International Journal of Innovative
Computing, Information and Control
Volume **15**, Number **1**, February **2019**

ICIC International ©2019 ISSN 1349-4198
pp. **291–304**

A BOTTOM-UP HIERARCHICAL CLUSTERING ALGORITHM WITH INTERSECTION POINTS

Zahra Nazari[1], Masooma Nazari[1] and Dongshik Kang[2]

[1]Graduate School of Engineering and Science
[2]Department of Information Engineering
University of the Ryukyus
1 Senbaru, Nishihara-cho, Nakagami-gun, Okinawa 903-0213, Japan
zahra.amin.nazari@gmail.com; nazarimasooma@yahoo.com; kang@ie.u-ryukyu.ac.jp

Received April 2018; revised August 2018

Abstract. *A pattern classification problem that does not have labelled data points requires a method to assort similar points into separated clusters before the training and testing can be performed. Clustering algorithms place most similar data points into one cluster with highest intra-cluster and lowest inter-cluster similarities. Purpose of this paper is to suggest a bottom-up hierarchical clustering algorithm which is based on intersection points and provides clusters with higher accuracy and validity compared to some well-known hierarchical and partitioning clustering algorithms. This algorithm starts with pairing two most similar data points, afterwards detects intersection points between pairs and connects them like a chain in a hierarchical form to make clusters. To show the advantages of paring and intersection points in clustering, several experiments are done with benchmark datasets. Besides our proposed algorithm, seven existing clustering algorithms are also used. Purity as an external criterion is used to evaluate the performance of clustering algorithms. Compactness of each cluster derived by clustering algorithms is also calculated to evaluate the validity of clustering algorithms. Eventually, the results of experiments show that in most cases the error rate of our proposed algorithm is lower than other clustering algorithms that are used in this study.*
Keywords: Data mining, Clustering algorithm, Pattern recognition, Machine learning

1. **Introduction.** Since the amount of data that we have to deal increases day by day, the methods that can detect structures in data and identify interesting subsets in datasets become more important. One of these methods is clustering. Clustering or cluster analysis is an unsupervised learning task which organizes data into homogeneous groups based on similarities among the individual points. Clustering is a fundamental problem that has been the focus of considerable studies in machine learning, data mining and statistics [1,2]. Clustering is a widely used algorithm in different areas such as business and retail, psychology, computational biology, astronomy and social media network analysis, to name just a few. Clustering differs from classification by the lack of a predetermined target value to be predicted; therefore, the resulting clusters are not known before the execution of a clustering algorithm. On the other hand, clustering can be thought as an unsupervised classification, because it can produce the same result as classification methods but without having predefined classes [3,4].

Over the years a range of different clustering methods have been proposed and each of them varies with the understanding of what a cluster can be thought. According to Farely and Raftery (1998) clustering algorithms are mainly divided into two groups of hierarchical and partitioning [5], but Han and Kamber [1] classified clustering algorithms into five

DOI: 10.24507/ijicic.15.01.291

major categories: hierarchical methods, partitioning methods, density-based, grid-based, and model-based methods [1,6]. Generally partitioning and hierarchical methods are widely used and more famous than other categories. Partitioning methods split datasets into K partitions and each partition represents a cluster. Partition is crisp if each point belongs to exactly one cluster, or fuzzy if one point is allowed to be in more than one cluster to a different degree [7]. Hierarchical method defined by Johnson [8] arranges data points into an underlying hierarchy which then determines the various clusters.

Among hierarchical algorithms, bottom-up approaches tend to be more accurate, but have higher computational cost than top-down approaches. According to different linkage methods, agglomerative algorithms can be subdivided into single linkage, complete linkage, average linkage, centroid linkage and ward linkage methods which are briefly explained in the next section [5]. Fast agglomerative hierarchical clustering algorithm using locality-sensitive hashing (LSH) link by Koga et al. [9] is proposed for single linkage method. This algorithm utilizes an approximate nearest neighbor search algorithm LSH and is faster than single linkage method for large data [9]. Zahoránszky et al. [10] have introduced a new level independent clustering selection method called LInCS. This method provides a new clustering mechanism that allows computing overlapping clusters, which is good for biological and chemical data sets [10]. Gagolewski et al. [11] presented a new hierarchical clustering linkage criterion which is based on the notion of an inequity (poverty) index. This algorithm depends on a free parameter, the inequity index merge threshold, and user can select its value to suit her/his needs [11].

In this paper we suggest a new bottom-up hierarchical clustering algorithm that uses intersection point as linkage criterion. This approach provides more accurate clustering result since none of nearest neighbors of a data point can be missed. Likewise reduce the clustering steps by combining more than two clusters at each stage. While other well-known bottom-up hierarchical methods (single, average, complete, centroid and ward linkages) combine only two clusters at each stage. This algorithm starts by finding nearest neighbor for each data point to make pairs and then finds the intersection points between pairs to form primary clusters. By finding intersection points between pairs, at each stage we can put all closest data points in a same cluster; therefore, more than two pairs or clusters can be merged in one step. Meanwhile, none of nearest neighbors of a data point will be missed, since there is no need to determine the number of nearest neighbors. Hence, the intra-cluster similarity or compactness of clusters will increase and more intersection points lead to better clustering result. To validate our proposed algorithm we have performed several experiments with benchmark datasets of Iris flowers, Appendicitis, Heart disease and Breast cancer. Purity of our proposed method is compared to the purity of agglomerative hierarchical clustering method with single, average, complete, centroid and ward linkages, as well as purity of K-means and fuzzy c-means methods from partitioning category. Experimental results prove that the new clustering algorithm with intersection points outperforms other clustering methods in most cases and especially in case of high correlated datasets and provides more accurate clustering results.

