
Interval Data Clustering with Applications

Authors and affiliations are hidden for review purpose

Abstract

Interval data is described by a group of variables, each of which contains a range of continuous values instead of the traditional single continuous or discrete value. Traditional data analysis simply replaces each interval by its representative (e.g., center or mean) and ignores the structure information of intervals. In this paper, we study the problem of clustering interval data using the modified or extended interval data dissimilarity measures. Our contributions are two-fold. First, we discuss various approaches for measuring the dissimilarities/distances between interval data, investigate the relations among them, and present a comprehensive experimental study on clustering interval data. We show that the extended interval data clustering achieves better performance than traditional ones and produces more meaningful and explanatory results. Second, we propose a two-stage approach for clustering interval data by exploiting the relations between the traditional distances and the modified distances. Experimental results show the effectiveness of our approach.

1. Introduction

Much of previous clustering work has been based on the two-dimensional tabular data presentation where each data sample is represented as a vector of quantitative/numerical measurements. In real world applications, however, many complex data types such as intervals and histograms have been widely used [1, 19, 24]. Table 1 shows an example of interval data table where each cell element contains the range of meat prices in a certain supermarket. Note that each element is an interval of real numbers rather than a single value. Interval data appear naturally in many applications. In addition, interval data can be generated to represent the variations/distributions of attributes by summarizing large datasets [8, 17].

Traditional clustering techniques can be easily applied to

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

| | CHICKEN | BEEF | PORK |
|---|------------|------------|------------|
| 1 | [2.3, 3.6] | [4.8, 6.9] | [3.2, 5.7] |
| 2 | [3.8, 5.7] | [5.1, 6.5] | [2.8, 5.2] |
| 3 | [2.6, 3.9] | [5.3, 7.8] | [4.2, 4.5] |

Table 1. An example of interval data table

interval data types by replacing each interval with a representative (e.g. the median of the points in the interval). However, this approach ignores the structure information of the interval. As shown in Figure 1, if we use the centroids to represent the intervals, we can not distinguish between A and B . Very limited work has been reported on clustering interval data.

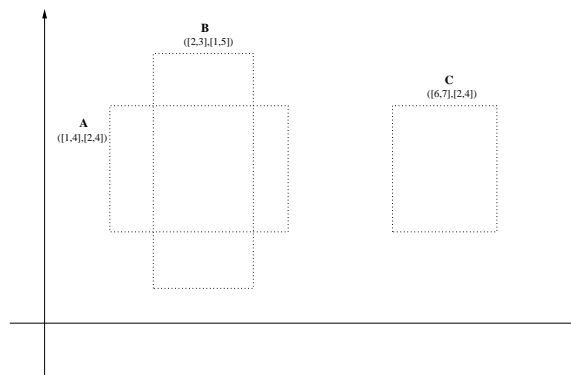


Figure 1. Limitations of using representative centroids to replace intervals. A, B, C are three data objects with two interval attributes. A and B have the same centroids. The representative centroid of C has the same Euclidean distance to the representative centroids of both A and B .

In this paper, we study the problem of clustering interval data, an important yet largely under-addressed problem. First, we discuss various approaches for measuring the dissimilarities/distances between interval data and investigate the relations among them. We also present a comparative study on various datasets. We show that the extended interval data clustering achieves better performance than traditional ones and produces more meaningful and explanatory results. Second, observe that there exists a natural two-level hierarchical representation for interval data: at the first level, the representative (e.g., the median of the points in the interval) can be used to generate a coarse representation of the interval data; at the second level, a fine representation is given by the interval to show its structure infor-

mation. The relationship between the coarse and fine representations motivates a two-stage approach for clustering: at the first stage, the coarse representation is used to obtain a rough partition of the data; at the second stage, the fine representation is employed to refine the partition and generate fine clusterings. Experimental results show that the two-stage approach reduces the computation costs while maintaining the clustering quality. The rest of the paper is organized as follows: Section 2 presents various distance measures for interval data and investigates the connections among them. Section 3 describes the K-mean type algorithm for clustering interval data. In particular, it discusses the objective criterion and illustrates the optimization procedure. Section 4 introduces the two-stage approach for clustering interval data, Section 5 shows the comparative clustering results on real data sets, Section 6 provides the conclusion and discusses future work.

2. Interval Distance Measures and Their Relationships

In this section, we discuss various distance measures for interval data and investigate the connections among them. It should be pointed out that: although the following discussion is based on datasets having only interval type data, it can be easily generalized to datasets having interval data type as well as traditional single-value data type.

2.1. Interval Distance Measures

Interval data can be represented by a vector of interval values. Let $A = (A^1, A^2, \dots, A^p)$ and $B = (B^1, B^2, \dots, B^p)$ be two interval objects with p attributes(variables) where $A^i = [a^i, b^i]$ and $B^i = [c^i, d^i]$ indicate the values of the interval for the i^{th} variable.

First, as mentioned in Section 1, a naive way for measuring the distance between the intervals is to compute the distances between their representatives. We refer to the naive approach as *traditional method*. Hence based on different choices of distance, we obtain *traditional L1 distance* and *traditional L2 distance* for interval data. Second, there are other distance measures which explicitly consider the boundary or the structure of the intervals. Typical examples include Hausdorff distance [5], city-block distance [24], and Minkowski(or Euclidean) distance [4]. We refer to these type of measures as extended/modified dissimilarity measures for interval data.

