MMeR: an algorithm for clustering heterogeneous data using rough set theory

Prakash Kumar and B.K. Tripathy*

School of Computing Sciences, VIT University,
Vellore-632 014, Tamilnadu, India
E-mail: manohar_vit@yahoo.co.in
E-mail: tripathybk@rediffmail.com
*Corresponding author

Abstract: Several cluster analysis techniques have been developed so far to group objects having similar characteristics. Clustering of categorical data is more challenging than that of numerical data. Most of the early cluster analysis techniques face problems due to the fact that much of the data contained in today's databases is categorical in nature. This necessitated the development of some algorithms for clustering categorical data. Uncertainty is an integral part of databases. The algorithms put forth either lack the capability to handle uncertainty or do not reach a steady state in a few iterations, which gives rise to the stability issues. Recently, an algorithm, termed MMR was proposed (Parmar et al., 2007), which uses the rough set theory to deal with the above problems in clustering categorical data. In this paper, we modified MMR to develop an improved algorithm and call it MMeR. This takes care of both numerical and categorical data simultaneously besides handling uncertainty. Also, this new algorithm provides much better performance than most of the existing algorithms including MMR. Some well known data sets are taken to test and illustrate the superiority of MMeR over most of the existing algorithms.

Keywords: clustering; min-min-roughness; MMR; MMeR; rough sets; heterogeneous data; purity.

Reference to this paper should be made as follows: Kumar, P. and Tripathy, B.K. (2009) 'MMeR: an algorithm for clustering heterogeneous data using rough set theory', *Int. J. Rapid Manufacturing*, Vol. 1, No. 2, pp.189–207.

Biographical notes: Prakash Kumar received his BTech from VIT University, Vellore, India, in 2009. At present he is working as an Associate Lecturer in the Col. D.S. Raju Polytechnic, A.P., India. The present paper is an extension work in connection with his final semester project under the supervision of Dr. B.K. Tripathy. He has published four papers so far and his fields of interest include rough set theory, clustering techniques and social network analysis.

B.K. Tripathy is working as a Senior Professor in the School of Computing Sciences of VIT University, Vellore, Tamil Nadu, India. He has supervised eight PhDs so far and has published over 75 technical articles in various international/national journals/proceedings of national/international conferences. He is in the Editorial Board/Review Committee of several international journals. Recently, he has contributed two chapters for a Springer international edited research volume on knowledge representation. His current fields of interest include fuzzy sets and systems, knowledge representation, rough set theory and applications, granular computing, soft computing, data mining, social network analysis and data clustering.

1 Introduction

The basic objective of cluster analysis is to discover natural groupings of objects; that is, forming groups of data having similar characteristics. Cluster analysis is used in data mining tasks such as unsupervised classification, data summation and data segmentation. Segmentation is the process of decomposing large heterogeneous data sets into smaller homogeneous subsets that can be easily managed, separately modelled and analysed. Cluster analysis techniques have been used in many application areas such as manufacturing, medicine, nuclear science, radar scanning and research and development planning (see for instance Jiang et al., 2004). Most of the early clustering methods are applicable only to data having numerical values for attributes. Unlike numerical data, categorical data may have multi-valued attributes. A number of algorithms for clustering categorical data have been proposed (Andritsos et al., 2003; Ganti et al., 1999; Gibson et al., 2000; Guha et al., 2000; He et al., 2004; He et al., 2002; Huang, 1998; Kim et al., 2004; Zhang et al., 2000). These methods are not designed to handle uncertainty in the clustering process. But, this is an important issue in many real world applications where there is often no sharp boundary between clusters (Halkidi et al., 2001). Recent algorithms in the area of applying fuzzy sets in clustering categorical data have been proposed by Huang (1998) and Kim et al. (2004). However, these algorithms require multiple runs to establish the stability needed to obtain a satisfactory value for one parameter used to control the membership value of fuzziness. Therefore, there was a need for a robust clustering algorithm that can handle uncertainty in the process of clustering categorical data. Rough set theory, since its inception in the year 1982 by Pawlak, has proved itself to be a natural tool to model and analyse uncertainty in data. Most importantly, in rough set theory uncertainty is expressed in term of certain concepts and as such uncertainty is inherent in the definition of rough sets, unlike other approaches and models available in the literature for this purpose (Pawlak, 1991; Pawlak and Skowron, 2007a, 2007b, 2007c). Such a clustering algorithm, which uses rough set theory to manage impreciseness, was put forth in Lingras et al. (2003) and studied further in Lingras and West (2004). An algorithm, called min-min-roughness (MMR) proposed by Parmar et al. (2007), has the potential for clustering categorical attributes in data mining. It is capable of handling the uncertainty in the clustering process. Unlike other algorithms, MMR requires only one input, the number of clusters, and has been tested to be stable. However, there are certain points in which MMR needs improvement. Our proposed algorithm MMeR improves the efficiency and applicability of MMR and has the following important features:

1 like MMR, it provides a user the ability to handle uncertainty in data in the clustering process

2 like MMR, the user is able to obtain stable results with only one input; the number of clusters

3 like MMR, it is capable of handling large data sets

4 it has better efficiency than MMR and other algorithms in different features

5 we provide a technique so that it can be used for heterogeneous data sets.

In the following sections, we present the new algorithm, analyse its superiority and provide comparison with other algorithms with suitable test data to solve the purpose.