This paper is organized as follows. In Section 2, we briefly review clustering, (dis)similarity measures and some widely used clustering algorithms. In Section 3, the new bottom-up clustering algorithm with a two dimensional example is introduced. Experimental results which compare our new hierarchical clustering algorithm to seven well-known existing clustering algorithms are presented in Section 4. Conclusion and topics for future research are presented in Section 5.

2. **Clustering Algorithms.**

Clustering or unsupervised classification is a convenient method for identifying homogeneous groups of data points called clusters. Cluster analysis

groups data points based on information found in the data which describes the points and their relationships. Unlike classification and prediction methods that analyze class labelled points, clustering analysis data points without presence of class labels in data and clustering even can be used to generate such labels. Clustering or grouping of data points is based on principle of maximizing the intra-cluster and minimizing the inter-cluster similarities. Therefore, points within a cluster share many characteristics and have high similarity in comparison to one another, but are very distinct from points in other clusters. Each cluster can be viewed as a class of points, from which rules can be derived [1,2].

2.1. **(Dis)similarity measure.** With the data points and their specified attributes, a means is needed to do comparison between them. This comparison can be performed by measuring the similarity or distance between two data points. Distances and similarities play an important role in cluster analysis and the function used to measure the similarity or distance is one of the key components in clustering [12]. There are many measures to calculate either the distance or the similarity between data points. The most popular similarity measures fall into two categories. 1) Difference based measures, which transform and aggregate the differences between attribute values of two compared points. Difference based measures are particularly common and often adopted as default for clustering algorithms unless existing domain knowledge suggests that they may be inappropriate. Euclidean distance, Minkowski distance, Manhattan distance and Mahalanobis distance are some of popular difference based measures. 2) Correlation based measures, which detect the common pattern of low and high attribute values for the two compared points. Pearson's correlation similarity, Spearman's correlation similarity and Cosine similarity are some of popular correlation based measures [13,14]. Euclidean distance is the most popular and widely used distance measure which is used in this study, too. For two data points x and y it can be calculated as:

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{1}$$

There are many algorithms which are proposed for the clustering task and none of them is universally applicable. Different algorithms are in favor for different clustering purposes, so an understanding of both clustering problem and clustering algorithm is required to apply a suitable method to a given problem. In the following, some of the well-known clustering algorithms are explained.

2.2. **Hierarchical clustering.** Hierarchical clustering divides a dataset into a sequence of nested partitions. Hierarchical algorithms are subdivided into agglomerative and divisive algorithms shown in Figure 1. Agglomerative algorithm starts with every single point in a single cluster and then repeatedly merges the closest clusters according to some similarity criteria until all data points are in one cluster [1,2]. Unlike the agglomerative method, divisive algorithm starts with all data points in one cluster and repeatedly splits large clusters to smaller clusters. If we treat agglomerative algorithms as a bottom-up method, then divisive can be viewed as a top-down method. Divisive methods are not generally available and rarely have been applied.

Dendrogram is a special type of tree structure which is used to visualize a hierarchical clustering shown in Figure 2. Same as other methods, there are some disadvantages for hierarchical clustering algorithms, for example: a) if a data point is placed in a group incorrectly at an early stage, it cannot be reallocated, b) different similarity measures for

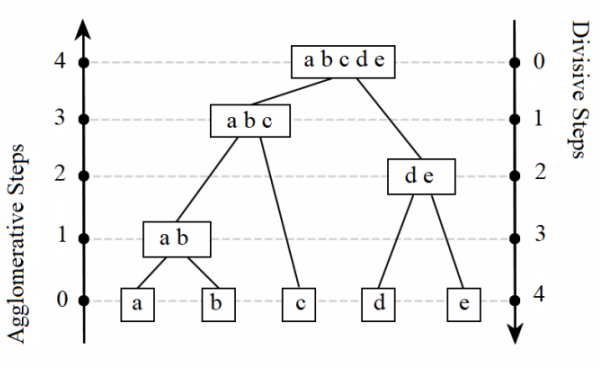


FIGURE 1. This diagram represents the step by step hierarchical clustering applied on data points {a, b, c, d, e}. The up arrow shows steps of agglomerative clustering and down arrow shows the divisive method.

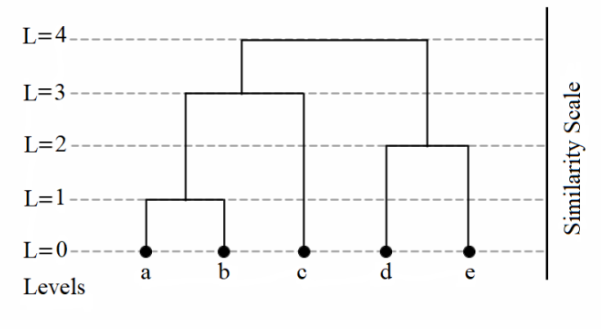


FIGURE 2. Dendrogram, a special type of tree which represents the agglomerative clustering method. Height of links indicates the similarity between data points.

measuring the similarities between points (clusters) may lead to different results. The basic process of hierarchical clustering is as below [14].

1) Start by assigning each item to a cluster, so that if there are n items, we will have n clusters, each cluster containing only one item.

2) Find the closest (most similar) pair of clusters and merge them into a single cluster, so now we have one cluster less.

3) Compute distances (similarities) between the new cluster and of the old clusters.

4) Repeat steps 2) and 3) until all items are clustered into a single cluster or size n.

