Clustering Algorithm of Similarity Segmentation based on Point Sorting

Hanbing Li, Yan Wang*, Lan Huang, Mingda Li, Ying Sun, Hanyuan Zhang
College of Computer Science and Technology, Jilin University
Changchun, Jilin, 130012, China
Corresponding author: wy6868@hotmail.com

Abstract—We propose a clustering algorithm of similarity segmentation based on point sorting to improve the clustering performance. Taking full advantage of segmentation sorting of the clustering algorithm based on minimum spanning tree, the algorithm use a variety of methods for different situations to sort these cluster elements with their similarity and segment them where there are large changes in their similarity to obtain cluster results. In order to compare the performance of the method, we select some traditional cluster analysis methods like k-means, hierarchical clustering and density clustering with noise data, etc. In the experimental testing, we select three sets of two-dimensional artificial data sets and four sets of real data sets as test data. And three evaluation indexes are applied to measure the quality of clustering. The simulation results in test data show that this algorithm can improve the accuracy of the algorithm effectively and achieved good clustering performance.

Keywords-similarity; point sorting; segmentation clustering; Wavelet filter

I. INTRODUCTION

Cluster analysis [1, 2] method is a procedure that all the elements in a set will be divided into multiple sets considering some criteria, and the elements are divided into the same set have more significant similarity than those in different sets. Namely, the elements concentrated in the same set, which is usually called a cluster, are more similar with their similarity or distance as the division standard. Commonly used cluster analysis methods can be roughly divided into five categories that are classified clustering, hierarchical clustering [3, 4, 5], density clustering, grid clustering [6], clustering model. These categories are represented as one or more of specific analysis methods, such as k-means clustering [7, 8, 9] is a representative of classified clustering, and DBSCAN [10, 11] is a representative of density clustering, etc. These clustering methods are widely used in the previous studies, and are applied to different situations considering their own specially advantages and disadvantages.

In the article, by our understanding of the nature of cluster analysis, the clustering problem is converted to arrange the data elements in a certain order to a one-dimensional array based on the similarity and segment them to cluster groups according to certain rules.

II. SIMILARITY SEGMENTATION BASED ON POINT SORTING

A. Algorithmic Thinking

As mentioned in the introduction, the cluster analysis is a procedure that all the elements will be divided into multiple sets, and the data in the same set after the division has more significant similarity, while the data in different sets has a lower degree of similarity. The mathematical description of the concept is as follows:

Definition 1. Set U as a limited data set, the set contains n elements $\{X_1, X_2, X_3, ..., X_n\}$, each element X_i contains m properties. Namely X_i can be expressed as the vector form $X_i = (X_1, X_2, X_3, ..., X_m)$, $d(X_1, X_2)$ is the distance between X_i, X_j.

Clustering is the process of dividing set $U = \{X_1, X_2, X_3, ..., X_n\}$ into K non-empty set, $C_1, C_2, C_3, ..., C_k$, by a certain rule, and the set, $C_1, C_2, C_3, ..., C_k$ need to satisfy the following conditions:

1) $C_i \neq \varnothing (i = 1, 2, 3, ..., k)$

2) $C_i \cap C_j = \varnothing \ (i = 1, 2, 3, ..., k,\ j = 1, 2, 3, ..., k, i \neq j)$

3) $\bigcup_{i=1}^{k} C_i = U$

According to the definition 1 and the data in the same set after the division has more significant similarity, while the data in different sets has low similarity. Clustering should be subject to the following constraints in the ideal conditions:

$$\forall X_m, X_p \in C_i, \forall X_n, X_q \in C_j, d(X_m, X_n) > d(X_m, X_p)$$
$$i = 1, 2, 3, ..., k,\ j = 1, 2, 3, ..., k, i \neq j,$$
$$m, p = 1, 2, 3, ..., card(C_i),\ n, q = 1, 2, 3, ..., card(C_j).$$

The distance of any two elements in the same class should be shorter than these in different classes.

Assuming that one-dimensional array orderly formed with cluster labels by sorting the clustered elements under ideal conditions is as follows:

C_1	C_2	C_3	C_4	.....	C_k

Now suppose the cluster C_1 has m elements, namely $X1_1, X1_2, ..., X1_m \in C_1$, C_2 has n elements, namely $X1_1, X1_2, ..., X1_n \in C_2$, the sorting order of the elements in

adjacent clusters C_1 and C_2 in the above array is as follows:

.....	$X1_{m-1}$	$X1_m$	$X2_1$	$X2_2$	$X2_3$	.....

After the continuous calculation of the distance between the adjacent element nodes in the array, it is obvious to conclude that $d(X1_m, X2_1) > \forall d(X2_i, X2_j)$, $(i, j = 1,2,3,...,card(C_2), i \neq j)$. In other words, calculating all the distances between every adjacent elements and segmenting the largest one will lead to get n+1 clusters.

