*Journal of Engg. Research* **Vol. 4 No. (2) June 2016** pp. 1-16

# معيار جديد متعدد الموضوعية يقوم على تعظيم النمطية

***محمد خدابرستی، **محدثه جنجی، ***سعید امیرقلی بور و***ابوصالح محمد شریفي**

*قسم الكمبيوتر، جامعة ازاد الإسلامية، فرع رامسر، رامسر، ايران

**قسم الحساب وأنظمة المعلومات، جامعة ملبورن، ملبورن، استراليا

***نادي النخبة والباحثون الشباب، جامعة ازاد الإسلامية، فرع رامسر، رامسر، ايران

## الخلاصة

خوارزميات التجميع التقليدية تستخدم معيار واحد فقط الذي قد لا يتوافق مع الأشكال المتنوعة للمجموعات الأساسية. ولكن في هذه الورقة، نحن نستخدم اثنين من المعايير المهمة واقتراح نموذج جديد لفرقة عنقودية متعددة الموضوعية لتمكين العثور على مجموعات من أنواع مختلفة. المعيار الأول هو مجموع مربعات الخطأ المعروفة. المعيار الثاني هو النمطية وهي في الأصل مقياسا لتقييم المجتمعات المحلية في الشبكات الاجتماعية. نحن نعظم النمطية بوصفها وظيفة إجماع الفرقة العنقودية. ومن أجل إضافة المزيد من التحسين، فنحن أيضا نعدل الخوارزمات الجينية الغير مهيمنة الفرز ونقترح لها مشغل عام متخصص. وقد اظهرت النتائج لأكثر من ثلاثة عشر مجموعة بيانات حقيقية أن الطريقة المقترحة تتفوق على التكتلات الأخرى الملتقية.

# A new multi-objective cluster ensemble based on modularity maximization

Mohammad Khodaparasti*, Mohadeseh Ganji**, Saeed Amirgholipour*** and Aboosaleh Mohammad Sharifi***

**Department of Computer, Islamic Azad University, Ramsar Branch, Ramsar, Iran*

***Department of Computing and Information Systems, The University of Melbourne, Melbourne, Australia*

****Young Researchers and Elite club, Islamic Azad University, Ramsar Branch, Ramsar, Iran*

**Corresponding author: m.khodaparasti@iauramsar.ac.ir*

## ABSTRACT

Conventional clustering algorithms utilize only one single criterion that may not conform to diverse shapes of the underlying clusters. But in this paper, we use two important criteria and propose a new multi-objective cluster ensemble model to empower finding clusters of different types. The first criterion is the well-known sum of squared error. The second criterion is modularity, which is originally a measure of evaluating communities in social networks. We maximize modularity as a consensus function of cluster ensemble. In order to add further improvement, we also modify the Non-dominant sorting genetic algorithm (NSGAII) and propose a specialized crossover operator for it. Experimental results over thirteen UCI real data sets show that the proposed method outperforms other clustering methods.

**Keywords:** Cluster ensemble; genetic algorithm; modularity; multi-objective clustering; non-dominant sorting.

## INTRODUCTION

Clustering as an important task of data mining has attracted a lot of attention in practice (Celebi, 2015). Clustering is the problem of assigning a set of unlabeled objects into groups or clusters. Clustering algorithms have been widely applied to many different application areas, such as image processing, text categorization, market segmentation, etc.

In real-world problems, clusters can appear with different shapes, sizes, sparseness, and degrees of separation. Most traditional clustering methods optimize only one criterion and are often very effective on those data sets that match the objective function used therein (Alizadeh *et al.,* 2013). Therefore, selection of the best clustering algorithm for a given data set is one of the main challenges in cluster analysis. One way of overcoming this problem is to combine several clustering criteria to improve

the quality of the final solution. Nowadays, there are two main approaches, which address the use of multiple clustering criteria: multi-objective clustering and cluster ensemble.

