Data Clustering Using the Bees Algorithm

D.T. Pham, S. Otri, A. Afify, M. Mahmuddin and H. Al-Jabbouli
Intelligent Systems Laboratory, Manufacturing Engineering Centre, Cardiff University, Cardiff CF24 3AA, UK

Abstract

Clustering is concerned with partitioning a data set into homogeneous groups. One of the most popular clustering methods is k-means clustering because of its simplicity and computational efficiency. K-means clustering involves search and optimisation. The main problem with this clustering method is its tendency to converge to local optima. The authors' team have developed a new population-based search algorithm called the Bees Algorithm that is capable of locating near-optimal solutions efficiently. This paper proposes a clustering method that integrates the simplicity of the k-means algorithm with the capability of the Bees Algorithm to avoid local optima. The paper presents test results to demonstrate the efficacy of the proposed algorithm.

1 INTRODUCTION

Clustering is a typical unsupervised learning technique for grouping similar data points. A clustering algorithm assigns a large number of data points to a smaller number of groups such that data points in the same group share the same properties while, in different groups, they are dissimilar. Clustering has many applications, including part family formation for group technology, image segmentation, information retrieval, web pages grouping, market segmentation, and scientific and engineering analysis [1].

One of the best known and most popular clustering algorithms is the k-means algorithm [2]. The algorithm is efficient at clustering large data sets because its computational complexity only grows linearly with the number of data points. However, the algorithm may converge to solutions that are not optimal [3].

This paper describes the application of a new optimisation algorithm called the Bees Algorithm [4] to find global solutions to the clustering problem. The Bees Algorithm performs a kind of neighbourhood search combined with random search in a way that is reminiscent of the food foraging behaviour of swarms of honey bees. The algorithm has been successfully applied to different optimisation problems including the training of neural networks for control chart pattern recognition [5] and wood defect identification [6].

The paper is organised as follows. Section 2 briefly reviews different clustering methods. Section 3 describes the foraging behaviour of bees and the core ideas of the proposed clustering method. Results of different clustering experiments are reported in section 4. Section 5 concludes the paper and gives suggestions for future work.

2 CLUSTERING METHODS

Many clustering methods have been proposed [7, 8]. They can be broadly classified into four categories [9]: partitioning methods, hierarchical methods, density-based methods and grid-based methods. Other clustering techniques that do not fit in these categories have also been developed. These are fuzzy clustering, artificial neural networks and genetic algorithms. A discussion of different clustering algorithms can be found in references [1, 9].

K-means is the simplest and most commonly used partitioning algorithm. It represents each cluster by the mean value of the data points within the cluster. It attempts to divide a data set S into k clusters to minimise the sum of the Euclidean distances between data points and their closest cluster centres. This criterion is defined formally by Equation 1, where $x_i^{(j)}$ is the *i*th data point belonging to the *j*th cluster, c_j is the centre of the *j*th cluster, *k* is the number of clusters and n_j is the number of data points in cluster *j*.

$$E = \sum_{j=1}^{k} \sum_{i=1}^{n_j} \left\| x_i^{(j)} - c_j \right\|^2 \qquad (1)$$

As mentioned above, the implementation of k-means clustering involves optimisation. First, the algorithm takes *k* randomly selected data points and makes them the initial centres of the *k* clusters being formed. The algorithm then assigns each data point to the cluster with centre closest to it. In the second step, the centres of the *k* clusters are recomputed, and the data points are redistributed. This step is repeated for a specified number of iterations or until there is no change to the membership of the clusters over two successive iterations.