- Single linkage: In single linkage, the distance between two clusters is the minimum distance between any single data point in the first cluster and any single data point in the second cluster. Therefore, at each stage of the process the two clusters that have the smallest single linkage distance will be combined.

$$D(c_1, c_2) = \min_{x_1 \in c_1, x_2 \in c_2} D(x_1, x_2) \tag{2}$$

- Complete linkage: In complete linkage, the distance between two clusters is the maximum distance between any single data point in the first cluster and any single data point in the second cluster. Therefore, at each stage of the process the two clusters that have the smallest complete linkage distance will be combined.

$$D(c_1, c_2) = \max_{x_1 \in c_1, x_2 \in c_2} D(x_1, x_2) \tag{3}$$

- Average linkage: In average linkage, the distance between two clusters is the average distance between data points in the first cluster and data points in the second cluster. According to this definition of distance between clusters, at each stage of the process the two clusters that have the smallest average linkage distance will be combined.

$$D(c_1, c_2) = \frac{1}{|c_1|}\frac{1}{|c_2|}\sum_{x_1 \in c_1}\sum_{x_2 \in c_2} D(x_1, x_2) \tag{4}$$

- Centroid linkage: In this method, the distance between two clusters is the distance between the two mean vectors of the clusters. At each stage of the process the two clusters that have the smallest centroid distance will be combined.

$$D(c_1, c_2) = D\left[\left[\frac{1}{|c_1|}\sum_{x \in c_1}\overrightarrow{x}\right], \left[\frac{1}{|c_2|}\sum_{x \in c_2}\overrightarrow{x}\right]\right] \tag{5}$$

- Ward linkage: This method is an ANOVA (analysis of variance) based approach and looks at cluster analysis as an analysis of variance problem, instead of using distance measures of association. Therefore, at each stage those two clusters merge, which provides the smallest increase in the combined error sum of squares from one-way univariate ANOVAs that can be done for each variable with groups defined by the clusters at that stage of the process [7,14].

2.3. **Partitioning clustering.** Partitioning clustering generates a single partition of the data to recover natural groups present in the data. Partitioning clustering determines a partition of points into K groups or clusters such that the points in a cluster are more similar to each other than to points in different clusters. K-means and FCM algorithms are most well-known and widely used partitioning methods [15]. In K-means algorithm the objective function (Equation (6)) normally is chosen to be the total distance between all data points from their respective cluster center.

$$J = \sum_{j=1}^{k}\sum_{i=1}^{n}\left\|\left(x_i^{(j)} - c_j\right)\right\|^2 \tag{6}$$

where n is number of points, k is number of clusters and $\left\|\left(x_i^{(j)} - c_j\right)\right\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster center c_j. The K-means algorithm has some drawbacks, for example: a) the performance of algorithm heavily depends on the initial starting conditions; b) it does not work effectively on high dimensional data. And also it works only on numerical data that restricts some applications of K-means algorithm [4,16]. Fuzzy C-means is another widely used partitioning clustering algorithm which is based on minimization of the following objective function [17].

$$j_m = \sum_{i=1}^{n}\sum_{j=1}^{c} U_{ij}^m \|(x_i - c_j)\|^2, \quad 1 \ll m \ll \infty \tag{7}$$

where m is any real number greater than 1, U is the degree of membership of x_i in the cluster j, x_i is the ith of d-dimensional measured data, c_j is the d-dimensional center of the cluster, and $\|(x_i - c_j)\|$ is norm expressing the similarity between x_i and c_j. Fuzzy partitioning is carried out through an iterative optimization of the objective function (Equation (7)) shown above, with the update of membership U_{ij} and the cluster centre c_j by following Equations (8) and (9). To run this procedure c the number of clusters and

m the weighting coefficient must be specified [18,19].

$$U_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \tag{8}$$

$$c_j = \frac{\sum_{i=1}^{n} u_{ij}^m \cdot x_i}{\sum_{i=1}^{N} u_{ij}^m} \tag{9}$$

3. **New Bottom-up Hierarchical Clustering Algorithm with Intersection Points.** Purpose of this algorithm is to provide more accurate and valid clusters with the same computational complexity $O(n^2)$ compared to other methods of bottom-up hierarchical clustering which are used in various domains. This algorithm performs hierarchical clustering by taking advantages of nearest neighbors and intersection points. By finding intersection points between pairs, at each step all closest data points will be placed in a same cluster; meanwhile, none of nearest neighbors of a data point will be missed, since there is no need to determine the number of nearest neighbors.

3.1. **Basic concept.** Intersection is the basic concept which is used in this algorithm. In mathematics, the intersection of two sets A and B denoted $\mathrm{A} \cap \mathrm{B}$ is the set that contains all elements of A that also belong to B, defined by $\mathrm{A} \cap \mathrm{B} = \{x | x \in \mathrm{A} \wedge x \in \mathrm{B}\}$. In plain language intersection means two sets have some elements in common. Therefore, intersection between two sets can be interpreted as similarity between them. Meanwhile, $\mathrm{A} \cap \mathrm{B} = \varnothing$ means there is no element in common between A and B and they are disjoint [20]. A simple example of intersection between two sets of A and B is shown in Figure 3.

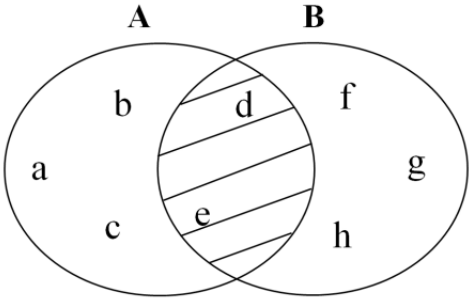


FIGURE 3. Elements {d, e} are intersection points between sets A and B.

