Hybrid Self Organizing Map using Monte Carlo Computing

[1]Sung-Hae Jun, [2]Minjae Park, [3]Kyung-Whan Oh

[1]Department of Statistics, Cheongju University, Chungbuk, Korea
shjun@cju.ac.kr
[2]Pentech, Seoul, Korea
pmi219@hotmail.com
[3]Department of Computer Science, Sogang University, Seoul, Korea
kwoh@sogang.ac.kr

ABSTRACT

Self Organizing Map(SOM) is a powerful neural network model for unsupervised learning. In many clustering works with exploratory data analysis, it has been popularly used. But it has a weakness which is the poorly theoretical base. A lot more researches for settling the problem have been published. Also, our paper proposes a method to overcome the drawback of SOM. As compared with the presented researches, our method has a different approach to solve the problem. So, a hybrid SOM is proposed in this paper. Using Monte Carlo computing, a hybrid SOM improves the performance of clustering. We verify the improved performance of a hybrid SOM according to the experimental results using UCI machine learning repository. In addition to, the number of clusters is determined by our hybrid SOM.

Key words : Hybrid Self Organizing Maps, MCMC, Optimal Computing

I. Introduction

Automatic determination of the number of population clusters is needed in the clustering like K-means algorithm, hierarchical clustering method, etc. Usually we have determined the number of clusters subjectively. In this paper, we proposed an method for automatic determination of the number of clusters using Bayesian Self Organizing Map(SOM) based fuzzy clustering. That is, the SOM, Bayesian learning, and fuzzy set logic were used in proposed algorithm for the determination of optimal number of clusters. The existing methods have had an uncertainty because they have determined the number of clusters with subjectivity. One of the gates to eliminate the uncertainty is through the fuzzy set theory[13]. If X is a collection of objects denoted generally by x, then a fuzzy set A in X consists of a set of x and its membership function. The membership function can be expressed by values from 0 to 1 as the degree of truth that maps X to A. However it is difficult to choose a suitable form for the membership function. Nowadays, it is common to determine the membership function subjectively. Then this may make the problems more ambiguous in the machine learning, which should resolve the uncertainty. In this paper, we proposed an objective method to select the membership function for determining the number of clusters by heuristic approach using Bayesian SOM. For illustration of our proposed method, we considered examples with comparing other clustering methods which are the SOM, K-means algorithm and statistical clustering methods using the data sets from UCI machine learning repository.

II. Related Works

The cluster is a set of adjacent objects in training data. Objects in the same cluster have close similarity and objects in other clusters have dissimilarity. We use distance as a measure of similarity between objects. The first problem to consider in clustering is to determine the number of clusters. K-means algorithm requires an initial number of clusters and hierarchical clustering method also requires an optimal number of clusters for stopping clustering process[4]. But it is hard to find any objective algorithm to determine the number of clusters. Most of them are determined subjectively. So, we propose a hybrid SOM using Monte Carlo computing for solving the problems.

Let X is a nonempty set and x is an element of X. A fuzzy set A is defined as follows[14].

$$A = \{(x, \mu_A(x)) \mid x \in X\} \qquad (1)$$

Where, $\mu_A(x)$ is a membership function which expresses a degree of inclusion of x into A. In this paper, fuzzy set is used to determine the number of clusters[2]. The membership functions of fuzzy set for clustering are computed repeatedly by our hybrid SOM in given training data. That is, X becomes a set of all possible numbers of clusters and A becomes a fuzzy set of appropriate cluster size. $\mu_A(x)$ is a membership function for each possible cluster size. So, we decide the element with the largest membership function in fuzzy set A to optimal number of clusters. The neural network model by Kohonen has two types which are SOM and LVQ(learning vector quantization)[7]. We used SOM in this paper because SOM is an efficient algorithm for clustering[5]. Though SOM requires a size of feature maps, it does not need the number of clusters. The SOM algorithm is expressed in the followings[5].

III. A Hybrid Self Organizing Maps

Monte Carlo methods are computational techniques that make use of random numbers. The aims of Monte Carlo methods are to solve one or both of the following problems. One is to generate samples $\{x^{(i)}\}_{i=1}^{n}$ from a given probability distribution *P(x)*. The other is to estimate expectations of functions under *P(x)*. Markov Chain Monte Carlo(MCMC) methods have been used for many years to solve problems in statistical physics, machine learning, bioinformatics, and so forth[6]. MCMC methods were especially introduced for computation in Bayesian statistics. The assumptions concerning the form of the distribution, such as normal approximation are not made in MCMC. Each node of the output layer achieves clustering by competitive learning from training data. Each object crisply belongs to only one exclusive cluster after the last training. And the clustering result is only one type because the weights have fixed values in nodes of SOM after final training. This result is usually not optimal[8, 9, 10,11] and it is impossible to repeat the different experiments to determine membership function of fuzzy set. In this paper, we'll get a fuzzy set with repeated experiments by using Bayesian inference[12, 13] that consists of prior probability distribution, posterior probability distribution, and likelihood distribution to SOM. The proposed Bayesian SOM updates parameters of probability distribution without having the fixed values of weights on each node of output layer. This strategy makes it possible to create the membership function by performing repeated experiments with same data to get different results. The proposed method doesn't always offer same results for the same training data because it uses a random number from the last updated distribution for clustering. The membership function of fuzzy set is determined by Bayesian learning[7] based SOM that computes a posterior by combining prior and likelihood. We summarized the proposed algorithm in this paper as following.