It is known that the k-means algorithm may become trapped at local optimal solutions, depending on the choice of the initial cluster centres. Genetic algorithms have a potentially greater ability to avoid local optima than is possible with the localised search employed by most clustering techniques. Maulik and Bandyopadhyay [10] proposed a genetic algorithm-based clustering technique, called GA-clustering, that has proven effective in providing optimal clusters. With this algorithm, solutions (typically, cluster centroids) are represented by bit strings. The search for an appropriate solution begins with a population, or collection, of initial solutions. Members of the current population are used to create the next-generation population by applying operations such as random mutation and crossover. At each step, the solutions in the current population are evaluated relative to some measure of fitness (which, typically, is inversely proportional to E), with the fittest solutions selected probabilistically as seeds for producing the next generation. The process performs a generate-and-test beam search of the solution space, in which variants of the best current solutions are most likely to be considered next.

In the next section, an alternative clustering method to solve the local optimum problem of the k-means algorithm is described. The new method adopts the Bees Algorithm as it has proved to give a more robust performance than other intelligent optimisation methods for a range of complex problems [4].

3 CLUSTERING USING THE BEES ALGORITHM

3.1 Bees in Nature

A colony of honey bees can extend itself over long distances in order to exploit a large number of food sources at the same time [11, 12].

The foraging process begins in a colony by scout bees being sent to search for promising flower patches. Flower patches with large amounts of nectar or pollen that can be collected with less effort tend to be visited by more bees, whereas patches with less nectar or pollen receive fewer bees [13]. During the harvesting season, a colony continues its exploration, keeping a percentage of the population as scout bees [12]. When they return to the hive, those scout bees that found a patch rated above a certain quality threshold deposit their nectar or pollen and go to the “dance floor” to perform a dance known as the “waggle dance” [11].

This mysterious dance is essential for colony communication, and contains three pieces of information regarding a flower patch: the direction in which it will be found, its distance from the hive and its quality rating (or fitness) [11, 13]. This information helps the colony to send its bees to flower patches precisely, without using guides or maps. Each individual’s knowledge of the outside environment is gleaned solely from the waggle dance. This dance enables the colony to evaluate the relative merit of different patches according to both the quality of the food they provide and the amount of energy needed to harvest it [13]. After waggle dancing on the dance floor, the dancer (i.e. the scout bee) goes back to the flower patch with follower bees that were waiting inside the hive. More follower bees are sent to more promising patches. This allows the colony to gather food quickly and efficiently.

While harvesting from a patch, the bees monitor its food level. This is necessary to decide upon the next waggle dance when they return to the hive [13]. If the patch is still good enough as a food source, then it will be advertised in the waggle dance and more bees will be recruited to that source.

3.2 The Proposed Clustering Method

The proposed clustering method exploits the search capability of the Bees Algorithm to overcome the local optimum problem of the k-means algorithm. More specifically, the task is to search for appropriate cluster centres $(c_1, c_2, \ldots, c_k)$ such that the clustering metric *E* (Equation 1) is minimised. Figure 1 shows the basic steps of the proposed clustering operation, which are essentially those of the Bees Algorithm. These steps are described in detail below.

```
1. Initialise the solution population.
2. Evaluate the fitness of the population.
3. While (stopping criterion is not met)
   //Forming new population.
4. Select sites for neighbourhood search.
5. Recruit bees for selected sites (more bees for
   the best e sites) and evaluate fitnesses.
6. Select the fittest bee from each site.
7. Assign remaining bees to search randomly and
   evaluate their fitnesses.
8. End While.
```

Figure 1. Basic steps of the Bees-Algorithm-based clustering method

The algorithm requires a number of parameters to be set, namely: number of scout bees (n), number of sites selected for neighbourhood searching (out of n visited sites) (m), number of top-rated (elite) sites among m selected sites (e), number of bees recruited for the best e sites (nep), number of bees recruited for the other (m-e) selected sites (nsp), and the stopping criterion.

The algorithm starts with an initial population of n scout bees. Each bee represents a potential clustering solution as set of k cluster centres. The initial locations of the centres are randomly assigned.

The Euclidean distances between each data object and all centres are calculated to determine the cluster to which the data object belongs (i.e. the cluster with centre closest to the object). In this way, initial clusters can be constructed.

After the clusters have been formed, the original clusters centres are replaced by the actual centroids of the clusters to define a particular clustering solution (i.e. a bee).