The essence of multi-objective clustering algorithms is to find clusters in the given data set, based on different objective functions. Multi-objective clustering is a two-step process (Law *et al.,* 2004): (i) independent discovery of clusters using different clustering algorithms, and (ii) construction of an optimal partition from the discovered clusters. In the context of multi-objective clustering, Law *et al.* (2004) developed a bi-objective clustering model, which clusters data by a set of candidate objective functions and their integration into the target partition. An essential ingredient of their approach is a cluster evaluation function that assesses the number of re-occurrences of clusters using re-sampling techniques. Then, final clustering is obtained as a solution to a discrete optimization problem. Handl & Knowles (2007) described a Pareto-based multi-objective evolutionary algorithm, which was named as multi-objective clustering with automatic K-determination (MOCK). The algorithm optimizes two complementary clustering criteria: overall deviation and connectivity. MOCK returns a number of different trade-off partitions over a range of different cluster numbers that are an approximation of the Pareto optimal set. However, as the number of alternatives increases, the analysis becomes harder. To overcome this difficulty, MOCK includes a mechanism to automatically select the best partitions from the set of solutions. The selection is based on the shape of the approximation of the Pareto front. Zheng *et al.* (2012) proposed a multi-objective gene expression programming for data clustering (MGEPC), which could automatically determine the number of clusters and the appropriate partitioning of the data set.

In cluster ensemble methods after establishing an ensemble of single criterion partitions in the first step, the final solution is derived by aggregation using a consensus function. Strehl & Ghosh, (2003) introduced the concept of cluster ensemble. They also proposed three hypergraph-based consensus functions. They transformed the data partitions into a hypergraph representation in which the vertices are data points and the hyperedges are clusters. After that, they employed minimum cut hypergraph algorithms in order to partition vertices or data points. The minimum k-cut of this hypergraph gives the final consensus partition. Later, Topchy *et al.* (2004) suggested a probabilistic model of consensus partitions using a finite mixture of multinomial distributions. In another work, Fred & Jain (2006) offered a new clustering ensemble method, which learns the pairwise similarities between points in order to facilitate a proper partitioning of data without a prior knowledge of the number and shape of the clusters.

The clustering problem in the context of complex network analysis is called network community detection. Community in community detection problem has

same meaning as cluster in clustering. In simple words, the community detection problem seeks for disjoint communities in which connections within the communities are stronger than connections to members of different communities. Newman & Girvan (2004) proposed a new measure, called modularity, to evaluate goodness of communities in a given network. Soon, the modularity maximization method became a well-known network community detection approach (Newman, 2012). It compares edge distribution of the network with a randomized graph of the same characteristics and extracts the final communities based on the information gained by this comparison. Because of the conceptual similarities between clustering and community detection problems, it is expected that modularity maximization methods can perform well in clustering problem. As the definition of modularity is based on graph theory, some adaptations are needed to use it in the new context. Alizadeh *et al.* (2013) and Alizadeh (2013) introduced such adaptations and used modularity to improve the performance of cluster ensemble methods. They supposed data points as nodes of a full edged graph in which the weights of edges are derived from information of an ensemble. According to this definition, modularity measure can also be used for evaluating data clustering in continuous space.

In this paper, we take advantages of multi-objective clustering, cluster ensemble, and modularity criterion, so as to develop a new multi-objective cluster ensemble model. The proposed model has two objectives; maximizing modularity and minimizing sum of squared error. We use modularity measure in a cluster ensemble framework. After modeling, we have developed a modified NSGAII to solve the model. Using single criterion clustering methods for generating initial population and defining a new specialized crossover operator are our novelties in this section.

The rest of this paper is organized as follows: Section 2 introduces the proposed model. The modified NSGAII algorithm is presented in Section 3. Section 4 reports computational results. Conclusions as well as some possible future directions are given in Section 5.

## THE PROPOSED MODEL

In this section we explain the proposed model of multi-objective cluster ensemble after introducing some preliminaries.

### Notations and definitions

*Clustering problem:* Given a data set D = (x1, x2, . . . ,xn), where xi is the i-th data point in a d dimensional feature space, clustering solution C = (C1,C2, . . . , Ck) represents a partition consisting k clusters. The clustering algorithm assigns these n data instances to k clusters such that each instance xi must be assigned to exactly one cluster and every cluster must include at least one member. In most of the clustering tasks the number of clusters is known in advance.

