Relational Visual Cluster Validity (RVCV)

Yunfei Ding [a,*] Robert F Harrison [a]

[a] *Department of Automatic Control and Systems Engineering, The University of Sheffield, Mappin Street, Sheffield, S1 3JD, UK*

Abstract

The assessment of cluster validity plays a very important role in clustering analysis. Most commonly used clustering validity methods are based on statistical hypothesis testing or finding the best clustering scheme by computing a number of different cluster validity indices. A few visual methods of clustering validity have been produced to display directly the validity of clusters by mapping data into two- or three- dimensional space. However, these methods may lose too much information to correctly estimate the results of clustering algorithms. Although the visual cluster validity (VCV) method of Hathaway and Bezdek can successfully solve this problem, it can only be applied for object data. There are very few validity methods which can be used to analyze the clustering validity of relational data. To tackle this problem, this paper presents a relational visual cluster validity (RVCV) method to assess the validity of clustering relational data by combining the results of the non-Euclidean relational fuzzy c-means (NERFCM) algorithm with a modified VCV display method. RVCV can cluster complete and incomplete relational data and adds to the visual cluster validity theory. Numeric examples using synthetic and real data are presented.

Key words: Clustering, Cluster validity, Relational data, Non-Euclidean fuzzy c-means, Visual cluster validity

1 Introduction

The aim of data classification is to disclose unknown information about a new object or phenomenon. Data classification is composed of supervised and unsupervised processes according to whether the available data is labeled or unlabeled. Clustering is one of the most important unsupervised classification

* Corresponding author. Tel: +44-114-2225679
Email address: Y.Ding@sheffield.ac.uk (Yunfei Ding).

processes that assign objects into clusters whose members are in some way similar to and in others, dissimilar from objects in other clusters. Generally speaking, clustering seeks to identify natural partitions of a finite unlabeled data set.

The typical clustering procedure consists of three steps. Clustering tendency: to decide whether the data actually does contain significant clusters by examining the raw data. Most methods of cluster tendency focus on the true number of clusters in the data (Dubes, 1987) on the basis of statistical hypotheses and parameter estimation e.g. Bootstrap procedure (Sahmer, Vigneau and Qannari, 2005), Bayesian Ying-Yang Model (Guo, Chen and Lyu, 2002) and Hopkins algorithm (Fernández Pierna and Massart, 2000). Another non-parametric approach using Adaptive Resonance Theory (ART) neural networks has been developed to examine clustering tendency without dependence on additional similarity metric and optimization procedure (Massey, 2002). Currently, a method named visual assessment of cluster tendency (VAT) is a very useful and convenient tool to analyze raw data clustering tendency (Bezdek and Hathaway, 2002) and has the advantage of presenting a 2-D image, which is easily evaluated by the user.

The second step is the clustering algorithm itself. Clustering techniques are broadly divided into hierarchical and partitioning methods, based on the properties of the clusters generated. Partitioning algorithms assign samples into some clusters directly while hierarchical algorithms build a nested series of partitions gradually, according to a proximity matrix. Clustering by partitioning is considered in three categories: error-based clustering, mixture density-based clustering and graph-theoretic clustering. The K-means method is the most common and simplest error-based clustering algorithm (MacQueen, 1967; Forgy, 1965). Expectation-maximization (EM) is a general mixture density-based algorithm to cluster an incomplete data set by modelling its density function (Dempster, Laird and Rubin, 1977; McLachlan and Krishnan, 1997). Graph-theoretic clustering is based on the minimum spanning tree (MST) algorithm (Prim, 1957; Friedman and Rafsky, 1979) or an extension of the MST that relates to regions of influence or directed trees (Zahn, 1971). Overall, clustering algorithms have been comprehensively reviewed in two good surveys (Xu and Wunsch, 2005; Jain, Murty and Flynnet, 1999).

The clusters produced by each clustering algorithm may be different. How to evaluate and assess the results of a clustering algorithm is the problem of cluster validation which is concerned with the third step in the process. Effective assessment criteria can be built by applying statistical methods and testing hypotheses. There are three types of assessment measures: external indices, internal indices and relative indices which determine the optimal partitions and quality in different groups (Halkidi, Batistakis and Vazirgiannis, 2002). In order intuitively to understand the results of validity there are a few

visual methods that display validity in two- or three-dimensional graphics. Most of them transform raw data to two-dimensional space by some mapping algorithms such as the Fastmap algorithm or α-mapping and then combine expert knowledge in corresponding areas and allow users to examine and refine clusters (Huang, Cheung and Ng, 2001; Chen and Liu, 2003). Regardless of whether a scalar measure using indices or a visual method of mapping is used, it always leads to the loss of more or less information during scaling or transformation. To combat this problem, Hathaway and Bezdek present a visual approach to assess cluster validity – visual cluster validity (VCV) (Hathaway and Bezdek, 2003). For a sample of size, N, VCV is a visual display of an $N \times N$ intensity image after utilizing and organizing all information produced by "prototype generator" clustering methods as mentioned above, e.g. K-means and fuzzy c-means (FCM) (Bezdek, 1981) methods. VCV gives good results for some synthetic and real data sets for the type of data called "object data" i.e. feature measurements.

Sometimes object data is not available. Instead some relation between samples is known, leading to "relational data" (Hathaway and Bezdek, 1994). If the objects are connected by links, i.e. persistent relationship, we can employ relational data to describe them through pairwise relations, which are similarities or dissimilarities between the objects. Similarity or dissimilarity is usually measured by some proximity associations such as correlation (Pearson's coefficient, Spearman's coefficient and Kendall's coefficient etc.), covariance, distance (Euclidean, Manhattan etc.) or difference (Pearson's dissimilarity, Spearman's dissimilarity and Percent disagreement etc.) functions (Rosen, 1988; Lee, 1999). Sometimes we only have relational data consisting of similarity or dissimilarity values without object data ever having existed, e.g. from expert/opinion or summary statistics. Although both hierarchical and partitional algorithms can all be revised to cluster relational data, partitional methods are focused on in this report. Hathaway, Davenport and Bezdek modify the objective functions to be able to work on relational data by reformulating a family of functionals, membership vectors and relational squared Euclidean distance algorithms instead of using the prototypes of objective clustering algorithms, and present relational duals of HCM and FCM clustering methods – relational hard c-means (RHCM) and relational fuzzy c-means (RFCM) methods (Hathaway,Davenport and Bezdek, 1989). RHCM and RFCM can perform the same job as HCM and FCM only when available relational data satisfies the assumptions of Euclidean distance, that is, there exists a Euclidean realization of the relational data matrix R which is some set of N points in $\mathbb{R}^{N-1}$ (Hathaway and Bezdek, 1994). Without the existence of such points, RHCM and RFCM may be unsuccessful in clustering relational data because there might be "negative distances" occuring during iterative computation. Therefore, Hathaway and Bezdek apply a "spreading" transformation to change non-Euclidean relational data to a matrix that has the properties of the Euclidean relationship to avoid this limitation, and they call

this method the non-Euclidean relational fuzzy c-means (NERFCM) clustering algorithm (Hathaway and Bezdek, 1994). Moreover, they give two simple transformations to convert similarity data into dissimilarity data to make the NERFCM method applicable to any kind of numerical relational data.