This initialisation process is applied each time new bees are to be created.

In step 2, the fitness computation process is carried out for each site visited by a bee by calculating the clustering metric E (Equation 1) which is inversely related to fitness.

In step 4, the m sites with the highest fitnesses are designated as “selected sites” and chosen for neighbourhood search. In steps 5 and 6, the algorithm conducts searches around the selected sites, assigning more bees to search in the vicinity of the best e sites. Selection of the best sites can be made directly according to the fitnesses associated with them. Alternatively, the fitness values are used to determine the probability of the sites being selected. Searches in the neighbourhood of the best e sites – those which represent the most promising solutions - are made more detailed. As already mentioned, this is done by recruiting more bees for the best e sites than for the other selected sites. Together with scouting, this differential recruitment is a key operation of the Bees Algorithm.

In step 6, for each patch, only the bee that has found the site with the highest fitness (the “fittest” bee in the patch) will be selected to form part of the next bee population. In nature, there is no such a restriction. This restriction is introduced here to reduce the number of points to be explored. In step 7, the remaining bees in the population are assigned randomly around the search space to scout for new potential solutions.

At the end of each iteration, the colony will have two parts to its new population: representatives from the selected patches, and scout bees assigned to conduct random searches. These steps are repeated until a stopping criterion is met.

4 TEST RESULTS

This section presents the results of testing the Bees-Algorithm-based clustering method against the k-means and GA-clustering algorithms. The algorithms were applied to five real data sets (*Vowel*, *Iris*, *Crude Oil*, *Control Charts* and *Wood Defects*). Table 1 summarises the main characteristics of these data sets. Details of the *Vowel*, *Iris* and *Crude Oil* data sets are given in reference [8]. The *Control Charts* data set is explained in reference [5] and the *Wood Defects* data set in [6].

The clustering criterion *E* (Equation 1) is used to evaluate the performance of the tested algorithms: the smaller the value of this metric, the better the clustering results. Table 2 shows the parameter values for each algorithm used in this test and

Table 3 the results obtained for each algorithm. The algorithms were executed 10 times and the average, minimum and maximum values of E are given.

Table 1. Data sets used in the experiments

Data Set Name	Number of Objects	Number of Features	Number of Classes
Vowel	871	3	6
Iris	150	4	3
Crude Oil	56	5	3
Control Charts	1500	60	6
Wood Defects	232	17	13

Table 2. Parameters used in the clustering experiments

Algorithm	Parameters	Value
k-means	Maximum number of iterations	1000
GA	Crossover probability, μ_c	0.8
	Mutation probability, μ_m	0.001
	Population size, P	100
Bees Algorithm	Number of scout bees, n	21
	Number of sites selected for neighbourhood search, m	8
	Number of best “elite” sites out of m selected sites, e	2
	Number of bees recruited for best e sites, *nep*	5
	Number of bees recruited for the other (m-e) selected sites, *nsp*	2
	Number of iterations, R	300

As can be seen in Table 3, the proposed clustering method outperforms the other two algorithms in all cases. For example, for the *Control Charts* data set, Bees-Algorithm-based clustering produced an optimum mean value that is 22.3 per cent and 31.1 per cent better than both GA-clustering and k-means clustering, respectively. For the *Wood Defects* data set, the Bees Algorithm gave 3.2 and 40.7 per cent better results than GA-clustering and k-means clustering, respectively.

