International Journal of Soft Computing 2 (5): 624-627, 2007
ISSN: 1816-9503

A Co-Evolutionary K-means Algorithm

[1]Sung-Hae Jun and [2]Im-Geol Oh
[1]Department of Bioinformatics and Statistics,
Cheongju University, Chungbuk, Korea
[2]Department of Internet Engineering, Hanseo University, Chungnam, Korea

Abstract: Clustering is an important tool for data mining. Its aim is to assign the points into groups that are homogeneous within a group and heterogeneous between groups. Many works of clustering methods have been researched in diverse machine learning fields. An efficient algorithm of clustering is K-means algorithm. This is a partitioning method. Also K-means algorithm has offered good clustering results. As well other clustering methods, K-means algorithm has some problems. One of them is optimal selection of the number of clusters. In K-means algorithm, the number of cluster K is determined by the art of researchers. In this study, we propose a co-evolutionary K-means(CoE K-means) algorithm for overcoming the problem of K-means algorithm. Our CoE K-means algorithm combines co-evolutionary computing into K-means algorithm. In our experimental results, we verify improved performances of CoE K-means algorithm using simulation data.

Key words: K-means clustering, the number of clusters, co-evolutionary computing

INTRODUCTION

K-means clustering algorithm is a good clustering algorithm based on portioning method (Everitt *et al*., 2001; Hastie *et al*., 2001). This has been applied to solve diverse clustering problems (Bock, 1985; Everitt, 1979; Everitt *et al*., 2001; Han and Kamber, 2001; Wang *et al*., 2005). The basic concept of K-means algorithm is to assign the points to the cluster with the smallest distance by non-hierarchical clustering. K-means algorithm is one of the works which have been researched in unsupervised machine learning. Generally in clustering fields, there are some problems. Optimal determination of the number of clusters is a problem of the problems of clustering a-lgorithms (Bock, 1985; Everitt, 1979; Everitt *et al*., 2001). But, it is difficult to determine the optimal number of clusters. Also, in K-means clustering, the number of cluster K is determined by the art of researchers. So, we propose a co-evolutionary K-means al-gorithm (CoE K-means algorithm) which combines competitive co-evolving into K-means clustering algorithm. Using competitive co-evolutionary computing, we overcome the problem about the selection of the number of clusters. That is, the number of clusters is able to be efficiently determined in our CoE K-means algorithm. We verify improved performances of a CoE K-means algorithm by experimental results of simulation data.

Co-evolutionary computing: In this study, using co-evolutionary computing, we overcome the problem of K-means clustering algorithm which is the selection of the number of clusters. Co-evolving is based on the competitive or cooperative evolutions between the species. A consequence of co-evolution comes from another population. The population influences the fitness of the main population. The main population also affects the fitness of the other one by turns (Eiben and Smith, 2003). So, in the case without co-evolution, the fitness landscape is fixed. Also the same individual always gives the same fitness. But, in the case with co-evolution, the fitness landscape is not fixed. Moreover, the fitness of an individual depends on other individuals. Therefore, the same individual may not have the same fitness in different populations. That is, co-evolution is able to be regarded as a kind of landscape coupling where adaptive moves by one individual will deform landscape of others. For our CoE K-means algorithm, we use competitive co-evolutionary computing with host and parasites by Hill (1990).

K-means clustering: K-mean algorithm is a popular non-hierarchical clustering method (Everitt *et al*., 2001; Hastie *et al*., 2001). This is to find clusters and their centers in unsupervised learning works. We determine the number of clusters K subjectively. K-means clustering is performed with repeat by moving the centers to minimize the total variance within clusters. Initially K is given then following two steps are performed iteratively until convergence. Firstly the points are assigned to closer group by distance measure. Next new center of each

Corresponding Author: Sung-Hae Jun, Department of Bioinformatics and Statistics, Cheongju University, Chungbuk, Korea

group is computed by averaging its points. In the following, K-means algorithm is shown (Han and Kamber, 2001).