VCV doesn't assess the clustering results of relational data because it needs object prototype parameters from prototype generator clustering methods. Specifically, the mean vectors which are not available from the relational algorithms. To solve this problem, we present a relational visual cluster validity (RVCV) method based on VCV. RVCV utilizes two relational prototype parameters – prototype distances and membership values which are available from relational clustering methods. Our method follows the steps of VCV but permits the re-ordering of clusters in the instead stage, which is not possible using the original algorithm. VCV requires knowledge of clusters in feature space which can't be computed directly for relational data. Instead we infer the necessary information from parameters that are available in VCV. This permits the generalization to relational data. In addition, when one or more components of relational data are missing, NERFCM offers a simpler solution than any of the object clustering approaches (Hathaway and Bezdek, 2002). In order to see the cluster validity of the results from NERFCM naturally, RVCV is presented to undertake this task and complete the entire visual cluster validity theory. The remainder of this paper is arranged as follows. The next part introduces the relational clustering method NERFCM, and the existing object clustering validity method, VCV. In the third part, the RVCV approach is described in detail and the specific RVCV algorithm steps are given. The fourth part includes results from four experiments for synthetic and real data and a description of the results. These are then discussed and conclusions are drawn.

2 Theoretical background

2.1 Non-Euclidean Relational Fuzzy Clustering

Prototype clustering algorithms for object data usually minimize an objective function defined as $J_m(U,v) = \sum_{i=1}^{c}\sum_{k=1}^{N} U_{ik}^m \left\|x_k - v_i\right\|^2$ to partition object data $x \in \mathbb{R}^n$, $k = 1, ..., N$, where m is a predefined fuzzification parameter which is usually 2, c is the assumed number of clusters set by the user before applying algorithm, U is the $c \times N$ membership matrix every element of which belongs to a set M_U that satisfies

(a) $0 \leq u_{ij} \leq 1$, for $i = 1, .., c, j = 1, ..., N$;

(b) $u_{1j} + ... + u_{cj} = 1$, for $j = 1, ..., N$;
(c) $u_{i1} + ... + u_{iN} > 0$, for $i = 1, .., c$;

and $v = (v_1, ..., v_c)$ is a matrix of prototype cluster centers. $d_{ki} = \|x_k - v_i\|$ describes the distance between x_k and prototype center v_i.

A relational data matrix $R = [R_{ij}], i, j = 1, ..., N$ measures the relationship between objects i and j. Normally we use S or D to denote similarity or dissimilarity relationships, respectively. Dissimilarity is more often used than similarity in clustering algorithms. If a relational data matrix is called Euclidean, D must satisfy the following four conditions :

(a) Symmetry. $D_{ij} = D_{ji}$, for i=1,...,N, j=1,...,N;
(b) Positivity. $D_{ij} \geq 0$, for i=1,...,N, j=1,...,N;
(c) Reflexivity. $D_{ii} = 0$, for i=1,...,N;
(d) Triangle inequality. $D_{ij} \leq D_{ik} + D_{kj}$, for all i,j and k=1,...N;

where $D_{ij} = \| x_i - x_j \|$ is the Euclidean norm. This kind of relational data can be realized by some set of points $\{x_1, ..., x_N\}$ in $\mathbb{R}^{N-1}$ (Hathaway and Bezdek, 1994).

Compared with prototype clustering algorithms for object data, most relational clustering methods consider squared Euclidean relational data as input data and minimize a corresponding objective function represented by (Hathaway,Davenport and Bezdek, 1989)

$$K_m(U) = \sum_{i=1}^{c} \Big(\sum_{j=1}^{N} \sum_{k=1}^{N} (u_{ij}^m u_{ik}^m r_{jk}^2) \Big) \Big/ 2 \sum_{t=1}^{N} u_{it}^m \Big) \tag{1}$$

where $m \geq 1$, $U \in M_U$, and for $1 \leq k \leq N$, r_{jk}^2 is some numerical relation between objects. In relational clustering algorithms, of the two key modifications used, one is the relational mean vector v_i calculated from U only according to

$$v_i = (u_{i1}^m, u_{i2}^m, ..., u_{in}^m)^T \Big/ \sum_{k=1}^{n} (u_{ik}^m)^m \tag{2}$$

while the other is the distance d_{ik} between object vector and cluster center which is computed from

$$d_{ik}^2 = (Rv_i)_k - (v_i^T R v_i) \big/ 2 \tag{3}$$

in which v_i is the corresponding relational vector. These equations are employed to update U in relational clustering algorithms RHCM or RFCM when

input data is Euclidean relational data. But if the relational data are non-Euclidean their relationships may be out of accord with the triangle inequality. Such data can result in negative d_{ik} when running relational clustering algorithms. To overcome this problem, Hathaway and Bezdek present the NERFCM (Hathaway and Bezdek, 1994) method.

NERFCM is a nonparametric approach to group the data into clusters. As stated, it is developed from RFCM algorithm to avoid the strong limitation that RFCM can only be used for relational data satisfying the Euclidean assumption.

A dissimilarity matrix D satisfying the symmetry, positivity and reflexivity conditions can be used to express non-Euclidean relational data. In NERFCM the main idea is to find a transformation to convert D to a relational matrix $[D_{ij}] = [\|x_i - x_j\|^2]$, each x is in $\mathbb{R}^{N-1}$, which satisfies the Euclidean conditions $(a) - (d)$ and then apply RFCM. The so-called β-spread transformation is competent for the task (Hathaway and Bezdek, 1993). That is: $D = D_0 \rightarrow D_\beta = D + \beta * (M - I)$ where β is a chosen scalar, $I \in \mathbb{R}^{N \times N}$ is the identity matrix and $M \in \mathbb{R}^{N \times N}$ satisfies $M_{ij} = 1$ for $1 \leq i, j \leq N$. Two simple transformations are used to transfer a similarity matrix to a dissimilarity matrix if this is how the data are given. Firstly, set $D_{ii} = 0$ for $1 \leq i \leq N$, then $D_{ij} = (1/S_{ij} - \min_{r \neq t}[1/S_{rt}])$, for $i \neq j$, or $D_{ij} = \max_{r \neq t}[S_{rt}] - S_{ij}$,for $i \neq j$.

The detailed NERFCM algorithm is given by (Hathaway and Bezdek, 1994):

Step1: Given relational data D satisfying (1). Fix c, $2 \leq c \leq N, m > 1$, and initialize $\beta = 0$ and $U^{(0)} \in M_U$. Then for $r = 0, 1, 2, ...$

Step2: Calculate the c mean vectors $v_i = v_i^{(r)}$ using $U = U^{(r)}$ and the equations, for $1 \leq i \leq c$:

$$v_i = (u_{i1}^m, u_{i2}^m, \ldots, u_{in}^m)/(U_{i1}^m + U_{i2}^m + \cdots + U_{in}^m)$$

Step3: Calculate $d_{ik} = (D_\beta v_i)_k - (v_i^T D_\beta v_i)\big/2$, for $1 \leq i \leq c$ and $1 \leq k \leq N$, if $d_{ik} < 0$ for any i and k, then:
calculate $\Delta\beta = \max\{-2 * d_{ik}\big/(\|v_i - e_k\|^2)\}$
update $d_{ik} \leftarrow d_{ik} + (\Delta\beta/2) * \|v_i - e_k\|^2\}$, for $1 \leq i \leq c$ and $1 \leq k \leq N$,
update $\beta \leftarrow \beta + \Delta\beta$

Step4: Update $U^{(r)}$ to $U = U^{(r+1)} \in M_U$ to satisfy, for each $k = 1, ..., n$: if $d_{ik} > 0$ for all i, then:

$$u_{ik} = 1\Big/(d_{ik}/d_{1k} + d_{ik}/d_{2k} + \cdots + d_{ik}/d_{ck})^{1/(m-1)}$$

otherwise: $u_{ik} = 0$ if $d_{ik} > 0$, $u_{ik} \in [0, 1]$, and $(u_{1k} + \cdots + u_{ck}) = 1$

Step5: Check for convergence using any convenient matrix norm $\|\ \|$: if $\left\|U^{(r+1)} - U^{(r)}\right\| \leq \varepsilon$, then stop; otherwise: set $r = r + 1$ and return to step 2.