3.2. **Algorithm.** The step by step explanation of new bottom-up hierarchical algorithm with intersection points is as follows.

1) Find the nearest neighbor (NN) for each data point and pair them; hence, for n data points we will have n pairs (some will be duplicates). Euclidean distance (Equation (1)) will be used to measure the distance between all data points to make a $(n \times n)$ distance matrix, and then each data point will make a pair with its nearest neighbor. Indexes of data points IoP and their NN $IoNN$ will be used for making pairs.

$$NN = \min_v \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{10}$$

$$Pairs = [\{IoP_1, IoNN_1\}\{IoP_2, IoNN_2\} \ldots \{IoP_n, IoNN_n\}]$$

where min is taken over all neighbors v of data point, and IoP_1 is the index of the 1st data point of the dataset and $IoNN_1$ is the index of NN for the 1st data point.

2) Consider each pair as a set containing index of one data point and index of its NN and find intersection (intersection points) between pairs and join them to make clusters. Suppose that we have a set of 5 data points = {a, b, c, d, e} and NN is already found for each data point in the previous step. Pairs = $[\{1,2\}\{2,3\}\{3,2\}\{4,5\}\{5,4\}]$. The first pair/set $\{1,2\}$ shows that the 2nd data point is the NN for the 1st data point.

To find intersections between pairs, each time we consider two pairs, e.g., for the first two pairs, $\{IoP_1, IoNN_1\}$ and $\{IoP_2, IoNN_2\}$ intersection is $\{1,2\} \cap \{2,3\} = \{2\}$, and we can find this answer by the following calculation:

First pair: $\{IoP_1, IoNN_1\} = \{1,2\}$
Second pair: $\{IoP_2, IoNN_2\} = \{2,3\}$
Intersection = $(IoP_1 - IoP_2) = (1-2) = -1$
$(IoP_1 - IoNN_2) = (1-3) = -2$
$(IoNN_1 - IoNN_2) = (2-3) = -1$
$(IoNN_1 - IoP_2) = (2-2) = 0$ ✓

The $(IoNN_1 - IoP_2) = (2-2) = 0$ ✓ shows that 2(b) is the intersection point between the first pair and second pair and we can join them (see Figure 4). By considering the first pair with others we can find all pairs that have intersection points and join them to make a cluster.

3) Calculate the mean value (center) of each cluster made by step 2).

4) Repeat steps 1) and 2) for mean values continuously to achieve the desired number of clusters. That means we find the NN (mean) for each mean value of our primary clusters, then we make pairs of mean values and in the following we look for the intersections between pairs of mean values to form our clusters. This procedure should be repeated to achieve our desired number of clusters or until all data points are in one cluster. In the following Figure 4 shows the difference between number of steps of new algorithm and two other hierarchical algorithms which are shown in Figure 1.

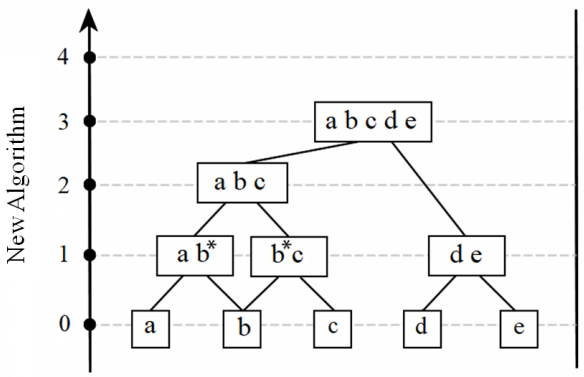


FIGURE 4. The new bottom-up clustering algorithm is used to cluster the dataset which is used in Figure 1 (b* indicates intersection point).

Suppose that we have a 2-dimensional dataset which is shown in Figure 5. Part 1 shows how data points are spread out. Part 2 shows each data point is paired with its nearest neighbour and some of them are nearest neighbour for more than one data point. Those intersection points shown in part 2 make chains of data points that are near to each other and we consider them as primary clusters as shown in part 3. The mean value for each primary cluster is calculated which is also shown in part 3. Subsequently, we consider only mean values and nearest means are found to join nearest clusters which are shown in parts 4 and 5. We repeat this procedure to achieve our desired number of clusters or until all data points are in one cluster as shown in part 6 of Figure 5.

Pseudo code for the new bottom-up hierarchical clustering algorithm with intersection points:

```
A set of n objects, X = {x_1, ..., x_n}, so that x(i) = c(i)
 1. for i = 1 : n do
 2.     for j = 1 : n do
 3.         d(i, j) = distance function (c(i), c(j))          /* distance matrix n × n */
 4.     end for
 5.     pair(i) = min(d(i, j = 1 : n))                        /* pair n × 2 */
 6. end for
 7. for k = 1 : n do
 8.     temp1 = pair(i)
 9.     pair(i) = 0
10.     if temp1 ≠ 0 then
11.         for j = 1 : size (pair) – 1 do
12.             temp2 = pair(j)
13.             intersect = intersection (temp1, temp2)       /* 1 if yes, else 0 */
14.             if intersect = 1 then
15.                 temp1 = merge (temp1 & temp2)
16.                 pair(j) = 0
17.             end if
18.         end for
19.     end if
20.     cluster(i) = temp1
21. end for
22. if number of cluster ≠ 1 then calculate mean value for each cluster, set them as
objects and go to line 1.
23. end if
```