Step 1: (Initialization)
(1-1) determining the number of clusters, K points
(1-2) initial centers of K clusters shown
Step 2: (Calculation of New Centers)
(2-1) calculating the distance between each point and a center of cluster
(2-2) the points are assigned to the cluster which has minimum distance
(2-3) new centers of the clusters are calculated by new points of the clusters
Step 3: (Repeating step2 until convergence)

Though K-means clustering algorithm is good clustering algorithm, the number of clusters is optimally determined for expecting good clustering results of K-means clustering algorithm. So, in this study, we propose a CoE K-means algorithm to settle the problem of traditional K-means clustering algorithm.

CO-EVOLUTIONARY K-MEANS ALGORITHM

K-means clustering algorithm is good tool for unsupervised learning. That is, this is an efficient clustering method (Ham and Kamber, 2001; Bezdek *et al*., 1994). In the clustering, the number of clusters has been significantly considered for good clustering results. But there are not completely satisfactory method for determining the number of clusters for any type of clustering (Bock, 1985; Everitt, 1979; Hartigan, 1985). The number of clusters has been subjectively determined by the art of researchers. However, this approach was not only an inefficient approach but also an annoying problem in clustering (Han and Kamber, 2001; Mitchell, 1997, 1998). So, the objective criteria have been needed for determining the number of clusters. The goal of our research is to solve the problems in K-means clustering algorithm. In this study, we propose CoE K-means algorithm which combines competitive co-evolving into K-means clustering algorithm. It is a method for determining the optimal number of clusters and clustering with good accuracy. Clustering process of K-means algorithm depends on initial number of clusters. Also, the number of clusters plays on important role in the clustering results of K-means algorithm. Though, the researches for determining it have been proposed (Bock, 1985; Everitt, 1979; Everitt *et al*., 2001; Han and Kamber, 2001; Wang *et al*., 2005). most determining processes are performed by the art of researchers (Everitt *et al*., 2001; Han and Kamber, 2001; Hastie *et al*., 2001). So, we work about the solution of optimal determination of the number. CoE K-means algorithm is constructed by combining competitive co-evolution into K-means clustering algorithm. Using our algorithm, we are able to determine the optimal number of clusters and cluster given data set efficiently. To develop CoE K-means algorithm, we use evolutionary algorithm (Eiben and Smith, 2003). This is mainly based on Genetic Algorithm (GA). GA has provided a analytical method motivated by an analogy to biological evolution (Mitchell, 1997). Traditional GA computes the fitness of given environment where is fixed. Distinguished from traditional GA, co-evolving approach is evolutionary mechanism of the natural world with competition or cooperation. The organism and the environment including organism evolve together (Mitchell, 1997). We apply not cooperation but competition to our proposed model. Our competitive co-evolving approach uses host-parasites co-evolution. The host and parasites are used for modeling CoE K-means model and training data set. Our CoE K-means model and training data set are considered as the organism and the environment including it. That is, the evolving CoE K-means algorithm is followed the evolution of host. The initial parameters for CoE K-means algorithm are determined as uniform random numbers from -1 to 1. A good result of clustering has high intra-cluster similarity and low inter-cluster similarity (Jun, 2005).

Step 1 (competitive co-evolving): In this study, we introduce a criterion for evaluating the results of clustering. The criterion is composed of two parts which are the variance of points in clusters and the penalty of excessive increasing the number of clusters. Using this criterion, we define the fitness function of CoE K-means algorithm as following.
(I-1) Host evolution

$$F_{HOST}(M) = \frac{1}{M}\sum_{i=1}^{M}\bar{V}_i + \frac{1}{\bar{V}_M}M \quad (1)$$

In the above, M is the number of clusters and $\bar{V}_i$ is the average of variances of points in the ith cluster. $\bar{V}_M$ is the variance of M clusters. The smaller the f_{host} (M) value is, the better the clustering result is. The fitness function of parasites is defined by the inverse form of the fitness of host as the following.
(I-2) Parasites evolution

$$F_{PARASITES}(M) = \frac{K}{\frac{1}{M}\sum_{i=1}^{M}\bar{V}_i + \frac{1}{\bar{V}_M}M} \quad (2)$$

Where, K is a constant. We are able to control the competitive levels between host and parasites. Our evolutionary approach of CoE K-means algorithm and training data set are competitive. In other words,

```
BEGIN
1.  Initialize population (K) from U[-1, 1]
2.  Competitive Co-Evolving,
 1)   CoE K-means model by F_host(x);
 2)   Training data set by F_parasites (x);
REPEAT UNTIL
 (Termination condition is satisfied)
DO
1.  Select Parents;
2.  Mutate the resulting offspring;
3.  Evaluate new candidates'
4.  Select individuals for the next generation;
LOOP
Determining K
 The number of clusters
K-means clustering
 Using K-means clustering,
 Assign the point to the nearest group
END
```

Fig. 1: Pseudo-code of CoE K-means algorithm

proposed model is competitive co-evolving of two different groups. One is the parasites evolution of given training data set. Another is the host evolution of CoE K-means algorithm. In this step, we determine the parameters which are kernel parameter and regularization constant.

