Evolutionary Rough K-Means Clustering

Pawan Lingras

Department of Mathematics and Computing Science, Saint Mary's University
Halifax, Nova Scotia, B3H 3C3, Canada
pawan@cs.smu.ca

Abstract. Rough K-means algorithm and its extensions have been useful in situations where clusters do not necessarily have crisp boundaries. Experimentation with the rough K-means algorithm has shown that it provides a reasonable set of lower and upper bounds for a given dataset. Evaluation of clustering obtained from rough K-means using various cluster validity measures has also been promising. However, rough K-means algorithm has not been explicitly shown to provide optimal rough clustering. This paper proposes an evolutionary rough K-means algorithm that minimizes a rough within-group-error. The proposal is different from previous Genetic Algorithms (GAs) based rough clustering, as it combines the efficiency of rough K-means algorithm with the optimization ability of GAs. The evolutionary rough K-means algorithm provides flexibility in terms of the optimization criterion. It can be used for optimizing rough clusters based on different criteria.

1 Introduction

Clustering in relation to rough set theory is attracting increasing interest among researchers [3,11,12,15,16,20]. Lingras [5] described how a rough set theoretic clustering scheme can be represented using a rough set genome. Rough set genomes were used to find an optimal balance between rough within-group-error and precision. However, the space requirement for rough set genomes as well as the convergence of the evolutionary process can be an issue for a large dataset. In subesquent publications [8,9], modifications of K-means and Kohonen Self-Organizing Maps (SOM) were proposed to create intervals of clusters based on rough set theory. Rough K-means algorithm and its variations [11,15] have been most popular methods for rough set clustering due to their simplicity and efficiency. However, rough K-means has not been shown to explicitly find an optimal clustering scheme for a particular cluster quality measure.

This paper combines the ability of genetic algorithms to evolve a near optimal solution based on a specified set of criteria along with the efficiency of rough K-means algorithm. The proposed evolutionary rough K-means algorithm will be used to optimize a distance based rough cluster quality measure. However, the proposal is capable of optimizing a clustering scheme for any other cluster quality measure such as the ones discussed in [7].

P. Wen et al. (Eds.): RSKT 2009, LNCS 5589, pp. 68–75, 2009.

2 Adaptation of Rough Set Theory for Clustering

Due to space limitations familiarity with rough set theory is assumed [14]. Let U be a set of objects. Rough sets were originally proposed using equivalence relations on U. However, it is possible to define a pair of upper and lower bounds $\left(\underline{A}(C), \overline{A}(C)\right)$ or a rough set for every set $C \subseteq U$ as long as the properties specified by Pawlak [13,14] are satisfied. Yao *et al.* [21] described various generalizations of rough sets by relaxing the assumptions of an underlying equivalence relation. Such a trend towards generalization is also evident in rough mereology proposed by Polkowski and Skowron [17] and the use of information granules in a distributed environment by Skowron and Stepaniuk [19]. The present study uses such a generalized view of rough sets. If one adopts a more restrictive view of rough set theory, the rough sets developed in this paper may have to be looked upon as interval sets.

Let us consider a hypothetical clustering scheme

$$U/P = \{C_1, C_2, \ldots, C_k\} \tag{1}$$

that partitions the set U based on an equivalence relation P. Let us assume that due to insufficient knowledge it is not possible to precisely describe the sets $C_i, 1 \leq i \leq k$, in the partition. However, it is possible to define each set $C_i \in U/P$ using its lower $\underline{A}(C_i)$ and upper $\overline{A}(C_i)$ bounds based on the available information. We will use vector representations, $\boldsymbol{u}, \boldsymbol{v}$ for objects and $\boldsymbol{c}_i$ for cluster C_i.

We are considering the upper and lower bounds of only a few subsets of U. Therefore, it is not possible to verify all the properties of the rough sets [13,14]. However, the family of upper and lower bounds of $\boldsymbol{c}_i \in U/P$ are required to follow some of the basic rough set properties such as:

(P1) An object $\boldsymbol{v}$ can be part of at most one lower bound

(P2) $\boldsymbol{v} \in \underline{A}(\boldsymbol{c}_i) \Longrightarrow \boldsymbol{v} \in \overline{A}(\boldsymbol{c}_i)$

(P3) An object $\boldsymbol{v}$ is not part of any lower bound

$$\Updownarrow$$

$\boldsymbol{v}$ belongs to two or more upper bounds.