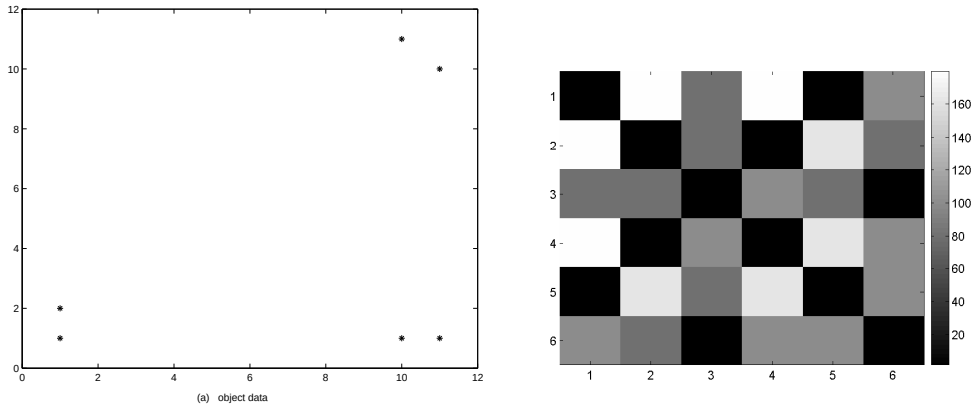

Fig. 1. original data and its corresponding dissimilarity image

When input data is Euclidean relational data, we can still use NERFCM the result of which is the same as that for RFCM and in a such situation $\beta = 0$. Therefore, we choose NERFCM as a representative of relational clustering algorithms because of its universality of practical application which means not only that it can cluster Euclidean and non-Euclidean relational data but also that it can be applied for dissimilarity or similarity relational data.

2.2 Visual cluster validity

To determine whether clusters are useful and valid, VCV (Hathaway and Bezdek, 2003) is reported to estimate the validity of clusters displaying as an intensity image in order to avoid the disadvantage of losing information which exists in most cluster validity indices or visualization methods.

For example it is easy to compute a squared Euclidean relational data matrix D in terms of $[D_{ij}] = [\|x_i - x_j\|^2]$, where x_i is the set of six object points in figure 1(a). The dissimilarity matrix D is

$$D = \begin{bmatrix} 0 & 181 & 81 & 181 & 1 & 100 \\ 181 & 0 & 82 & 2 & 164 & 81 \\ 81 & 82 & 0 & 100 & 82 & 1 \\ 181 & 2 & 100 & 0 & 162 & 101 \\ 1 & 164 & 82 & 162 & 0 & 101 \\ 100 & 81 & 1 & 101 & 101 & 0 \end{bmatrix} \tag{4}$$

and its corresponding image is shown in figure 1(b). Each element of the matrix D corresponds to one intensity or gray level image of pixel (i, j). VCV makes

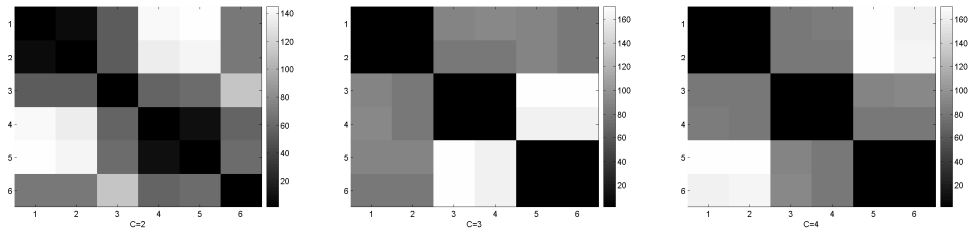

Fig. 2. VCV of six object data with FCM when c=2, 3, 4

use of results obtained from prototype clustering algorithms including cluster centers and membership values and the raw object data itself to execute two steps of reordering data. The first stage is to reorder the cluster centers. The distance between cluster centers is defined as the Euclidean distance between the cluster prototypes (i.e. mean vectors). The first cluster center is chosen arbitrarily. Then one of the remaining centers is chosen as the next reordered cluster $i+1$ based on which one is closest to the last reordered cluster i. The rest may be deduced by analogy until all clusters are rearranged in order of proximity. The second step is to reorder the data in each cluster when the prototype method is fuzzy (Bezdek, 1981). In this situation, each datum x_j is assigned to cluster i in terms of its largest membership and the data in each cluster is arranged corresponding to decreasing membership values in U. Afterwards we can build a new relationship matrix according to the measured inter-datum distance between two object data. The key process is to define a so called pairwise dissimilarity R^*_{ij} which can be calculated through

$$R^*_{ik} = \min_{1 \le j \le c} \{d_{ij} + d_{jk}\} \tag{5}$$

The R^*_{ij} is symmetric and in accord with the triangle inequality although it is not metric. R^*_{ij} is displayed as an intensity image $I(R^*)$.

As an example, we assess the cluster validity of the object data mentioned above. Firstly we select FCM as one generally applicable method of prototype clustering algorithm to look for any natural clusters in the data set. The fuzzification parameter m in FCM is set to 2 and the routine will terminate when the maximum change of all membership values is less than 0.0001. The experiments use different initial numbers of clusters c equal to 2, 3 or 4 (for $c = 2$ and $c = 3$, each cluster has been assigned an equal number of samples; when $c = 4$, two of the four clusters have 2 samples and two have one). The results of VCV are displayed in figure 2.

In VCV images, dark shading means less dissimilarity between two patterns in the data set while light shading shows more dissimilarity and corresponding deep dark diagonal blocks imply high quality clusters in the data. In figure 2, it is apparent that VCV fails to show three diagonal blocks when $c = 2$ but images for $c = 3$ and 4 bring a very obvious visualization of three clusters

along the diagonal line which is consistent with the true number of clusters in the original data. VCV can keep the correct number of clusters even when a large value of c is predefined and it can also be applied to both spherical and linear data (Hathaway and Bezdek, 2003).

3 Relational visual cluster validity (RVCV)

The VCV method can only estimate cluster validity for object data with the help of prototype clustering algorithms because it needs prototype parameters such as cluster centers v, membership U and the object data x. However if we only know the relationship between objects without knowing what exactly the object pattern is we can use NERFCM to partition D. If D is non-Euclidean, NERFCM can find an appropriate β to make D a Euclidean matrix D_β by spread transformation. Therefore all relational parameters produced by NERFCM are based on Euclidean relational data which means we can analyze these parameters through some Euclidean property. Another specially important feature of NERFCM is that it can cluster incomplete relational data using simple triangle inequality-based approximations (TIBA) method to "complete" the relational data matrix (Hathaway and Bezdek, 2002). This is a particular interest of ours. Suppose $\tilde{R}_{ij}$ is a missing value and corresponding index set K_{ij} is defined as: $K_{ij} = \{k | R_{ik}$ and R_{kj} are available$\}$. Minimax TIBA satisfies: $\tilde{R}^U_{ij} = \tilde{R}^U_{ji} = \min\{R_{ik} + R_{jk} | k \in K_{ij}\}$. Maximin TIBA is: $\tilde{R}^L_{ij} = \tilde{R}^L_{ji} = \max\{|R_{ik} - R_{kj}| | k \in K_{ij}\}$. Maximin/minimax average is the average of the sum of the former two TIBA. These three TIBA schemes replace the missing values to produce a complete relational data which can be clustered by NERFCM. Consequently, RVCV based on NERFCM is convenient to analyze the cluster validity of nearly all relational data with or without missing values.