Table 2 lists various distance measures for interval data. The first row in Table 2, $U1$, is one of most common dissimilarity measures for interval data [9]. It computes the distance between symbolic data by comparing their positions, spans, and contents. Specifically, the distance $U1$ between these two interval objects A and B consists of

three types of dissimilarity measures(normalized to $[0, 1]$) $D_\pi(A^i, B^i)$, $D_s(A^i, B^i)$, and $D_c(A^i, B^i)$. D_π indicates the relative positions of two attribute values on the real line, where $|Y^i|$ is the maximum interval length along variable i . D_s computes the span of interval data where $|A^i| = b^i - a^i$, $|B^i| = d^i - c^i$, and $\max(b^i, d^i) - \min(a^i, c^i)$ (also denoted by $A^i \cup B^i$) is the span length of A^i and B^i . D_c considers the non-common parts of A^i and B^i , where *inter* is the length of $|A^i \cap B^i|$. Actually D_c is the normalized length of non-common part of A^i and B^i . When A^i and B^i intersect each other, *inter* can be represented by $\min(b^i, d^i) - \max(a^i, c^i)$, otherwise it is zero.

Note that $|A^i \cup B^i| - |A^i \cap B^i|$ computes the outer-side nearness between A^i and B^i . and $2|A^i \cap B^i| - |A^i| - |B^i|$ computes the inner-side nearness between A^i and B^i . Hence, $U2$ dissimilarity measure computes the length of non-common parts of interval values with a parameter γ that controls the effect of the inner-side nearness and the outer-side nearness and it can be thought as an approximation to D_c in $U1$. In traditional L_1 and L_2 , $\frac{a^i+b^i}{2}$ and $\frac{c^i+d^i}{2}$ are centroids of A^i and B^i respectively. Modified L_1 and L_2 distances are the natural generalization of traditional L_1 and L_2 distances by taking into account the interval boundaries. The last dissimilarity measure in Table 2 is the Hausdorff distance which was initially defined to compare two sets [5].

2.2. Relations Among Various Measures

In this section, we investigate the non-trivial relationships among various distance measures. The relationships among various measures are summarized in Figure 2.

First, the $U2$ dissimilarity measure can be viewed as an approximation to D_c in $U1$ as it computes the length of non-common parts of interval values with a parameter γ that controls the effect of the inner-side nearness and the outer-side nearness.

Second, different choices of γ yield different distance measures. When $\gamma = 0$, ϕ_{u2} becomes $|A^i \cup B^i| - |A^i \cap B^i|$. It can be easily shown that in this case, with $q = 1$, the object-wise dissimilarity measure $U2$ is equivalent to the city-block distance (i.e., modified L_1 distance). When $\gamma = 0.5$, component-wise dissimilarity $\phi_{u2}(A^i, B^i)$ can be denoted as

$$n = \phi_{u2}(A^i, B^i) = \frac{b^i - a^i}{2} - \frac{d^i - c^i}{2} = \frac{|A^i| - |B^i|}{2}. \quad (1)$$

where $A^i \subseteq B^i$ or $B^i \subseteq A^i$ (one component contains the other component). Similarly, when A^i and B^i intersect, $\phi_{u2}(A^i, B^i)$ becomes

$$m = \phi_{u2}(A^i, B^i) = \frac{a^i + b^i}{2} - \frac{c^i + d^i}{2}, \quad (2)$$

| Name | Object-wise dissimilarity measure | Component-wise dissimilarity measure |
|-------------------|---|---|
| U1 | $d_{u1}(A, B) = \sum_{i=1}^p D(A^i, B^i)$ | $D_{u1}(A^i, B^i) = D_\pi(A^i, B^i) + D_s(A^i, B^i) + D_c(A^i, B^i)$, where $D_\pi(A^i, B^i) = \frac{ a^i - c^i }{ Y^i }$, $D_s(A^i, B^i) = \frac{ A^i - B^i }{\max(b^i, d^i) - \min(a^i, c^i)}$, and $D_c(A^i, B^i) = \frac{ A^i + B^i - 2 \cdot \text{inter}}{\max(b^i, d^i) - \min(a^i, c^i)}$. |
| U2 | $d_{u2}(A, B) = \sqrt[q]{\sum_{i=1}^p [\phi_{u2}(A^i, B^i)]^q}$ | $\phi_{u2}(A^i, B^i) = A^i \cup B^i - A^i \cap B^i + \gamma(2 A^i \cap B^i - A^i - B^i)$. |
| Traditional L_1 | $d_{TraL1}(A, B) = \sum_{i=1}^p D_{TraL1}(A^i, B^i)$ | $D_{TraL1}(A^i, B^i) = \frac{ a^i + b^i }{2} - \frac{c^i + d^i}{2}$. |
| Modified L_1 | $d_{ModL1}(A, B) = \sum_{i=1}^p D_{ModL1}(A^i, B^i)$ | $D_{ModL1}(A^i, B^i) = (a^i - c^i + b^i - d^i)$. |
| Traditional L_2 | $d_{TraL2}(A, B) = \sum_{i=1}^p D_{TraL2}(A^i, B^i)$ | $D_{TraL2}(A^i, B^i) = \left(\frac{ a^i + b^i }{2} - \frac{c^i + d^i}{2}\right)^2$. |
| Modified L_2 | $d_{ModL2}(A, B) = \sum_{i=1}^p D_{ModL2}(A^i, B^i)$ | $D_{ModL2}(A^i, B^i) = ((a^i - c^i)^2 + (b^i - d^i)^2)$. |
| Hausdorff | $d_{Hau}(A, B) = \sum_{i=1}^p D_{Hau}(A^i, B^i)$ | $D_{Hau}(A^i, B^i) = \max(a^i - c^i , b^i - d^i)$. |