Property (P1) emphasizes the fact that a lower bound is included in a set. If two sets are mutually exclusive, their lower bounds should not overlap. Property (P2) confirms the fact that the lower bound is contained in the upper bound. Property (P3) is applicable to the objects in the boundary regions, which are defined as the differences between upper and lower bounds. The exact membership of objects in the boundary region is ambiguous. Therefore, property (P3) states that an object cannot belong to only a single boundary region. Note that (P1)-(P3) are not necessarily independent or complete. However, enumerating them will be helpful in understanding the rough set adaptation of evolutionary, neural, and statistical clustering methods.

3 Genetic Algorithms

The origin of Genetic Algorithms (GAs) is attributed to Holland's [4] work on cellular automata. There has been significant interest in GAs over the last two decades. The range of applications of GAs includes such diverse areas as job shop scheduling, training neural nets, image feature extraction, and image feature identification [1]. This section contains some of the basic concepts of genetic algorithms as described in [1].

A genetic algorithm is a search process that follows the principles of evolution through natural selection. The domain knowledge is represented using a candidate solution called an *organism*. Typically, an organism is a single *genome* represented as a vector of length n:

$$c = (c_i \mid 1 \ \leq \ i \ \leq \ n)\,, \tag{2}$$

where c_i is called a *gene*.

Genetic Algorithm:
generate initial population, $G(0)$;
evaluate $G(0)$;
for(t = 1; solution is not found, t++)
generate $G(t)$ using $G(t-1)$;
evaluate $G(t)$;

Fig. 1. Abstract view of a generational genetic algorithm

An abstract view of a generational GA is given in Fig. 1. A group of organisms is called a *population*. Successive populations are called *generations*. A generational GA starts from initial generation $G(0)$, and for each generation $G(t)$ generates a new generation $G(t+1)$ using genetic operators such as *mutation* and *crossover*. The mutation operator creates new genomes by changing values of one or more genes at random. The crossover operator joins segments of two or more genomes to generate a new genome.

4 Existing Rough Clustering Approaches

Lingras [5] proposed a rough set genome, which consists of n genes, one gene per object in U. A gene for an object is a string of bits that describes which lower and upper approximations the object belongs to. The gene was partitioned into two parts, *lower* and *upper*. Both the lower and upper parts of the string consist of k bits each. The i^{th} bit in lower/upper string tells whether the object is in the lower/upper approximation of $\boldsymbol{c}_i$. The fitness function was a combination of the within-group-error [18] modified for the rough set representation of the clusters and precision of rough sets [13].

One of the major issues with the rough set genome based clustering was that the length of a genome was a function of the number of objects. For n objects and k clusters, there will be a total of $2 \times n \times k$ bits. For a large dataset, this not only increases the space requirements, but also makes it difficult for the evolution process to converge to an optimal solution. Experiments indicated that the rough genomes were practical for datasets with less than 1000 objects.

Lingras and West [8] provided an efficient alternative based on an extension of the K-means algorithm. K-means clustering is one of the most popular statistical clustering techniques [2,10]. Incorporating rough sets into K-means clustering requires the addition of the concept of lower and upper bounds. The incorporation required redefinition of the calculation of the centroids to include the effects of lower and upper bounds. The next step was to design criteria to determine whether an object belongs to the lower and upper bounds of a cluster.

The rough K-means approach has been a subject of further research. Peters [15] discussed various refinements of Lingras and West's original proposal [8]. These included calculation of rough centroids and the use of ratios of distances as opposed to differences between distances similar to those used in the rough set based Kohonen algorithm described in [9]. The rough K-means [8] and its various extensions [11,15] have been found to be effective in distance based clustering. However, there is no theoretical work that proves that rough K-means explicitly finds an optimal clustering scheme. Moreover, the quality of clustering that is maximized by the rough clustering is not precisely defined. The evolutionary rough K-means clustering described in the following section attempts to overcome the shortcomings of both the rough genome clustering and rough K-means clustering.

5 Combining Rough K-Means and Genetic Algorithms

This section proposes an evolutionary modification of the rough K-means algorithm. The objective of the proposed approach is to explicitly evolve an optimal clustering scheme. We demonstrate the optimization process with the help of a distance based measure, but the proposal can be used for optimization of any other cluster validity measure such as the ones discussed in [7].

The genome for the evolutionary algorithm has a total of $k \times m$ genes, where k is the desired number of clusters and m is the number of dimensions used to represent objects and centroids. The first m genes represent the first centroid. Genes $m+1, \ldots, 2 \times m$ give us the second centroid, and so on. Finally, $((k-1) \times m)+1, \ldots, k \times m$ corresponds to the k^{th} centroid.