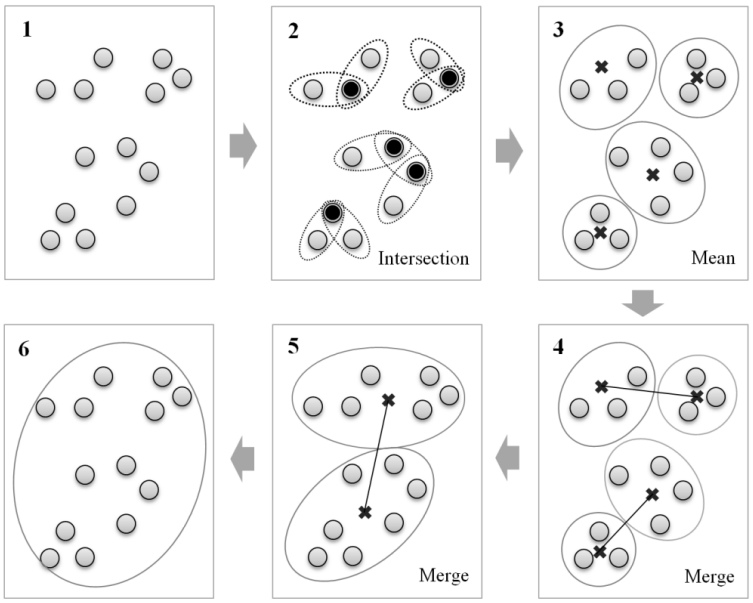


FIGURE 5. A 2-dimensional dataset clustered by new bottom-up hierarchical clustering algorithm (× are mean values, and • are intersection points)

4. **Experiments.** To validate efficiency of our proposed algorithm we applied it to four benchmark datasets of Iris Flowers, Appendicitis, Breast Cancer and Heart Disease datasets from The University of California, Irvine (UCI) repository [21]. The characteristics of datasets are shown in Table 1. All of the mentioned datasets have predefined labels, but we use them without presence of their class labels to form clustering problems. Agglomerative hierarchical clustering with single linkage, complete linkage, average linkage, centroid linkage and ward linkage, K-means clustering and FCM clustering methods which are explained in Section 2 are used for clustering these data and their results are compared to the new algorithm.

TABLE 1. Characteristics of benchmark datasets that are used in experiments

Dataset Name	Dataset Characteristics	Number of Instances	Number of Attributes	Number of Categories
Iris Flowers	Multivariate	150	4	3
Appendicitis	Multivariate	106	7	2
Breast Cancer	Multivariate	286	9	2
Heart Disease	Multivariate	303	13	5

Clustering evaluation or cluster validity is a necessary but challenging task and has become a core task in cluster analysis. Therefore, a great number of validation measures have been proposed. Generally, validation measures are classified to internal and external criteria. Internal criteria are based on intrinsic information of data and analyze the goodness of a clustering structure, but external criteria are based on previous knowledge about data and analyze how close are clustering to a reference, e.g., predefined class labels [22]. Internal clustering evaluation measures often make latent assumption on the formation of cluster structure and usually have high computational complexity. Therefore, researchers prefer to use external criterion when the purpose is only to assess clustering algorithms and class labels are available.

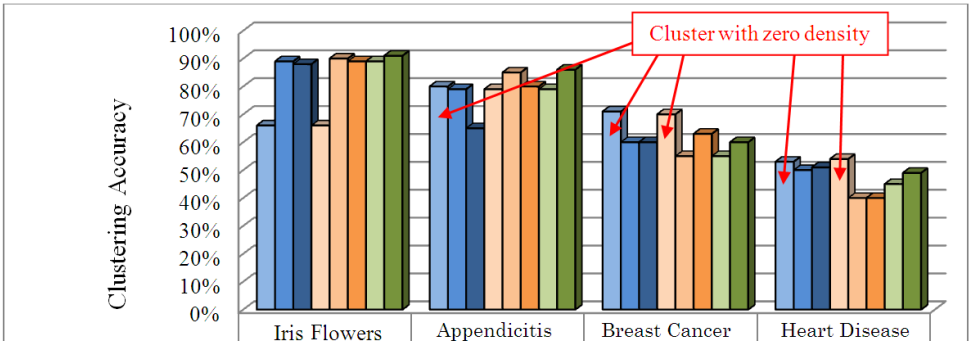
Since the purpose of this study is to introduce and assess a new clustering algorithm and class labels are also available for our datasets, we can use an external criterion. Purity is a popular external evaluation criterion which is used to assess clustering algorithms that are used in this study. To compute purity, each cluster is assigned to the class which is most frequent in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned data points and dividing by N number of data points [23,24].

$$Purity = \frac{1}{N}\sum_{i=1}^{K} \max_j |c_i \cap t_j| \tag{11}$$

where N is number of objects (data points), K is number of clusters, c_i is a cluster in C, and t_j is the classification which has the max count for cluster c_i.

As mentioned in previous section, we have predefined labels for all datasets, so we can use them to calculate the purity of clustering algorithms and rank their results according to their accuracy. The accuracies of clustering methods are calculated and shown in Figure 6.