Of course, the above situation is under ideal conditions, but the actual situation which is often encountered with is that the distance between the elements cannot work well to explain the similarity between each pair elements. For example, the elements in the two-dimensional plane in the cluster, the most intuitive and commonly used Euclidean distance is not very good to reflect element density distribution, while the density distribution of the elements is a key reference for a lot of clustering data. And when using two-dimensional data is processed by the Euclidean distance, it often happens that some elements are likely to have the same distance to two or more classes, namely $d(X_m, X_n) = d(X_m, X_p)$, it is not in conformity with the assumption of ideal conditions. Even so, we usually think that it still meets the condition that is $d(X1_m, X2_1) \geq d(X1_m, X1_t)$, $(t = 1,2,3,...,card(C_1))$ and the average distance between the element and all the other elements in same the cluster should be shorter than the average distance between the elements and all elements in a different cluster, which is concluded as follows:

$$\frac{\sum_{i=1}^{card(C_1)} d(X1_m, X1_t)}{card(C_1)} < \frac{\sum_{i=1}^{card(C_2)} d(X1_m, X1_r)}{card(C_2)} \quad (1)$$
$$(t = 1,2,3...,card(C_1), r = 1,2,3...,card(C_2))$$

In the above conditions, clustering can be performed according to this method that the nodes in a array are sorted by the distance, and then the distance between the adjacent nodes is calculated, the segmentation is applied in the maximum distance. This method can be realized by a two-category algorithm as a result of the assumption that the distance between elements in the same cluster is shorter or equal to the distance between the elements in different clusters. Firstly, the distance of each pair elements and the sums of all distances from every element to others are calculated successively, and then the maximum value of the sums of the distance from an element to all the others is added in the head of an one dimensional array, the element is noted as E1, then another element, noted as En which has the maximum distance to E1 has been found out and put into the tail of the one-dimensional array. Under these conditions, E1 and En, should not be divided in the same class in any case. In the next step, an element which is the nearest distance from array head like E1(or an array of tail) is found out and inserted into the position where is adjacent to array head (or the tail of the array), noted as E2. An element which has the nearest average distance from all elements of the array head (or the tail of the array) is found out and inserted into the position where is adjacent to array head (or the tail of the array), until all the elements in the array are arranged into the appropriate position in this order. Calculate the distance of the adjacent nodes in the sorted array, segmentation in the maximum distance, and two clusters are formed at last. If there is a need to get more clusters, run the algorithm again in one of the existed clusters, then there will be one more cluster obtained, and so on.

The above algorithm is based on the ideal conditions that the distance between elements in the same cluster is shorter or equal to the distance between the different clusters. However, in many non-convex data, especially in the uneven density distribution of the data set, the constraints cannot be effectively guaranteed. Therefore, in the actual situation, according to two strategies, which are that the distance between the adjacent the clustering center is minimum and the average density of the adjacent elements is maximum, to sort all elements in different clusters to the array by one of the strategies, and the clustering segmentation point is still in the longest distance between the adjacent nodes in the array. All the cluster centers can be regarded as a complete graph, the reciprocal of the distance between centers is used as the weight to get minimum spanning tree [12], use the weight to traverse the minimum spanning tree in depth first or breadth first, the traversal results will be put in a array. The procedure mentioned above is to complete the process of point sorting. The distance between the adjacent elements in a sorted array is calculated, segment in the maximum distance. Then the clustering results are obtained.

B. Algorithmic Description

Similarity segmentation clustering algorithm based on points sorting has two steps including points sorting and segmentation clustering, different strategies can be adopted in each step for different situations.

1) Point Sorting: The multidimensional distance between the elements can be mapped into a one-dimensional distance in an array by points sorting. Different sorting methods can be adopted for different data sets, the sorting result from the existing cluster analysis method can also be used. For example, hierarchical clustering produces nested and disjoint tree (Fig. 1) while it is also sorting the element nodes in the data set. Although sometimes this kind of sorting is arbitrary, it can still reflect the degree of similarity between nodes to a large extent. The sorting result is to put into a one-dimensional array and the adjacent elements nodes in the array usually have a high similarity. The application based on this method will be further described later. Point sorting can also be sorted again by the sorting results of the later cluster analysis method. The ultimate goal of points sorting is to make two nodes with a more closely distance between adjacent nodes. At this point, some of the adjacent nodes are close enough to each other can be thought in the same cluster. The maximum distance area between adjacent nodes is called clustering segmentation.

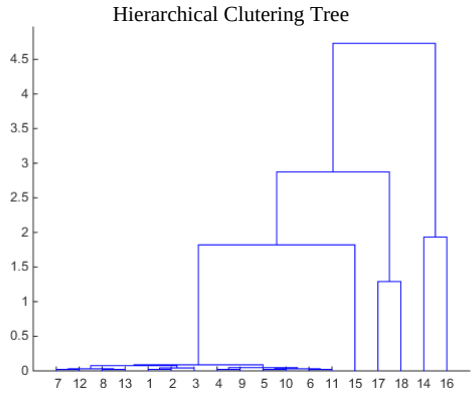

Figure 1. The order of 7, 12, 8, 13, 1, 2...which is below the hierarchical clustering can be input one dimensional array to form a sort of element nodes.