VCV cannot deal with relational data using membership U only because it fails the first step to reorder clusters without information of prototype centers because these are not produced by NERFCM. Our contribution is to present relational visual cluster validity (RVCV) to sort clusters depending only on one critical parameter d_{ik} which measures the distance between object data and cluster centers. Note that this distance can be obtained directly from NERFCM and is required to update the membership matrix as discussed in section 2. Suppose d and U are both known; instead of the true distance between data and cluster we define a pairwise distance as:

$$\hat{d}_{ij} = \min_{1 \le k \le N} (d_{ik} + d_{jk}) \tag{6}$$

for $i, j = 1, ..., c$ and $k = 1, ..., N$, where c is the number of clusters assumed and N the unlabeled patterns. Using $\hat{d}_{ij}$ to approximate inter-cluster distances

satisfies one Euclidean property – triangle inequality.

The RVCV approach is divided into three steps – reordering clusters, reorganizing data and calculating the pairwise dissimilarity matrix. The first step is very important for intensity image display because it is the basis of the second step. Without reordering clusters, the corresponding dark shading of each cluster may be torn into several strips dispersing from the diagonal since the corresponding membership values of each cluster might stand in the position of more than one cluster. To realize the first step, the first cluster is arbitrarily set as the initial reordered cluster. Then every d between the ith and each of the remaining clusters is calculated through equation (6) and the index of the smallest d is chosen as the $(i+1)$th reordered cluster. Next, the $(i+1)$th cluster is removed from the remainder and the remaining clusters are reordered until all are arranged in turn. To illustrate, we can see from figure 3 examples with spherically and linearly distributed samples both of which have clearly chained clusters labeled $c = 1$, $c = 2$ and $c = 3$. d_{ij} for $i, j = 1, ..., c$ is used to describe inter-cluster distances which are shown by dot line in figure 3(a). $d(x_k, c_i)$ for $k = 1, ..., N$ and $i = 1, ..., c$ is used to measure the distance between object data x_k and cluster center c_i which is denoted by the solid line in figure 3(a).

For spherical samples, the first cluster c_1 is determined as the initial reordered cluster. Then we compute pairwise distances d_{12} and d_{13} between cluster c_1 and the other two clusters c_2 and c_3 corresponding to the definition of equation (6). It is easy to find one point x_1 which makes $\hat{d}_{12} = d(x_1, c_1) + d(x_1, c_2)$ be the smallest distance between all points and clusters c_1 and c_2, while x_2 makes $\hat{d}_{13} = d(x_2, c_1) + d(x_2, c_3)$. Comparing $\hat{d}_{12}$ with $\hat{d}_{13}$, we choose c_2 as next reordered cluster after c_1 because $\hat{d}_{12} < \hat{d}_{13}$ and so c_3 is the last one in this case. According to the triangle inequality, $\hat{d}_{12} \geq d_{12}$ and $\hat{d}_{13} \geq d_{13}$ so it is reasonable to find a *minimum* distance between object and clusters instead of the actual inter-cluster distance. It is apparent from figure 3(a) that $\hat{d}_{13}$ is roughly equal to d_{13} because x_2 is almost on the dotted line of d_{13}. For a larger number of samples this is more likely to occur. If the number of samples is large enough to make some points just fall on the inter-cluster lines, the $\hat{d}_{ij}$ is exactly d_{ij}. For linearly distributed samples, $d(x_k, c_i)$ is denoted as the vertical distance in figure 3(b). Through the same reordering method we get $\hat{d}_{12} = d(x_1, c_1) + d(x_1, c_2) < \hat{d}_{13} = d(x_2, c_1) + d(x_2, c_3)$. Here $\hat{d}_{12}$ and $\hat{d}_{13}$ are defined as the inter-cluster distances of linear samples (Hathaway and Bezdek, 2003). Therefore, the final order of clusters is c_1, c_2 and c_3.

The second step of RVCV is to reorder data in every cluster. Because NER-FCM gives a fuzzy membership U, we have to find data that belong to the cluster with the largest membership values and arrange these data in an order of decreasing membership values. So every datum is assigned to a corresponding cluster and similar points are close together to keep a smoother display

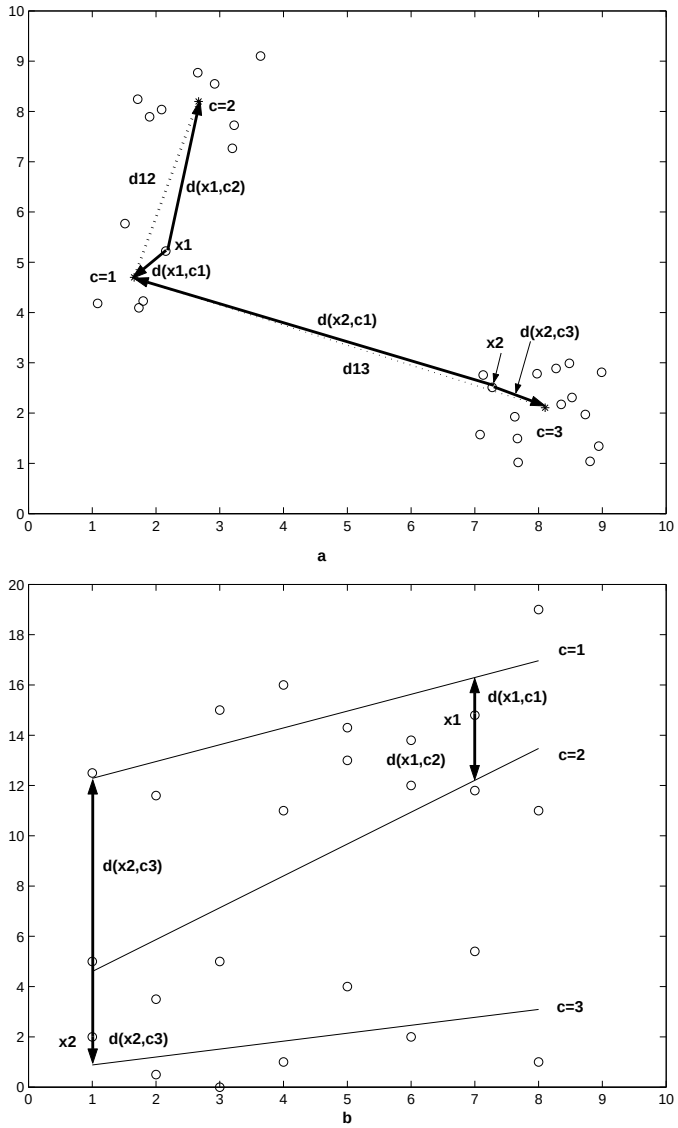

Fig. 3. Distance between object and cluster: (a) spherically distributed samples and (b) linearly distributed samples (after ((Hathaway and Bezdek, 2003)))

in image. It is convenient to get d that is already given in RVCV without expensive computation in VCV.