1.1 Initial approaches

In this section, we present an overview of all earlier methods available in the literature to cluster categorical data. Dempster et al. (1977) presents a partitional clustering method, called the expectation-maximisation (EM) algorithm. EM first randomly assigns different probabilities to each class or category, for each cluster. These probabilities are then successively adjusted to maximise the likelihood of the data given the specified number of clusters. Since the EM algorithm computes the classification probabilities, each observation belongs to each cluster with a certain probability. The actual assignment of observations to a cluster is determined based on the largest classification probability. After a large number of iterations, EM terminates at a locally optimal solution. K-modes (He et al., 2002), extending K-means introduces a new dissimilarity measure for categorical data. The dissimilarity measure between two objects is calculated as the number of attribute values in which they differ. This is a generalisation of the concept of hamming distance. The K-modes algorithm introduced modes for means to reduce the cost function. K-modes must be run multiple times with different starting values of modes to test the stability of the clustering solution. Huang (1998) also proposes the K-prototypes algorithm, which allows clustering of objects described by a combination of numeric and categorical data.

1.2 Handling uncertainty

One of the first algorithms to deal with uncertainty is fuzzy K-means (Ruspini, 1969). In this algorithm, each pattern or object is allowed to have membership functions to all clusters rather than having a distinct membership to exactly one cluster. Krishnapuram and Keller (1993) propose a probabilistic approach to clustering in which the membership of a feature vector in a class has nothing to do with its membership in other classes and modified clustering methods are used to generate membership distributions. Krishnapuram et al. (1995) have presented several fuzzy and probabilistic algorithms to detect linear and quadratic shell clusters. It may be noted that the initial work in handling uncertainty was based on numerical data. Huang (1998) proposes a fuzzy K-modes algorithm with a new procedure to generate the fuzzy partition matrix from categorical data within the framework of the fuzzy K-means algorithm. This method finds fuzzy cluster modes when a simple matching dissimilarity measure is used for categorical objects. By assigning confidence to objects in different clusters, the core and boundary objects of the clusters can be decided. This helps in providing more useful information for dealing with boundary objects. More recently, Kim et al. (2004) have extended the fuzzy K-modes algorithm by using fuzzy centroids to represent the clusters of categorical data instead of the hard-type centroids used in the fuzzy K-modes algorithm. The use of fuzzy centroids makes it possible to fully exploit the power of fuzzy sets in representing the uncertainty in the classification of categorical data. However, fuzzy K-modes and fuzzy centroids algorithms suffer from the same problem as K-modes that they require multiple runs with different starting values of modes to test the stability of the clustering solution. In addition, these algorithms have to adjust one control parameter for membership fuzziness to obtain better solutions. This necessitates the effort for multiple runs of these algorithms to determine an acceptable value of this parameter. Therefore, there is a need for a categorical data clustering method, having the ability to handle uncertainty in the clustering process while providing stable results.

1.3 Rough set theory

Most of our traditional tools for formal modelling, reasoning and computing are deterministic and precise in character. Real situations are very often not deterministic and they cannot be described precisely. For a complete description of a real system often one would require by far more detailed data than a human being could ever recognise simultaneously, process and understand. This observation led to the extension of the basic concept of sets so as to model imprecise data which can enhance their modelling power. The fundamental concept of sets has been extended in many directions in the recent past. The notion of fuzzy sets, introduced by Zadeh (1965) deals with the approximate membership and the notion of rough sets, introduced by Pawlak (1982) captures indiscernibility of the elements in a set. These two theories have been found to complement each other instead of being rivals. The idea of rough set consists of approximation of a set by a pair of sets, called the lower and upper approximations of the set. The basic assumption in rough set is that, knowledge depends upon the classification capabilities of human beings. Since every classification (or partition) of a universe and the concept of equivalence relation are interchangeable notions, the definition of rough sets depends upon equivalence relations as its mathematical foundations (Pawlak and Skowron, 2007a, 2007b, 2007c).

Let U $(\neq \phi)$ be a finite set of objects, called the universe and R be an equivalence relation over U. By U / R, we denote the family of all equivalence classes of R (or classification of U) referred to as *categories* or *concepts* of R and $[x]_R$ denotes a category in R containing an element $x \in U$. By a Knowledge base, we understand a relation system $K = (U, \Re)$, where U is as above and $\Re$ is a family of equivalence relations over U.

For any subset $\mathrm{P}(\neq \phi) \subseteq \Re$, the intersection of all equivalence relations in P is denoted by *IND*(P) and is called the *indiscernibility relation over* P. The equivalence classes of *IND*(P) are called P-*basic knowledge* about U in K. For any $Q \in \Re$, Q is called an Q-elementary knowledge about U in K and equivalence classes of Q are called Q-*elementary concepts of knowledge* $\Re$. The family of P-basic categories for all $\phi \neq \mathrm{P} \subseteq \Re$ will be called the *family of basic categories* in knowledge base K. By *IND*(K), we denote the family of all equivalence relations defined in K. Symbolically, $IND(K) = \{IND(\mathrm{P}) : \phi \neq \mathrm{P} \subseteq \Re\}$.