Table 2. Dissimilarity measures for interval data. U1 denotes Gowda and Diday's dissimilarity measure [9] and U2 denotes Ichino and Yaguchi's first formulation of a dissimilarity measure [13]. Modified L_1 is also known as city-block distance. $|X|$ denotes the length of the interval X .

In this case, the object dissimilarity measure d_{u2} is thus equal to traditional Minkowski dissimilarity measures for interval data. In particular, when $q = 1$, d_{u2} is equivalent to traditional L_1 distance d_{TraL1} ; when $q = 2$, d_{u2} is equivalent to traditional L_2 distance d_{TraL2} . Hausdorff distance synthesizes two possible situations of U2 in the case of $\gamma = 0.5$ at the same time. It can be represented as

$$d_{Hau}(A, B) = \sum_{i=1}^p \max(|m - n|, |m + n|) = |m| + |n|. \quad (3)$$

Observe that when A^i and B^i intersect, the numerator of D_c can be replaced by $|a^i - c^i| + |b^i - d^i|$. So in this situation D_c is a normalized version of the modified L_1 distance. When $A^i \subseteq B^i$ or $B^i \subseteq A^i$, D_s is also the normalized version of modified L_1 distance. D_π is the normalized Hausdorff distance when $|a^i - c^i| > |b^i - d^i|$. In addition, from formulas of D_{TraL1} , D_{ModL1} , D_{TraL2} , and D_{ModL2} , it is easy to deduce that $D_{ModL1} > 2 \times D_{TraL1}$ and $D_{ModL2} > 2 \times D_{TraL2}$.

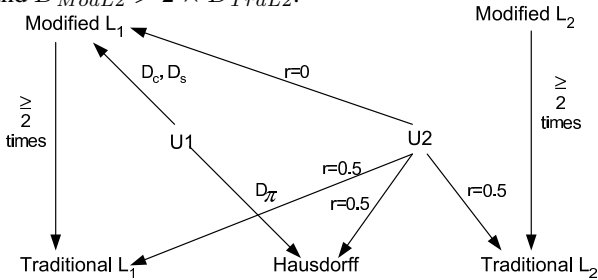


Figure 2. The Relations Among Various Dissimilarity Measures for Interval Data

3. Clustering Interval Data

In this section, we present an alternative optimization procedure for clustering interval data. This procedure is a natural extension of the popular K-means type algorithm.

Note that the following discussion can be easily generalized to datasets having interval data type as well as traditional single-value data type.

3.1. Introduction

Interval data can be represented by a vector of interval values. Let $A = \{A_1, A_2, \dots, A_n\}$ be a set of interval objects. Each object A_i can be represented by a vector $A_i = (A_i^1, A_i^2, \dots, A_i^p)$, where there are p interval values that $A_i^j = [a_i^j, b_i^j]$ and $a_i^j \leq b_i^j$. Suppose we want assign the symbolic objects in A into K clusters $C = (C_1, C_2, \dots, C_K)$, where $C_k, 1 \leq k \leq K$ denotes the k -th cluster. We also use $i \in C_k$ to denote that the i -object is in cluster C_k . The clusters have their corresponding representations or prototypes $G = (G_1, G_2, \dots, G_K)$, where G_k can be also represented as vectors of interval values such that $G_k = (g_k^1, g_k^2, \dots, g_k^p)$, and $g_k^j = [x_k^j, y_k^j]$.

As discussed in Section 1, the clustering problem is determined by four basic components: the (physical) data representation, the distance/dissimilarity measures, the objective criterion, and the optimization procedure. The data representation for interval data is a vector of interval values and the distance measures are studied in Section 2. We now present the objective criterion and describe the optimization procedure.

3.2. Objective Criterion

The goal of clustering is to find the representation for each cluster such that a corresponding criterion $\delta(k)$, defined as the sum of distances between the representation and all objects in that cluster, is minimized. Let the representation of cluster C_k be g_k , and interval objects in cluster C_k be A_i ($i \in C_k$). Based on different distance measures, $\delta(k)$ has

different representations as follows ¹:

1. $\delta(k) = \sum_{i \in C_k} d_{TraL1}(A_i, g_k)$;
2. $\delta(k) = \sum_{i \in C_k} d_{ModL1}(A_i, g_k)$;
3. $\delta(k) = \sum_{i \in C_k} d_{TraL2}(A_i, g_k)$;
4. $\delta(k) = \sum_{i \in C_k} d_{ModL2}(A_i, g_k)$;
5. $\delta(k) = \sum_{i \in C_k} d_{Hau}(A_i, g_k)$.

3.3. Clustering Procedure

The optimization procedure is a variant of K-means type algorithm. The clustering is carried out by an iterative procedure that alternates between identification of the cluster representations to minimize δ and allocation of interval data to the closest cluster.