*Multi-objective clustering:* Suppose that there is a data set D = (x1, x2, . . . ,xn) and L clustering algorithms Ai, i = 1, ... , L, such that each algorithm Ai returns a partition C(i) of D which maximizes its corresponding objective function fi. Formally

$$C(i) = \{C_1^{(i)}, C_2^{(i)}, ..., C_{M_i}^{(i)}\} = \underset{C(D)}{\operatorname{argmax}} \; f_i(C(D)) \tag{1}$$

Where C(D) denotes an arbitrary partition of D and Cj(i) is the j-th cluster in C(i). Mi stands for the number of clusters in the partition generated by the Ai. Let C ≡ ∪i C(i) be a collection of all the clusters generated by a candidate algorithm { Ai }:

$$C = \{C_1^{(1)}, ..., C_{M_1}^{(1)}, C_1^{(L)}, ..., C_{M_L}^{(L)}\} \tag{2}$$

The goal of multi-objective clustering is to find a "compromise" partition C*≡{C1*,…,Ck*} based on the partition C={C(1),…,C(L)}. In other words, the target collection of clusters, C*, is derived from the primary clusters in C. Hence each cluster in C* is equivalent to one of the partitions created by a primary clustering algorithm.

$$C^* = \{C_1^*, ..., C_k^* : \forall c \;\; \exists i, j \quad C_c^* \approx C_j^{(i)}\} \tag{3}$$

In order to interpret C* as a partition, the following conditions must be satisfied:

No data point xi is assigned to more than one cluster in C*.

Each data point in D is assigned at least to one cluster in partition C*.

*Modularity matrix:* Suppose that there is a weighted similarity graph in which Ai,j indicates weight of the edge connecting nodes i and j; $k_i = \sum_{j=1}^{n} A_{i,j}$ is the degree of node i; and $T = \sum_{i=1}^{n} k_i$ is the sum of the all degrees in the graph. It can be shown that the expected number of edges that fall between nodes i and j in its corresponding randomized graph is given by kikj/T (Newman & Girvan, 2004). The modularity matrix M=[mi,j] is defined as the actual number of edges between i and j minus the expected one, i.e. mi,j=Aij-kikj/T.

*Modularity measure:* Given a modularity matrix M and a vector (y1,…,yn), where yi is the cluster label of node i, the modularity measure Q is defined as:

$$Q = \frac{1}{T} \sum_{i,j} m_{i,j} \delta(y_i, y_j) \tag{4}$$

Where δ is the Kronecker delta that is 1 if its arguments are equal, and 0 otherwise. In simple words, modularity is defined to be the fraction of edges within the same communities minus the expected values. If there are more edges within clusters than what is expected in a randomized network, then the modularity will have a positive value. A large positive value indicates good community division.

*Sum of squared error:* Given a partition including k cluster centers, the sum of squared error measures total squared differences between every data point and its cluster center.

$$SSE = \sum_{p=1}^{k} \sum_{x_i \in C_p} \sum_{f=1}^{d} \left( c_f^p - x_{i,f} \right)^2 \tag{5}$$

Where cfp is f-th dimension of the p-th cluster center and xi,f is f-th dimension of the i-th data point.

## The proposed model

The objective function of our proposed multi-objective cluster ensemble model has two components. The first one is minimizing sum of squared error which is a well-known criterion for evaluating clusters. The second component is maximizing modularity measure in a cluster ensemble framework.

The proposed framework for calculating the objective functions is demonstrated in Figure 1. As it is shown in Figure 1, the sum of squared error (SSE) of a given a candidate solution is directly calculated from the input data set as defined in Equation (5). To figure the second objective, an ensemble of existing single objective algorithms including single linkage, weighted linkage, average linkage, fuzzy c-means and spectral clustering is generated. Then, we count the occurrence of samples in a same cluster in this ensemble based on evidence accumulation clustering method (Fred & Jain, 2005). The entry Ai,,j from this n×n similarity matrix reflects the number of times that the i-th and j-th data points appear in the same cluster. The range of values in this matrix is between [0, ensemble_length], where ensemble_length indicates maximum similarity. Given this similarity matrix, its corresponding modularity matrix is simply computed using the definition of modularity matrix, emphasizing that it is constant and is not updated during the optimization step.

