An Improved Algorithm for Clustering

***Tsan-Jung He[1], Zhao-Yu Wang[1], †Shie-Jue Lee[1], and Shing-Tai Pan[2]**

[1]Department of Electrical Engineering National Sun Yat-Sen University, Kaohsiung, Taiwan
[2]Department of Electrical Engineering National University of Kaohsiung, Kaohsiung, Taiwan

*Presenting author: zlhe@water.ee.nsysu.edu.tw
†Corresponding author: leesj@mail.ee.nsysu.edu.tw

Abstract

In this paper, we propose a new clustering algorithm which is an improvement to a self-constructing clustering (SCC) method. The SCC processes all the data points incrementally. If the input data point is similar enough to an existing cluster, the point is added to the cluster. Otherwise, the data point forms a new cluster of its own. The method ends up with a set of clusters after it runs through the whole dataset once. However, once a data point is assigned to a cluster, there is no way to change the assignment afterwards. This may cause assignment errors and the efficacy of the clustering is reduced. In this paper, we adopt an iterative approach to overcome this shortcoming. A data point can be re-assigned to another cluster. Adding points into and removing points from a cluster are allowed to be done iteratively in the clustering process. The clustering work stops when all the assignments are stable, i.e., no assignment would be changed. The proposed approach can result in better clusters, and experimental results show that it performs better than SCC for real world datasets.

Keywords: data mining, clustering, self-constructing clustering, similarity, classification.

I. Introduction

In the field of artificial intelligence, clustering techniques play a very important role [3][9]. Clustering is an unsupervised classification technology, with a purpose of forming meaningful clusters for the objects under consideration. Usually, similar objects are grouped in the same cluster, and different objects are grouped in different clusters. The clustering concept is widely applied in a variety of different areas, such as bio-engineering [6][14], environmental monitoring [8], economic applications [12], and so on.

In the electronic text applications [4], the dimensionality of the data can be reduced to improve the efficiency of the operation through the clustering technology. In the recommendation applications of e-commerce [7], clustering is used to reduce the size of the information matrix to enhance the efficiency of the operation. In the application of regression, the reduction of information dimension is used. In power system, clustering is used to predict the electrical trend in the future [1]. In other areas, such as stock market and data regression [11][13], the clustering technology is an important and indispensable core key. Therefore, developing a better clustering technology is a very critical issue.

Ouyang et al. [5][10] proposed a clustering method, self-constructing clustering (SCC), which has been applied in various applications. It considers all the data points one by one. For an input point, its similarity to each existing cluster is calculated. If the point is similar enough to an existing cluster, the point is added in the cluster. On the other hand, if the point is not similar enough to any of the existing clusters, the point forms a new cluster. The algorithm proceeds until all the points have been processed once. SCC offers several advantages. First, since the algorithm runs through the data points once, it is fast. Second, it considers the

variation of the data under consideration. Third, the number of clusters is not to be specified in advance.

However, SCC has one disadvantage. Once a data point is assigned to a cluster, there is no way to change the assignment afterwards. This may cause assignment errors and the efficacy of the clustering is reduced. In this paper, we adopt an iterative approach to overcome this shortcoming of SCC. A data point can be re-assigned to another cluster. Adding points into and removing points from a cluster are allowed to be done iteratively in the clustering process. The clustering work stops with a desired number of clusters when all the assignments are stable, i.e., no assignment would be changed. The proposed approach can result in better clusters, and experimental results show that it performs better than SCC for real world datasets.

The rest of this paper is organized as follows. SCC is briefly reviewed in Section II. Our proposed improvement presented in Section III. Experimental results are shown in Section IV. Section V gives a conclusion.

II. Related Work

SCC [5][10] is a progressive clustering method using the Gaussian function as the membership function of the resulting clusters. For each cluster, its center and distribution are described by the mean and standard deviation, respectively, of the contained data points.