The detailed RVCV clustering validity display algorithm is:

Step1: Set $N = 1, ..., n$, $C = 1, .., c$, $I(1) = 1$, structure arrays $\underset{k \in C}{S}.(k) = \emptyset$, $J = 2, ..., c$

Step2: For $k = 1, ..., c$, $i = I(k)$, select $(i, j) \in \underset{j \in C, p, q \in \mathbb{N}}{\arg\min}(d_{ip} + d_{jq})$, replace $I \leftarrow I \cup \{j\}$ and $J \leftarrow I - \{j\}$, $\underset{p \in \mathbb{N}}{U}(k+1, p) = \underset{p \in \mathbb{N}}{U}(j, p)$ Next k

Step3: For $p = 1, ..., n$, select $U(j, p) = \underset{i \in C}{\arg\max}\{U(i, p)\}$, for $k = 1, ..., c$, if $j = k$,$S.(k) \leftarrow S.(k) \cup \{j\}$, next k; next p

Step4: For $k = 1, ..., c$, sort $S.(k)$ in decreasing order of $\underset{j \in S.(k)}{U}(k, j)$, next k; $S = \underset{k \in C}{\cup} S.(k)$

Step5: Calculate $R^*_{ij} = \min_{k \in C} \{d_{ik} + d_{jk}\}$ using the ordering $d(i, j) = d(I(k), S(p))$, for $k \in C$, and $p \in N$

Step6: Display the reordered matrix R^* as an intensity image $I(R^*)$.

4 Numerical Examples

4.1 2-D Examples

We begin with two simple 2-D examples – the six points given by equation (4) and a set of three spherically distributed clusters shown in figure 5. Here each cluster includes 50 samples. We calculate their squared Euclidean distance matrix by $d^2_{ij} = \|x_i - x_j\|^2$ for use as relational data input. We examine the effect of different initial c values on RVCV through the resulting intensity images.

The experimental conditions are as follows: the fuzzification constant is $m = 2$, and the stopping criterion for successive partitions is 0.0001. The initial number of clusters c of the first experiment using six points is equal to $2, 3$ and 4 (for $c = 2$ and $c = 3$, each cluster has an equal number of samples; for $c = 4$, two of the four clusters have 2 samples and two have one). For the second experiment of three spherically distributed clusters, c equals to $2, 3, 4$ and 10 by proportionately allocating data to each cluster (for $c = 4$, the numbers in the clusters are $37, 37, 38$ and 38). The results of figure 4 show that when $c = 2$ RVCV fails to give the correct data structure. However a quite explicit diagonal dark shading appears in RVCV images when c is equal to or more than the true number of clusters.

There are three true clusters in each of these two examples and we can see clearly three dark blocks along the diagonal in which data with a strong similarity relationship is assigned to each cluster. The light shadings imply strong dissimilarity between the data in different clusters. RVCV has the same important property as VCV so that even if c is very much larger than its true value, RVCV can still show the right number of clusters in the data. We do not need to worry about which c should be chosen and just increase c until the main dark structure stabilizes.

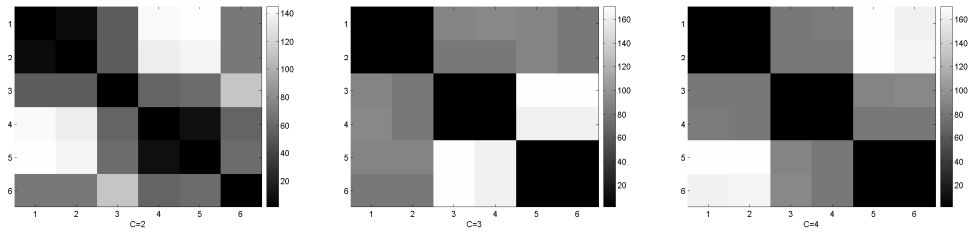

Fig. 4. RVCV of six points data with NERFCM when c=2,3 and 4

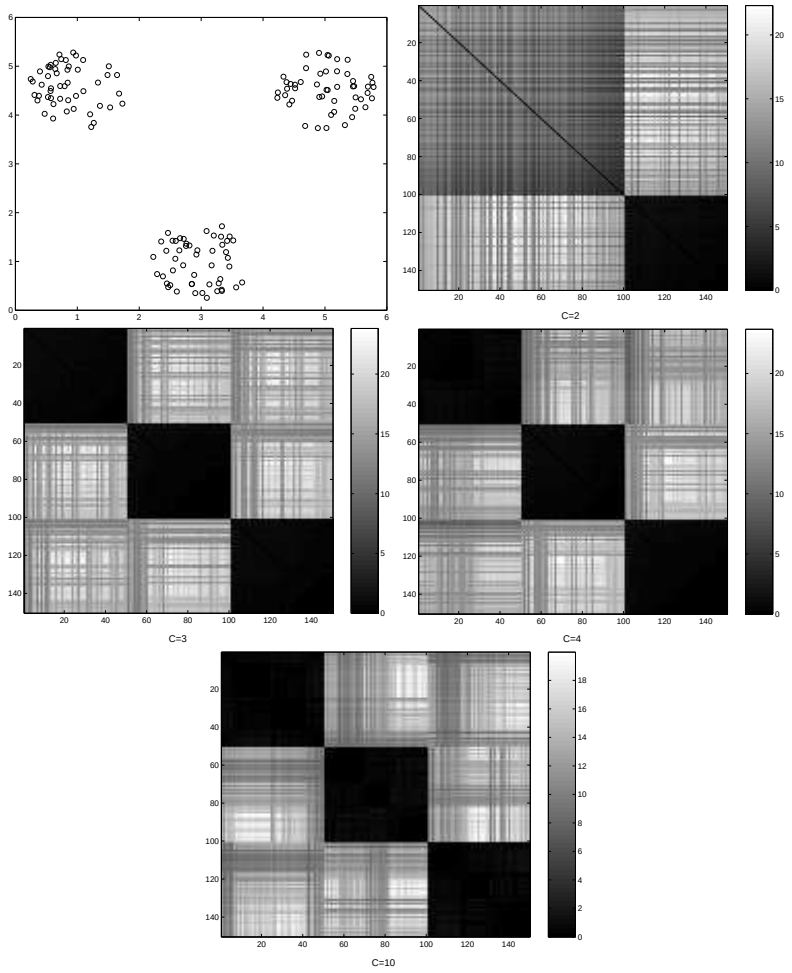

Fig. 5. RVCV of spherically distributed data with NERFCM when c=2,3,4 and 10

4.2 High Dimension Examples

To explain the capability of the proposed method it is necessary to conduct a systematic study using data with known cluster structure. In a third experiment, three separated multivariate normal distributed clusters are respectively generated in 2, 5, 10, 20, and 30 dimensional spaces while keeping the Euclidean distances unchanged between any two cluster centers. The two dimensional data are shown in figure 6. One of the three clusters has 100 sam-

ples and the other two have 200 and 300 samples in each. The samples are randomly arranged in each dataset. All datasets are studied by RVCV with NERFCM when the initial numbers of clusters $c = 3$ and relational Euclidean dissimilarities are used as input. The experimental conditions are the same as before. It is very clear to see from figure 6 that there are three darkly shaded diagonal blocks in all intensity images corresponding to the three clusters in the data. It is easy to identify the size of each block from the axes. The orders of the dark blocks for 5-D, 10-D and 20-D images are the same but different from 2-D and 30-D because the final configuration of blocks depends on the initial choice of sample – this is made at random. No matter which block is first selected, the following block represents the cluster that is the nearest to it. When the data dimension increases to 20 or 30, intensity images still show three clear diagonal blocks.

