Density-Sensitive Evolutionary Clustering

Maoguo Gong, Licheng Jiao, Ling Wang, and Liefeng Bo

Institute of Intelligent Information Processing, Xidian University,
Xi'an 710071, China
Maoguo_Gong@hotmail.com

Abstract. In this study, we propose a novel evolutionary algorithm-based clustering method, named density-sensitive evolutionary clustering (DSEC). In DSEC, each individual is a sequence of real integer numbers representing the cluster representatives, and each data item is assigned to a cluster representative according to a novel density-sensitive dissimilarity measure which can measure the geodesic distance along the manifold. DSEC searches the optimal cluster representatives from a combinatorial optimization viewpoint using evolutionary algorithm. The experimental results on seven artificial data sets with different manifold structure show that the novel density-sensitive evolutionary clustering algorithm has the ability to identify complex non-convex clusters compared with the K-Means algorithm, a genetic algorithm-based clustering, and a modified K-Means algorithm with the density-sensitive distance metric.

1 Introduction

Many clustering approaches, such as the K-Means Algorithm[1], partition the data set into a specified number of clusters by minimizing certain criteria. Therefore, they can be treated as an optimization problem. As global optimization techniques, Evolutionary algorithms (EAs) have been used for clustering tasks commonly in literature.[2][3][4] The solution representation and dissimilarity measure are the main difficulties in designing EA for clustering. Many researchers have used a representation approach that borrows from the K-Means algorithm: the representation codes for cluster center only, and each data item is subsequently assigned to a cluster representative according to an appointed dissimilarity measure.[5] The most popular dissimilarity measure is the Euclidean distance. By using Euclidean distance as a measure of dissimilarity, these evolutionary clustering methods as well as the K-Means algorithm have a good performance on the data set with compact super-sphere distributions, but tends to fail in the data set organized in more complex and unknown shapes, which indicates that this dissimilarity measure is undesirable when clusters have random distributions. As a result, it is necessary to design a more flexible dissimilarity measure for clustering. Su and Chou [6] proposed a nonmetric measure based on the concept of point symmetry, according to which a symmetry-based version of the K-Means algorithm is given. This algorithm assigns data points to a cluster center if they present a symmetrical structure with respect to the cluster center. Therefore, it is suitable to clustering data sets with clear symmetrical structure. Charalampidis [7] recently developed a dissimilarity measure for directional patterns

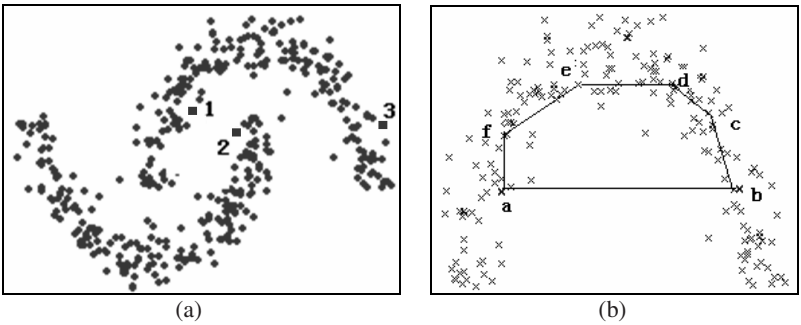
Z.-H. Zhou, H. Li, and Q. Yang (Eds.): PAKDD 2007, LNAI 4426, pp. 507–514, 2007.

represented by rotation-variant vectors and further introduced a circular K-Means algorithm to cluster vectors containing directional information.

In this study, we propose a novel evolutionary algorithm-based clustering technique, named density-sensitive evolutionary clustering (DSEC), by using a novel representation method and a density-sensitive dissimilarity measure. In DSEC, each string is a sequence of the cluster representatives selected from all the data items. The density-sensitive dissimilarity measure can describe the distribution characteristic of data clustering. The experimental results on seven artificial data sets show that the novel density-sensitive evolutionary clustering algorithm is very suitable to identify complex non-convex clusters compared with the K-Means algorithm [1], a genetic algorithm-based clustering [3], and a modified K-Means algorithm with the density-sensitive distance metric [8].