Data points are considered one by one sequentially. When the first data point comes in, the first cluster is created for it. Then, for each of the rest data points, SCC calculates the similarity between the input data point and each existing cluster. If the input data point is similar enough to an existing cluster, the data point is added into this cluster. Otherwise, a new cluster is created for the input data point. Given the input data point x, the similarity to cluster G for x is calculated as follows:

$$\mu_G = \prod_{i=1}^{p} exp[-(\frac{x_i - m_i}{\sigma_i})^2] \tag{1}$$

where p is the number of dimensions of the input data, and m_i and σ_i are the center and standard deviation of the i th dimension of cluster G, defined respectively by

$$m_i = \frac{\sum_{j=1}^{|G|} y_{ji}}{|G|}, \tag{2}$$

$$\sigma_i = \sqrt{\frac{\sum_{j=1}^{|G|} (y_{ji} - m_{ji})^2}{|G| - 1}}. \tag{3}$$

Note that $|G|$ is the total number of data points contained in cluster G, y_j, $j=1,\ldots,|G|$, are the data points contained in cluster G, and y_{ji} represents the ith dimension of y_j. When the similarity between the input data point x and the existing clusters is greater than a default threshold, the input is added into the cluster with the largest similarity. Let the cluster be G_t and its size be S_t. The mean and deviation of cluster G_t are then updated.

After all the data points are considered, SCC stops and a set of clusters are obtained. The algorithm can be described below.

Table 1. SCC

Algorithm 1 SCC	
1:	**Input:** dataset D
2:	**for** each data point x **do**
3:	compute similarity to each existing cluster
4:	**if** some similarity bigger enough **then**
5:	add x to the cluster with largest similarity
6:	**else**
7:	form a new cluster
8:	**end if**
9:	**end for**
10:	**Output:** all the clusters obtained

SCC offers several advantages. First, since the algorithm runs through the data points once, it is fast. Second, it considers the variation of the data under consideration. Third, the number of clusters is not to be specified in advance.

III. Proposed Method

However, SCC has one disadvantage. Once a data point is assigned to a cluster, there is no way to change the assignment afterwards. This may cause assignment errors and the efficacy of the clustering is reduced. We adopt an iterative approach to overcome this shortcoming of SCC. A data point can be re-assigned to another cluster. Adding points into and removing points from a cluster are allowed to be done iteratively in the clustering process. The clustering work stops with a desired number of clusters when all the assignments are stable, i.e., no assignment would be changed.

For convenience, our proposed approach is called New-SCC. New-SCC consists of several rounds of iteration. In the first round, the algorithm of the original SCC is performed.
Then we perform the second round and beyond, each round considering all the data points sequentially. In each succeeding round, for an input data point x, we first remove it from the cluster G_t, which x belongs to. Then we calculate the similarity between x and each existing cluster by Eq. (1). If the max similarity occurs with cluster G_a and it is higher than a specified threshold, x is added into G_a. However, if the max similarity is not higher than the specified threshold, a new cluster is created for x. A round of iteration ends when all the data points are gone through once. If one of the assignments has been changed for the data points in the current round, the next round of iteration begins. Otherwise, the assignments are stable and

New-SCC stops with a desired number of clusters. Let's have an example here to illustrate how New-SCC works. Suppose we have the following 12 data points:

$$x_1 = \langle 0.30, 0.60 \rangle;$$
$$x_2 = \langle 0.70, 0.35 \rangle;$$
$$x_3 = \langle 0.50, 0.52 \rangle;$$
$$x_4 = \langle 0.78, 0.20 \rangle;$$
$$x_5 = \langle 0.62, 0.25 \rangle;$$
$$x_6 = \langle 0.40, 0.65 \rangle;$$
$$x_7 = \langle 0.35, 0.38 \rangle;$$
$$x_8 = \langle 0.28, 0.48 \rangle;$$
$$x_9 = \langle 0.19, 0.89 \rangle;$$
$$x_{10} = \langle 0.24, 0.81 \rangle;$$
$$x_{11} = \langle 0.29, 0.89 \rangle;$$
$$x_{12} = \langle 0.24, 0.89 \rangle.$$