There is no doubt that the lowest inter-cluster and highest intra-cluster similarities are always desired for clustering results. In other words, the member of each cluster should be as close as possible to each other, which is defined as compactness. A common measure of compactness is the variance. Hence, variances between attribute values in each cluster of a dataset are also calculated. By comparing variance values of each cluster, we can



	Iris Flowers	Appendicitis	Breast Cancer	Heart Disease
Single Linkage	66%	80%	71%	53%
Average Linkage	89%	79%	60%	50%
Complete Linkage	88%	65%	60%	51%
Centroid Linkage	66%	79%	70%	54%
Ward Linkage	90%	85%	55%	40%
K-means	89%	80%	63%	40%
FCM	89%	79%	55%	45%
New Algorithm	91%	86%	60%	49%

FIGURE 6. Purity (accuracy) of clustering algorithms applied on Iris Flower, Appendicitis, Breast Cancer, and Heart Disease datasets

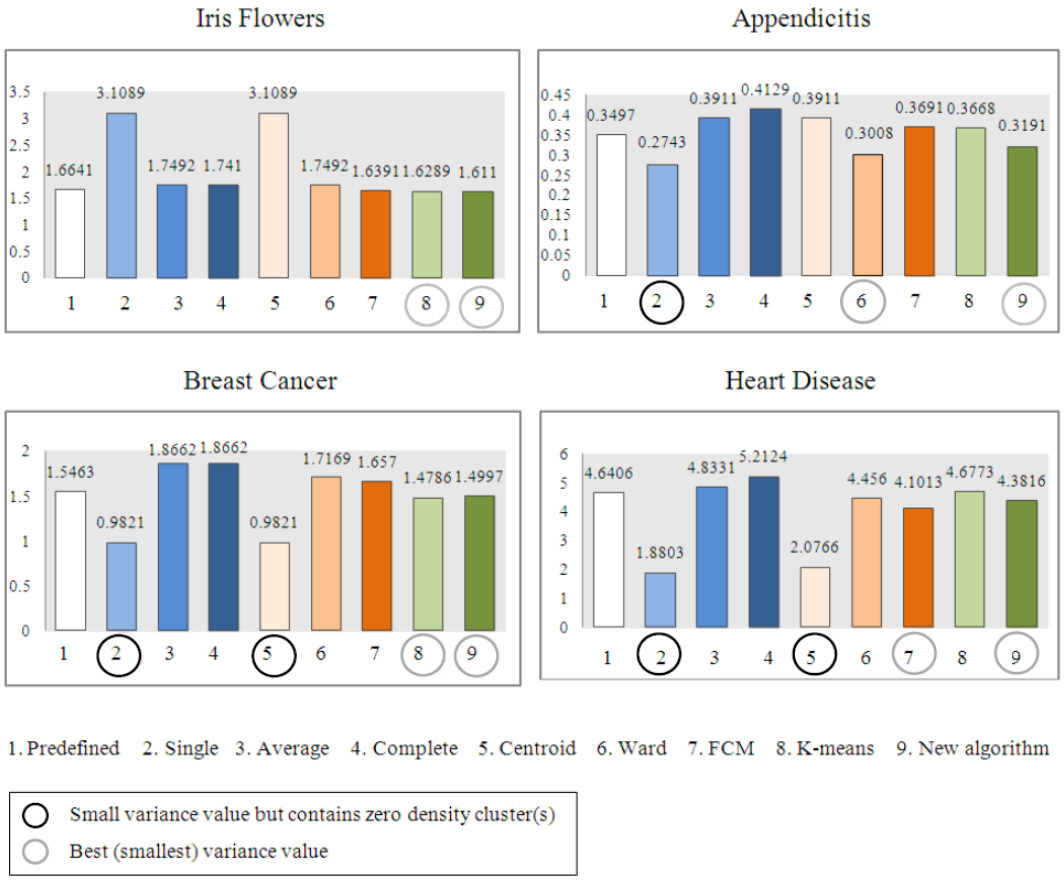


FIGURE 7. The total variance value resulted by each clustering algorithm. Black circles mean total variance value is small but there are one or more clusters with zero density in the clustering results which are not involved in ranking and gray circles show the best variance values. An example of zero density cluster is shown in Table 2 by an asterisk.

measure how similar data points are set in one cluster by each clustering algorithm. In addition to purity, analysis of variances can also help us to evaluate the clustering quality. The total variance values of Iris Flowers dataset, Appendicitis dataset, Breast Cancer dataset and Herat Disease dataset are presented in Figure 7. To clarify that how total variance values are calculated details of variance values of Appendicitis dataset (as an example) are shown in Table 2.

TABLE 2. Variance values of Appendicitis dataset in detail

Labels	**Clusters**	**Attributes**							**Total**
		1	2	3	4	5	6	7	
Predefined	1	0.0213	0.028	0.0293	0.022	0.03	0.0315	0.0213	0.1834
	2	0.0251	0.0302	0.0319	0.0216	0.0058	0.0311	0.0206	0.1663
Total (clusters 1, 2): 0.3497									
Single linkage	1	0.0367	0.0432	0.0425	0.0342	0.0302	0.0482	0.0393	0.2743
	2	0	0	0	0	0	0	0	0
Total (clusters 1, 2): 0.2743*									
Average linkage	1	0.0344	0.0436	0.0403	0.0368	0.0215	0.0482	0.0377	0.2626
	2	0.0002	0.0015	0.0012	0.0138	0.0233	0.0696	0.019	0.1286
Total (clusters 1, 2, 3): 0.3911									
Complete linkage	1	0.0199	0.0497	0.0263	0.036	0.0126	0.0511	0.0222	0.2178
	2	0.0162	0.0077	0.0162	0.0509	0.0748	0.0141	0.0152	0.1951
Total (clusters 1, 2): 0.4129									
Centroid linkage	1	0.0344	0.0436	0.0403	0.0368	0.0215	0.0482	0.0377	0.2625
	2	0.0002	0.0015	0.0012	0.0138	0.0233	0.0696	0.019	0.1286
Total (clusters 1, 2): 0.3911									
Ward linkage	1	0.0316	0.0164	0.0318	0.042	0.0332	0.0188	0.0305	0.2043
	2	0.0182	0.0117	0.0142	0.0114	0.0033	0.0138	0.0239	0.0965
Total (clusters 1, 2): 0.3008									
K-means	1	0.0225	0.0108	0.0205	0.0268	0.0342	0.0106	0.0207	0.1461
	2	0.0117	0.0432	0.0103	0.0745	0.0217	0.0534	0.0082	0.223
Total (clusters 1, 2): 0.3691									
FCM	1	0.0227	0.0105	0.0205	0.0271	0.0347	0.0102	0.0207	0.1464
	2	0.0125	0.0424	0.0111	0.0721	0.021	0.0523	0.009	0.2204
Total (clusters 1, 2): 0.3668									
New algorithm	1	0.0225	0.0135	0.0021	0.0026	0.0311	0.0111	0.021	0.1039
	2	0.0192	0.0462	0.049	0.0411	0.0215	0.0311	0.0071	0.2152
Total (clusters 1, 2): 0.3191									