2) Segmentation Clustering: For the segmentation clustering of point sorted array, calculate the distance between each adjacent nodes in the sorted array, the distance between adjacent nodes become drawn curve, as shown in Fig. 2. The distance between the adjacent nodes with strong amplitude curve means this distance between several nodes on both sides of the change is longer enough to regard this position, as a clustering segmentation point. This corresponding relationship between the distance curve and the similarity matrix of adjacent nodes are shown in Fig. 3.

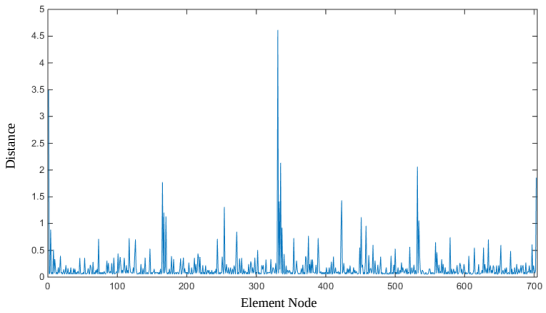

Figure 2. Distance curve between the adjacent nodes.

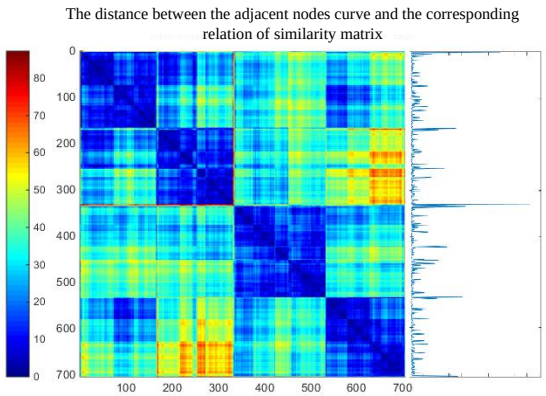

Figure 3. The maximum distance between adjacent nodes which is also regarded as the maximum amplitude is located in similarity matrix for the most suitable for clustering segmentation.

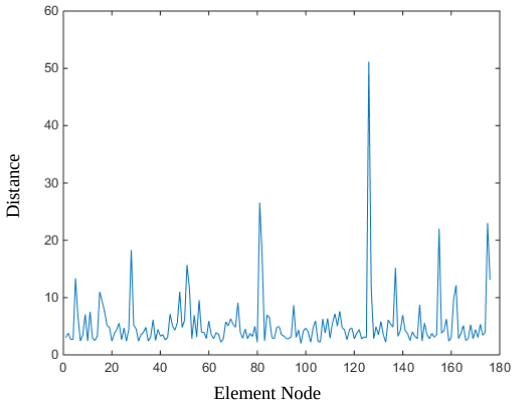

Figure 4. Distance curve between the adjacent nodes filtered by using db2 wavelet.

After wavelet de-noising, the curve has become relatively smooth, so that the distance between the adjacent nodes changes obviously. You can give a certain threshold and clear the part of the curve which is less than the threshold. That means that the several elements whose distance between the adjacent nodes is less than a certain threshold are in the same cluster and they should not break up into others. The threshold value is typically the average of the amplitude of the curve. This curve has been divided into a number of segments which are discrete, as shown in Fig. 5.

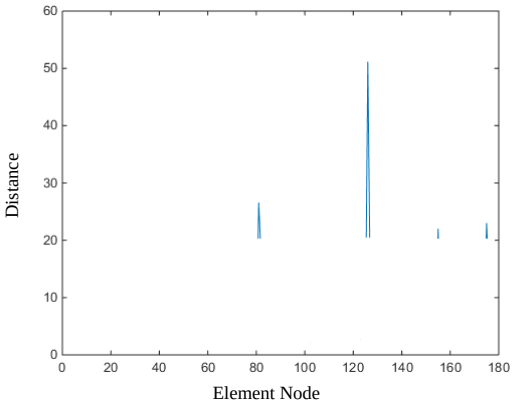

Figure 5. After filtered, distance curve between the adjacent nodes is divided into several disconnected segments.

In this case, to find the clustering segmentation points is to search the maximum distance between the adjacent nodes. All segmentation points are sorted and they will be the preferred point of division when the distance is larger. In other words, clustering for dichotomous classes is to select the first sorted points for segmentation and for three classes is to select the top two sorted points for segmentation and so on, until the required number of clusters are obtained.

III. SIMULATION EXPERIMENT

A. *Experiment Test Data Set*

For experimental input data format, the rows represent different elements and the columns represent different attributes of the element. The experimental output data are one-dimensional array by consisting of cluster labels with corresponding to the row elements of the input data. For non-standard partition data and two-dimensional data, the result is estimated by subjective judgment of the observer by a graphical representation of the results.