2 A Novel Density-Sensitive Dissimilarity Measure

For real world problems, the distribution of data points takes on a complex manifold structure, which results in the classical Euclidian distance metric can only reflect the local consistency which refers that data points close in location will have a high affinity, but fail to describe the global consistency which refers that data points locating in the same manifold structure will have a high affinity. We can illustrate this problem by the following example. As shown in Fig. 1(a), we expect that the affinity between point 1 and point 3 are higher than that of point 1 and point 2. In other words, point 1 is much closer to point 3 than to point 2 according to some distance metric. In terms of Euclidian distance metric, however, point 1 is much closer to point 2, thus without reflecting the global consistency. Hence for complicated real world problems, simply using Euclidean distance metric as a dissimilarity measure can not fully reflect the characters of data clustering.



(a) (b)

Fig. 1. (a) An illustration of that the Euclidian distance metric can not reflect the global consistency; (b) An illustration of that the global consistency of clustering does not always satisfy the triangle inequality under the Euclidean metric

Here, we want to design a novel dissimilarity measure with the ability of reflecting both the local and global consistency. As an example, we can observe from the data distribution in Fig. 1(a) that data points in the same cluster tend to lie in a region of

high density, and there exists a region of low density where there are a few data points. We can design a data-dependent dissimilarity measure in terms of that character of local data density.

At first, data points are taken as the nodes V of a weighted undirected graph $G=(V,E)$. Edges $E=\{W_{ij}\}$ reflect the affinity between each pair of data points. We expect to design a dissimilarity measure that ascribes high affinity to two points if they can be linked by a path running along a region of high density, and a low affinity if they cannot. This concept of dissimilarity measure has been shown in experiments to lead to significant improvement in classification accuracy when applied to semi-supervised learning [9][10]. We can illustrate this concept in Fig 1(a), that is, we are looking for a measure of dissimilarity according to which point 1 is closer to point 3 than to point 1. The aim of using this kind of measure is to elongate the paths that cross low density regions, and simultaneously shorten those that not cross.

To formalize this intuitive notion of dissimilarity, we need first define a so-called density adjusted length of line segment. We have found a property that a distance measure describing the global consistency of clustering does not always satisfy the triangle inequality under the Euclidean metric. In other words, a direct connected path between two points is not always the shortest one. As shown in Fig 1(b), to describe the global consistency, it is required that the length of the path connected by shorter edges is smaller than that of the direct connected path, i.e. $\overline{af}+\overline{fe}+\overline{ed}+\overline{dc}+\overline{cb}<\overline{ab}$. Enlightened by this property, we define a density adjusted length of line segment as follows.

Definition 1. The density adjusted length of line segment (x_i,x_j) is defined as

$$L(x_i,x_j)=\rho^{dist(x_i,x_j)}-1 \tag{1}$$

where $dist\left(x_i,x_j\right)$ is the Euclidean distance between x_i and x_j, $\rho>1$ is the flexing factor.

Obviously, this formulation possesses the property mentioned above, thus can be utilized to describe the global consistency. In addition, the length of line segment between two points can be elongated or shortened by adjusting the flexing factor ρ.

According to the density adjusted length of line segment, we can further introduce a new distance metric, called density-sensitive distance metric, which measures the distance between a pair of points by searching for the shortest path in the graph.

Definition 2. Let data points be the nodes of graph $G=(V,E)$, and $p\in V^l$ be a path of length $l=\left|p\right|-1$ connecting the nodes p_1 and $p_{|p|}$, in which $(p_k,p_{k+1})\in E$, $1\le k<\left|p\right|$. Let $\boldsymbol{P}_{i,j}$ denote the set of all paths connecting nodes x_i and x_j. The density-sensitive distance metric between x_i and x_j is defined as

$$D(x_i,x_j)=\min_{p\in\boldsymbol{P}_{i,j}}\sum_{k=1}^{|p|-1}L(p_k,p_{k+1}) \tag{2}$$

Thus $D(x_i, x_j)$ satisfies the four conditions for a metric, i.e. $D(x_i, x_j) = D(x_j, x_i)$; $D(x_i, x_j) \geq 0$; $D(x_i, x_j) \leq D(x_i, x_k) + D(x_k, x_j)$ for all x_i, x_j, x_k; and $D(x_i, x_j) = 0$ if and only if $x_i = x_j$.

As a result, the density-sensitive distance metric can measure the geodesic distance along the manifold, which results in any two points in the same region of high density being connected by a lot of shorter edges while any two points in different regions of high density are connected by a longer edge through a region of low density, thus achieving the aim of elongating the distance among data points in different regions of high density and simultaneously shortening that in the same region of high density. Hence, this distance metric is data-dependent, and can reflect the data character of local density, namely, what is called density-sensitive.