For any $X \subseteq U$ and an equivalence relation $R \in IND(K)$, we associate two subsets, $\underline{R}X = \bigcup\{Y \in U / R : Y \subseteq X\}$ and $\overline{R}X = \bigcup\{Y \in U / R : Y \cap X \neq \phi\}$, called the *R-lower* and *R-upper approximations* of X respectively. The *R-boundary* of X is denoted by $BN_R(X)$ and is given by $BN_R(X) = \overline{R}X - \underline{R}X$. The elements of $\underline{R}X$ are those elements of U which can be certainly classified as elements of X with the knowledge of R and $\overline{R}X$ is the set of elements of X which can be possibly classified as elements of X employing knowledge of R. The borderline region is the undecidable area of the universe. We say X is *rough* with respect to R if and only if $\underline{R}X \neq \overline{R}X$, equivalently $BN_R(X) \neq \phi$. X is said to be *R-definable* if and only if $\underline{R}X = \overline{R}X$, or $BN_R(X) = \phi$. So, a set is rough with respect to R if and only if it is not R-definable.

Unlike fuzzy set based approaches, rough sets have no requirement on domain expertise to assign the fuzzy membership. Still, it may provide satisfactory results for

rough clustering (Voges et al., 2002). Keeping this point in view, we proceeded further in developing an algorithm, which can handle uncertainty as well as heterogeneous data, by the way generalising the existing algorithms in this direction, which we named MMeR.

2 Definitions, notations and the algorithm

In this section, we shall state the notations and definitions to be used throughout the paper and put forth the clustering algorithm, which we term as MMeR.

2.1 Nomenclature

We denote by U the universe or the set of all objects and X as a subset of U. Let A be the set of all the attributes of objects in U and B be a non-empty subset of A.

Definition 2.1.1 – Indiscernibility relation

Given two objects x, y $\in$ U we say that x and y are indiscernible by the set of attributes B in A if and only if a(x) = a(y) for every a $\in$ B. This relation is an equivalence relation on U and decomposes it into disjoint equivalence classes. We denote it by Ind (B). For any $x_i \in U$, the set of objects x_j having the same values as x_i for the set of attributes in B consists of the equivalence class of x_i, with respect to Ind (B) and is denoted by $[x_i]_{Ind(B)}$.

It is also known as the *elementary set* of x_i with respect to B.

Example 1

We shall consider the following example, which is a characterisation of various animals in terms of size, animality and colour.

Table 1 Example data

Animals	*Size*	*Animality*	*Colour*
A1	small	bear	black
A2	medium	bear	black
A3	large	dog	brown
A4	small	cat	black
A5	medium	horse	black
A6	large	horse	black
A7	large	horse	brown

$$\text{Equivalence classes of the attribute set } \{\text{size, animality}\} = \begin{Bmatrix} \{A1\},\{A2\},\{A3\}, \\ \{A4\},\{A5\},\{A6,A7\} \end{Bmatrix}.$$

$$\text{Equivalence classes of the attribute set } \{\text{animality, colour}\} = \begin{Bmatrix} \{A1,A2\},\{A3\}, \\ \{A4\},\{A5,A6,\{A7\} \end{Bmatrix}.$$

Definition 2.1.2 – Approximations

Given the set of attributes B in A, set of objects X in U, the *lower approximation* of X is defined as the union of all the elementary sets which are contained in X. That is,

$$\underline{X_B} = \bigcup\{x_i / [x_i]_{Ind(B)} \subseteq X\}.$$

The *upper approximation* of X is defined as the union of the elementary sets which have a non-empty intersection with X. That is,

$$\overline{X_B} = \bigcup\{x_i / [x_i]_{Ind(B)} \bigcap X \neq \phi\}.$$

Example 2

Lower approximation of the set of animals X = {A1, A2, A4, A5} with respect to the attribute set B = {animality, colour} is $\underline{X_B} = \{A1, A2, A4\}$ and upper approximation is $\overline{X_B} = \{A1, A2, A4, A5, A6\}$.

Definition 2.1.3 – Roughness

The ratio of the cardinality of the lower approximation and the cardinality of the upper approximation is defined as the accuracy of estimation. The measure of roughness is denoted by $R_B(X)$ and is defined as

$$R_B = (X) = 1 - \frac{\left|\underline{X_B}\right|}{\left|\overline{X_B}\right|}$$

Example 3

Continuing with example 2, $R_B(X) = 1 - 3/5 = 2/5$.

Definition 2.1.4 – Relative roughness

Given $a_i \in A$, X is a subset of objects having one specific value *a* of attribute a_i, $\underline{X_{a_j}(a_i = \alpha)}$ and $\overline{X_{a_j}(a_i = \alpha)}$ refer to the lower and upper approximation of X with respect to $\{a_j\}$, then $R_{a_j}(X)$ is defined as the roughness of X with respect to $\{a_j\}$, that is

$$R_{a_j}(X/a_i = \alpha) = 1 - \frac{\left|X_{a_j}(a_i = \alpha)\right|}{\left|X_{a_j}(a_i = \alpha)\right|}, \quad \text{where } a_i, a_j \in A \text{ and } a_i \neq a_j.$$

Example 4

Taking a_i as 'size', a_j as 'colour', α = '*l*arg*e*' and X = {A3, A6, A7}, we get

$$X_{a_j}(a_i = \alpha) = \{A3, A7\} \text{ and } \overline{X_{a_j}(a_i = \alpha)} = \{A1, A2, A3, A4, A5, A6, A7\}.$$

Definition 2.1.5 – Mean roughness (MeR)

Let A have n attributes, and $a_i \in A$. X be the subset of objects having a specific value α of the attribute a_i. Then we define the mean roughness for the equivalence class $a_i = \alpha$, denoted by $\text{MeR}(a_i = \alpha)$ as

$$\text{MeR}(a_i = \alpha) = \sum_{\substack{j=1 \\ j \neq 1}}^{n} R_{a_j}(X/a_i = \alpha) \big/ (n-1).$$

Example 5

We continue with example 4. Taking a_i as 'animality', we have

$$X_{a_j}(a_i = \alpha) = \{A3\} \text{ and } \overline{X_{a_j}(a_i = \alpha)} = \{A3, A5, A6, A7\}.$$

Hence, $R_{a_j}(X/a_i = \alpha) = 1 - 2/7 = 5/7$.