In the experiment, in order to compare the feasibility and performance of segmentation clustering algorithm based on point sorting, specially selected three groups of two-dimensional artificial data sets, which are three rectangular data with visible boundary (Fig. 6), a data set based on Gaussian distribution (Fig. 7), and a clustering data sets based on the distinction of density (Fig. 7), and the other four groups are real data sets commonly used in studies of cluster analysis. The four groups of data are iris, alcohol, breast cancer , heart disease data sets which are from the University of California campuses ear Bay (UCI) machine learning [13, 14] database. For the data defined by existed criterion, the quality of clustering results was measured by Rand Index [15] and Adjusted Rand index [16]. In experiments made by simple point sorting, we choose the data set more easily divided to verify the feasibility of point sorting algorithm, and then use the algorithm to process the real data sets with the comparison result to other methods.

The next experiment mainly tests the effectiveness of clustering algorithm of similarity segmentation based on adjacent nodes, using three (single connection, average and, complete connection) hierarchical clustering method, and using two kinds of distance (Euclidean distance [17], the Standard Euclidean distance [17] respectively for the three methods. For the two-dimensional data use Euclidean distance and for high-dimensional data use Euclidean distance and the Standard Euclidean distance to deal with. To compare the influence on the result, we use the clustering method of segmentation based on adjacent nodes similarity, not only when comparing with classification number for the standard partition, but also in the number of non-standard classification division. Finally, clustering algorithm of similarity segmentation based on point sorting will be verified by the experiment.

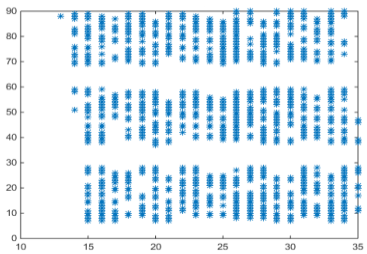

Figure 6. Three rectangular data set.

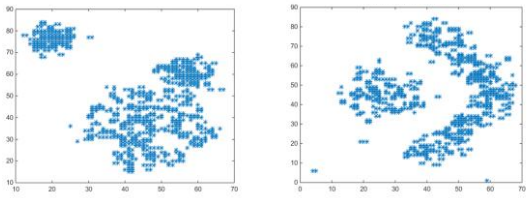

Figure 7. The data set based on the Gaussian Distribution and the distinction of density.

B. *Simple Point Sorting Method*

As mentioned earlier, the applied point sorting method is varied for different data and the similarity. In this section, this article will use a relatively simple method of point sorting for cluster analysis to prove the feasibility of the point sorting method itself.

This simple sorting method is firstly to find the farthest node from all of the nodes in the data set and put the node into the first position of sorting array. Then put the node which is the farthest away from the first sorted node into the last position of sorting array. Next insert the node which is the closest to the first or the last node in the sorting array into the adjacent side of the first or the last node in the sorting array, in order to form an orderly array and complete the point sorting process. Finally find the maximum distance between adjacent nodes to segment in sorted array to form two clusters. As a result, the clusters have been formed through the iterative process for multi-classification. The specific process is as follows:

1) Generate an empty array whose size is the number of all elements in the data set.
2) Find the farthest node from the sum of all the elements of a data set and put it in the first position of the array to store.
3) Find the farthest node from the first sorted node and put it in the last position of the array to store.
4) Judge whether the array is full. If it is full, go to step 5, otherwise go to step 6 .
5) Calculate the distance between adjacent nodes in the array and segment the maximum distance, then judge whether the number of clusters has been reached to the requirements, if meet the requirements, then the algorithm ends, otherwise go to step 7.
6) Find out the node which is the closest to the first or the last node in the sorting array and insert into the adjacent side of the first or the last node in the sorting array, repeat step 4.
7) Compare the distance between adjacent element nodes on both sides of the clustering segmentation points. Find out the maximum distance as the next iteration of the data set, return to step 1.

Using the data sets of three rectangular and data sets based on Gaussian distribution which are more easily to distinguish in the experiment, manually set the clustering results for 3 clusters and the similarity measure using the Euclidean distance, the clustering results are shown in Fig. 8.

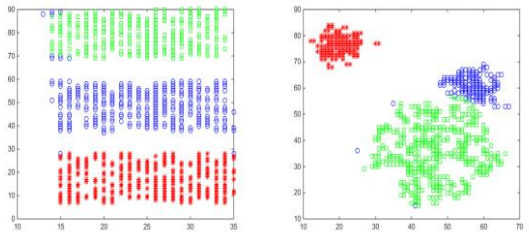

Figure 8. Treatment of the three rectangular data set and the data set based on the Gaussian Distribution using simple points sorting algorithm.