Proposition 1 *The prototype $G_k = (g_k^1, g_k^2, \dots, g_k^p)$ of cluster C_k that minimizes δ , defined in Section 3.2, is given as follows:*

1. For traditional L_1 distance, $g_k^j = x_k^j$, where x_k^j is the median of the set $\{\frac{a_i^j + b_i^j}{2} | i \in C_k\}$;
2. For modified L_1 distance, $g_k^j = [x_k^j, y_k^j]$, where x_k^j is the median of $\{a_i^j | i \in C_k\}$ and y_k^j is the median of $\{b_i^j | i \in C_k\}$;
3. For traditional L_2 distance, $g_k^j = x_k^j$, where x_k^j is the mean of the set $\{\frac{a_i^j + b_i^j}{2} | i \in C_k\}$;
4. For modified L_2 distance, $g_k^j = [x_k^j, y_k^j]$, where x_k^j is the mean of $\{a_i^j | i \in C_k\}$ and y_k^j is the mean of $\{b_i^j | i \in C_k\}$;
5. For Hausdorff distance, the representation interval data is $g_k^j = [x_k^j, y_k^j]$, where $x_k^j = \text{median}\{\frac{a_i^j + b_i^j}{2} | i \in C_k\} - \text{median}\{\frac{a_i^j - b_i^j}{2} | i \in C_k\}$, and $y_k^j = \text{median}\{\frac{a_i^j + b_i^j}{2} | i \in C_k\} + \text{median}\{\frac{a_i^j - b_i^j}{2} | i \in C_k\}$.

Remark 1 *Note that the representation prototypes for traditional L_1 distance and L_2 distance are shown in [11]. The representation prototypes for modified L_1 distance and L_2 distance are natural generalizations of the traditional ones. The derivation of the representation prototype for Hausdorff distance follows from Equation 3.*

Proposition 1 provides the basis of the **Identification Step** for the clustering procedure, i.e., to identify the representations of clusters to minimize δ .

¹We don't include $U1$ and $U2$ distance measures here as, in practice, they are usually reduced to other measures [3].

Proposition 2 *An interval object A_j is assigned to the cluster m with the prototype which is nearest to that object:*

1. $m = \text{argmin}_{m=1, \dots, K} d_{TraL1}(A_j, g_m)$;
2. $m = \text{argmin}_{m=1, \dots, K} d_{ModL1}(A_j, g_m)$;
3. $m = \text{argmin}_{m=1, \dots, K} d_{TraL2}(A_j, g_m)$;
4. $m = \text{argmin}_{m=1, \dots, K} d_{ModL2}(A_j, g_m)$;
5. $m = \text{argmin}_{m=1, \dots, K} d_{Hau}(A_j, g_m)$.

Proposition 2 establishes the **Allocation Step** of the clustering procedure, i.e., assigning each interval object A_j to the cluster m with the prototype which is nearest to that object.

3.4. Clustering Procedure

Based on the above Proposition 1 and Proposition 2, the clustering procedure can be described as follows: An initial cluster configuration is first generated. This can be done by randomly assigning interval objects into K clusters or by choosing K interval objects as the initial representations of the clusters. Then the clustering procedure iterates between the identification step and allocation step until it converges or some stopping criterion is met.

4. A Two-stage Approach

4.1. Motivation

There exists a natural two-level hierarchical representation for interval data: at the first level, the representative (e.g., the median of the points in the interval) can be used to generate a coarse representation of the interval data; at the second level, a fine representation is given by the interval to show its structure information. Since the traditional distance is obtained by computing the distance between the representatives of two intervals, sometimes we also refer the coarse representation as traditional distance and the fine representation as modified distance. The relationship between the coarse and fine representations motivates a two-stage approach for clustering: at the first stage, the coarse representation is used to obtain a rough partition of the data; at the second stage, the fine representation is employed to refine the partition and generate fine clusterings. The two-stage approach reduces the computation costs while maintaining the clustering quality.

4.2. The Two-Stage Approach

Note that the traditional distance ignores the structure of the interval. However, it is simple and easy to compute. On

the other hand, the modified distance considers the interval structure and needs more computations. Let traditional interval distance such as traditional d_{TraL1} and d_{TraL2} between object x and y be $d_{Tra}(x, y)$, and the corresponding modified interval distance be $d_{Mod}(x, y)$. From Section 2, we know that $d_{Mod}(x, y) > d_{Tra}(x, y)$. This relationship plays an important role in the two-stage approach for clustering interval data.

Assume we want to cluster the interval data such that when two objects x and y are in the same cluster then $d_{Mod}(x, y) \leq \delta$ (Note that the clustering objective here is a little bit different from the objective criterion discussed in Section 3. The relations of the two objective criteria are discussed in [14]). Let's denote the final clustering as $C^M = \{C_1^M, C_2^M, \dots, C_t^M\}$.