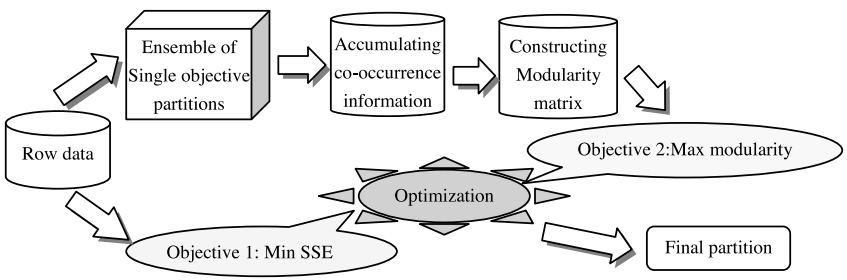


**Fig. 1.** The proposed framework for calculating objective functions

It is worth mentioning that we do not use k-means algorithm in the ensemble. Because the conceptual goal of k-means is to minimize sum of squared error, using it

in the ensemble leads to a bias toward the first objective. Using these two objective functions in the multi-objective clustering model can be an effective combination since they are heterogeneous. For example, sum of squared error is minimized by increasing the number of clusters, whereas, modularity may reach to its optimum value when data points are grouped in a few number of clusters. The use of modularity for clustering is a novel idea which, leads to better clustering.

The proposed multi-objective cluster ensemble model is presented below by Equations (6)-(8).

$$\min \sum_{p=1}^{k} \sum_{x_i \in C_p} \sum_{f=1}^{d} \left(c_p^f - x_{i,f}\right)^2 - \sum_{i,j} m_{i,j} \delta(y_i, y_j) \tag{6}$$

$$s.t$$

$$c_p^f = \frac{\sum_{x_i \in C_p} x_{i,f}}{|C_p|} \qquad f = 1,...,d \quad p = 1,...,k \tag{7}$$

$$1 \le y_i \le k \qquad i = 1,...,n \tag{8}$$

where constraint (7) calculates d-th feature of p-th cluster center. The model variables, (y1,…,yn), are the desired clustering labels. Thus, each yi has a value between 1 and the number of clusters, k. The objective function in this model is nonlinear and the second part is not explicit. We propose a specialized NSGAII algorithm to solve the proposed model. The detailed description of the proposed algorithm is presented next.

## PROPOSED ALGORITHM

We use the well-known NSGAII algorithm with two modifications. We specialize the initial population and crossover operator of the original NSGAII algorithm. In our representation, each chromosome is denoted by a vector of n integer values (y1,…,yn) as the genes which shows the cluster labels of the corresponding data points, where $y_i \in \{1,...,k\}$.

### Initial population

Unlike most genetic algorithms that begin with a random population, we take advantage of a partially guided initial population in our proposed algorithm. To be more precise, it is consisted of some base partitions resulting from single objective algorithms. Each of the hired algorithms has a unique criterion for clustering that performs well in special types of data. We use six single objective algorithms to generate the initial population namely k-means, single linkage, weighted linkage, average linkage, fuzzy C-means, and spectral clustering (Ng *et al.,* 2002) algorithms. Although effective initialization approaches for K-means is discussed by Celebi *et al.* (2013), in these experiments, we randomly seed k-means and fuzzy C-means algorithms. We use these primary

algorithms to generate half of the initialization solutions. In addition, the other half of the initial population is generated randomly to assure that the initial solutions are diverse enough.

## Crossover operator

Most of the traditional operators of genetic algorithms used for clustering problems manipulate gene values without taking into account their connections with the other genes. Our proposed crossover operator for NSGAII algorithm considers similarities between parents and generates a child, which is not violating characteristics, which both the parents share in common. It inherits parents' similar genes and it accepts other gens from each of them randomly. More precisely, given P1 and P2 as the parents, the crossover operator re-labels P2 to find its best match to P1. Remind that a same cluster number in different partitions does not mean an identical cluster. Thus, the relabeling process is needed to find the best matching between clusters of different partitions. We use a state-of-the-art relabeling method, which is recently presented in Alizadeh (2013) and Alizadeh *et al.* (2013). After relabeling, the identical genes in both the parents are fixed in the corresponding child's genes. The remaining genes are randomly inherited from each of the parents. In the case that both the parents are totally identical, the operator hires a local search mechanism which, looks the neighborhood of the parent for a better solution. It changes only one element of the parent to achieve an improvement according to the defined objective function. Pseudo code of proposed crossover operator is as follows.