C. *Hierarchical Clustering Combined with Segmentation*

Hierarchical clustering method itself has a function of points sorting, although this kind of points sorting may be arbitrary to some extent. On the whole, it still meets the requirements that the distance between the adjacent nodes should be close in points sorting. In this section, in order to validate the feasibility and actual effect on similarity segmentation clustering between adjacent nodes using the existing point sorting method, we get the sorting result from single connection hierarchical clustering method and average hierarchical clustering method, then split sequencing nodes array to use of similarity segmentation clustering method and obtain the clustering results finally. Its algorithm process is listed as follows:

1) Firstly form hierarchical cluster tree using hierarchical clustering method, get a group result of points sorting after preorder traversal for the leaf node of the tree.
2) Put the sorting result into an array.
3) Draw distance curve of adjacent nodes.
4) Wavelet filtering and de-noise processing of the distance curve.
5) Return to zero for the parts below the global average and get the distance curve of sectional adjacent nodes.
6) Find out the position of maximum distance between the adjacent nodes in every curve and divide it, namely clustering segmentation point.
7) Order all clustering segmentation points according to the distance, larger distance points segmented firstly.
8) Select N segmentation points of the largest distance and go on segmentation to form N+1 clusters.

In this experiment, we use three sets of real data sets, namely iris, alcohol and breast cancer data. For the distance between nodes, we use the Euclidean distance and the standard Euclidean distance to perform the experiments in each set of data. The experimental evaluation index is measured according to accuracy. The number of the clusters we have taken is 2 to 5, then calculate the accuracy for every cluster. Compare the algorithm performance even if the number of clusters does not equal to the standard division (which is often encountered in practice, because the number of standard division clusters cannot always be predicted in advance). Making a comparison between the clustering results obtained by using adjacent nodes based on similarity segmentation clustering method and original hierarchical clustering method, the results are shown in Fig. 9.

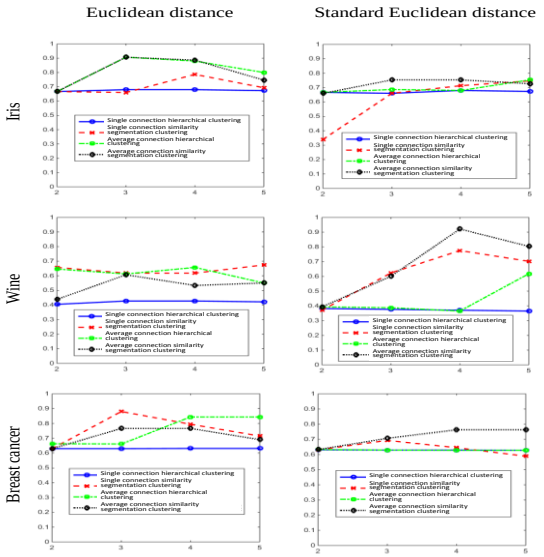

Figure 9. Comparison of accuracy with Hierarchical clustering and similarity segmentation clustering between the adjacent nodes.

From the above experimental results, without changing the point sorting, the result of using segmentation clustering of the similarity between adjacent nodes is significantly better than the division result of hierarchical clustering method in most cases. That is mainly because the use of wavelet filtering for smoothing treatment of distance curve between adjacent nodes can weaken the noise and the impact of abnormal nodes on clustering results. We do not make a separate division of abnormal nodes for the small size. We think it can have a better grasp of the whole for such data sets, but it will also weaken the potential of algorithms in anomaly detection and other aspects.

D. *K-means algorithm combined with point sorting and segmentation Clustering*

Point sorting clustering methods not only can be used as a single cluster analysis method, but also can be combined with other clustering methods to complement and optimize the existing methods. In this section, combined the point sorting clustering method with the K-means algorithm, the experiment is carried on to compare with the results of K-means algorithm. The specific process combined with the K-means for point sorting and clustering segmentation is as follows:

1) Use k-means algorithm to get N clustering groups.
2) Calculate the center of these N clusters.
3) Choose two closest centers of these N clusters, marked as a and b. The two clusters are marked as A and B.
4) Judge if A is from other clusters' merging, if it is, then go to step 6, else go to step 5.
5) Calculate and merge A and B, sort the nodes inside the clusters to form a new cluster C by the formula $C=[A,B]$, then go to step 7.

6) Calculate the distance of each pair of b, a' and b' (a' and b' are formed from last iteration), merge cluster A and B, if $a'b \le bb' and\ a'b \ge a'b'$, sort the nodes inside the clusters to form a new cluster C by the formula $C = [B, A', B',]$; when $a'b > bb' and\ a'b \ge a'b'$, sort the nodes inside the clusters to form a new cluster C by the formula $C = [A', B', B]$, when $a'b < a'b'$, sort the nodes inside the clusters to form a new cluster C by the formula $C = [A', B, B']$.
7) Check if there are other clusters needed to be merged, if yes, return to step 1; else, ordered array has been formed, then go to step 8.
8) Go on segmentation clustering for the ordered array and get the clustering result.