In Figures 6 and 7, we present the results of eight clustering algorithms applied on benchmark datasets. Figure 6 shows purity (accuracy) of clustering algorithms and accuracy of our proposed clustering algorithm is higher than others in case of Iris Flower and Appendicitis datasets. In Figure 7 we present the total variance values for each attribute of all datasets. Total variance values of those algorithms that have zero density clusters are shown by black circle and they are not involved in ranking. The smaller total variance value (gray circle) indicates higher intra-cluster similarity which is resulted by joining most similar data points. Finding intersection points help us to cluster most similar data points and reduce the variance of the cluster. Therefore, number of intersection points in a dataset has effects on clustering result. Numbers of intersection points found in Iris

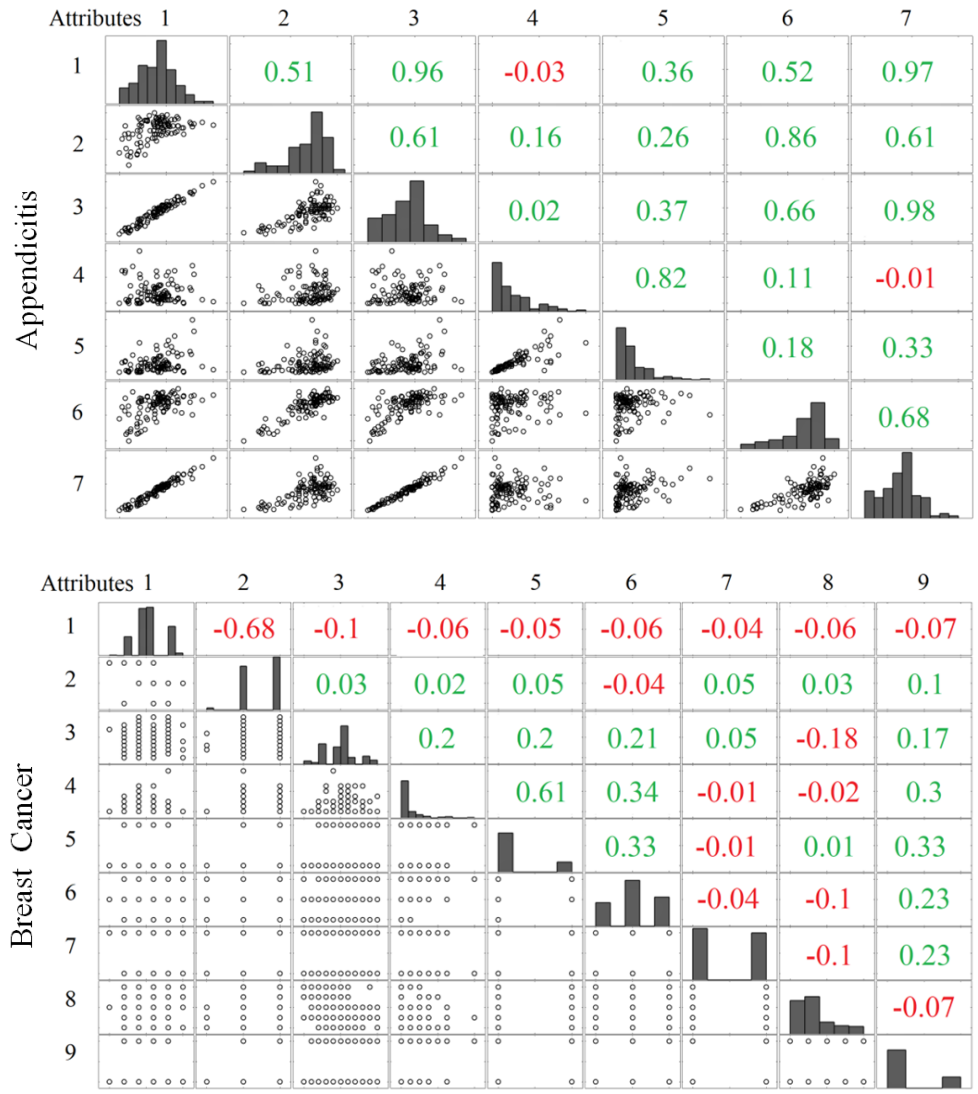


FIGURE 8. Scatter plots, histogram and correlation coefficient

Flowers, Appendicitis, Heart Disease and Breast Cancer datasets are 64 (43%), 50 (47%), 134 (44%) and 115 (20%) respectively.