Proposition 3 *Suppose we first use traditional dissimilarity measures d_{Tra} to partition the data into a number of partitions such that: if a and b are in the same partition, $d_{Tra}(a, b) \leq \delta$, otherwise $d_{Tra}(a, b) > \delta$. Then, if two objects x and y are in the same cluster of C^M , x and y must be in the same partition obtained using d_{Tra} .*

Remark 2 *Proposition 3 follows from the fact that $d_{Mod}(x, y) > d_{Tra}(x, y)$ as described in Section 2.*

Proposition 3 serves as the foundation for the two-stage approach for clustering interval data. Its key idea is to make clustering both efficient and exact. In the first stage, we utilize traditional dissimilarity measures (as rough and cheap distance measures) to partition the data into a certain number of overlapping partitions (similar to that of [18]). Then modified interval dissimilarity measures (rigorous and expensive) are used to perform clustering with the constraint that the interval objects in the same final cluster should also be in the same partition obtained in the first stage. This could reduce the computation of distances between the objects not in the same partitions [18]. Formally, assume after the first stage, if two objects x and y are not in the same partition, then $d_{Tra}(x, y) > \delta$. Since $d_{Mod}(x, y) > d_{Tra}(x, y)$, we have $d_{Mod}(x, y) > \delta$. On the other hand, if x and y are located in the same partition, $d_{Tra}(x, y) \leq \delta$. Then $d_{Mod}(x, y)$ may be less than or equal to δ , or greater than δ . So when $d_{Mod}(x, y) \leq \delta$, they will be assigned into the same cluster. Otherwise, new cluster will be derived from the current partition. In a word, objects in the same final cluster must be in the same partition.

4.3. Related Work on Multi-level Clustering

The two-stage approach can be thought of as a simple multi-level clustering approach. The work that are closely related to multi-level clustering can be characterized as scale-space clustering [16, 26, 22], annealing for clustering [20, 2, 7, 12] and multi-resolution approaches [6, 21,

23]. The scale-space theory models the blurring effect of lateral retinal interconnection by applying Gaussian filters to a digital image [16]. In a nutshell, scale-space clustering performs a blurring process in which smaller blobs merge into larger ones until the whole image contains only one light blob at a low enough level of resolution. This blurring process is thus leading to a hierarchical clustering process.

Clustering via annealing provides clustering solutions at different scales where the scale is directly related to the temperature parameter which models the rate of distortion. The phase transition in the annealing process indicates the effective number of clusters in the solution which grows as the temperature is lowered.

The multi-resolution approach includes Wavecluster [23] and Multi-resolution instance-learning [6, 21]. Wavecluster applies wavelet transform on the spatial data feature space for detecting arbitrary shape clusters at different scales. The high frequency parts correspond to the cluster boundary while the low frequency parts correspond to the clusters. Finding dense (connected) regions in the transformed space is equivalent to finding the clusters. The multi-resolution instance-learning tries to utilize the KD-tree structure to improve the learning efficiency.

Most of these work aims for spatial clustering where the data sets are images. For the spatial data, clustering at different resolutions/scales usually corresponds to different number of clusters and there is an induced hierarchical procedure associated with the clustering. In our work, we do not assume a nested hierarchical clustering across different resolutions. In fact, there is a natural hierarchy representation for the interval dataset and it is not obtained by Gaussian filtering or annealing process. Second, we have a novel clustering procedure which utilizes the relationship between the coarse and fine representations. The two-stage approach reduces the computation costs while maintaining the clustering quality.

5. Experiments

In this section, we conduct three sets of experiments: i) we present a comprehensive experimental study to compare interval clustering with traditional clustering on real datasets; ii) we apply the interval data clustering to cluster replicated microarray data; iii) we evaluate the two-stage approach. The toolkits developed for interval data clustering can be downloaded from the author's homepage. The software is written in J# in Microsoft .NET framework. It has been tested on Windows XP operating systems.

5.1. Comparing Interval Clustering With Traditional Clustering on Interval Data

5.1.1. DATASETS DESCRIPTION

We use two datasets in our experiments: fish dataset and waveform dataset. The fish dataset is based on a sample of 67 fishes whose species and mercury concentrations in 6 organs have been recorded. It is collected by researchers from LEESA and can be downloaded from <http://www-rocq.inria.fr/sodas/WP6/data/data.html>. The dataset is classified into 4 groups. The waveform dataset is taken from [15]. The problem is based on the three different waveforms h1, h2 and h3 that are the shifted triangular distributions. The interval data representation of the dataset contains 30 individuals where each individual is described by 21 interval attributes. The interval objects 1 to 10 for the wave-1 group, the interval objects 11 to 20 for the wave-2 group, the interval objects 21 to 30 for the wave-3 group.

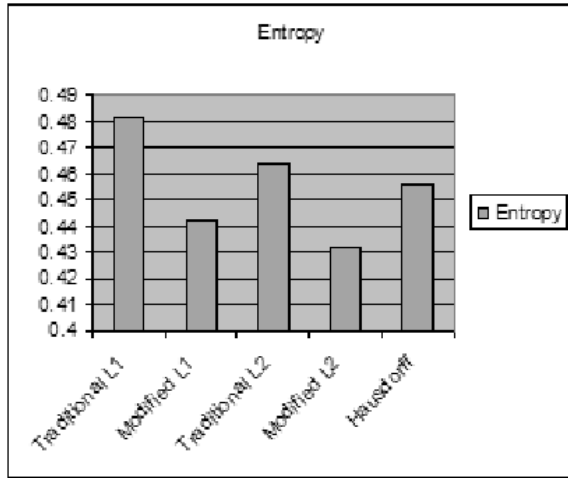


Figure 3. Entropy Comparisons of the Clustering Results for Fish Dataset

5.1.2. EXPERIMENTAL RESULTS

We use entropy which measures how classes distributed on various clusters to evaluate clustering quality of these traditional and extended interval methods [29]. Generally, the smaller the entropy value, the better the clustering quality is. The entropy results for fish and waveform datasets based on different dissimilarity measures are shown in Figure 3 and Figure 4, respectively. It can be observed that, in general, modified interval approaches are better than traditional approaches.