Step1: Initialize

(n: data size, p: the dimension of input vectors)

Normalization of input vectors

$x_i = (x_{i1}, \cdots, x_{ip})$ represents the *i*th input pattern

$$x_i^{normal} = \left(\frac{x_{i1}-\mu_1}{\sigma_1}, \cdots, \frac{x_{ip}-\mu_{1p}}{\sigma_p}\right) = \left(x_{i1}^{normal}, \ldots, x_{ip}^{normal}\right)$$

$x_i^{normal} \sim N(0,1)$, $(i = 1, \ldots n)$; likelihood

1.2 Initialize the weights vectors: Prior of weights

1.2.1 determine the distribution type of $f(\cdot)$

$f(\cdot)$ is any probability density function(pdf)

$w \sim f(\theta)$

optionally, $\theta \sim g(\varphi)$: φ is the hyper-parameter of θ, $g(\cdot)$ is also pdf

Step2: Determine winner node

(m: feature map dimension)

2.1 Weights sampling from current prior

2.2 Compute the $dist(x_i^{normal}, w_j)$

(Euclidean distance of x_i^{normal} and w_j)

$$dist(x_i^{normal}, w_j) = \sqrt{(x_{i1}^{normal} - w_{j1})^2 + \cdots + (x_{ip}^{normal} - w_{jp})^2}$$

$(i = 1, \ldots, n$, $j = 1, \ldots, m^2)$

2.3 Determine winner node

w_k is winner node if

$$dist(x, w_k) < dist(x, w_j) \text{ , } j = 1, \ldots, m^2$$

that is,

$$w_k = \arg\min_j \{dist(x, w_j)\}$$

Step3: Update distribution of weights

3.1 Compute posterior of winner node using Bayes' rule

3.2 Replace current posterior by new prior

Repeat phase2 and phase3 until given conditions are satisfied

Step4: Extract Fuzzy Set for the number of Clusters

4.1 Repeat experiments until given number

4.1 Determine the membership function of fuzzy set

IV. Experimental Results and Conclusion

For our experiments, we used Iris plants, Glass identification, and Abalone data in UCI machine learning repository[10]. We verified our model with other clustering methods. The CCVP measure is good when it is smaller. And the clustering is good when the s.d.(standard deviation) is smaller. Because the smaller s.d. of clustering is the more similar objects of data are. We used the k of K-means clustering and the stopping cluster size of hierarchical clustering by the number of labels of target variable.

Table 1. The evaluation of comparative models

Data set	Methods	# of clusters	CCVP mean	s.d.
Iris plants	SOM	5	0.017	0.146
	K-means	3	0.093	0.583
	Hierarchical	3	0.121	0.912
	Our model	3	0.002	0.058
Glass identification	SOM	11	0.184	0.364
	K-means	7	0.312	0.986
	Hierarchical	7	0.498	1.014
	Our model	6	0.105	0.215
Abalone	SOM	24	2.515	6.311
	K-means	29	4.319	11.358
	Hierarchical	29	5.914	12.984
	Our model	20	1.313	4.560

Above result showed the CCVP mean and CCVP s.d. of Bayesian learning SOM is the smallest among comparative clustering methods.

V. Reference

[1] Bishop, C. M., Svensen, M., Williams, C. K. I.: GTM: A Principled Alternative to the Self Organizing Map, ICANN 96, Volume1112, Bochum, Germany, pp. 165-170, (1996)

[2] Dumitrescu, D., Lazzerini, B., Jain, L. C.: Fuzzy Sets and Their Application to Clustering and Training, CRC Press, (2000)

[3] Gelman, A., Carlin, J. B., Stern, H. S., Rudin, D. B.: Bayesian Data Analysis, Chapman & Hill, (1995)

[4] Han, J., Kamber, M.: Data Mining Concepts and Techniques, Morgan Kaufmann, (2001)

[5] Kohonen, T.: Self Organizing Maps, Second Edition, Springer, (1997)

[6] Neal, R. M.: Bayesian Learning for Neural Networks, Springer, (1996)

[7] Pandya, A. S., Macy, R. B.: Pattern Recognition with Neural Networks in C++, IEEE Press, (1995)

[8] Park, M. J., Jun, S. H., Oh, K. W.: Determination of Optimal Cluster Size Using Bootstrap and Genetic Algorithm, Journal of Fuzzy Logic and Intelligent Systems, Vol. 13, No. 1, pp. 12-17, (2003)

[9] Tanner, M. A.: Tools for Statistical inference, Springer, (1996)

[10] UCI Machine Learning Repository, http://www1.ics.uci.edu/~mlearn

[11] Utsugi, A.: Topology selection for self-organizing maps, Network: Computation in Neural Systems, Vol. 7, No. 4, pp. 727-740, (1996)

[12] Utsugi, A.: Hyperparameter selection for self-organizing maps, Neural Computation, Vol. 9, No. 3, pp. 623-635, (1997)

[13] Zadeh, L.: Fuzzy Sets, Information and Control, (1965)

[14] Zimmermann, H. J.: Fuzzy Set Theory and its Applications, Third Edition, (1996)

[15] Mackay, D. J. C.: Information Theory, Inference, and Learning Algorithms, Cambridge University Press, (2003).