So, $\text{MeR}(a_i = \alpha) = (5/7 + 3/4)/2 = 41/56$.

Definition 2.1.6 – Min-mean roughness (MMeR)

We define MMeR as

$$\text{MMeR} = \min_{1 \le i \le n} \min\{MeR(a_i = \alpha_1), \ldots MeR(a_i = \alpha_{k_j})\},$$

where k_j is the number of equivalence classes in Dom (a_i).

Example 6

1 Taking a_i as 'size' and a_j as 'colour', we get

$$\min\{MeR(a_i = 'large'), MeR(a_i = 'medium'), MeR(a_i = 'small')\} = \min\{5/7, 1, 5/6\} = 5/7.$$

2 Taking a_i as 'colour' and a_j as 'size', we get

$$\min\{MeR(a_i = 'black'), MeR(a_i = 'brown')\} = \min\{3/14, 17/24\} = 3/14.$$

3 Taking a_i as 'size' and a_j as 'animality', we get

$$\min\{MeR(a_i = 'small'), MeR(a_i = 'medium'), MeR(a_i = 'large')\} = \min\{5/6, 41/56\} = 41/56.$$

4 Taking a_i as 'animality' and a_j as 'size', we get

$$\min\{MeR(a_i = 'bear'), MeR(a_i = 'dog'), MeR(a_i = 'cat'), MeR(a_i = 'horse')\} = \min\{1, 1, 1, 1\} = 1.$$

5 Taking a_i as 'colour' and a_j as 'animality', we get

$$\min\{MeR(a_i = 'black'), MeR(a_i = 'brown')\} = \min\{13/28, 7/8\} = 13/28.$$

6 Taking a_i as 'animality' and a_j as 'colour', we get

$$\min\{MeR(a_i = 'black'), MeR(a_i = 'brown')\} = \min\{1,1,1,1\} = 1.$$

So, MMeR = min {5/7, 3/14, 41/56, 1, 13/28, 1} = 3/14.

Definition 2.1.6 – Distance of relevance (DR)

Given two objects B and C of categorical data with n attributes, DR for relevance of objects is defined as follows:

$$DR(B,C) = \sum_{i=1}^{n} DR(b_i, c_i).$$

Here, b_i and c_i are values of objects B and C respectively, under the ith attribute a_i. Also, we have

1 $DR(b_i, c_i) = 1$ if $b_i \neq c_i$

2 $DR(b_i, c_i) = 0$ if $b_i = c_i$

3 $DR(b_i, c_i) = \dfrac{\left|eq_{B_i} - eq_{C_i}\right|}{no_i}$, if a_i is a numerical attribute

where 'eq_{B_i}' is the number assigned to the equivalence class that contains b_i. 'eq_{C_i}' is similarly defined and 'no_i' is the total number of equivalence classes in numerical attribute a_i.

Example 7

Continuing with the same example, we have DR (A3, A4) = 3 and DR (A5, A7) = 1.

2.2 MMeR: min-mean roughness

In this section, we present the algorithm of MMeR.

Procedure

Initially, the value of number of clusters is set to one. The data available in the cluster is the given input data. Now the equivalence classes in each attribute will be calculated. The relative roughness of the equivalence classes of different attributes will be calculated. Since the relative roughness of a class is given with respect to other classes is individual value, we need a mean value of them to compare among the classes. This is termed as MeR of a class and minimum of the MeRs is MMeR. The class that is defined well defined, or whose roughness is MMeR is formed as a cluster. All other classes of the same attribute are grouped to form the second cluster.

Algorithm

Procedure MMeR(U, k)
Begin
Set current number of cluster CNC = 1
Set ParentNode = U
Loop 1:
If CNC < k and CNC ≠ 1 then
ParentNode = Proc ParentNode (CNC)
End if
// Clustering the ParentNode
For each $a_i \in A$ (i = 1 to n, where n is the number of attributes in A)
Determine $[x_m]_{Ind(ai)}$ (m = 1 to number of objects)
For each $a_j \in A$ (j = 1 to n, where n is the number of attributes in A, $j \neq i$)
Calculate $Rough_{a_j}(a_i)$
Next

$$MeR(a_i = \alpha) = \sum_{\substack{j=1 \\ j \neq 1}}^{n} R_{a_j}(X/a_i = \alpha) \Big/ (n-1).$$

Next
Set $MMeR = \min_{1 \le i \le n} \min\{MeR(a_i = \alpha_1), ...MeR(a_i = \alpha_{k_j})\}$, where k_j is the number of equivalence classes in Dom (a_i).
Determine splitting attribute a_i corresponding to the min-mean-roughness
Do binary split on the splitting attribute a_i
CNC = the number of leaf nodes
Go to loop 1:
End

Proc ParentNode (CNC)
Begin
Set i = 1
Do until i < CNC
If Avg-distance of cluster I is calculated
    Go to label
else
    n = count (set of elements in cluster i).