In order to determine the fitness of a genome, we need to first assign an object to lower and/or upper bound of one of the clusters. For each object vector, $\boldsymbol{v}$, let $d(\boldsymbol{v}, \boldsymbol{c}_j)$ be the distance between itself and the centroid of cluster $\boldsymbol{c}_j$. Let $d(\boldsymbol{v}, \boldsymbol{c}_i) = \min_{1 \leq j \leq k} d(\boldsymbol{v}, \boldsymbol{c}_j)$. The ratios $d(\boldsymbol{v}, \boldsymbol{c}_i)/d(\boldsymbol{v}, \boldsymbol{c}_j)$, $1 \leq i, j \leq k$, are used to determine the membership of $\boldsymbol{v}$. Let $T = \{j : d(\boldsymbol{v}, \boldsymbol{c}_i)/d(\boldsymbol{v}, \boldsymbol{c}_j) \leq threshold \text{ and } i \neq j\}$.

1. If $T \neq \emptyset$, $\boldsymbol{v} \in \overline{A}(\boldsymbol{c}_i)$ and $\boldsymbol{v} \in \overline{A}(\boldsymbol{c}_j), \forall j \in T$. Furthermore, $\boldsymbol{v}$ is not part of any lower bound. The above criterion gurantees that property (P3) is satisfied.
2. Otherwise, if $T = \emptyset$, $\boldsymbol{v} \in \underline{A}(\boldsymbol{c}_i)$. In addition, by property (P2), $\boldsymbol{v} \in \overline{A}(\boldsymbol{c}_i)$.

It should be emphasized that the approximation space A is not defined based on any predefined relation on the set of objects. The lower and upper bounds are constructed based on the criteria described above.

The next step in calculating the fitness of a genome is to measure the validity of a clustering scheme. We will use one of the most intuitive distance based validity measure. The measure will accumulate the distances of the objects assigned to a cluster and its centroid as determined by the GAs:

$$\triangle = \sum_{i\ =\ 1}^{k} \sum_{\boldsymbol{u} \in \boldsymbol{c}_i} d(\boldsymbol{u}, \boldsymbol{c}_i), \tag{3}$$

where the function d provides the distance between two vectors. The distance $d(\boldsymbol{u}, \boldsymbol{v})$ is given by:

$$d(\boldsymbol{u}, \boldsymbol{v}) \ = \ \sqrt{\frac{\sum_{j=1}^{m} (u_j - v_j)^2}{m}}. \tag{4}$$

We need to adapt the above measure for the rough set theory by creating lower and upper versions of the error as:

$$\underline{\triangle} = \sum_{i\ =\ 1}^{k} \sum_{\boldsymbol{u} \in \underline{A}(\boldsymbol{c}_i)} d(\boldsymbol{u}, \boldsymbol{c}_i), \text{ and} \tag{5}$$

$$\overline{\triangle} = \sum_{i\ =\ 1}^{k} \sum_{\boldsymbol{u} \in \overline{A}(\boldsymbol{c}_i) - \underline{A}(\boldsymbol{c}_i)} d(\boldsymbol{u}, \boldsymbol{c}_i). \tag{6}$$

The rough error is then calculated as a combination of the lower and upper error:

$$\triangle_{rough} = w_l \times \underline{\triangle} + w_u \times \overline{\triangle}. \tag{7}$$

The rough within-group-error defined above is computationally more efficient than a similar measure used for rough set genome [5]. The rough set genome clustering accumulated errors between objects belonging to lower and upper bounds of a cluster, which requires quadratic computational time. The rough within-group-error given by Eq. (7) requires linear time with respect to the number of objects in the lower and upper bounds of a cluster.

We used the synthetic data set developed by Lingras *et al.* [6] to test the validity of the evolutionary rough K-means. In order to visualize the data set, it was restricted to two dimensions as can be seen in Fig. 2. There are a total of 65 objects. It is obvious that there are three distinct clusters. However, five objects do not belong to any particular cluster. We performed rough clustering on the synthetic data set for $threshold = 1.4, w_l = 0.6,$ and $w_u = 0.4$. The GAs used