After performing SCC in the first round, we have 6 clusters: G_1, G_2, G_3, G_4, G_5, and G_6, with

$$m_1 = \langle 0.35, 0.625 \rangle, \sigma_1 = \langle 0.2707, 0.2354 \rangle;$$
$$m_2 = \langle 0.66, 0.3 \rangle, \sigma_2 = \langle 0.2566, 0.2707 \rangle;$$
$$m_3 = \langle 0.5, 0.52 \rangle, \sigma_3 = \langle 0.2, 0.2 \rangle;$$
$$m_4 = \langle 0.78, 0.2 \rangle, \sigma_4 = \langle 0.2, 0.2 \rangle;$$
$$m_5 = \langle 0.315, 0.43 \rangle, \sigma_5 = \langle 0.2495, 0.2407 \rangle;$$
$$m_6 = \langle 0.24, 0.81 \rangle, \sigma_6 = \langle 0.2408, 0.24 \rangle.$$

Note that G_1 contains data points 1 and 6, G_2 contains data points 2 and 5, G_3 contains data point 3, G_4 contains data point 4, G_5 contains data points 7 and 8, and G_6 contains data points 9, 10, 11, and 12. After the second round, we have 4 clusters: G_1, G_2, G_3, and G_4, with

$$m_1 = \langle 0.37, 0.5625 \rangle, \sigma_1 = \langle 0.3013, 0.2768 \rangle;$$
$$m_2 = \langle 0.7, 0.2667 \rangle, \sigma_2 = \langle 0.28, 0.2764 \rangle;$$
$$m_3 = \langle 0.35, 0.38 \rangle, \sigma_3 = \langle 0.2, 0.2 \rangle;$$
$$m_4 = \langle 0.24, 0.87 \rangle, \sigma_4 = \langle 0.2408, 0.24 \rangle.$$

Note that G_1 contains data points 1 3, 6, and 8, G_2 contains data points 2, 4, and 5, G_3 contains data point 7, and G_4 contains data points 9, 10, 11, and 12. After the third round, we have 3 clusters: G_1, G_2, and G_3, with

$$m_1 = \langle 0.366, 0.526 \rangle, \sigma_1 = \langle 0.2882, 0.3053 \rangle;$$
$$m_2 = \langle 0.7, 0.2667 \rangle, \sigma_2 = \langle 0.28, 0.2764 \rangle;$$
$$m_3 = \langle 0.24, 0.87 \rangle, \sigma_3 = \langle 0.2408, 0.24 \rangle.$$

Note that G_1 contains data points 1 3, 6, 7, and 8, G_2 contains data points 2, 4, and 5, and G_3 contains data points 9, 10, 11, and 12. After the fourth round, no assignment has been changed. New-SCC stops with three clusters G_1, G_2, and G_3, with

$$m_1 = \langle 0.366, 0.526 \rangle, \sigma_1 = \langle 0.2882, 0.3053 \rangle;$$
$$m_2 = \langle 0.7, 0.2667 \rangle, \sigma_2 = \langle 0.28, 0.2764 \rangle;$$
$$m_3 = \langle 0.24, 0.87 \rangle, \sigma_3 = \langle 0.2408, 0.24 \rangle.$$

IV. Experimental Results

In this section, some experimental results are presented. Comparisons between SCC and New-SCC have been done. Several real world datasets taken from the UCI Machine Learning Repository are used in experiments [2]. The characteristics of these datasets are listed in Table 2.

Table 2. Accuracy results with different window sizes

Dataset	# instances	# features	# classes
Breast	569	30	2
Ecoli	336	7	8
Glass	214	9	6
Heart	270	13	2
Iris	150	4	3
Libras	360	90	15
Wine	178	13	3
Yeast	1484	8	10

In this table, column 1 indicates the name of the dataset, and the remaining columns indicate the number of instances, the number of features, and the number of classes, respectively, associated with each dataset. For example, the Breast dataset contains 569 data instances, each instance has 30 features (or dimensions) and belongs to one of 2 classes. Note that these datasets are single-labeled, i.e., an instance belongs to only one class.