Step 2: (K-means Clustering).

(II-1) Assign the point into K groups

(II-2) Calculate the centers of K groups by averaging the points of each group

(II-3) Re-assign each point to the group with the nearest distance measure

(II-4) Go to (II-2), Stop until convergence (no more new assignment)

In our method, CoE K-means model and training data set are co-evolved, respectively. During evolution for weight optimization of CoE K-means model, the competitive co-evolving is occurred between evolving CoE K-means model and evolving training data set. In this place, our model uses co-evolutionary computation for determining the parameters for traditional K-means clustering algorithm for optimal clustering. The following is a pseudo-code of CoE K-means algorithm (Fig. 1).

Therefore, using CoE K-means algorithm, we are able to perform the optimal clustering.

EXPERIMENTAL RESULTS OF SYNTHETIC DATA

To verify improved performance of CoE K-means algorithm, we make experiments using data sets from simulation data (Martinez and Zartinez, 2002). For usage of synthetic data sets, we generate multivariate random data from Finite Mixture Density (FMD) (Evevitt *et al*., 2001). FMD is a probability density function as the following form (Evevitt *et al*., 2001).

$$f(x;p,\theta)=\sum_{i=1}^{c}p_i g_i(x;\theta_i) \tag{3}$$

Where, x, π and θ are random vector, mixing proportions and model parameters respectively. Also, density g is parameterized by θ. In FMD, each cluster comes from a population with a different probability distribution (Evevitt *et al*., 2001). So, we get random data sets from the following expression.

$$f(\text{cluster } i|x_i)=\frac{\hat{\pi}_i g_i(x_i,\hat{\theta}_i)}{f(x_i;\hat{\pi},\hat{\theta})} \tag{4}$$

In this experiment, we need to generate multivariate random vector x. Based on a d-dimensional vector of standard normal random numbers, the following transformation is performed (Martinez and Zartinez, 2002).

$$x_{(d\times 1)}=R^{T}_{(d\times d)}z_{(d\times 1)}+\mu_{(d\times 1)} \tag{5}$$

Where, z is the standard normal random vector and μ is a mean vector. $R^TR = \Sigma$ is a covariance matrix. Using different Σs, we get two synthesis data sets which are high and low correlated data. In the following illustration, Σ_{high}, Σ_{middle} and Σ_{low} are covariance matrices for high, middle and low correlated data between attributes. The data set with low correlation are independent.

$$\Sigma_{high}=\begin{pmatrix}1&&&&\\0.87&1&&&\\0.96&0.78&1&&\\0.68&0.84&0.86&1&\\0.92&0.77&0.93&0.88&1\end{pmatrix},$$

$$\Sigma_{middle}=\begin{pmatrix}1&&&&\\0.24&1&&&\\0.35&0.19&1&&\\0.28&0.35&0.40&1&\\0.25&0.18&0.22&0.43&1\end{pmatrix}, \tag{6}$$

$$\Sigma_{low}=\begin{pmatrix}1&&&&\\0.05&1&&&\\0.03&0.06&1&&\\0.15&0.19&0.09&1&\\0.09&0.11&0.08&0.08&1\end{pmatrix}$$

Table 1: Accuracy results

	High Corr.		Middle Corr.		Low Corr.	
Algorithms	Training	Validation	Training	Validation	Training	Validation
CoE K-means	94.5	93.6	95.8	95.1	96.1	96.0
SVC	93.4	92.1	93.1	91.8	94.4	92.8
SOM	91.3	90.7	92.4	91.0	93.6	92.5
K-NN	92.5	90.8	93.7	93.0	94.6	93.2
Hierarchical agg.	89.3	88.7	90.7	89.4	91.5	90.2
Hierarchical div.	89.6	88.9	91.4	90.0	92.3	90.9