---

*Algorithm 1: Proposed Crossover Operator*

---

*Begin*
*- Input: P1 and P2 // the two input parents*
*- Generate a CH with the same dimensions to the parents*
*- Rp2 = Relabel P2 according to P1*
*- Find I = (P1 ∩ RP2) //similar genes in parents (where RP2==P1)*
*- If I = P1 = RP2 //P1 and RP2 are completely the same*
*- CH = Algorithm2 (P1) // Algorithm2 is a local search mechanism*
*Else*
*- CH = I // Copy I to the new child*
*- For each remaining genes*
*- copy the gene from P1or P2 randomly*
*- End for*
*End if*
*- Output: CH*
*End*

---

As we mentioned before, a local search mechanism is employed, when both the parents are totally the same. Pseudo code of the suggested local search mechanism is as follows.

*Algorithm 2: Proposed Local Search Algorithm*

*Begin*

*- Input: P // the parent vector*

*- OB = Choose one of the objectives randomly*

*- for all genes in P*

*- $g_i$ =i-th gene of P, where i is picked by subsampling from {1,2,3,…,n}*

*- Change $g_i$ to $g_i$' // $g_i$' is a new feasible value*

*- P'=new P with $g_i$' replaced with $g_i$*

*- If OB(P') > OB(P)*

*- Output = P'*

*- Go to the End*

*- End if*

*- End for*

*Output = P*

*End*

## Mutation

The mutation function changes a chromosome slightly. It exchanges one randomly picked element of the input chromosome to a different randomly-generated value which retains feasibility. In other words, the mutation operator picks a data instance and changes its cluster label to another label randomly.

## Next generation selection

To select members of the next generation, all members of current generation and produced children are ranked. The ranking is done in two levels. At first the non-dominant sorting method sorts them into different non-domination levels (fronts). Then, members of each front are ranked according to crowd distance criterion which has been introduced in Deb *et al.* (2002). Pseudo code of the modified NSGAII algorithm is as follows.

*Algorithm 3: Modified NSGAII Algorithm*

*Begin*

*Input: The Modularity and SSE objectives*

*- t =0*

*- P(t) = Generate an initial population of size N by clustering methods*

*- M = Construct modularity matrix using P(t)  // it is used in objective calculation*

*- While t ≤ MAX_GENERATION*

*- CAND= {P(t)}  // candidate solutions*

*- Calculate each objective values for all members of P(t)*

*- Rank P(t) using non-dominant sorting method and crowd distance*

*- OFFS = Produce N offsprings using defined crossover and mutation operators*

*- Calculate each objective values for OFFS*

*- CAND= {P(t) ∪ OFFS}*

*- Rank CAND using non-dominant sorting method and crowd distance*

*- t = t + 1*

*- P(t) = N top-ranked members of the CAND  //the next population*

*- Endwhile*

*Output: A set of non-dominant solutions*

*End*

## EXPERIMENTAL RESULTS

This section presents the experimental results obtained by applying the proposed method as well as other well-known clustering algorithms to a diverse set of standard benchmarks. All the methods are implemented in MATLAB 7.7. Each of the results in this paper is the average of fifty independent runs of the algorithms. The population size for implementing the proposed genetic algorithm is chosen to be 50. The maximum number of generations is set to 150 and run time limitation is 3 hours. Regarding the time needed for MOCE algorithm, as it is a population based algorithm, it takes more time than single criterion algorithms in general. However, the required time depends on the size of the data set and the number of initial population and number of generations.

In our first experiment, we compare average performance of the proposed scheme with different sorts of initial population. In Figure 2, the average sum of squared error is shown in all 50 generations on Iris data set from UCI repository. The triangle sign

shows performance of totally random initialization. The circle represents 90% guided initialization and the plus sign shows our applied initialization, which contains 50% guided initialization. Results shows that complete guided initialization results in premature convergence and the partially guided initialization (50%) performs much better than random initialization in terms of quality of the solutions.

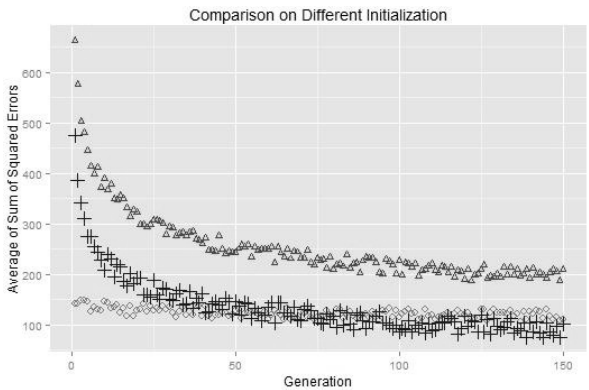


**Fig. 2.** Comparing different initialization schemes

Now that choice of partially random initialization is properly justified, we present an experiment demonstrating priority of the proposed crossover scheme over random crossover.