Avg-distance (i) = $2 * (\sum_{j=1}^{n-1} \sum_{k=j+1}^{n}$ (Distance of relevance between objects a_j and a_k))/ (n * (n – 1))

Algorithm (continued)

```
label:
increment i
Loop
Determine Max (Avg-distance (i))
Return (Set of elements in cluster i) corresponding to Max (Avg-distance (i))
End
```

3 Comparison with MMR

In this section, we compare the different features of the two algorithms MMR and MMeR to show the overall superiority of MMeR.

3.1 Relative roughness

In MMeR algorithm, the mean of roughness values for all the equivalence classes in each of the attributes is computed and the attribute possessing the equivalence class with least mean is selected as the splitting attribute rather than choosing the attribute which has the least roughness with respect to any attribute. On the other hand, in MMR, we do not consider the roughness behaviour with all the remaining attributes and hence the clusters formed may not be stable, that is the objects may not be properly related as required, which we shall see in Section 3.3.1.

3.2 Cluster selection

After splitting the objects into two parts, we find the average distance between every two tuples in each cluster. For finding the distance we shall use a new metric called 'distance of relevance (DR)', derived from the notion of Hamming distance, which provides the difference between two objects by taking the equality or otherwise of the respective attributes into consideration. If the respective attribute values are equal we add zero and add one otherwise.

Comparing the average distances in the clusters, the cluster having the least average distance is selected. If clusters with same average distances are found then we choose the one which has more number of objects. In case of a tie a random selection is made.

3.3 Empirical comparison

3.3.1 Relative roughness

We shall consider the following data set in all discussions to follow.

Relative roughness of equivalence classes of each attribute:

Attribute 1 Big – 1.0, Medium – 0.76666665, Small – 0.6681818

Attribute 2 Blue – 0.8, Red – 1.0, Yellow – 0.7272727, Green – 1.0

Attribute 3 Hard – 0.90500003, Moderate – 0.76666665, Soft – 0.95500004

Attribute 4 Indefinite – 0.9666666, Smooth – 0.7838384, Fuzzy 0.9666666

Attribute 5 Plastic – 0.9236363, Wood – 0.9272727, Plush – 1.0, Metal – 1.0

Attribute 6 Negative – 0.8, Neutral – 0.9818182, Positive – 0.9636364

Considering the MMR algorithm Attr-3 will be the splitting attribute resulting in two clusters {2, 4, 10} and {1, 3, 5, 6, 7, 8, 9, 11}. Using DR average distance in {2, 4, 10} and {1, 3, 5, 6, 7, 8, 9, 11} are 3 and 3.857 respectively.

But taking equivalence class into consideration we will select Attr-1 as splitting attribute and clusters obtained in first iteration are {3, 5, 7, 8, 11} & {1, 2, 4, 6, 9, 10}. Using DR the average distances in {3, 5, 7, 8, 11} and {1, 2, 4, 6, 9, 10} are 2.6 and 3.8666 respectively. So this approach provides a cluster with objects of higher closeness.

3.3.2 Cluster selection

The selection of cluster in MMeR for splitting is much more logical and efficient than that in MMR. In MMR, the cluster with maximum number of objects is chosen for clustering. This selection does not reflect the internal structure of elements in the cluster. Moreover, the basic requirement for a good cluster, that is closeness of elements may be violated.

Let us consider the data set {1, 2, 6, 7, 8, 9, 10}. Obviously, the splitting attribute is 'a4' as its roughness with respect to attribute 'a2' is 0. So, the next clusters are: {1, 7, 8} and {2, 6, 9, 10}.

Now, if more number of clusters are required then using MMR cluster-2 is to be chosen; whereas using DR between the objects, we find that cluster-1 and cluster-2 have the average distances 3.33 and 2.83 respectively. As objects in cluster-2 are more similar to each other than the objects in cluster-1, it is more logical to select cluster-1 for splitting.

MMR is ambiguous about the choice of splitting attribute when the number of elements in the clusters is same. However, MMeR provides a logical approach to arrive at a conclusion.

4 Further extension to handle heterogeneous data

Many of the data sets in the real world applications cannot be simply categorical or numerical by nature. Rather, we come across heterogeneous data sets, which are combinations of both types. In this section, we propose a procedure, which allows us to consider numerical data alongside categorical data in a similar manner as done in MMeR.

We need to have classes for clustering of numerical data. For this, we consider mean 'n' of the number of equivalence classes of all attributes containing categorical data. and does not divide the number of objects then its ceiling or floor is selected, whichever leads to merging of less number of objects. So, we merge the elements which are nearest possible. This iterative step ends when the above condition is satisfied. The merged set is termed as an element and all its elements are included in same class.

Example: Consider the data set from Table 2

Table 2 Sample data

Rows	*a1*	*a2*	*a3*	*a4*	*a5*	*a6*
1	Big	Blue	Hard	Indefinite	Plastic	Negative
2	Medium	Red	Moderate	Smooth	Wood	Neutral
3	Small	Yellow	Soft	Fuzzy	Plush	Positive
4	Medium	Blue	Moderate	Fuzzy	Plastic	Negative
5	Small	Yellow	Soft	Indefinite	Plastic	Neutral
6	Big	Green	Hard	Smooth	Wood	Positive
7	Small	Yellow	Hard	Indefinite	Metal	Positive
8	Small	Yellow	Soft	Indefinite	Plastic	Positive
9	Big	Green	Hard	Smooth	Wood	Neutral
10	Medium	Green	Moderate	Smooth	Plastic	Neutral
11	Small	Yellow	Soft	Smooth	Wood	Neutral

The average number of equivalence classes of all attributes = (3 + 4 + 3 + 3 + 3 + 3)/6 = 3.16.