Next RVCV is applied to a more complicated and overlapped synthetic dataset. In figure 7 three well-separated clusters are generated by different shape and distribution. The dot cluster has 300 samples with a spherical distribution. The square cluster has 200 samples with a square distribution while there are 100 spherically distributed samples in the circle cluster. The center of the circle cluster is fixed in figures 8-10 but the centers of the other two clusters are changed while maintaining their distributions. The squared Euclidean distance matrix of each dataset is used for relational data input to RVCV. The experimental conditions and initial number of clusters c are set by the same method as that of the previous experiment. When $c = 2$ figures 7 and 8 give an ambiguous cluster structure. When $c = 3$ and 4 three dark diagonal blocks are shown quite clearly in figure 7 with the clearest corresponding to the correct number of clusters (= 3). When $c = 10$ the three cluster structure is preserved but is less coherent. In figure 8 the situation is less clear owning to the overlap between two of the clusters. When the initial number of clusters is set to the correct value three dark blocks can be identified but there are strongly shaded off-diagonal strips between two bigger blocks corresponding to the two overlapped clusters. This can be interpreted as suggesting two or three clusters corresponding to one's visual impression. If c increases to 4 and 10 clarity degrades. However the sooth dark blocks when $c = 3$ still can provide a hint for cluster validity.

Because the dot cluster mixes extensively with the circle cluster in figure 9 there are only two clusters in this dataset and there are two clear dark diagonal blocks in each image when $c = 2, 3, 4$ and 10. However, in figure 10 all three clusters of different shapes are strongly overlapped and this reflected in the lack of diagonal structure in the intensity images.

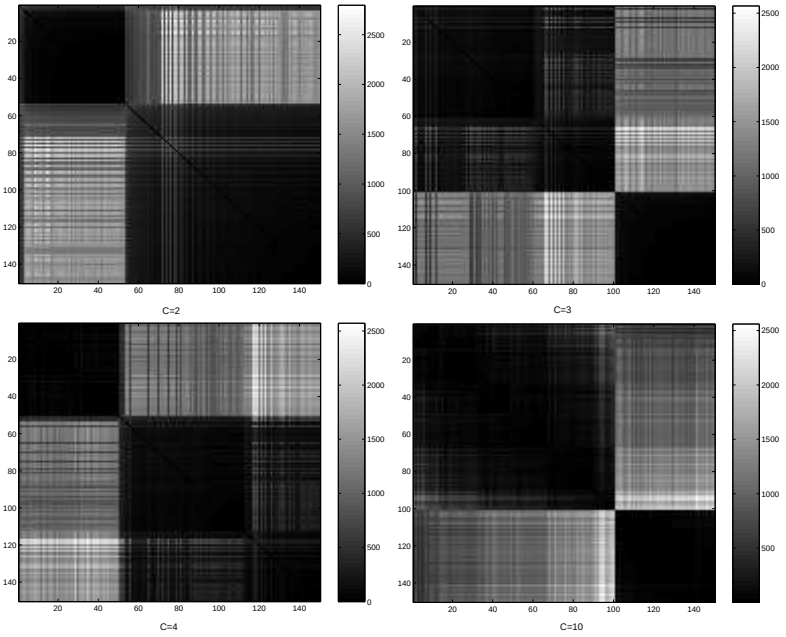

Fig. 6. RVCV of Iris data with NERFCM when c=2,3,4 and 10

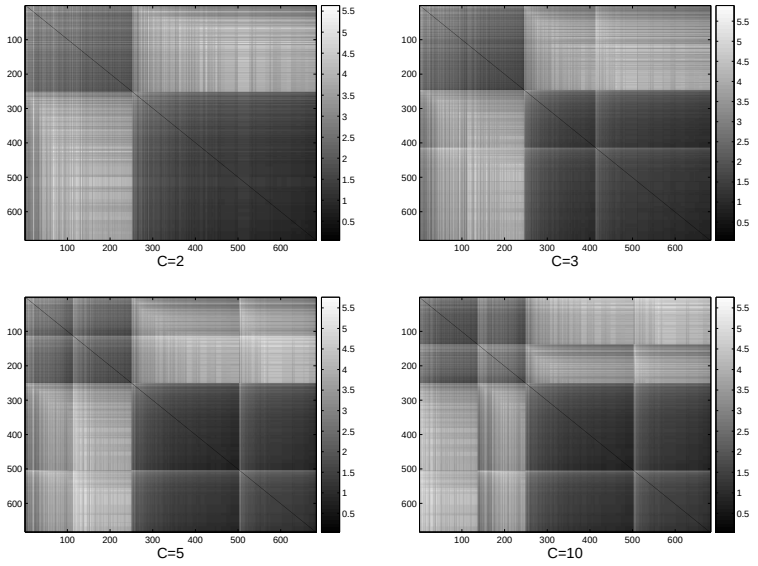

Fig. 7. RVCV of Wisconsin Breast Cancer data with NERFCM when c=2,3,5 and 10

4.3 Real Example

In the following experiments, we employ two real datasets to test our RVCV method. One is the well-known iris data and the other is Wisconsin Breast Cancer Data (WBCD), both of which are available at the UCI Machine Learning Repository (UCI, 1998). Five widely applicable clustering algorithms: K-means, Partition Around Medoids (PAM), Hierarchical, Self-Organizing Map

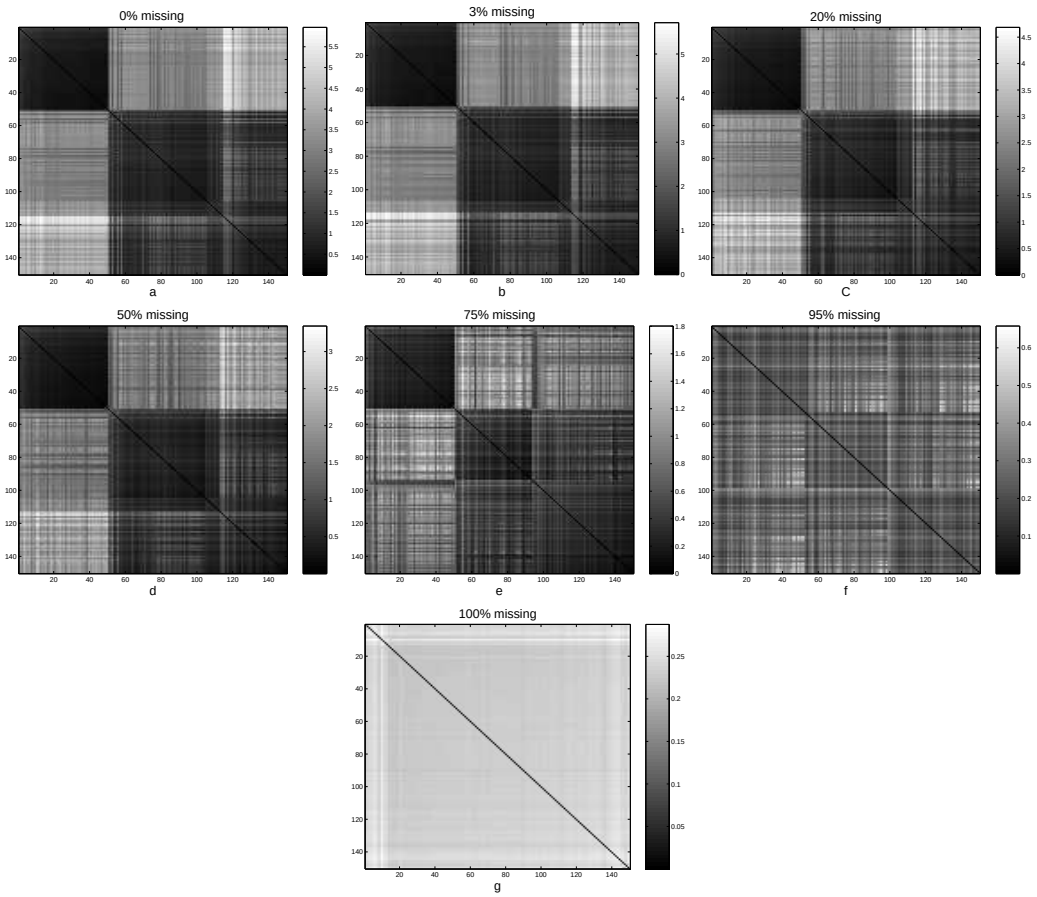

Fig. 8. RVCV of zero replacement with different missing values

(SOM) and Nuero-Gas, are used to cluster these two object datasets. Then nine popular internal cluster validity indices are employed to estimate their performances, comparing with the results of RVCV. These cluster validity methods can be obtained from Cluter validity analysis platform (Wang, 2007).