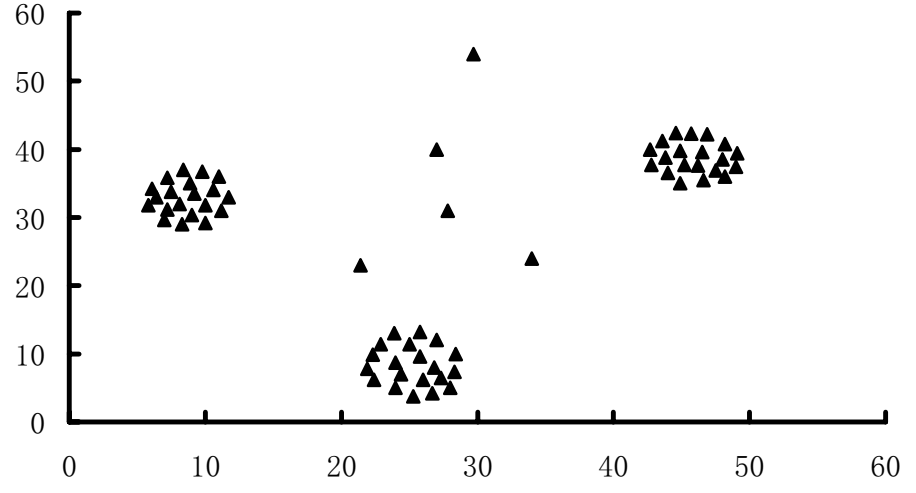

Fig. 2. Synthetic data

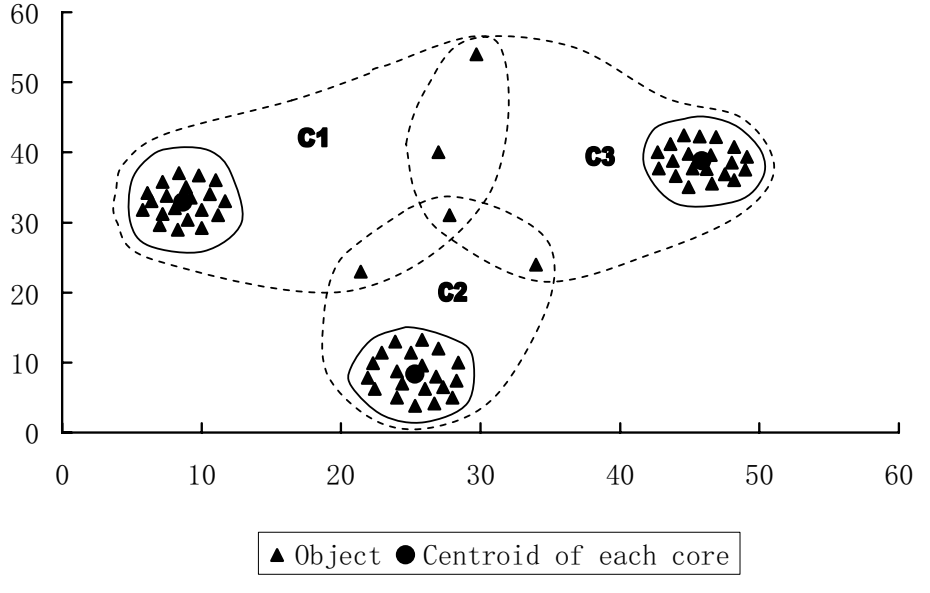

Fig. 3. Rough clusters for the synthetic data

the crossover probability of 70% and mutation probability of 10%. Population size for each generation was set at 100, and the GAs managed to evolve to the same solution as rough K-means within 50 generations. The evolved rough clustering is shown in Fig. 3. The rough clustering is the same as obtained by Lingras *et al.* [6] using rough K-means with a $threshold = 1.4$.

The size of a genome used in the proposed evolutionary rough K-means is $k \times m$, which compares favourably with $2 \times n \times k$ for the previous rough set genome clustering. Usually, $k, m \leq 50$, while n can be as high as a million. The smaller size of a genome increases the chances of convergence to a near optimal solution. We have used rough within-group-error given by Eq. (7) as a cluster quality measure that is optimized by the GAs. However, the proposed evolutionary rough K-means algorithm allows us to substitute a different cluster validity measure depending on the application. This ability to optimize clustering for different cluster quality measures is an advantage over the conventional rough K-means algorithm. Moreover, preliminary experiments with a small dataset indicate the tendency of the proposed algorithm to converge to an optimal solution within relatively few generations.