5.2. Clustering Replicated Microarray Data

Clustering techniques has been widely applied in microarray to identify patterns of gene co-expression. To improve the precision of inherently noisy microarray data and to assess the reproducibility of observed patterns, experimen-

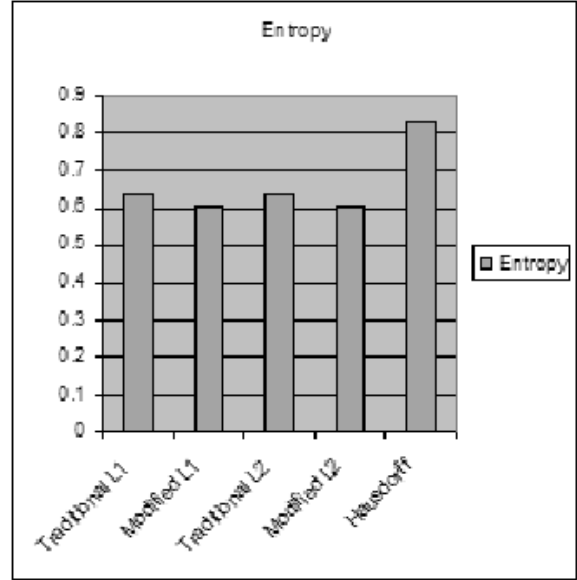


Figure 4. Entropy Comparisons of the Clustering Results for Waveform Dataset

tal replicates are usually performed in microarray measurements. However, the majority of current clustering techniques are not able to accommodate appropriately replicated microarray data [28]. In the section, we investigate the use of interval data clustering for replicated microarray data.

5.2.1. DATASETS DESCRIPTION

Yeast galactose data of Ideker et al. [25] which is expression gene data with repeated measurements was used in our experiments. 205 genes galactose data are described by 20 experiments or expression levels (nine deletions and one wild-type without galactose and raffinose, nine single-gene deletions and one wild-type experiment with galactose and raffinose). Each experiment contains four replicate hybridization expression values based on four different measurements. The expression patterns of these genes reflect four functional categories which is used to calculate our clustering qualities. The missing data values are preprocessed by KNN impute [28]. This dataset can be downloaded from <http://expression.microslu.washington.edu/expression/kayee/medvedovic2003/medvedovicbioinf2003.html>.

5.2.2. INTERVAL DATA REPRESENTATION

Let $A_j = (a_j^1, a_j^2, \dots, a_j^i)$ be the value set of j -th attribute with i repeated values inside, $mean_j$ be the average/mean of set A_j , and δ be the standard deviation. Let min_j , max_j , min'_j , and max'_j be the minimum value, the maximum value, the second minimum value, and the second maximum value of A_j , respectively. There are four ways to transform sets of repeated values to inter-

vals. They are: (i) MinMax: $A_j = [min_j, max_j]$; (ii) MinMax': $A_j = [min'_j, max'_j]$; (iii) MeanVar1: $A_j = [mean_j - \delta, mean_j + \delta]$; and (iv) MeanVar2: $A_j = [mean_j - 2 \times \delta, mean_j + 2 \times \delta]$.

5.2.3. EXPERIMENTAL RESULTS

By summarizing and computing on the raw yeast galactose dataset, we got nine alternative datasets of raw data. Four datasets, denoted as Measure 1, Measure 2, Measure 3 and Measure 4, are derived by extracting a particular value from the four experiment, respectively. One is composed by selecting means of repeated values as representative experiment values, one by selecting MeanVar1, one by MeanVar2, one by MinMax, and one by MinMax'.

To measure the clustering performance, we use entropy, purity and adjusted Rand Index as they are widely used in microarray data analysis. The larger the Purity value, the better the clustering quality is [29]. Adjusted Rand Index is a statistic to assess the clustering quality compared against assigned known classes. The Rand Index is defined as the number of pairs of objects which are both located in the same cluster and the same class, or both in different clusters and different classes, divided by the total number of objects [27]. Adjusted Rand Index which adjusts Rand Index is set between $[0, 1]$ [10]. The higher the Adjusted Rand Index, the better the clustering quality is.

The Adjusted Rand Index, Entropy, Purity of clustering results are shown in Figure 5. MeanVar1, MeanVar2, MinMax, and MinMax' represent raw yeast galactose data by interval data. Mean is the mean value of four repeated experiments. Measure 1, Measure 2, Measure 3, and Measure 4 represent a single experimental value based on four measurements, respectively. We use modified interval k-means to cluster the former four representatives by modified L_2 dissimilarity, and traditional k-means to cluster the rest ones by traditional L_2 .