To evaluate the effectiveness of SCC and New-SCC, the following performance measures are adopted:

1. F-score. It is defined as

$$F-score = \sum_{j=1}^{k} \frac{n_j}{n} * max_{1 \le l \le L} \{ \frac{2 * \frac{n_{jl}}{n_j} * \frac{n_{jl}}{n_l}}{\frac{n_{jl}}{n_j} + \frac{n_{jl}}{n_l}} \} \quad (8)$$

where **k** is the number of classes, **L** is the number of clusters, **n** is the size of the entire data set, n_{jl} is the number of data instances belonging to class j in cluster l, n_l is the size of cluster l, and n_j is the size of class j.

2. RI. It is defined as

$$RI = \frac{a+b}{n(n-1)/2} \tag{9}$$

where **a** is the number of pairs of data objects having different class labels and belonging to different clusters, **b** is the number of pairs of data objects having the same cluster labels and belonging to the same clusters, and **n** is the size of the entire data set.

3. NMI. It is defined as

$$NMI = \frac{\sum_{j=1}^{k}\sum_{l=1}^{L} n_{jl} log(\frac{n * n_{jl}}{n_j * n_l})}{\sqrt{(\sum_{j=1}^{k} n_j log \frac{n_j}{n}) * (\sum_{l=1}^{L} n_l log \frac{n_l}{n})}} \tag{10}$$

where **k** is the number of classes, **L** is the number of clusters, **n** is the size of the entire data set, n_{jl} is the number of data instances belonging to class j in cluster l, n_l is the size of cluster l, and n_j is the size of class j.

All these measures have a common property: a higher measure indicates a better clustering performance.

Table 3 shows performance comparisons between SCC and New-SCC. In this table, the values for the three measures, F-score, RI, and NMI, are listed, and the CPU time elapsed in clustering is also listed in the last column. For the sake of fairness, we compare SCC and New-SCC under the condition of producing the same number of clusters for each dataset. Evidently, New-SCC performs better than SCC in F-score, RI, and NMI for most of the datasets. For example, for the Breast dataset, SCC has F-score = 0.7691, RI=0.6742, and NMI = 0.3349, while New-SCC has F-score = 0.9260, RI=0.8630, and NMI=0.6049. We can see that New-SCC provides a very significant improvement to SCC in this case. However, not all the datasets offer so much difference.

For some datasets, New-SCC is even inferior to SCC in some measure or another. For example, for the Ecoli dataset, SCC has F-score = 0.7333, RI = 0.8229, and NMI = 0.6261, while New-SCC has F-score = 0.7212, RI = 0.8339, and NMI=0.6447. NEW-SCC is better in RI and NMI, but is worse in Fscore. Note that, in general, New-SCC takes more CPU time in clustering than SCC. This is reasonable, since SCC performs one round of iteration while New-SCC performs two or more rounds of iteration. For example, for the Breast dataset, SCC takes 0.02 seconds while New-SCC takes 0.25 seconds for clustering.

Table 3. Accuracy results with different dimension sizes

Dataset		F-score	RI	NMI	# clusters	CPU time
Breast	SCC	0.7691	0.6742	0.3349	2	0.02
	New-SCC	0.9260	0.8630	0.6049	2	0.25
Ecoli	SCC	0.7333	0.8229	0.6261	8	0.015
	New-SCC	0.7212	0.8339	0.6447	8	0.29
Glass	SCC	0.5209	0.5506	0.3440	6	0.01
	New-SCC	0.5593	0.6472	0.4519	6	0.13
Heart	SCC	0.6506	0.5273	0.0938	2	0.01
	New-SCC	0.6470	0.5342	0.1075	2	0.056
Iris	SCC	0.8639	0.8589	0.7351	3	0.006
	New-SCC	0.8901	0.8781	0.7472	3	0.04
Libras	SCC	0.3871	0.7619	0.4434	15	0.023
	New-SCC	0.4767	0.8787	0.5603	15	0.37
Wine	SCC	0.7201	0.7073	0.5525	3	0.007
	New-SCC	0.7634	0.7546	0.6024	3	0.075
Yeast	SCC	0.4184	0.5398	0.1985	10	0.07
	New-SCC	0.4300	0.7288	0.2653	10	5.6