3 Evolutionary Clustering Based on the Density-Sensitive Dissimilarity Measure

3.1 Representation and Operators

In this study, we consider the clustering problem from a combinatorial optimization viewpoint. Each individual is a sequence of real integer numbers representing the sequence number of K cluster representatives. The length of a chromosome is K words, where the first gene represents the first cluster, the second gene represents the second cluster, and so on. As an illustration, let us consider the following example.

Example 1. Let the size of the clustered data set be 100 and the number of clustering being considered be 5. Then the individual (6, 19, 91, 38, 64) represents that the 6-th, 19-th, 91-th, 38-th, and 64-th points are selected to represent the five clusters, respectively.

So this representation method does not mention the data dimension. If the size of the data set is N and the number of clustering is K, then the search space is N^K.

Crossover is a probabilistic process that exchanges information between two parent individuals for generating offspring. In this study, we choose the uniform crossover [11] because it is unbiased with respect to the ordering of genes and can generate any combination of alleles from the two parents.[12][5] An example of the operation of uniform crossover on the encoding employed is shown in example 2.

Example 2. Let the two parent individuals be (6, 19, 91, 38, 64) and (3, 29, 17, 61, 6), random generate the mask (1, 0, 0, 1, 0), then the two offspring after crossover are (6, 29, 17, 38, **64**) and (3, 19, 91, 61, 64). In this case, the first offspring is not (6, 29, 17, 38, **6**) because the 6 in bold is repeat, we keep it unchanged.

Each individual undergoes mutation with probability p_m as example 3.

Example 3. Let the size of the clustered data set be 100 and the number of clustering being considered be 5. Then the individual (6, 19, 91, 38, 64) can mutate to (6, 19+*floor*((100-19)**random*+1), 91, 38, 64) or (6, 19-floor((19-1)**random*+1), 91, 38, 64) equiprobably, where the second gene is selected to mutate, *random* denotes a uniformly distributed random number in the range [0,1), and *floor* denotes rounding towards minus infinity.

3.2 Objective Function

Each point is assigned to the cluster whose density-sensitive distance of its representative to the point is minimum. As an illustration, let us consider the following example.

Example 4. Let the 6-th, 19-th, 91-th, 38-th, and 64-th points represent the five clusters, respectively. For the first point, we compute the density-sensitive distance between it and the 6-th, 19-th, 91-th, 38-th, and 64-th points, respectively. If the density-sensitive distance between the first point and the 38-th point is the minimum one, then the first point is assigned to the cluster represented by the 38-th point. All the points are assigned in the same way.

Subsequently, the objective function is computed as follows:

$$Dev(C) = \sum_{C_k \in C} \sum_{i \in C_k} D(i, \mu_k) \tag{3}$$

where C is the set of all clusters, μ_k is the representative of cluster C_k, and $D(i, \mu_k)$ is the density-sensitive distance between the *i*-th data item of cluster C_k and μ_k.

3.3 Density-Sensitive Evolutionary Clustering Algorithm

The processes of fitness computation, roulette wheel selection with elitism [13], crossover, and mutation are executed for a maximum number of generations $G_{\max}$. The best individual in the last generation provides the solution to the clustering problem.

Algorithm 1. Density-Sensitive Evolutionary Clustering (DSEC)

```
Begin
1.  t=0
2.  random initialize population P(t)
3.  assign all points to clusters according to the density-
    sensitive dissimilarity measure and compute the objective
    function values of P(t)
4.  t=t+1
5.  if t< Gmax
6.     select P(t) from P(t-1)
7.     crossover P(t)
8.     mutate P(t)
9.     go to step 3
10. end if
11. output best and stop
end
```

Fig. 2. Density-Sensitive Evolutionary Clustering

The initial population in step 2 is initialized to *K* randomly generated real integer number in [1, *N*], where *N* is the size of the data set. This process is repeated for each of the *P* chromosomes in the population, where *P* is the size of the population.

4 Experimental Results

In order to validate the clustering performance of DSEC, here we give the experimental results on seven artificial data sets, named Line-blobs, Long1, Size5, Spiral, Square4, Sticks, and Three-circles, with different manifold structure. The distribution of data points in these data sets can be seen in Fig. 3. The results will be compared with the K-Means algorithm (KM)[1], a modified K-Means algorithm using the density-sensitive dissimilarity measure (DSKM)[8], and the genetic algorithm-based clustering technique (GAC) [3]. In all the algorithms, the desired clusters number is set to be known in advance. The parameter settings used for DSEC and GAC in our experimental study are given in Table 1. For DSKM and KM, the maximum iterative number is set to 500, and the stop threshold 1e-10.