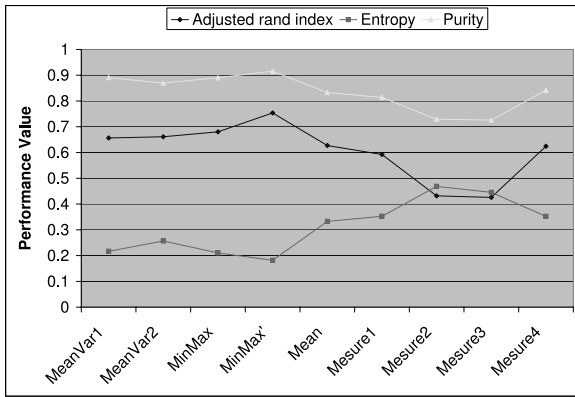


Figure 5. Performance Comparisons of the Clustering Results on Yeast galactose Dataset. Results are obtained by averaging 15 trials. From performance comparisons, we observe that gen-

erally all interval approaches yield better results than traditional approaches. To get a better understanding of the advantages of interval approaches, we take a closer look at some examples from experimental results. For instance, Gene *RPS8A* and Gene *RPL23A* are supposed to be categorized into the same class while Gene *RPS6B* stays in different class according to external knowledge. Using the modified interval approach on MinMax, Gene *RPS8A* and Gene *RPL23A* are perfectly assigned to the same cluster. However, using the traditional approach based on Mean, Gene *RPL23A* and Gene *RPS6B* are grouped together and Gene *RPS3A* is separated from Gene *RPL23A*. In other word, $D_{Inter}(RPS8A, RPL23A) = 18.7492 < D_{Inter}(RPL23A, RPS6B) = 21.5498$, whereas $D_{Tra}(RPL23A, RPS6B) = 9.47405 < D_{Tra}(RPS8A, RPL23A) = 20.3835$.

In general, modified interval dissimilarity measures explicitly consider the structure of interval data and yield better clustering results. Moreover, another advantage of modified interval approaches over traditional ones is that the output cluster prototypes are represented by intervals which are more descriptive than simple quantitative values.

5.3. Two-Stage Approach

In this section, we illustrate the two-stage approach for clustering interval data. Figure 6 illustrate a small example of the two-stage approach on the fish dataset. In the first stage, we partition the dataset based on traditional L_1 distance(rough and cheap distance metric) into three clusters (partitions) as shown in the second bar of Figure 6. In the second stage, B is further divided into into two clusters using modified L_1 distance(rigorous and expensive distance metric) as shown in the third bar of Figure 6.

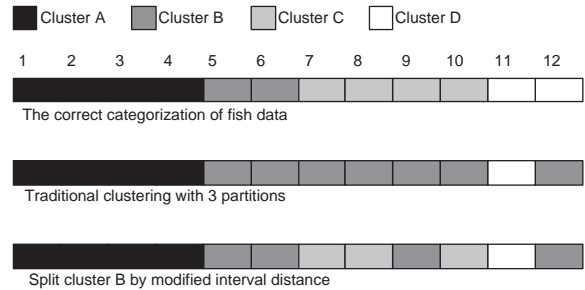


Figure 6. An Example of the Two-stage Approach.

We also apply the two-stage approach on Yeast Galactose Data. We compare the direct clustering approach and two-stage approach to cluster yeast galactose data. Direct clustering approach partitions MinMax' interval objects based on modified L_2 until the desired number of clusters is formed. In this experiment, as the number of known classes is four, we then cluster data into four clusters by direct clustering. Two-stage clustering combines the traditional distance and the modified interval distance to make clus-

tering both effective and efficient. In our experiment, we firstly partition the dataset into three groups based on traditional L_2 , and then modified interval distance is used to divide the cluster with the largest diameter further into two clusters. Table 3 compares the clustering quality of direct clustering and two-stage clustering. Furthermore, the average running time of these two clustering approaches are also compared. As you can see from the Table 3, the clustering performance of the two-stage approach is very close to that of the direct clustering. The performance difference is very small: the entropy of two-stage clustering is 0.204 and the entropy of direct clustering is 0.182; the purity of two-stage clustering is 0.763 and the entropy of direct clustering is 0.754; the ARI of two-stage clustering is 0.900 and the entropy of direct clustering is 0.915. However, the time saving is significant: the running time for two-stage is 74.8ms while the running time for direct clustering is 255.1ms. In summary, the two-stage clustering saves computation time without losing clustering quality.

| Methods | Entropy | ARI | Purity | Time |
|-------------------|---------|-------|--------|-------|
| Two-Stage | 0.204 | 0.763 | 0.900 | 74.8 |
| Direct Clustering | 0.182 | 0.754 | 0.915 | 255.1 |

Table 3. Comparisons of Direct Clustering with Two-Stage Approach. ARI stands for Adjusted Rand Index. Time is reported in millisecond.

6. Conclusion

In this paper, we study the problem of clustering interval data. We discuss various interval data distance measures and present a comparative study on clustering interval data. Our experimental results show that extended interval data clustering achieves better performance than traditional ones, and extended interval approaches excel the traditional ones by taking fully advantages of repeated values. In addition, two-stage approach makes clustering both efficient and exact. Interval data analysis can be extended to summarize standard objects to a single interval object. For example, objects in the same cluster, or the same class, can be combined to form one object. The values which describe the same attribute of these objects can be dealt with as repeated values as in 5.2. The methods for extracting interval objects from classical data include clustering, discrimination, factorial analysis, or decision tree.