V. Conclusion

We have presented a new clustering algorithm, New-SCC, which is an improvement to the SCC clustering algorithm. SCC considers all the data points one by one sequentially. Clusters are created incrementally and automatically. If the input data point is similar enough to an existing cluster, the point is added to the cluster. Otherwise, the data point forms a new cluster of its own. SCC ends up with a set of clusters after it runs through the whole dataset once. However, once a data point is assigned to a cluster, there is no way to change the assignment afterwards. This may cause assignment errors and the efficacy of the clustering is reduced. New-SCC is aimed to overcome this shortcoming. A data point can be re-assigned to another cluster. Adding points into and removing points from a cluster are allowed to be done iteratively in the clustering process. New-SCC stops when all the assignments are stable, i.e., no assignment would be changed. As a result, New-SCC can produce better clusters. Experimental results have shown that NEW-SCC performs better than SCC for real world datasets.

References

[1] Alvarez, F. M., Troncoso, A., Riquelme, J. C. and Ruiz, J. S. A. (2011) Energy time series forecasting based on pattern sequence similarity. *IEEE Transactions on Knowledge and Data Engineering*, **23**, 1230–1243.

[2] Asuncion, A. and Newman, D. (2007) UCI machine learning repository.

[3] Haykin, S. (1999) *Neural Networks -- A Comprehensive Foundation*. Prentice-Hall Upper Saddle River, NJ, USA.

[4] Lee, S. J. and Jiang, J. Y. (2014) Multilabel text categorization based on fuzzy relevance clustering. *IEEE Transactions on Fuzzy Systems*, **22**, 1457–1471.

[5] Lee, S. J., Ouyang, C. S. and Du, S.H. (2003) A neuro-fuzzy approach for segmentation of human objects in image sequences. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, **33**, 420–437.

[6] Li, W., Jaroszewski, L. and Godzik, A. (2001) Clustering of highly homologous sequences to reduce the size of large protein databases. *Bioinformatics*, **17**, 282–283.

[7] Liao, C. L. and Lee, S. J. (2016) A clustering based approach to improving the efficiency of collaborative filtering recommendation. *Electronic Commerce Research and Applications*, **18**, 1–9, 2016.
[8] Wang, M., Yu, Y. and Lin, W. (2009) Adaptive neural-based fuzzy inference system approach applied to steering control. *In Proceedings of International Symposium on Neural Networks*, Springer, 1189–1196.
[9] Olson, D. L. and Shi, Y. (2007) *Introduction to business data mining*, **10**, McGraw-Hill/Irwin Englewood Cliffs.
[10] Ouyang, C. S., Lee, W. J. and Lee, S. J. (2005) A TSK-type neurofuzzy network approach to system modeling problems. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, **35**, 751–767.
[11] Wang, Z. Y. and Lee, S. J. (2014) A neuro-fuzzy based method for TAIEX forecasting. *In Proceedings of International Conference on Machine Learning and Cybernetics (ICMLC)*, **2**, 579–584.
[12] Wei, C. C., Chen T. T. and Lee, S. J. (2013) K-NN based neuro-fuzzy system for time series prediction. *In Proceedings of 14th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, 569–574.
[13] Xu, R. F. and Lee, S. J. (2015) Dimensionality reduction by feature clustering for regression problems. *Information Sciences*, **299**, 42–57.
[14] Xu, Y., Olman, V. and Xu, D. (2002) Clustering gene expression data using a graph-theoretic approach: an application of minimum spanning trees. *Bioinformatics*, **18**, 536–545.