (coded as SA, ISA and IISA). Two different neighborhood structures are employed within the proposed approaches i.e., (N1 and N2).

Experimental results demonstrated that, IISA algorithm in both minimal distance (i.e. between objects and between centers) calculation has performed better compared to the rest of the algorithms tested here that employed (N1 and N2) neighborhood structures, which extend to test on combination of between centers and between objects calculation. Generally, it can be concluded that, the algorithms behave differently due to the different measurements imposed during the search process.

REFERENCES

Al-Shboul, B. and S.H. Myaeng, 2009. Initializing K-means using genetic algorithms. World Acad. Sci. Eng. Technol., 54: 114-118.

Barthelemy, J.P. and F. Brucker, 2001. NP-hard approximation problems in overlapping clustering. J. Classification, 18: 159-183.

Berkhin, P., 2002. Survey of Clustering Data Mining Techniques. Accure Software Inc., San Jose, CA., USA.

Cao, D. and B. Yang, 2010. An improved k-medoids clustering algorithm. Proceedings of the 2nd International Conference on Computer and Automation Engineering, February 26-28, 2010, Singapore, pp: 132-135.

Dasgupta, S. and Y. Freund, 2009. Random projection trees for vector quantization. IEEE Trans. Inform. Theory, 55: 3229-3242.

Davidson, I. and A. Satyanarayana, 2003. Speeding up k-means clustering by bootstrap averaging. Proceedings of the IEEE Workshop on Clustering Large Data Sets, November 19, Melbourne, FL., USA., pp: 16-25.

Frackiewicz, M. and H. Palus, 2008. Clustering with K-harmonic means applied to colour image quantization. Proceedings of the IEEE International Symposium on Signal Processing and Information Technology, December 16-19, 2008, Sarajevo, pp: 52-57.

Halkidi, M., Y. Batistakis and M. Vazirgiannis, 2001. On clustering validation techniques. J. Intell. Inform. Syst., 17: 107-145.

Hong, S., 2006. Experiments with K-means, fuzzy C-means and approaches to choose K and C. Ph.D. Thesis, University of Central Florida, Orlando, FL., USA.

Jain, A.K. and R.C. Dubes, 1988. Algorithms for Clustering Data. Prentice-Hall Inc., Englewood Cliffs, NJ., USA.

Jain, A.K., 2010. Data clustering: 50 years beyond K-means. Pattern Recogn. Lett., 31: 651-666.

Kanungo, T., D.M. Mount, N.S. Netanyahu, C.D. Piatko, R.S. Angela and Y. Wu, 2002. An efficient k-means clustering algorithm: Analysis and implementation. IEEE Trans. Pattern Anal. Mach. Intell., 24: 881-892.

Kirkpatrick, S., 1984. Optimization by simulated annealing: Quantitative studies. J. Stat. Phys., 34: 975-986.

Li, X., X. Lu, J. Tian, P. Gao, H. Kong and G. Xu, 2009. Application of fuzzy c-means clustering in data analysis of metabolomics. Anal. Chem., 81: 4468-4475.

Lin, J.L. and M.C. Wei, 2009. Genetic algorithm-based clustering approach for k-anonymization. Expert Syst. Appl., 36: 9784-9792.

Liu, Y., Z. Yi, H. Wu, M. Ye and K. Chen, 2008. A tabu search approach for the minimum sum-of-squares clustering problem. Inform. Sci., 178: 2680-2704.

Mahajan, M., P. Nimbhorkar and K. Varadarajan, 2009. The planar k-means problem is NP-hard. WALCOM: Algorithms Comput., 5431: 274-285.

Maulik, U. and S. Bandyopadhyay, 2000. Genetic algorithm-based clustering technique. Pattern Recogn., 33: 1455-1465.

Metropolis, N., A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller and A.E. Teller, 1953. Equation of state calculations by fast computing machines. J. Chem. Phys., 21: 1087-1091.

Niknam, T. and B. Amiri, 2010. An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis. Applied Soft Comput., 10: 183-197.

Saha, I., D. Lewczynski, U. Maulik and S. Bandyopadhyay, 2010. Consensus multi objective differential crisp clustering for categorical data analysis. Rough Sets Curr. Trends Comput., 6086: 30-39.

Selim, S.Z. and M.A. Ismail, 1984. K-means type algorithms: A generalized convergence theorem and characterization of local optimality. IEEE Trans. Pattern Anal. Mach. Intell., 6: 81-87.

Selim, S.Z. and K. Alsultan, 1991. A simulated annealing algorithm for the clustering problem. Pattern Recognit., 10: 1003-1008.

Wang, X., 2006. Fuzzy clustering in the analysis of fourier transform infrared spectra for cancer diagnosis. Ph.D. Thesis, School of Computer Science and Information Technology, University of Nottingham.

Zhang, C., D. Ouyang and J. Ning, 2010. An artificial bee colony approach for clustering. Exp. Syst. Appl., 37: 4761-4767.