References

- [1] G. Biswas, J.B. Weiberg, and D.H. Fisher. ITERATE: A conceptual clustering algorithm for data mining. *IEEE Transactions On Systems, Man and Cybernetics, Part C*, 28:219–230, 1998.
- [2] Marcelo Blatt, Shai Wiseman, and Eytan domany. Data clustering using a model granular magnet. *Neural Computation*, 9(8):1805–1842, 1997.
- [3] H.H. Bock and E. Diday. *Analysis of Symbolic Data*. Analysis of Symbolic Data. Exploratory methods for extracting Statistical Information from Complex Data, Series: Studies in Classification, Data Analysis, and Knowledge Organisation, Vol.15. Springer-Verlag, Berlin, 2000.
- [4] F. De Carvalho, P. Brito, and Bock H. H. Dynamic clustering for interval data based on l_2 distance. Technical report, Cidade Universitaria, 2004.
- [5] M. Chavent and Y. Lechevallier. Dynamical clustering algorithm of interval data: Optimization of an adequacy criterion based on hausdorff distance. In: *Sokolowsky and H.H. Bock Eds., Classification, Clustering and Data Analysis*. Springer, Heidelberg, 1:53–59, 2002.
- [6] K. Deng and A.W. Moore. Multiresolution instance-based learning. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI '95)*, pages 1233–1239, 1995.
- [7] Yiu fai Wong. Clustering data by melting. *Neural computation*, (5):89–104, 1993.
- [8] A.D. Gordon. An iterative relocation algorithm for classifying symbolic data. In W. Gaul et al., editor, *Data Analysis: Scientific modeling and practical application*, pages 17–23. Springer, 2000.
- [9] K.C. Gowda and E. Diday. Symbolic clustering using a new dissimilarity measure. In *Pattern Recognition*, 24:567–578, 1991.
- [10] Milligan GW and Cooper MC. A study of the comparability of external criteria for hierarchical cluster analysis. *Multivar Behav Res*, 21:846–850, 1986.
- [11] John A. Hartigan. *Clustering Algorithms*. Wiley, 1975.
- [12] Thomas Hofmann and Joachim M. Buhmann. Pairwise data clustering by deterministic annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(1):1–14, 1997.
- [13] M. Ichino and H. Yaguchi. Generalized minkowski metrics for mixed feature-type data analysis. *IEEE Transactions on Systems, Man, and Cybernetics*, 24:698–707, 1994.
- [14] Anil K. Jain and Richard C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.
- [15] L. Breiman, J.H. Friedman, R.A. Oshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth, 1984.
- [16] Yee Leung, Jiang she Zhang, and Zong-Ben Xu. Clustering by scale-space filtering. *IEEE Transactions on pattern analysis and machine intelligence*, 22(12):1396–1410, December 2000.
- [17] Kalyani Mali and Sushmita Mitra. Clustering and its validation in a symbolic framework. *Pattern Recognition Letters*, 24:2367–2376, 2003.
- [18] Andrew McCallum, Kamal Nigam, and Lyle H. Ungar. Efficient clustering of high-dimensional data sets with application to reference matching. In *KDD '00: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 169–178. ACM Press, 2000.
- [19] R. Michalski and R.E. Stepp. Automated construction of classifications: conceptual clustering versus numerical taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5:396–410, 1983.
- [20] David Miller and Kenneth Rose. Hierarchical, unsupervised learning with growing via phase transitions. *Neural Computation*, 8(2):425–450, 1996.
- [21] A. Moore. Very fast em-based mixture model clustering using multiresolution kd-trees. In *In Neural Information Processing Systems Conference, 1998.*, 1998.
- [22] Stephen J. Roberts. Parametric and non-parametric unsupervised cluster analysis. *Pattern Recognition*, (4):261–272, 1996.
- [23] Gholamhosein Sheikholeslami, Surojit Chatterjee, and Aidong Zhang. WaveCluster: A multi-resolution clustering approach for very large spatial databases. In *Proc. 24th Int. Conf. Very Large Data Bases, VLDB*, pages 428–439, 1998.
- [24] Renata M.C.R.de Souza and Francisco de A.T.de Carvalho. Clustering of interval data based on city-block distances. *Pattern Recognition Letters* 25, pages 353–365, 2004.
- [25] Ideker T, Thorsson V, Ranish JA, Christmas R, Buhler J, Eng JK, Bumgarner RE, Goodlett DR, Aebersold R, and Hood L. integrated genomic and proteomic analyses of a systemically perturbed metabolic network. *Science* 2001, 292:929–934.
- [26] R. wilson and M.Spann. A new approach to clustering. *Pattern Recognition*, 23(12):1413–1425, 1990.
- [27] Rand WM. Objective criteria for the evaluation of clustering methods. *J Am Stat Assoc*, 66:846–850, 1971.
- [28] Ka Yee Yeung, Mario Medvedovic, and Roger E Bumgarner. Clustering gene-expression data with repeated measurements. *Genome Biology*, 4(5):R34, 2003.
- [29] Ying Zhao and George Karypis. Empirical and theoretical comparisons of selected criterion functions for document clustering. *Machine Learning*, 55(3):311–331, 2004.