The experimental data is firstly used by intuitive two-dimensional data based on the Gaussian distribution and the density-based division . In the experiments, the data is first to cluster into 10 clusters using K-means algorithm, and then the data based on the Gaussian distribution is clustered into 3 clusters, while the data base on the density into 2 clusters, the experimental results are shown in Fig. 10 and Fig. 11:

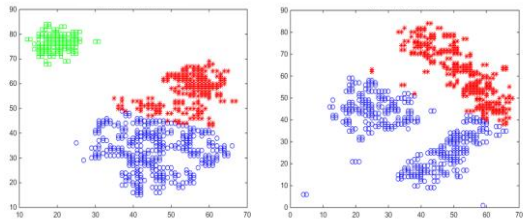

Figure 10. The clustering result of the data set based on the Gaussian Distribution and the data set based on distinction of density obtained by using K-means algorithm.

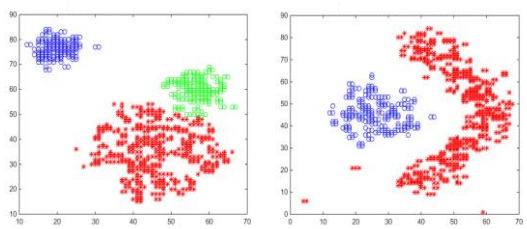

Figure 11. The clustering result of the data set based on the Gaussian Distribution and the data set based on distinction of density obtained by using K-means algorithm combined with points sorting segmentation clustering.

In Fig. 10 and Fig. 11, we clearly see that k-means algorithm, which was unable to originally handle the non-convex data, have been able to work well with density-based division of data sets after combining segmentation clustering of point sorting, and the performance in the data set based on the Gaussian distribution of is also greatly improved.

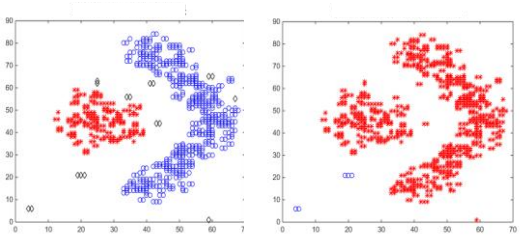

Figure 12. The clustering result of the data set based on distinction of density obtained by using DBSCAN algorithm and K-means algorithm combined with points sorting segmentation clustering.

Also it needs to pay attention that the k-means method itself is unstable, you cannot always get the best clustering results. But even so, the results from k-means algorithm combined with point sorting segmentation clustering are more satisfactory than the original method. It can be found from the comparison between Fig. 11 (right) and Fig. 12 (right) (in the figure using a density-based clustering algorithm for comparison).

At the end of this section, 4 groups of real data used in this paper using the K-means clustering segmentation method combined with point sorting. Usually the performance on heart disease data is worst with the K-means clustering algorithm. In the experiment, firstly, using the K-means method to cluster the heart disease data into 8 clusters, and then into 2 clusters with the segmentation clustering based on point sorting. The accuracy and Rand index reached respectively 71.62% and 63.77%. It has been great increased, compared with the 52.57% and 49.91% for heart disease data analyzed by the original K-means algorithm. The experimental results prove that performance and adaptability of the K-means algorithm combined with segmentation clustering of point sorting have greatly improved.

E. Hierarchical Cluster combined with the Methods of Points Sorting and Segmentation Clustering

Because the result of points sorting obtained from the hierarchical cluster tree is usually arbitrary, it can only show the successively order of merged clustering and can't change the order of nodes merged. In other words, it can't change the order of the nodes which are merged earlier and can't insert the new merging point of the cluster into the cluster having been merged. Then we can consider the changes of clustering center merged every time when they were merged. Thus before merging operation, we can firstly calculate the current clustering center, and compare the distance between the clustering center. When the current clustering center was needed to be merged at now locates between the two clustering centers be merged last time, we insert the current clustering nodes into them for nodes sorting. This is equivalent to backtrack the process of hierarchical clustering to a certain extent. In this experiment, we use average connection hierarchical clustering based on clustering order to modify sorting method. Centroid is used instead of clustering center for simplicity. Centroid of the clustering is as follows:

$$C = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad (2)$$

Its algorithm process is listed as follows:

1) Find out two clusters needed to be merged using average connection hierarchical clustering method and marked as A and B, then calculate the clustering center of the both clusters, marked as a and b.
2) Judge if A is from other clusters' merging, if it is, then go to step 4, else go to step 3.
3) Calculate and merge A and B, sort the nodes inside the clusters to form a new cluster C by the formula $C=[A,B]$, then go to step 5.
4) Calculate the distance of each pair of b, a' and b' (a' and b' are formed from last iteration), merge cluster A and B, if $a'b \le bb' and\ a'b \ge a'b'$, sort the nodes inside the clusters to form a new cluster C by the formula $C=[B,A',B',]$; when $a'b > bb' and\ a'b \ge a'b'$, sort the nodes inside the clusters to form a new cluster C by the formula $C=[A',B',B]$, when $a'b < a'b'$, sort the nodes inside the clusters to form a new cluster C by the formula $C=[A',B,B']$.
5) Check if there are other clusters needed to be merged, if yes, return to step 1; else, ordered array has been formed, then go to step 6.
6) Go on segmentation clustering for the ordered array and get the clustering result.