Now presume that there is another attribute a7 (size) with numerical values

{2, 5, 15, 3, 25, 6, 11, 12, 4, 8, 4, 25, 15, 12}

Arranging them in increasing order, we get 2, 3, 4, 5, 6, 8, 11, 12, 15 and 25.

We have ten elements

Case 1 Choosing the average number of equivalence classes as 3 (floor of 3.16). We have to go for merging of only two of them.

Case 2 Choosing the average number of equivalence classes as 4 (roof of 3.16). We have to go for merging of three of them.

So choosing case 1 the selection of data will be more uniform.

Here nearest possible terms are either {2, 3}, {3, 4}, {4, 5} or {5, 6}. Let us choose based on first come first serve since no external dependencies are acting on them. So, we get three equivalence classes:

1 {{2, 3}, 4, 5}, 2: {6, 8, 11} and 3: {12, 15, 25}.

Here 1, 2, 3 are the names given to the equivalence classes.

Let us apply the DR on few of the objects form Table 2. We have,

Object 4 Small, Yellow, Hard, Indefinite, Metal, Positive, 1

Object 5 Small, Yellow, Soft, Indefinite, Plastic, Positive, 3

Object 10 Small, Yellow, Soft, Indefinite, Plastic, Neutral, 2 and

Object 11 Small, Yellow, Soft, Smooth, Wood, Neutral, 1. So,

DR (4, 5) = 2.66 and DR (10, 11) = 2.33.

Hence, with this DR, it can be found that objects 10 and 11 are more relative to each other than objects 4 and 5. Even though the differences are found in three attributes in both pairs.

This DR is pretty interesting measure but we certainly are restricted by incomparable nature of categorical data. This measure can solve many cases of ambiguity in heterogeneous data, which is proved by previous example.

5 Empirical analysis

In order to compare MMeR with MMR and all other algorithms which have taken initiative to handle categorical data we developed an implementation. The data sets are taken from UCI machine learning repository*. The traditional approach for calculating purity of a cluster is given below.

$$\text{Purity (i)} = \frac{\text{the number of data occuring in both } i\text{th cluster and its corresponding class}}{\text{the number of data in the set}}$$

$$\text{Over all Purity (i)} = \frac{\sum_{i=1}^{\#\text{ of clusters}} \text{Purity(i)}}{\#\text{ of clusters}}$$

The same criterion for purity is used in Kim et al. (2004) and Guha et al. (2000). Purity of a cluster reflects the relevantness of objects in the cluster. The purity value ranges from zero to one.

5.1 Comparison of MMeR with MMR and algorithms based on fuzzy set theory

Till the development of MMR, the only algorithms which aimed at handling uncertainty in the clustering process were based upon fuzzy set theory (Huang, 1998; Kim et al., 2004; Krishnapuram et al., 1995; Zhang et al., 2000).These algorithms based on fuzzy set theory include fuzzy K-modes, fuzzy centroids. The K-modes algorithm replaces the means of the clusters (K-means) with modes and uses a frequency based method to update the modes in the clustering process to minimise the clustering cost function. Fuzzy K-modes generates a fuzzy partition matrix from categorical data. By assigning a

confidence to objects in different clusters, the core and boundary objects of the clusters

are determined for clustering purposes. The fuzzy centroids algorithm uses the concept of fuzzy set theory to derive fuzzy centroids to create clusters of objects which have categorical attributes.

First, we shall see the purity of MMeR by applying it on a small data set in experiment 1

Experiment 1

The soybean small data set contains 47 objects on diseases in soybeans. Each object can be classified as one of the four diseases namely, diaporthe stem canker, charcoal rot, rhizoctonia root rot, and phytophthora rot and is described by 35 categorical attributes.

The data set is comprised 17 objects for phytophthora rot disease and ten objects for each of the remaining diseases. So we have only four possible class values (four diseases).

Observation

The algorithms based on fuzzy set theory generate four clusters. The results are summarised in following tables. Out of 47 objects, 39 belong to the majority class label of the cluster in which they are classified. Thus, the overall purity of the clusters is 83%. The performance of MMeR is found to be same as MMR; as detailed in the following table.

Experiment 2

The zoo data set is comprised of 101 objects, where each data point represents information of an animal in terms of 18 categorical attributes. Animals are classified into seven classes. Therefore, stopping criterion for MMR is set at seven clusters. As we know that teaching a system need infinite data, the data that is given is a sample. It is observed that MMeR classified this data with 84 objects in majority class out of 101. This is an improvement over MMR, which classified 81objects in majority class. Also, the purity in case of MMeR is significantly higher than MMR. Tables 4 and 5 provide a comparative study of these experimental results.