Figure 3 demonstrates average SSE in each generation for Iris data set. The triangle sign represents using random crossover and the plus sign curve shows the result of using the proposed crossover. Based on the convergence curves in this figure, we can see the proposed crossover can achieve better results by exploring the solution space more efficiently that the total random crossover.

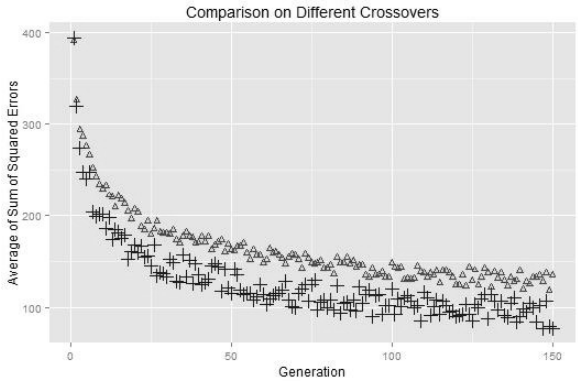


**Fig. 3.** Comparison convergence of the proposed crossover with random crossover

13 real world data sets have been taken from UCI repository of machine learning, which are varying in the number of clusters, features and samples. Table 1 gives brief information about the used data sets.

**Table 1.** Characteristics of the used data sets

| Formal name | #instances | #Features | #Clusters |
|---|---|---|---|
| Wine | 178 | 13 | 3 |
| Iris | 150 | 4 | 3 |
| Ecoli | 336 | 7 | 8 |
| Seeds | 210 | 7 | 3 |
| Vertebral Column 3C | 310 | 6 | 3 |
| Glass Identification | 214 | 9 | 6 |
| Breast Tissue | 106 | 9 | 6 |
| Ionosphere | 351 | 33 | 2 |
| KDD-Synthetic | 600 | 60 | 6 |
| Letters | 600 | 16 | 3 |
| Vehicle | 846 | 18 | 4 |
| WDBC | 569 | 30 | 2 |
| Yeast | 1484 | 8 | 10 |

For all of these data sets, the real class labels are already known. Thus, percentage of the samples that are correctly classified is used as the performance criterion for clustering method. More precisely, after re-labeling process of the obtained clusters with respect to the actual classes, accuracy of the method is given by calculating percentage of correct classified samples to the whole data points.

As we have discussed in the introduction, each single criterion clustering algorithm optimizes its own objective function and it can only perform well on specific data sets. Besides, there is no knowledge about the most appropriate algorithm for an unknown data set. Therefore, it is common in the literature to compare the performance of a new multi criteria method with average performance of single objective algorithms in order to evaluate the new method. Results of such comparison between the proposed method and a number of the most renowned single objective clustering algorithms are represented in Table 2.

**Table 2.** Results of the proposed method in comparison with average of six well known single criterion clustering algorithms in terms of accuracy

| Data sets | Proposed Method | Single Algorithms | # non-dominant |
|---|---|---|---|
| Breast Tissue | 34.6 | 28.8 | 1.7 |
| Iris | 88.9 | 85.4 | 1.5 |
| Wine | 69.5 | 56.4 | 17.75 |
| Seeds | 88.2 | 78.2 | 1.7 |
| Glass Identification | 46.0 | 43.2 | 1.5 |
| Vertebral Column 3C | 51.0 | 49.2 | 15.16 |
| Ecoli | 77.3 | 59.8 | 9.5 |
| Ionosphere | 69.8 | 68.0 | 4.6 |
| WDBC | 74.3 | 61.2 | 8.2 |
| KDD-Synthetic | 76.7 | 64.3 | 3.5 |
| Letters | 41.5 | 39.8 | 3.8 |
| Vehicle | 89.3 | 74.3 | 10.5 |
| Yeast | 44.7 | 34.6 | 14.7 |

The first column of Table 2 is the name of data sets. Average accuracy obtained from the proposed algorithm is presented in second column. In addition, the average accuracy of six single criterion algorithms is presented in the third column. These single criterion clustering algorithms are k-means, single linkage, average linkage, weighted linkage, spectral clustering and fuzzy C-means algorithm. The greater accuracy obtained from each data set is highlighted in bold. As it is seen in this table, the proposed method definitely outperforms the average results of the other six single-criterion algorithms in terms of accuracy. Applying the proposed multi-objective algorithm to a diverse set of data sets, confirms the excellent performance of our method in dealing with the clustering problem, especially in the case of facing with an unknown data set.