In order to test the algorithm presented in this section has a better grasp of clustering and the ability to resist noise, the data set based on density distinction is firstly used. Then we apply three hierarchical clustering methods, divide the data set into two to five clusters and display the result of classification. Compare it with the result obtained by using similarity segmentation algorithm based on point sorting presented in this section. The experimental result is showed in Fig. 13:

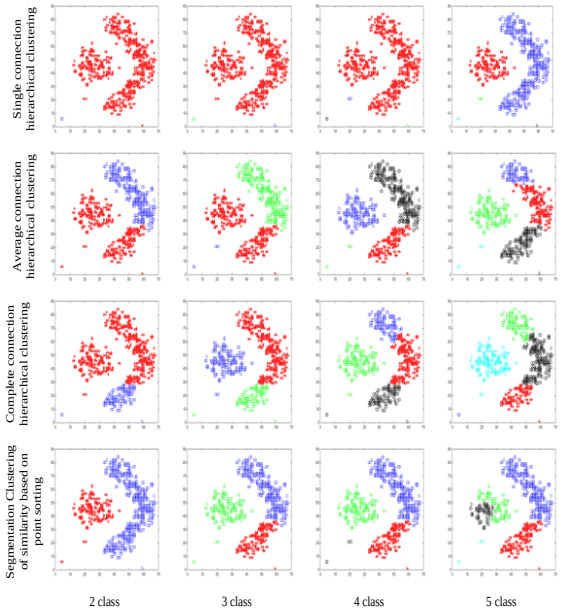

Figure 13. The data set based on distinction of density is divided into two to five clusters by using three hierarchical clustering methods and similarity segmentation clustering based on points sorting.

The experimental results in above show that the three hierarchical clustering result is satisfactory when dividing the data set into five clusters using single connection hierarchical clustering. However, no matter how many clusters are divided using the other two hierarchical clustering, the results cannot meet the requirements. By compared to single connection hierarchical clustering, and dividing the data set into two clusters using similarity segmentation clustering based on points sorting, we still can get a satisfying result. We can get a better result than the result of hierarchical cluster even when choosing wrong clustering number. After choosing proper points sorting algorithm, average hierarchical clustering method that can't be used to deal with non-convex data has a good ability of dealing with the experimental data. In addition, it can have a better grasp of clustering than single connection hierarchical clustering and are much less exposed to the noise interference.

Then we make a full test of performance of the algorithm proposed in this section using three groups of real data sets. We perform similarity measure for every data set employing Euclidean distance and standard Euclidean distance and calculate the accuracy respectively when divided into two to five clusters using all algorithms. Then we choose the one having the best performance and consider the results from the other several important clustering methods which are Single Connection Hierarchical Clustering(SCHC), Average Connection Hierarchical Clustering(ACHC), Complete Connection Hierarchical Clustering(CCHC), K-means Clustering(KMS), Density-based Spatial Clustering of Applications with Noise(DBSCAN), Markov Clustering(MC), Affinity Propagation Clustering(APC), Hierarchical Clustering combined with Point Sorting(HCCPS), Simple Point Sorting (SPS) as a comparison. At last, we draw the graph and the result is showed in Figure 14-16.

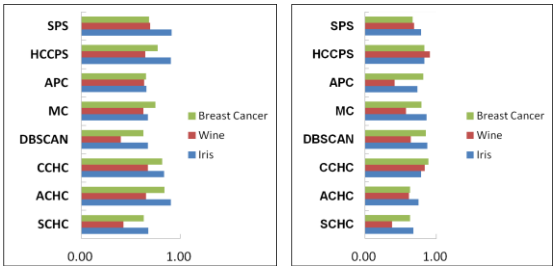

Figure 14. Comparison of the accuracy of similarity calculated by using Euclidean distance and Standard Euclidean distance.

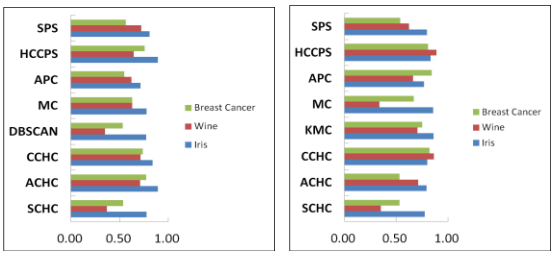

Figure 15. Comparison of the Rand index of similarity calculated by using Euclidean distance and Standard Euclidean distance.