Table 1. Parameter settings for DSEC and GAC

Parameter	DSEC	GAC
Maximum Number of generations	100	100
population size	50	50
Crossover probability	0.8	0.8
Mutation probability	0.1	0.1

Clustering quality is evaluated using two external measures, the Adjusted Rand Index [5] and the Clustering Error [8]. The adjusted rand Index returns values in the interval [0, 1] and is to be maximized. The clustering error also returns values in the interval [0, 1] and is to be minimized.

We perform 30 independent runs on each problem. The average results of the two metrics, clustering error and adjusted rand index, are shown in Table 2.

Table 2. Results of DSEC, GAC, DSKM and KM where the results in bold are the best ones

Problem	Clustering Error				Adjusted Rand Index			
	DSEC	GAC	DSKM	KM	DSEC	GAC	DSKM	KM
line-blobs	**0**	0.263	0.132	0.256	**1**	0.399	0.866	0.409
Long1	**0**	0.445	**0**	0.486	**1**	0.011	**1**	0.012
Size5	**0.010**	0.023	0.015	0.024	**0.970**	0.924	0.955	0.920
Spiral	**0**	0.406	**0**	0.408	**1**	0.034	**1**	0.033
Square4	0.065	**0.062**	0.073	0.073	0.835	**0.937**	0.816	0.816
Sticks	**0**	0.277	**0**	0.279	**1**	0.440	**1**	0.504
Three-circles	**0**	0.569	0.055	0.545	**1**	0.033	0.921	0.044

From Table 2, we can see clearly that DSEC did best on six out of the seven problems, while GAC did best only on the Square4 data set. DSKM also obtained the true clustering on three problems. KM and GAC only obtained desired clustering for the two spheroid data sets, i.e. Size5 and Square4. This is due to that the structure of the other five data sets does not satisfy convex distribution. On the other hand, DSEC and DSKM can successfully recognize these complex clusters, which indicate the density-sensitive distance metric are very suitable to measure complicated clustering structure. When comparisons are made between DSEC and DSKM, the two

algorithms can obtained the true clustering on the Long1, Spiral, Sticks in all the 30 runs, but DSKM can not do it on the Line-blobs and Three-circles. Furthermore, for the Size5 and Square4 problems, DSEC did a little better than DSKM in both the clustering error and the adjusted rand index. The main drawback of DSKM is that it has to recalculate the geometrical center of each cluster as the K-Means algorithm after cluster assignment which reducing the ability of reflecting the global consistency. DSEC made up this drawback by evolutionary searching the cluster representatives from a combinatorial optimization viewpoint. In order to show the performance visually, the typical simulation results on the eight data sets obtained from DSEC are shown in Fig. 3.