In the above covariance matrices, the number of attributes is four, respectively. We generate data sets which have 1000 data points randomly. In this study, we are able to consider the performance of CoE K-means algorithm according to the correlation coefficient between attributes. In the experiment, we compare CoE K-means algorithm with established machine learning algorithms which are Support Vector Clustering (SVC), Self Organizing Map (SOM), K-Nearest Neighbor (K-NN) and hierarchical clustering (Ben-Hur *et al*., 2001; Cherkassky and Mulier, 1998; Everitt *et al*., 2001; Haykins, 1999). In this section, we show the accuracy results by the correctly classified points in each learning algorithm. For the experiment, given data are divided into training and validation data sets. We use one-third of the given data for the validation set and other two-thirds for the training (Mitchell, 1997).

In this experiment, the kernel function and regularization constant of SVC are Gaussian kernel function and 1, respectively. The regularization constant of SVC is 1 means not to consider the influence of regularization constant. Also, the dimension of feature map in SOM is 5 normally. Hierarchical agg. and div. are the agglomerative and divisive methods in hierarchical clustering. From above Table 1, we find the accuracy rate of the points of CoE K-mean algorithm is better than other comparative methods. Also, the difference between training and validation about the accuracy of CoE K-means algorithm is the smallest in the models. So, we are able to verify improved performance of CoE K-means algorithm.

CONCLUSION

In this study, we propose a CoE K-means algorithm for optimal clustering. Our algorithm combines competitive co-evolving into K-means Clustering algorithm. In our CoE K-means algorithm, we are able to determine the number of clusters objectively. Also, we got improved performance of K-means clustering algorithm by CoE K-means algorithm. In future works, we will apply competitive co-evolution to other portioning methods which are K-medoids clustering algorithm, CLARANS (clustering large application) (Han and Kamber, 2001) and so forth.

REFERENCES

Ben-Hur, A., D. Horn, H.T. Siegelmann and V. Vapnik, 2001. Support Vector Clustering. J. Machine Learning Res., 2: 125-137.

Bezdek, J.C., S. Boggavarapu, L.O. Hall and A. Bensaid, 1994. Genetic algorithm guided clustering. IEEE. World Congress on Computational Intelligence, 1: 34-39.

Bock, H.H., 1985. On Some Significance Tests in Cluster Analysis. J. Classification, 2: 77-108.

Cherkassky, V. and F. Mulier, 1998. Learning from Data-Concepts. Theory and Methods, John Wiley and Sons, Inc.

Eiben, A.E. and J.E. Smith, 2003. Introduction to Evolutionary Computing, Springer.

Everitt, B.S., 1979. Unresolved Problems in Cluster Analysis. Biometrics, 35: 169-181.

Everitt, B.S., S. Landau and M. Leese, 2001. Cluster Analysis, Arnold.

Han, J. and M. Kamber, 2001. Data Mining Concepts and Techniques, Morgan Kaufmann.

Hartigan, J.A., 1985. Statistical Theory in Clustering. J. Classification, 2: 63-76.

Hastie, T., R. Tibshirani and J. Friedman, 2001. The Elements of Statistical Learning Data Mining, Inference and Prediction, Springer.

Haykin, S., 1999. Neural Networks, Prentice Hall.

Hillis, W.D., 1990. Co-evolving parasites improve simulated evolution as an optimization procedure. Physica D, 42: 228-234.

Jun, S.H., 2005. Web Usage Mining Using Support Vector Machine. Lecture Note in Computer Sci., 3512: 349-356.

Martinez, W.L. and A.R. Zartinez, 2002. Computational Statistics Handbook with MATRAB. Chapman and Hall.

Mitchell, T.M., 1997. Machine Learning and McGraw-Hill.

Mitchell, T.M., 1998. An introduction to Genetic Algorithms, MIT Press.

Vapnik, V.Z., 1998. Statistical Learning Theory, John Wiley and Sons, Inc.

Wang, J., X. Wu and C. Zhang, 2005. Support vector machine based on K-means clustering for real-time business intelligence systems. Int. J. Business Intelligence and Data Mining, 1: 54-64.