Number of intersection points is related to the relationship between data. Hence, we calculate correlation coefficient for each dataset to prove this viewpoint. Correlation coefficient, histogram and scatter plots of Appendicitis and Breast Cancer as datasets with biggest and smallest number of intersection points are shown in Figure 8. This figure shows that Appendicitis data are highly correlated but the correlation coefficients between Breast Cancer data are very low.

5. **Conclusion and Future Work.** This paper proposes a bottom-up hierarchical clustering algorithm with intersection points. Several experiments with benchmark datasets are performed to validate usefulness of our proposed clustering algorithm. Seven existing clustering algorithms are used in our experiments. Purity is used as external criterion to evaluate the clustering results of all clustering algorithms. Variance values of attributes of each cluster are also calculated to evaluate the clustering quality. In addition to the purity and variance of clusters, cluster density is also considered to rank the clustering results. Eventually, according to the results of experiments, the bottom-up hierarchical clustering

algorithm with intersection points provides good results with lower error rates for those datasets which are highly correlated, regardless of dimension and number of data points. However, other clustering algorithms with the same computational complexity perform well only in few cases.

As described in Section 2.1, there are several methods to measure the similarities between data points. Euclidean distance is a measure which is used frequently in different areas. We also use Euclidean distance to introduce the bottom-up hierarchical clustering algorithm with intersection points. Therefore, our future work is to use other similarity measures for our proposed clustering algorithm as well as using more datasets for experiments and more clustering algorithms for comparison.

Acknowledgment. Authors would like to thank the KAKENHI(C)-18K02901, University of the Ryukyus and The Mitsubishi Corporation for their supports. We would also like to thank the reviewer(s) for helpful comments and for pointing out some mistakes in the original draft of this paper.

REFERENCES

[1] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2nd Edition, Elsevier Inc., MA, 2006.
[2] B. S. Everitt, S. Landau, M. Leese and D. Stahl, *Cluster Analysis*, 5th Edition, John Wiley & Sons, Ltd., UK, 2011.
[3] M. Sarstedt and Mooi, *A Concise Guide to Market Research*, 2nd Edition, Springer-Verlag, New York, 2014.
[4] P. Tan, M. Steinbach and V. Kumar, *Introduction to Data Mining*, Pearson Education Limited, UK, 2014.
[5] Z. Nazari, D. Kang, M. R. Asharif, Y. Sung and S. Ogawa, A new hierarchical clustering algorithm, *International Conference on Intelligent Informatics and Biomedical Sciences*, Okinawa, Japan, 2015.
[6] F. Achcar, J. M. Camadro and D. Mestivier, AutoClass@IJM: A powerful tool for Bayesian classification of heterogeneous data in biology, *Nucleic Acids Research*, vol.37, no.2, pp.W63-W67, 2009.
[7] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice-Hall, Inc., NJ, 1988.
[8] S. C. Johnson, Hierarchical clustering scheme, *Psychometrika Journal*, vol.32, no.3, pp.241-254, 1967.
[9] H. Koga, T. Ishibashi and T. Watanabe, Fast agglomerative hierarchical clustering algorithm using Locality-Sensitive Hashing, *Knowl. Inf. Syst.*, vol.12, no.1, pp.25-53, 2007.
[10] L. A. Zahoránszky, G. Y. Katona, P. Hári, A. M. Csizmadia, K. A. Zweig and G. Z. Kohalmi, Breaking the hierarchy – A new cluster selection mechanism for hierarchical clustering methods, *Algorithms for Molecular Biology*, vol.4, no.12, pp.1-22, 2009.
[11] M. Gagolewski, M. Bartoszuk and A. Cena, Genie: A new, fast, and outlier-resistant hierarchical clustering algorithm, *Information Sciences*, vol.363, pp.8-23, 2016.
[12] B. Mirkin, *Clustering for Data Mining, a Data Recovery Approach*, Chapman & Hall/CRC, FL, 2012.
[13] P. Cichosz, *Data Mining Algorithms Explained Using R*, John Wiley & Sons Inc., UK, 2015.
[14] G. Gan, C. Ma and J. Wu, *Data Clustering: Theory, Algorithms, and Applications*, ASA-SIAM, VA, 2007.
[15] J. Wu, *Advances in K-Means Clustering. A Data Mining Thinking*, Beihang University, 2012.
[16] T. W. Liao, Clustering of time series data – A survey, *The Journal of Pattern Recognition Society*, vol.38, no.11, pp.1857-1874, 2005.
[17] P. S. Szczepaniak, P. J. G. Lisboa and J. Kacprzyk, *Fuzzy Systems in Medicine*, Springer-Verlag, Berlin, 2000.
[18] J. Abonyi and B. Feil, *Cluster Analysis for Data Mining and System Identification*, Birkhauser Verlag, Berlin, 2007.
[19] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data – An Introduction to Cluster Analysis*, John Wiley & Sons Inc., UK, 1990.
[20] *https://en.wikibooks.org/wiki/Discrete_Mathematics/Set_theory.*
[21] M. Lichman, *UCI Machine Learning Repository*, School of Information and Computer Science, University of California, Irvine, CA, http://archive.ics.uci.edu/ml, 2013.

[22] S. S. I. Walde, Experiments on the automatic induction of German semantic verb classes, *Computational Linguistics*, vol.32, no.2, pp.159-194, 2006.
[23] H. C. Romesburg, *Cluster Analysis for Researchers*, LuLu Press, NC, 2004.
[24] J. Kogan, *Introduction to Clustering Large and High Dimensional Data*, Cambridge University Press, New York, 2007.