Table 3. Results for the tested clustering algorithms

Data set	Algorithm	Mean	Min.	Max.
Vowel	*k*-means	107.721	97.205	124.022
	GA	97.101	97.101	97.101
	Bees Algorithm	96.764	96.728	96.787
Iris	*k*-means	107.721	97.205	124.022
	GA	97.101	97.101	97.101
	Bees Algorithm	96.764	96.728	96.787
Crude Oil	*k*-means	279.662	279.485	279.743
	GA	278.965	278.965	278.965
	Bees Algorithm	277.339	277.227	277.558
Control Charts	*k*-means	2490.267	2464.813	2528.663
	GA	2322.234	2289.450	2355.690
	Bees Algorithm	1898.991	1860.440	1938.970
Wood Defects	*k*-means	228126.662	199306.380	270584.630
	GA	168035.200	157508.000	174784.000
	Bees Algorithm	162193.157	153866.531	173523.000

5 CONCLUSIONS AND FUTURE WORK

This paper has presented a new clustering method based on the Bees Algorithm. The method employs the Bees Algorithm to search for the set of cluster centres that minimises a given clustering metric. One of the advantages of the proposed method is that it does not become trapped at locally optimal solutions. This is due to the ability of the Bees Algorithm to perform local and global search simultaneously. Experimental results for different data sets have demonstrated that the proposed method produces better performances than those of the k-means algorithm and the GA-clustering algorithm.

One of the drawbacks of the Bees Algorithm is the number of tunable parameters it employs. A possible line of research, therefore, is to find ways to help the user choose appropriate parameters.

6 ACKNOWLEDGEMENTS

This work was carried out within the ERDF (Objective One) project “Supporting Innovative Product Engineering and Responsive Manufacturing” (SUPERMAN), the EC-funded Network of Excellence on “Innovative Production Machines and Systems” (I*PROMS) and the EPSRC-funded Innovative Manufacturing Research Centre at Cardiff University.

7 REFERENCES

[1] Pham, D.T. and Afify, A.A. Clustering techniques and their applications in engineering. Submitted to Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 2006.

[2] Jain, A.K. and Dubes, R.C. Algorithms for Clustering Data, 1988 (Prentice Hall, Englewood Cliffs, New Jersey, USA).

[3] Bottou, L. and Bengio, Y. Convergence properties of the k-means algorithm. Advances in Neural Information Processing Systems, 1995, 7, 585-592.

[4] Pham, D.T., Ghanbarzadeh, A., Koç, E., Otri , S., Rahim , S. and Zaidi, M. The Bees Algorithm – A novel tool for complex optimisation problems. In: Proceedings of the 2nd Virtual International Conference on Intelligent Production Machines and Systems (I*PROMS-06), Cardiff, UK, 2006, 454-459.

[5] Pham, D.T., Ghanbarzadeh, A., Koç, E. and Otri , S. Application of the Bees Algorithm to the training of radial basis function networks for control chart pattern recognition. In: Proceedings of the 5th CIRP International Seminar on Intelligent Computation in Manufacturing Engineering (ICME-06), Ischia, Italy, 2006, 711-716.

[6] Pham, D.T., Soroka, A., Ghanbarzadeh, A., Koç, E., Otri, S. and Packianather, M. Optimising neural networks for identification of wood defects using the Bees Algorithm, In: Proceedings of the IEEE International Conference on Industrial Informatics, Singapore, 2006, 1346-1351.

[7] Jain, A.K., Murty, M.N. and Flynn, P.J. Data clustering: A review. ACM Computing Survey, 1999, 31 (3), 264-323.

[8] Grabmeier, J. and Rudolph, A. Techniques of cluster algorithms in data mining. Data Mining and Knowledge Discovery, 2002, 6, 303-360.

[9] Han, J. and Kamber, M. Data Mining: Concepts and Techniques, 2001 (Academic Press, San Diego, California, USA).

[10] Maulik, U. and Bandyopadhyay, S. Genetic algorithm-based clustering technique, Pattern Recognition, 2000, 33 (9), 1455-1465.

[11] Von Frisch, K. Bees: Their Vision, Chemical Senses and Language, 1976 (Cornell University Press, Ithaca).

[12] Seeley, T.D. The Wisdom of the Hive: The Social Physiology of Honey Bee Colonies, 1996 (Harvard University Press, Cambridge).

[13] Camazine, S., Deneubourg, J., Franks, N.R., Sneyd, J., Theraula, G. and Bonabeau, E. Self-Organization in Biological Systems, 2003 (Princeton University Press, Princeton).