As the proposed method is a multi-objective cluster ensemble one, the outputs of the presented method are non-dominant solutions. Another criterion to evaluate a multi-objective method is the number of non-dominant solutions that it finds. The fourth column in Table 2 reports average number of non-dominant solutions that the proposed method finds for each data set. The number of returned non-dominant solutions ranges from 1.5 to 17.75, which are acceptable results for such problems.

In addition to superior results of the proposed method in Table 2, we also compare our method's performance with the individual single objective algorithms in Table 3. Again, the best obtained accuracy in each data set is highlighted in bold. It is shown in this table that the proposed method achieves the best accuracy among other algorithms in four data sets and it gains admissible accuracies in the others. The closest algorithms that win the first ranked results in two benchmarks are k-means and average linkage. However, their striking drops in some other data sets make them unreliable. For example, k-means and average linkage perform disappointing in Ecoli and Glass data sets respectively. On the other hand, our proposed algorithm performs much more reliably in comparison with existing methods.

**Table 3.** Results of the proposed method in comparison with six well known single criterion clustering algorithms in terms of accuracy

| Data sets | Proposed Method | Kmeans | Single Linkage | Average Linkage | Complete Linkage | Spectral | Fuzzy |
|---|---|---|---|---|---|---|---|
| Breast Tissue | 34.6 | 34.0 | 23.6 | 28.3 | 24.5 | 29.3 | 33.1 |
| Iris | 88.9 | 84.3 | 68.0 | 90.7 | 90.0 | 90.7 | 88.4 |
| Wine | 69.5 | 67.7 | 42.7 | 61.2 | 56.2 | 41.2 | 69.1 |
| Seeds | 88.2 | 88.4 | 37.1 | 91.0 | 76.2 | 88.6 | 88.1 |
| Glass | 46.0 | 51.6 | 36.4 | 37.9 | 38.8 | 45.8 | 48.6 |
| Vertebral Column | 51.0 | 56.9 | 47.7 | 47.1 | 44.2 | 48.5 | 50.7 |
| Ecoli | 77.3 | 56.1 | 44.9 | 76.5 | 75.9 | 54.2 | 51.0 |
| Ionosphere | 69.8 | 71.2 | 64.4 | 64.4 | 68.7 | 68.5 | 70.9 |
| WDBC | 74.3 | 61.8 | 62.5 | 54.2 | 54.2 | 60.4 | 74.0 |
| KDD-Synthetic | 76.7 | 75.3 | 63.2 | 63.0 | 40.8 | 66.9 | 76.5 |
| Letters | 41.5 | 43.6 | 26.1 | 37.6 | 43.6 | 43.1 | 44.9 |
| Vehicle | 89.3 | 85.4 | 62.9 | 66.3 | 66.3 | 79.3 | 85.4 |
| Yeast | 44.7 | 36.9 | 32.4 | 31.7 | 32.3 | 39.9 | 34.3 |
| Average | 65.5 | 62.5 | 47.1 | 57.7 | 54.7 | 58.2 | 62.7 |

To have a better comparison, the results of each algorithm over the whole thirteen data sets are averaged and depicted in Figure 4. As it is shown in this figure, the proposed algorithm has the best performance in terms of averaged accuracy over all thirteen benchmarks. It again confirms the idea that employing multi-objective clustering methods perform better than even using very good single objective ones, as the average accuracy of our method is superior to the best of the others more than 2.7%.