Table 3 Applying MMeR on the soybean data set

Cluster number	*Disease 1*	*Disease 2*	*Disease 3*	*Disease 4*	*Purity*
1	0	10	0	0	1
2	10	0	0	0	1
3	0	0	8	17	0.68
4	0	0	2	0	1
Overall purity					0.83

Table 4 MMR results on the zoo data set

Cluster number	*C1*	*C2*	*C3*	*C4*	*C5*	*C6*	*C7*	*Purity*
1	0	0	3	0	3	0	0	0.50
2	39	0	0	0	0	0	0	1
3	0	0	1	0	1	0	0	0.50
4	0	0	1	13	0	0	0	0.93
5	0	0	0	0	0	2	10	0.83
6	2	0	0	0	0	6	0	0.75
7	0	20	0	0	0	0	0	1
Overall purity								0.787*

Note: *In the original paper of MMR the overall purity is misprinted as 0.91

Table 5 Applying MMeR on zoo data set

Cluster number	*C1*	*C2*	*C3*	*C4*	*C5*	*C6*	*C7*	*Purity*
1	0	20	0	0	0	0	0	1
2	2	0	0	0	0	0	0	1
3	39	0	3	0	4	0	0	0.829
4	0	0	1	13	0	0	0	0.93
5	0	0	0	0	0	0	1	1
6	0	0	0	0	0	7	9	0.563
7	0	0	0	0	0	1	0	1
Overall purity								0.902

The earlier algorithms for classification with uncertainty like K-modes, fuzzy K-modes and fuzzy centroid on one hand and MMR on the other hand were applied to both soybean and zoo data sets. Table 6 below provides the comparison of purity for these algorithms on these two datasets. It is observed that MMeR has a better purity than all other algorithms when applied on zoo data set. Also, except for fuzzy centroids algorithm it has better purity than all the rest of the algorithms. Since soybean data set is a very small data set and has no uncertainty involved, this performance seems to be unimportant.

As mentioned earlier, all the fuzzy set based algorithms face a challenging problem that is the problem of stability. These algorithms require great effort to adjust the parameter, which is used to control the fuzziness of membership of each data point. At each value of this parameter, the algorithms need to be run multiple times to achieve a stable solution.

MMR on the other hand has no such problem. MMeR continues to have the advantages of MMR over the other algorithms as mentioned above. But it has higher purity than MMR, which establishes its superiority over MMR.

Table 6 Purity comparison of MMeR with algorithms based on fuzzy set theory and MMR

Data set	*K-modes*	*Fuzzy K-modes*	*Fuzzy centroids*	*MMR*	*MMeR*
Soybean	0.69	0.77	0.97	0.83	0.83
Zoo	0.60	0.64	0.75	0.787	0.902

The next experimental data set is a large data set. Though it has no uncertainty involved in it, we apply MMeR on it to show its applicability on large data sets.

Experiment 3

The Mushroom data set has 8,418 objects, where each object contains information about a single mushroom. There are 22 categorical attributes; each attribute corresponds to a physical characteristic of a mushroom. An object also contains a poisonous or edible class label for a mushroom. The data has 4,208 edible mushrooms and 3,916 poisonous

mushrooms. Since we go for only two clusters the results of MMR and MMeR are same. But we can look out below when we go for 20 clusters the results that MMeR attained are quite astounding with 0.964 purity.

Observation

The ROCK algorithm, which is a traditional hierarchical algorithm, was used by Guha (2000) on mushroom data. However, it was found that the total number of mushrooms considered is less than the actual number. In case of ROCK 4,692 mushrooms belong to the majority class label of the cluster in which they are classified. In case of MMR, this number turns out to be 6,935 (Table 7) to give an overall purity ratio of 0.84, where as in case of MMeR the number comes out to be 8,120 (Table 8) for an overall purity ratio of 0.96. This shows the efficiency of MMeR is much higher on large data sets in comparison to all other existing algorithms.

Table 7 MMR results on the mushroom data set with 20 clusters

Cluster number	*Edible*	*Poisonous*	*Purity*
1	164	0	1
2	425	36	0.92
3	640	0	1
4	539	0	1
5	19	0	1
6	1	0	1
7	329	151	0.68
8	14	0	1
9	144	115	0.56
10	111	120	0.52
11	0	782	1
12	1177	321	0.79
13	0	6	1
14	0	1	1
15	0	28	1
16	40	0	1
17	120	323	0.73
18	88	1440	0.94
19	12	196	0.94
20	385	397	0.51
Over all purity			0.84

Table 8 MMeR results on the mushroom data set with 20 clusters

Cluster number	*Edible*	*Poisonous*	*Purity*
1	0	8	1
2	0	1296	1
3	0	24	1
4	0	144	1
5	0	72	1
6	336	0	1
7	192	0	1
8	1728	72	0.96
9	96	8	0.923
10	96	96	0.5
11	192	0	1
12	0	72	1
13	1024	96	0.9142
14	72	0	1
15	512	0	1
16	0	256	1
17	24	1760	0.9865
18	24	0	1
19	192	0	1
20	0	24	1
Over all purity			0.9641

Data set description

- It is a simple database having 22 Boolean-valued attributes.

6 Conclusions

In this paper, we introduced a new algorithm called MMeR, which improves the algorithm MMR, proposed and studied in Parmar et al. (2007) in two ways. First, it is more efficient than most of the earlier algorithms including MMR and handles uncertain data using rough set theory. Secondly, we have provided a method such that it can tackle both numerical and categorical data sets simultaneously. Like MMR, it requires only one input, which is the number of clusters desired.

Here, we have introduced a new entity criterion to find a distance between any two data objects by generalising hamming distance. Other advancements which we made are in choosing cluster for re-clustering and handling of heterogeneous data.

Thus, some of the proposals for future research stated in Parmar et al. (2007) have been taken care by our algorithm, MMeR. However, the study of the approach proposed by Voges and Pope (2004) in exploring the potential application of evolutionary algorithms with rough sets to help determine the number of clusters in advance can be studied. Future enhancements of this algorithm can also be made in the fields of selection of splitting attribute by introducing fuzzy properties. This will lead to development of rough-fuzzy concepts in clustering.