Iris data is a record of 150 measurements for three Iris plants with four variables each of which includes 50 samples. The input relational data is directly calculated from the iris data by Euclidean distances. We use the same method to initialize clusters c as for the previous experiment. The four images in figure 6 depict the results. However it is seen that there are only two clear cluster blocks in all pictures. It should be noted that, while the data are grouped by three plant varieties, this will not necessarily be reflected in unsupervised clustering, e.g. there may be insufficient features to permit the separation. The different ordering of dark blocks is caused by different initial clusters produced by NERFCM. We can also see vague area in the large dark block with inconspicuous boundaries which implies the cluster may include two or more overlapped clusters in it with very close relationship to each other. The validity indices of the iris data with five clustering algorithms are shown in table 1. The close equivalence of the numbers of 2 and 3 indicates the argument of

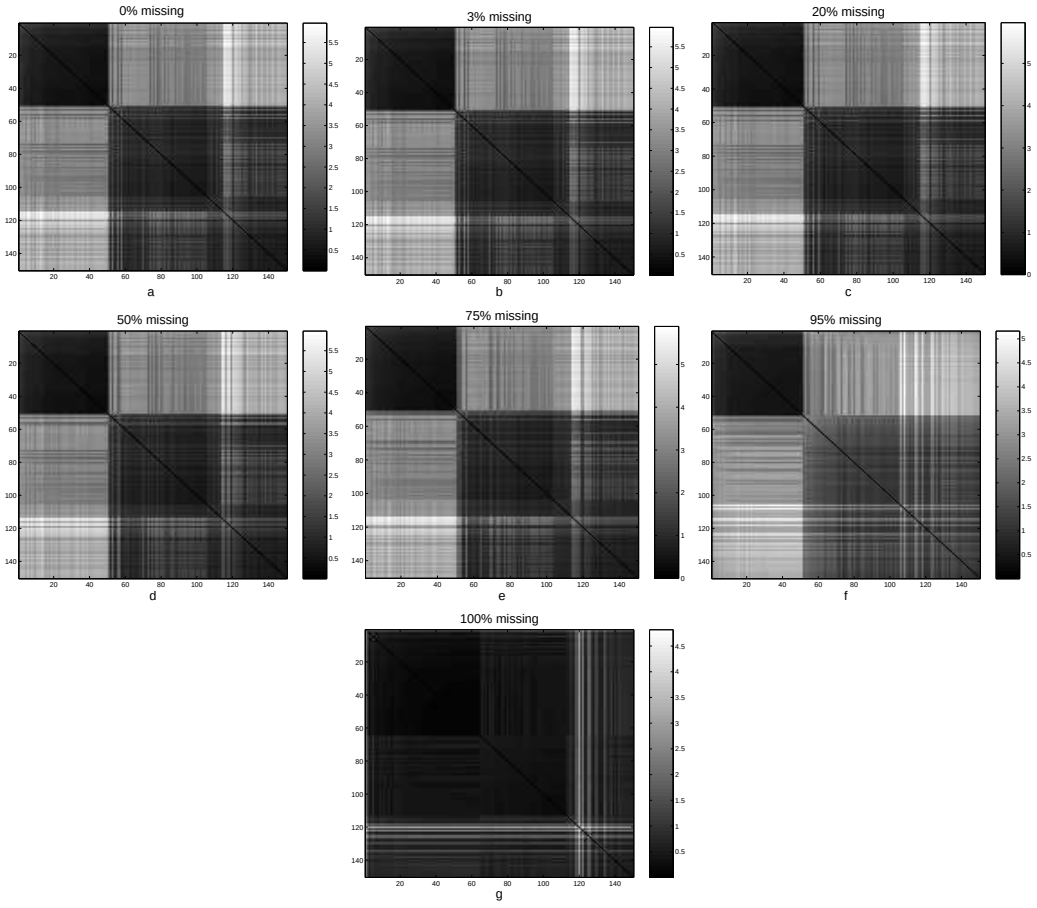

Fig. 9. RVCV of MaxiMin TIBA with different missing values

optimal cluster number between 2 and 3. Proposed RVCV method determines there are 2 clusters and can imply there is one more overlapped cluster which should be noted in one of the clusters.

WBCD contains 683 complete instances with 9 attributes from 2 classes: Benign and Malignant. We perform log-transformation to preprocess the data before computing Euclidean relational data as input. The same initializing clusters method is also applied. Figure 7 shows that the clear two dark blocks in each picture keep stable as c increases from 2 to 10. The two dark blocks prove that, as in most validity indices as shown in table 2, there are two clusters in WBCD. This is consistent with the two true classes in WBCD where benign group includes 444 samples and malignant group 239. The dark block size means the number of samples in one cluster. According to the results of RVCV, 430 benign samples are correctly assigned to the bigger block and 236 malignant to the smaller. The correct rates in each cluster respectively are 96.8% and 98.7%.

Although the cluster validity methods in the tables can give correct optimal cluster number, these methods can only be used in object data. However our

RVCV not only gets the same good performance as them but can also deal with relational data. The result of RVCV shown in a picture is very straightforward and easily understandable to find out the optimal c and how many instances in each cluster, or whether there are any overlapping clusters inside.

Data	n	d	c	Algorithms	S	DB	CH	D	C	K	H	W	R
Iris	150	4	2/3	K-means	2	2	3	2	3	8	8	4	3
				PAM	2	2	3	2	3	8	8	3	3
				Hierarchical	2	10	2	2	5	2	3	3	5
				SOM	3	6	3	3	3	3	1	3	3
				Neuro-Gas	2	2	3	2	3	7	7	5	3
BC	683	9	2	K	2	2	2	2	3	2	8	3	4
				PAM	2	2	2	2	4	4	2	5	4
				Hierarchical	2	10	2	2	7	2	1	2	5
				SOM	2	2	2	2	5	2	5	4	3
				Neuro-Gas	2	2	2	2	8	2	2	8	4