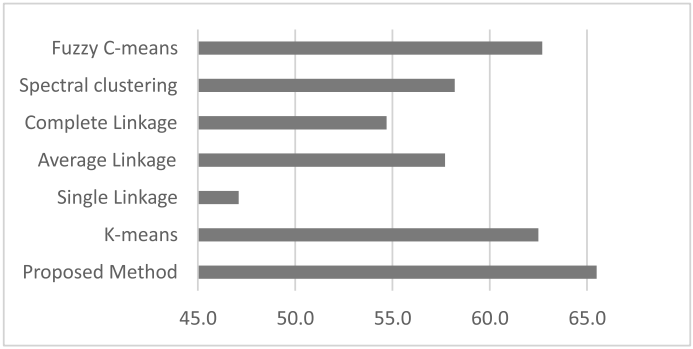


**Fig. 4.** Results of the proposed method in comparison with other clustering algorithms in terms of averaged accuracy over all thirteen data sets

## CONCLUSION

In this paper, we propose a new multi-objective model for clustering problem that uses the concept of cluster ensemble. The proposed model considers sum of squared error and modularity as two objectives. To benefit from modularity, the well-known measure for community detection in graph clustering, we have transformed our cluster ensemble problem to a network community detection problem. We have also used a modified NSGAII algorithm to solve the presented model. A new crossover operator specialized for the clustering problem is also suggested in this paper.

In our opinion, our new model and the customized genetic algorithm can efficiently solve the resulting clustering problem and the resulting clusters are much more reliable than those obtained by other single objective algorithms. The high performance of the method in comparison with a set of single criterion clustering algorithms over multiple standard data sets confirms our idea.

There is still plenty of room to improve our proposed model. One possibility is to develop a new multi-objective model using a set of heterogeneous cluster evaluation criteria that seems to have some promises in clustering unknown data sets.

## ACKNOWLEDGEMENT

This paper is extracted from research projects for Islamic Azad University, Ramsar branch, Ramsar, Iran.

## REFERENCES

**Alizadeh, H. 2013.** Cluster ensemble selection based on mathematical and social optimization, PhD thesis, Iran University of Science and Technology, Iran.

**Alizadeh, H., Minaei-Bidgoli, B. & Parvin, H. 2013.** Optimizing fuzzy cluster ensemble in string representation. International Journal of Pattern Recognition and Artificial Intelligence, **27**(02):1350005.

**Celebi, M.E. 2015.** Partitional clustering algorithms. Springer.

**Celebi, M.E., Kingravi, H.A. & Vela, P.A. 2013.** A comparative study of efficient initialization methods for the k-means clustering algorithm. Expert Systems with Applications, **40**(1):200-210.

**Deb, K., Pratap, A., Agarwal, S. & Meyarivan, T.A.M.T. 2002.** A fast and elitist multi objective genetic algorithm: NSGA-II. Evolutionary Computation, IEEE Transactions on, **6**(2):182-197.

**Fred, A.L. & Jain, A.K. 2005.** Combining multiple clusterings using evidence accumulation. Pattern Analysis and Machine Intelligence, IEEE Transactions on, **27**(6):835-850.

**Fred, A.L. & Jain, A.K. 2006.** Learning pairwise similarity for data clustering. 18th International Conference on pattern recognition, ICPR, 1: 925-928.

**Handl, J. & Knowles, J. 2007**. An evolutionary approach to multi objective clustering. Evolutionary Computation, IEEE Transactions on, **11**(1):56-76.

**Law, M.H., Topchy, A.P. & Jain, A.K. 2004.** Multi objective data clustering. In Proceedings of the Computer Vision and Pattern Recognition, CVPR, 2: II-424.

**Newman, M.E. 2012.** Communities, modules and large-scale structure in networks. Nature Physics, **8**, 25-31.

**Newman, M.E. & Girvan, M. 2004.** Finding and evaluating community structure in networks. Physical review E, **69**(2):026113.

**Ng, A.Y., Jordan, M.I. & Weiss, Y. 2002.** On spectral clustering: Analysis and an algorithm. Advances in neural information processing systems, **2**, 849-856.

**Strehl, A. & Ghosh, J. 2003.** Cluster ensembles- a knowledge reuse framework for combining multiple partitions. The Journal of Machine Learning Research, **3**, 583-617.

**Topchy, A., Jain, A. & Punch, W., 2004.** A mixture model for clustering ensembles, SIAM international conference on data mining, pp. 379– 390.

**Zheng, Y., Jia, L. & Cao, H. 2012.** Multi-objective gene expression programming for clustering. Information Technology and Control, **41**(3):283-294.

***Submitted:*** 29/1/2015

***Revised:*** 8/10/2015

***Accepted:*** 13/10/2015