6 Conclusions

This paper combines the efficiency of rough K-means algorithm with the ability of genetic algorithms to find a near optimal solution based on a cluster quality measure. The genome used in the proposed evolutionary rough K-means algorithm only has $k \times m$ genes, which makes for a reasonable memory requirement and increases the chances of evolving to a near optimal solution based on a specified criterion. The paper demonstrates the use of the proposed algorithm for a rough within-group-error measure. However, the proposal allows for optimization of other cluster quality measures such as the ones discussed in [7]. The algorithm was tested for a small synthetic dataset. The fact that the evolution only needed 50 generations indicates that the convergence of the evolutionary rough K-means may be comparable to the original rough K-means algorithm. We plan to test the evolutionary rough K-means clustering for optimization with different cluster quality measures and compare its efficiency with the rough K-means algorithm for large datasets. Results of our experiments will appear in future publications.

Acknowledgment

The author would like to thank the Natural Sciences and Engineering Research Council of Canada and the Faculty of Graduate Studies and Research, Saint Mary's University for funding.

References

1. Buckles, B.P., Petry, F.E.: Genetic Algorithms. IEEE Computer Press, Los Alamitos (1994)
2. Hartigan, J.A., Wong, M.A.: Algorithm AS136: A K-Means Clustering Algorithm. Applied Statistics 28, 100–108 (1979)
3. Hirano, S., Tsumoto, S.: Rough Clustering and Its Application to Medicine. Journal of Information Science 124, 125–137 (2000)

4. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
5. Lingras, P.: Unsupervised Rough Set Classification using GAs. Journal Of Intelligent Information Systems 16(3), 215–228 (2001)
6. Lingras, P., Chen, M., Miao, D.: Rough multi-category decision theoretic framework. In: Wang, G., Li, T., Grzymala-Busse, J.W., Miao, D., Skowron, A., Yao, Y. (eds.) RSKT 2008. LNCS, vol. 5009, pp. 676–683. Springer, Heidelberg (2008)
7. Lingras, P., Chen, M., Miao, D.: Rough Cluster Quality Index Based on Decision Theory. Submitted to IEEE Transactions on Knowledge and Data Enginering (2008)
8. Lingras, P., West, C.: Interval Set Clustering of Web Users with Rough K-means. Journal of Intelligent Information Systems 23(1), 5–16 (2004)
9. Lingras, P., Hogo, M., Snorek, M.: Interval Set Clustering of Web Users using Modified Kohonen Self-Organizing Maps based on the Properties of Rough Sets. Web Intelligence and Agent Systems: An International Journal 2(3) (2004)
10. MacQueen, J.: Some Methods for Classification and Analysis of Multivariate Observations. In: Proceedings of Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297 (1967)
11. Mitra, S., Bank, H., Pedrycz, W.: Rough-Fuzzy Collaborative Clustering. IEEE Trans. on Systems, Man and Cybernetics 36(4), 795–805 (2006)
12. Nguyen, H.S.: Rough Document Clustering and the Internet. Handbook on Granular Computing (2007)
13. Pawlak, Z.: Rough Sets. International Journal of Information and Computer Sciences 11(145-172) (1982)
14. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1992)
15. Peters, G.: Some Refinements of Rough k-Means. Pattern Recognition 39(8), 1481–1491 (2006)
16. Peters, J.F., Skowron, A., Suraj, Z., Rzasa, W., Borkowski, M.: Clustering: A rough set approach to constructing information granules. In: Soft Computing and Distributed Processing. Proceedings of 6th International Conference, SCDP 2002, pp. 57–61 (2002)
17. Polkowski, L., Skowron, A.: Rough Mereology: A New Paradigm for Approximate Reasoning. International Journal of Approximate Reasoning 15(4), 333–365 (1996)
18. Sharma, S.C., Werner, A.: Improved method of grouping provincewide permanent traffic counters. Transportation Research Record 815, 13–18 (1981)
19. Skowron, A., Stepaniuk, J.: Information granules in distributed environment. In: Zhong, N., Skowron, A., Ohsuga, S. (eds.) RSFDGrC 1999. LNCS (LNAI), vol. 1711, pp. 357–365. Springer, Heidelberg (1999)
20. Voges, K.E., Pope, N.K.L.l., Brown, M.R.: Cluster Analysis of Marketing Data: A Comparison of K-Means, Rough Set, and Rough Genetic Approaches. In: Abbas, H.A., Sarker, R.A., Newton, C.S. (eds.) Heuristics and Optimization for Knowledge Discovery, pp. 208–216. Idea Group Publishing (2002)
21. Yao, Y.Y.: Constructive and algebraic methods of the theory of rough sets. Information Sciences 109, 21–47 (1998)