Finally, we analyze the RVCV image of incomplete Iris data using NERFCM and TIBA methods. The complete Euclidean relational matrix R_{ij} is calculated from the iris data, where i and j are any two samples of the iris data. Then we randomly delete some off-diagonal values $\tilde{R}_{ij}$ in pairs to produce an incomplete relational matrix. It should be noted that each value in the tridiagonal part of R_{ij} can not be deleted in order to keep the relationship in the subsets of the data (Hathaway and Bezdek, 2002). That is, if the remaining $n^2 - 3n + 2$ values are deleted we consider that 100% values are missing, and if 0% value is missing the complete relational data is used in experiment. Two methods are used to impute the missing values. One is using zero instead of missing data directly and another is the maximin TIBA method which is the recommended approximation of the three TIBA strategies when computation is based on R (Hathaway and Bezdek, 2002). The new estimated complete matrix $\hat{R}_{ij}$ is clustered by K-means method with real initial clusters $k = 3$ to produce hard partitions which are used as the input initial membership U_0 of the NERFCM approach. Figures 12(a-g) show the RVCV images of zero replacement with different missing values and figure 13(a-g) are maximin TIBA. When 0% is missing, both RVCV images of zero replacement and maximin TIBA mean clustering the complete iris relational data and the results are the same. It is easy to see that as the missing values increase, both images are worse. Unsurprisingly, for over 95% cases missing, we can not get any useful information from these images. Comparing the RVCV images of zero replace-

ment with that of maximin TIBA under the same missing values, the diagonal blocks of the latter are clearer than those of the former which are blurred by some light strips. These light strips caused by the large error between real value and zero show how much dissimilarity exists between the two samples. In this situation, the RVCV images of maximin TIBA can show a more true relationship than the fuzzy images using zero replacement. Therefore, RVCV is also an easy and valid method to assess the cluster validity of data set with missing values.

5 Discussion and Conclusion

A visual clustering validity approach for relational data (RVCV) is described based on the visual cluster validity method (Hathaway and Bezdek, 2003). We still use the main idea of VCV to reorder inter-cluster distances and inter-datum distances and display a pairwise dissimilarity R in an intensity image. NERFCM is selected as a relational clustering algorithm to partition our original relational data. We make use of one parameter d and membership U produced by NERFCM. At first, d is used to reorder inter-cluster distances which are defined through equation (6) and then used to calculate a pairwise dissimilarity matrix by equation (5) to save execution time. Our method provides a visual validity results for relational clustering algorithms corresponding to the object data method. We demonstrate through several synthetic and real numeric experiments the effectiveness of our method. Overlapped relational data, incomplete relational data and high dimensional data are tested to demonstrate that RVCV is a valid method for estimating relational cluster validity. RVCV only needs U and d as input parameters which are easy to obtain from relational algorithms. The RVCV method fills a gap in the set of current VCV methods and both methods provide an integrated visual cluster validity approach to identify both object and complete or incomplete relational data. The calculation complexity of RVCV is dominated by the sorting algorithm. This is achievable in $O(n \log n)$ operations and must be done c times. Hence, for small c, runtime will be manageable even for quite large n.

References

Bezdek,J.C., 1981. Pattern Recognition with Fuzzy Objective Function Algoritms. Plenum Press, New York.

Bezdek, J.C and Hathaway, R.J., 2002. VAT: A tool for visual assessment of (cluster) tendency. In: Proc. IJCNN 2002. IEEE Press, Piscataway, NJ, pp: 2225–2230.

Bezdek J. and Pal N., "Some new indexes of cluster validity", IEEE Trans. on Syst., Man and Cybernetics, Part B. Vol.28, pp: 301–315.

Chen, K. and Liu, L., 2003. Validation and refining clusters via visual renderingm. Proc. of Intl. Conf. on Data Mining (ICDM03), Melborne, FL.

Dempster, A.P., Laird, N.M., and Rubin, D.B., 1977. Maximum likelihood estimation from incomplete-data via the EM algorithm (with discussion). Journal of the Royal Statistical Society, Series B. Vol. 39, pp: 1–38.

Dubes, R.C, 1987. How many clusters are best? – an experiment. Pattern Recognition. Vol. 20(No. 6), pp: 645–663.

Fernández Pierna, J.A. and Massart, D.L., 2000. Improved algorithm for clustering tendency. Analytica Chimica Acta 408, pp: 13–20.

Friedman, J.H. and Rafsky, L.C., 1979. Multivariate generalizations of the Wald-Wolfowitz and Smirnov Two-Sample tests. The Annals of Stattsttics. Vol.7, pp: 697–717.

Forgy, E.W., 1965. Cluster analysis of multivariate data: Efficiency versus interpretability of classifications. Biometrics. Vol. 21, No. 3, pp: 768–780.

Guo, P., Chen, C.L.P. and Lyu, M.R., 2002. Cluster Number Selection for a Small Set of Samples Using the Bayesian Ying-Yang Model. IEEE Transactions on Neural Networks. Vol. 13, No.3, pp: 757–763.

Halkidi, M., Batistakis, Y., and Vazirgiannis, M., 2002. Cluster validity methods: Part i, SIGMOD Record 31(2) 31(2): 40-45 and Part ii, SIGMOD Record 31(3), pp: 19–27.

Hathaway,R.J. and Bezdek,J.C., 1994. NERF c-MEANS: Non-Euclidean relational fuzzy clustering. Pattern Recognition. Vol. 27(No. 3), pp: 429–437.

Hathaway,R.J. and Bezdek,J.C., 2002. Clustering incomplete relational data using the non-Euclidean relational fuzzy c-means algorithm. Pattern Recognition Letters 23, pp: 151–160.

Hathaway,R.J. and Bezdek,J.C., 2003. Visual cluster validity for prototype generator clustering models. Pattern Recognition Letters 24, pp: 1565–1569.

Hathaway, R.J., Davenport, J. W. and Bezdek, J.C, 1989. Relational duals of hard and fuzzy c-means clustering algorithms. Pattern Recognition. Vol.22, No.2, pp: 205–212.

Huang, Z., Cheung, D.W. and Ng, M.K., 2001. An empirical study on the visual cluster validation method with Fastmap. Proceedings of the 7th International Conference on Database Systems for Advanced Applications, pp: 84–91.

Jain, A.K., Murty, M. N. and Flynnet, P.J., 1999. Data clustering: a review. TACM Computing Surveys. Vol.31, No.3, pp: 264–323.

Krishnapuram, R. and Keller, J., 1993. A possibilistic approach to clustering. IEEE Transactions on Fuzzy Systems. Vol.1, pp:98–110.

Lee, L., 1999. Measures of distributional similarity. In 37th Annual Meeting of the Association for Computational Linguistics (ACL), pp: 25–32.

MacQueen, J. B., 1967. Some Methods for classification and Analysis of Multivariate Observations, Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press.

Vol.1, pp:281–297.

Massey, L., 2002. Determination of clustering tendency with ART Neural Networks. Proceedings of 4th Intl. Conf. on Recent Advances in soft computing.

McLachlan, G.J. and Krishnan, T., 1997. The EM algorithm and extensions. New York: John Wiley.

Prim, R.C., 1957. Shortest connection networks and some generalizations. Bell System Technical Journal. Vol.36, pp:1389–1401.

Rosen, B.R., 1988. From fossils to earth history: applied historical biogeography. In: Myers, A.A. & Giller, P.S. (eds.). Chapman & Hall, New York. Analytical biogeography, pp: 437–481.

Sahmer, K., Vigneau, E. and Qannari, M. E., 2005. A cluster approach to analyze preference data: choice of the number of clusters. Food Quality and Preference.

UCI Benchmark repository: A huge collection of artificial and real world data sets, 1998. Available at http://www.ics.uci.edu/ mlearn.

Wang, K. J., Cluster Validation Toolbox CVAP, 2007. Available at http://www.mathworks.com/matlabcentral.

Xu, R and Wunsch, D., 2005. Survey of clustering algorithm. Transactions on neural networks. Vol.16, No.3, pp: 645–678.

Zahn, C.T., 1971. Graph-theoretic methods for detecting and describing gestalt clusters. IEEE Trans. Comp. Vol.20, pp: 68–86.