Contents lists available at Science-Gate

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.science-gate.com/IJAAS.html

# Improved minimum-minimum roughness algorithm for clustering categorical data

Do Si Truong, Nguyen Thanh Tung, Lam Thanh Hien *

*Faculty of Information Engineering Technology, Lac Hong University, Bien Hoa, Vietnam*

ARTICLE INFO

*Article history:*
Received 17 April 2021
Received in revised form
14 July 2021
Accepted 22 July 2021

*Keywords:*
Data mining
Categorical data
Rough set theory
Clustering category
IMMR

ABSTRACT

Clustering is a fundamental technique in data mining and machine learning. Recently, many researchers are interested in the problem of clustering categorical data and several new approaches have been proposed. One of the successful and pioneering clustering algorithms is the Minimum-Minimum Roughness algorithm (MMR) which is a top-down hierarchical clustering algorithm and can handle the uncertainty in clustering categorical data. However, MMR tends to choose the category with less value leaf node with more objects, leading to undesirable clustering results. To overcome such shortcomings, this paper proposes an improved version of the MMR algorithm for clustering categorical data, called IMMR (Improved Minimum-Minimum Roughness). Experimental results on actual data sets taken from UCI show that the IMMR algorithm outperforms MMR in clustering categorical data.



## 1. Introduction

Clustering is a fundamental technique in data mining and machine learning. It is actually the finding of groups of objects such that objects in the same group have high similarity, and those in different groups have low similarity (Han and Kamber, 2006). Clustering has been widely deployed in several fields such as data mining, machine learning, pattern recognition, bioinformatics, etc. Literally, many clustering techniques have been proposed and generally classified into two types: partial and hierarchical. Most of the clustering techniques focus on numeric data sets, where each category describing the objects has a value domain that is a continuous interval of real values, and each data object number is treated as a point in a multidimensional metric space with a metric that measures distances between objects, such as the Euclidean metric or the Mahalanobis metric. However, practical applications often encounter data sets classified as categories whose values are finite and unordered; for example, hair color, nationality, etc. fail to be defined with a distance function spontaneously.

Recently, clustering categorical data has attracted special attention from several researchers in data mining areas and several clustering algorithms have been proposed (Khandelwal and Sharma, 2015; Cao et al., 2009; Guha et al., 2000; Gibson et al., 2000; Huang, 1998; Kim et al., 2004; Mesakar and Chaudhari, 2012). Although these algorithms make important contributions to the problem of clustering categorical data, they still fail to handle uncertainty in the clustering process. Handling uncertainty during clustering is an important issue because in many practical applications there are often no clear boundaries between clusters. To handle the uncertainty in the clustering of categorical data, recently Huang (1998), and Kim et al. (2004) proposed two algorithms that apply fuzzy set theory, However, these algorithms require many runs to establish a stable value needed for the parameter used to control the degree of fuzzy membership. A popular approach to dealing with uncertainty is to use Rough Set Theory (RST), proposed by Pawlak (1991). The RST is an effective tool for machine learning and data mining from information systems with category values (Pawlak, 1991). It has been successfully applied in many fields (Zhang et al., 2016; Bello and Falcon, 2017) because it can effectively deal with data that require neither thresholds nor domain-specific expertise (Jensen and Shen, 2008).

Recently, some authors have proposed a new approach to solve the problem of clustering categorical data by using RST and divisive technique

* Corresponding Author.
Email Address: lthien@lhu.edu.vn (L. T. Hien)
https://doi.org/10.21833/ijaas.2021.10.006
Corresponding author's ORCID profile:
https://orcid.org/0000-0002-4539-3712

(Hassanein and Elmelegy, 2014; Herawan et al., 2010; Jyoti, 2013; Mazlack et al., 2000; Parmar et al., 2007). Its key principle is to choose the best category from many candidate categories to gradually divide the objects into clusters in each run. Specifically, Mazlack et al. (2000) proposed an algorithm that uses an index Total Roughness (TR) in RST to determine the clustering quality of the selected category, i.e., larger TR is better. Herawan et al. (2010) proposed another technique called Maximum Dependency Categories (MDA) which uses the dependency between categories in RST. Qin et al. (2012) argued that TR and MDA values are both determined mainly based on the number of elements in the lower approximation of a category for other categories; so, they often choose the same category as the clustering category in most cases.