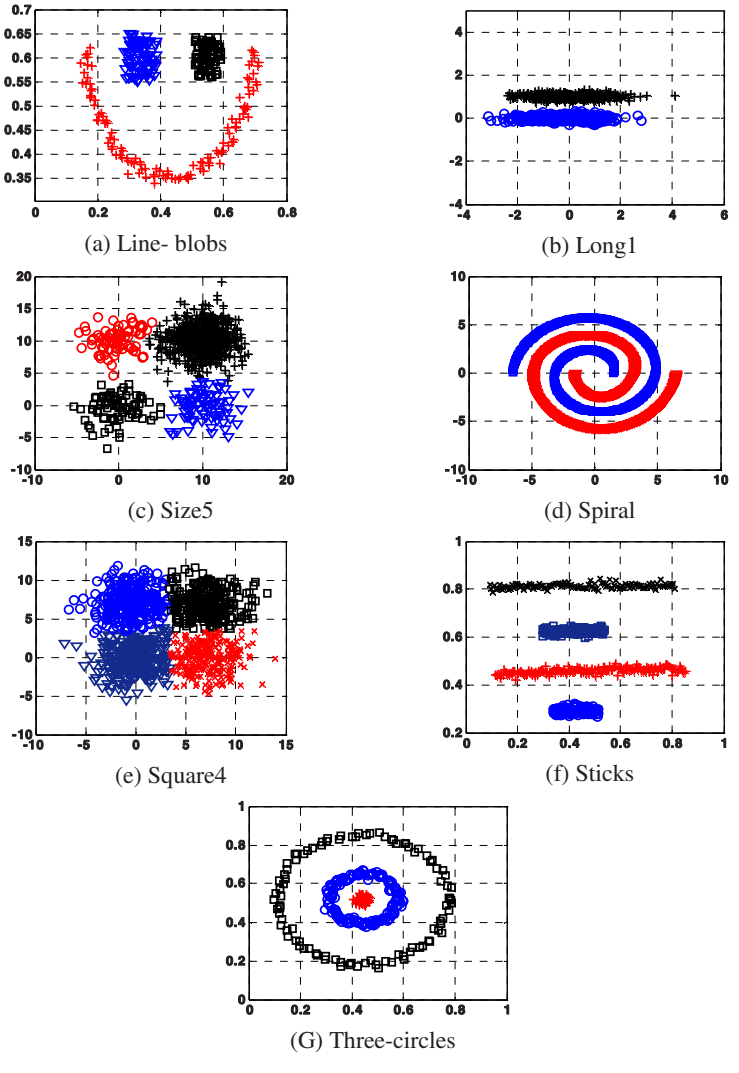


Fig. 3. The typical results on the artificial data sets obtained from DSEC

5 Concluding Remarks

In this paper, we proposed the density-sensitive evolutionary clustering by using a novel representation method and a density-sensitive dissimilarity measure. The experimental results on seven artificial data sets showed that in terms of cluster quality, DSEC outperformed GAC, DSKM and KM in partitioning most of the test problems.

The density-sensitive evolutionary clustering algorithm is a trade-off of flexibility in clustering data with computational complexity. The main computational cost for the flexibility in detecting clusters lies in searching for the shortest path between each pair of data points which makes it slower than KM and GAC.

Acknowledgements. This work was supported by the National High Technology Research and Development Program (863 Program) of China (No. 2006AA01Z107), the National Basic Research Program (973 Program) of China (No. 2006CB705700) and the Graduate Innovation Fund of Xidian University (No. 05004).

References

1. Hartigan, J.A., Wong, M.A.: A K-Means clustering algorithm. Applied Statistics, 28 (1979) 100-108
2. Hall, L.O., Ozyurt, I.B., Bezdek, J.C.: Clustering with a genetically optimized approach. IEEE Transactions on Evolutionary Computation, Vol. 3, No. 2 (1999) 103-112
3. Maulik, U., Bandyopadhyay, S.: Genetic algorithm-based clustering technique. Pattern Recognition, Vol. 33, No. 9 (2000) 1455-1465
4. Pan, H., Zhu, J., Han, D.: Genetic algorithms applied to multiclass clustering for gene expression data. Genomics, Proteomics & Bioinformatics, Vol. 1, No. 4 (2003) 279-287
5. Handl, J., Knowles, J.: An evolutionary approach to multiobjective clustering. IEEE Transactions on Evolutionary Computation, Vol. 11 (2007)
6. Su, M.C., Chou, C.H.: A modified version of the K-Means algorithm with a distance based on cluster symmetry. IEEE Transactions on PAMI, Vol. 23, No. 6 (2001) 674-680
7. Charalampidis, D.: A Modified K-Means Algorithm for Circular Invariant Clustering. IEEE Transactions on PAMI, Vol. 27, No. 12 (2005) 1856-1865
8. Wang, L., Bo, L.F., Jiao, L.C.: A modified K-Means clustering with a density-sensitive distance metric. RSKT 2006, Lecture Notes in Computer Science, Vol. 4062. Springer-Verlag, Berlin Heidelberg New York (2006) 544-551
9. Bousquet, O., Chapelle,O., Hein, M.: Measure based regularization. Advances in Neural Information Processing Systems 16 (NIPS), MIT Press, Cambridge, MA (2004)
10. Blum, A., Chawla, S.: Learning from labeled and unlabeled data using graph mincuts. Proceedings of the Eighteenth International Conference on Machine Learning (ICML) 18, (2001) 19-26
11. Syswerda, G.: Uniform crossover in genetic algorithms. In: Proceedings of the Third International Conference on Genetic Algorithms. Morgan Kaufmann Publishers, San Francisco, CA (1989) 2-9
12. Whitley, D.: A genetic algorithm tutorial. Statistics and Computing, Vol. 4 (1994) 65-85
13. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Reading, Massachusetts: Addison-Wesley (1989)