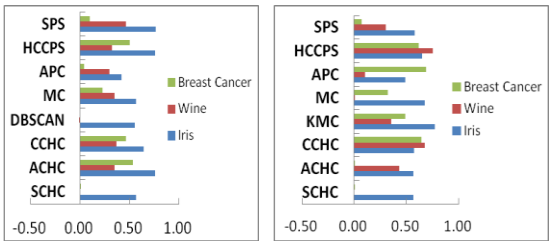

Figure 16. Comparison of the Adjust Rand index of similarity calculated by using Euclidean distance and Standard Euclidean distance.

As you can see from the above experiment results, the accuracy, Rand index and Adjusted Rand index in the treatment of the data sets of iris, alcohol, breast cancer with the algorithm of segmentation clustering based on point sorting is much better than the rest of the clustering algorithms. It well illustrates that the performance of the clustering algorithm based on point sorting is very good and can be used in the application and research. Especially in the structure of the data set with prior knowledge, it can use the corresponding point sorting method to get best function.

IV. CONCLUSIONS

The cluster algorithm of similarity segmentation based on point sorting was proposed in this paper, its key step is using certain rules for the dataset element nodes. The elements are mapped to a set of one-dimensional array in orderly arrangement and calculated the distance between neighboring nodes in the ordered set, and these distance are used to segment clustering in large variations of the similarity. Namely the two core steps that are point sorting and segmentation clustering can be used alone to improve the existing methods, and also can be used in combination with other cluster method into a new cluster analysis method. Experiments demonstrated the effectiveness of each of these two steps, respectively. But one of the major problems is the selection of the similarity in cluster analysis and data structures [18] matching. The obtained results may vary greatly for the same data set with the same algorithm processing, when using the different similarity measures. How to use the most appropriate point sorting for the data sets and similarity of the known characteristics to achieve a minimum measurement error is still pending further study.

ACKNOWLEDGMENT

This work was supported by the Natural Science Foundation of China (Grant No. 61472159) and Development Project of Jilin Province of China (Grant No. 20140101180JC).

REFERENCES

[1] Tryon R C. Cluster analysis: correlation profile and orthometric (factor) analysis for the isolation of unities in mind and personality. Edwards brother, Incorporated, lithoprinters and publishers, 1939.

[2] R. B, "The description of personality: basic traits resolved into clusters," J. Abnorm. Soc. Psychol., vol. 38, no. 4, pp. 476–506, 1943.

[3] R. Sibson, "SLINK: an optimally efficient algorithm for the single-link cluster method," Comput. J., vol. 16, no. 1, pp. 30–34, 1973.

[4] D. Defays, "An efficient algorithm for a complete link method," Comput. J., vol. 20, no. 4, pp. 364–366, Jan. 1977.

[5] J. H. W. Jr, "Hierarchical Grouping to Optimize an Objective Function," J. Am. Stat. Assoc., vol. 58, no. 301, pp. 236–244, 1963.

[6] M.-C. Su and C.-H. Chou, "A modified version of the K-means algorithm with a distance based on cluster symmetry," IEEE Trans. Pattern Anal. Mach. Intell., no. 6, pp. 674–680, 2001.

[7] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, 1967, vol. 1, pp. 281–297.

[8] S. Lloyd, "Least squares quantization in PCM," IEEE Trans. Inf. Theory, vol. 28, no. 2, pp. 129–137, 1982.

[9] E. W. FORGY, "Cluster analysis of multivariate data: efficiency versus interpretability of classifications," Biometrics, vol. 21, pp. 768–769, 1965.

[10] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise.," in Kdd, 1996, vol. 96, pp. 226–231.

[11] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "OPTICS: Ordering Points to Identify the Clustering Structure," in Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data, New York, NY, USA, 1999, pp. 49–60.

[12] Y. Xu, V. Olman, and D. Xu, "Clustering gene expression data using a graph-theoretic approach: an application of minimum spanning trees," Bioinformatics, vol. 18, no. 4, pp. 536–545, Apr. 2002.

[13] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: analysis and implementation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 7, pp. 881–892, 2002.

[14] M. Meilă and D. Heckerman, "An Experimental Comparison of Model-Based Clustering Methods," Mach. Learn., vol. 42, no. 1–2, pp. 9–29, Jan. 2001.

[15] E. B. Fowlkes and C. L. Mallows, "A Method for Comparing Two Hierarchical Clusterings," J. Am. Stat. Assoc., vol. 78, no. 383, pp. 553–569, 1983.

[16] L. Hubert and P. Arabie, "Comparing partitions," J. Classif., vol. 2, no. 1, pp. 193–218, Dec. 1985.

[17] Deza M M, Deza E. Encyclopedia of distances. Springer Berlin Heidelberg, 2009.

[18] Jain A K, Dubes R C. Algorithms for clustering data. Englewood Cliffs: Prentice hall, 1988.