One of the most successful and pioneering clustering algorithms based on RST is the Minimum-Minimum Roughness algorithm (MMR) proposed by Parmar et al. (2007). MMR, a top-down hierarchical clustering algorithm, uses Min-Roughness as the criterion to determine the clustering category at each iteration step. MMR allows handling uncertainty during the clustering of categorical data. However, MMR tends to choose the category with fewer values (Qin et al., 2012), i.e., if a category has only a single value, it is selected as a clustering category, resulting in the termination of clustering. Moreover, MMR chooses a leaf node that has more objects to split further, thus producing undesirable clustering results.

In this paper, we propose an improved MMR algorithm called IMMR (Improved Minimum-Minimum Roughness) to overcome the mentioned shortcomings. Besides the advantages of MMR, our proposed IMMR algorithm not only ignores all single-valued categories but also determines the next split node by considering the sum entropy of all categories on the nodes. Experimental results on actual data sets taken from UCI databases show that the IMMR algorithm can be used successfully in clustering analysis of categorical data with better clustering results.

## 2. Related concepts

A categorical data set can be represented as a table, where each row represents an object, case, or event, and each column represents a category, property, or a scale to be measured on each object. In RST, such a data table is called an information system. Formally, an information system is defined as follows.

**Definition 1:** An information system is a quadruple tuple $S = (U, A, V, f)$, where $U$ is a non-empty finite set of objects, $A$ is a nonempty finite set of categories, $V = \bigcup_{a \in A} V_a$ where $V_a$ is a set of all values of category $a$, and $f: U \times A \to V$ is a function, called information function, that assigns value a $f(u, a) \in V_a$ for every $(u, a) \in U \times A$.

**Definition 2** Let $S = (U, A, V, f)$ be an information system, $B \subseteq A$. Two elements $x, y \in U$ is said to be $B$-indiscernible in $S$ if and only if $f(x, a) = f(y, a)$, for every $a \in B$.

We denote the indiscernibility relation induced by the set of categories $B$ by $IND(B)$. Obviously, $IND(B)$ is an equivalence relation and it induces a unique partition (clustering) of $U$. The partition of $U$ induced by $IND(B)$ in $S = (U, A, V, f)$ denoted by $P_B$ and the equivalence class in the partition $P_B$ containing $x \in U$, denoted by $[x]_B$.

**Definition 3:** Let $S = (U, A, V, f)$ be an information system, $B \subseteq A$, and $X \subseteq U$. The $B$-lower approximation of $X$, denoted by $\underline{B}(X)$ and $B$-upper approximation of $X$, denoted by $\overline{B}(X)$, respectively, are defined by:

$$\underline{B}(X) = \{x \in U | [x]_B \subseteq X\}$$

and,

$$\overline{B}(X) = \{x \in U | [x]_B \cap X \neq \emptyset\}. \quad (1)$$

These definitions state that object $x \in \underline{B}X$ certainly belongs to $X$, whereas object $x \in \overline{B}X$ could belong to $X$. Obviously, there is $\underline{B}X \subseteq X \subseteq \overline{B}X$ and $X$ is said to be definable if $\underline{B}X = \overline{B}X$. Otherwise, $X$ is said to be rough with B-boundary $BN_B(X) = \overline{B}X - \underline{B}X$.

**Definition 4:** Let $S = (U, A, V, f)$ be an information system, $B \subseteq A$, and $X \subseteq U$. The accuracy of approximation of $X$ with respect to $B$ is defined as:

$$\alpha_B(X) = \frac{|\underline{B}X|}{|\overline{B}X|} \quad (2)$$

Throughout the paper, $|X|$ denotes the cardinality of $X$.

Obviously, $0 \leq \alpha_B(X) \leq 1$. If $\alpha_B(X) = 1$, then $\underline{B}X = \overline{B}X$. The $B$-boundary of $X$ is empty, and $X$ is crisp with respect to $B$. If $\alpha_B(X) < 1$, then $\underline{B}X \subset \overline{B}X$. The $B$-boundary of $X$ is not empty, and $X$ is rough with respect to $B$.