References

Andritsos, P., Tsaparas, P., Miller, R.J. and Sevcik, K.C. (2003) 'Clustering categorical data based on information loss minimisation', in *2nd Hellenic Data Management Symposium*, pp.334–344.

Dempster, A., Laird, N. and Rubin, D. (1977) 'Maximum likelihood from incomplete data via the EM algorithm', *Journal of the Royal Statistical Society*, Vol. 39, No. 1, pp.1–38.

Ganti, V., Gehrke, J. and Ramakrishnan, R. (1999) 'CACTUS – clustering categorical data using summaries', in *5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.73–83.

Gibson, D., Kleinberg, J. and Raghavan, P. (2000) 'Clustering categorical data: an approach based on dynamical systems', *The Very Large Data Bases Journal*, Vol. 8, Nos. 3–4, pp.222–236.

Guha, S., Rastogi, R. and Shim, K. (2000) 'ROCK: a robust clustering algorithm for categorical attributes', *Information Systems*, Vol. 25, No. 5, pp.345–366.

Halkidi, M., Batistakis, Y. and Vazirgiannis, M. (2001) 'On clustering validation techniques', *Journal of Intelligent Information Systems*, Vol. 17, Nos. 2–3, pp.107–145.

He, Z., Xu, X. and Deng, S. (2004) 'A link clustering based approach for clustering categorical data', *Proceedings of the WAIM Conference*, available at http://xxx.sf.nchc.org.tw/ftp/cs/papers/0412/0412019.pdf.

He, Z., Xu, X. and Deng, S. (2002) 'Squeezer: an efficient algorithm for clustering categorical data', *Journal of Computer Science & Technology* Vol. 17, No. 5, pp.611–624.

Huang, Z. (1998) 'Extensions to the k-means algorithm for clustering large data sets with categorical values', *Data Mining and Knowledge Discovery*, Vol. 2, No. 3, pp.283–304.

Jiang, D., Tang, C. and Zhang, A. (2004) 'Cluster analysis for gene expression data: a survey', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16, No 11, pp.1370–1386.

Kim, D., Lee, K. and Lee, D. (2004) 'Fuzzy clustering of categorical data using fuzzy centroids', *Pattern Recognition Letters*, Vol. 25, No. 11, pp.1263–1271.

Krishnapuram, R., Frigui, H. and Nasraoui, O. (1995) 'Fuzzy and possibilistic shell clustering algorithms and their application to boundary detection and surface approximation', *IEEE Transactions on Fuzzy Systems*, Vol. 3, No. 1, pp.29–60.

Krishnapuram, R. and Keller, J. (1993) 'A possibilistic approach to clustering', *IEEE Transactions on Fuzzy Systems*, Vol. 1, No. 2, pp.98–110.

Lingras, P. and West, C. (2004) 'Interval set clustering of web users with rough K-means', *Journal of Intelligent Information Systems*, Vol. 23, No. 1, pp.5–16.

Lingras, P., Yan, P.R. and Hogo, M. (2003) 'Rough set based clustering: evolutionary, neural, and statistical approaches', *Proceedings of the 1st Indian International Conference on Artificial Intelligence*, pp.1074–1087.

Mazlack, L., He, A., Zhu, Y. and Coppock, S. (2000) 'A rough set approach in choosing partitioning attributes', in *Proceedings of the ISCA 13th International Conference (CAINE-2000)*, pp.1–6.

Parmar, D., Wu, T. and Blackhurst, J. (2007) 'MMR: an algorithm for clustering categorical data using rough set theory', *Data & Knowledge Engineering*, Vol. 63, pp.879–893.

Pawlak, Z. (1982) 'Rough sets', *International Journal of Information and Computer Science*, Vol. 11, pp.341–356.

Pawlak, Z. (1991) *Rough Sets – Theoretical Aspects of Reasoning about Data*, Kluwer Academic Publishers, Boston.

Pawlak, Z. and Skowron, A. (2007a) 'Rudiments of rough sets', *Information Sciences*, Vol. 177, pp.3–27.

Pawlak, Z. and Skowron, A. (2007b) 'Rough sets: some extensions', *Information Sciences*, Vol. 177, pp.28–40.

Pawlak, Z. and Skowron, A. (2007c) 'Rough sets and Boolean reasoning', *Information Sciences*, Vol. 177, pp.41–73.

Ruspini, E. (1969) 'A new approach to clustering', *Information Control*, Vol. 15, No. 1, pp.22–32.

Voges, K., Pope, N. and Brown, M. (2002) 'Cluster analysis of marketing data examining on-line shopping operation: a comparison of K-means and rough clustering approaches', in Abbas, H.A., Sarkar, R.A. and Newton, C.S. (Eds.): *Heuristics and Optimization for Knowledge Discovery*, pp.207–224, Idea Group Publishing, Hershey, PA.

Voges, K. and Pope, N. (2004) 'Generating compact rough cluster descriptions using an evolutionary algorithm', in Deb, K. (Ed.): *GECCO 2004, Genetic and Evolutionary Algorithm Conference – LNCS*, pp.1332–1333, Springer-Verlag, Berlin.

Zadeh, L.A. (1965) 'Fuzzy sets', *Information and Control*, Vol. 11, pp.338–353.

Zhang, Y., Fu, A., Cai, C. and Heng, P. (2000) 'Clustering categorical data', in *Proceedings of the 16th International Conference on Data Engineering*, pp.305–324.