**Definition 5:** Let $S = (U, A, V, f)$ be an information system, $B \subseteq A$, and $X \subseteq U$. The roughness of $X$ with respect to $B$ is defined as:

$$\rho_B(X) = 1 - \frac{|\underline{B}(X)|}{|\overline{B}(X)|} \quad (3)$$

**Definition 6:** Let $S = (U, A, V, f)$ be an information system. For $P, Q \subseteq A$, it is said that $Q$ depends on $P$ in a degree $k$ $(0 \leq k \leq 1)$, denoted by $P \Longrightarrow_k Q$, if:

$$k = \frac{\sum_{X \in Q} |\underline{P}(X)|}{|U|} \quad (4)$$

**Definition 7:** Let $S = (U, A, V, f)$ be an information system, $X \subseteq A$ and $P_X = \{X_1, X_2, \ldots, X_m\}$. The entropy

of a partition $P_X$ is defined as:

$$E(P_X) = -\sum_{i=1}^{m} P(X_i)\log_2 P(X_i) \quad (5)$$

where $P(X_i) = |X_i|/|U|$, and we define $0\log_2 0 = 0$. Entropy is a measure of the degree of confusion (uncertainty) about the value of a category $a$ in an information system S. The smallest possible value of entropy is 0, which occurs when all the column vector components corresponding to the category a in S are the same, i.e., $\Pr(a = i) = 1$ and $\Pr(a = j) = 0$ for all $j \neq i$. In other words, there is no disturbance in this column vector. The larger the value of entropy, the more disordered the column vector associated with a The maximum possible value of entropy is $log_2|V_a|$, which is obtained when $\Pr(a)$ is uniformly distributed, i.e., $\Pr(a = i) = 1/m$ for all $i \in V_a$. Entropy depends only on probability and not on the specific value of a.

For the above reason, entropy has been used by many authors to determine how good a clustering operation is Ienco et al. (2012), Jyoti (2013), Parmar et al. (2007), and UCI (2013). Value entropy of a cluster of smaller extent smaller disturbances in clusters, i.e., clusters uniformity in increasingly large over. However, McCaffrey (2013) argued that it is uneasy to modify the entropy definition of a vector to apply to a cluster and a clustering result (essentially a set of tables or matrices). To evaluate the quality of a clustering, McCaffrey (2013) used the following definition.

**Definition 8:** Given the clustered data set in the form of an information system $S = (U, A, V, f)$, a clustering operation $\pi_a = \{X_1, X_2, \dots, X_m\}$ of the objects contained in U. The entropy of a cluster $X_i$ is determined by the sum of the entropy of each of the above categories $X_i$. The entropy of clustering $\pi_a$ is defined as the weighted sum of the entropies of each cluster, where the weight of each cluster is its probability and equals $\Pr(X_i) = |X_i|/|U|$.

The lower the entropy of the clustering, the higher the clustering quality, in the sense that the similarity of objects in the same cluster is high and the similarity between clusters is low.

## 3. MMR algorithm and its improved version IMMR

### 3.1. MMR algorithm

MMR is a top-down hierarchical clustering algorithm (Parmar et al., 2007). It is an iterative non-inverting process that progressively dichotomizes the original set U of objects with the goal of achieving a better clustering result. The algorithm takes the number of clusters to collect predetermined k as input and will terminate when this number of clusters k is reached. At each iteration, the two basic tasks that a top-down hierarchical clustering algorithm must perform include:

(1) Choose the best category from all candidate categories to partition the node to further bifurcate into equivalent classes.
(2) Among the obtained equivalent classes, determine a class that becomes a cluster (leaf node), merge all remaining classes into a node for bifurcation in the next step.

To perform the above two tasks, the MMR algorithm uses the roughness concept in the RST as presented in the following definitions.

**Definition 9:** Mean-Roughness: Given a clustered data set in the form of an information system $S = (U, A, V, f)$, two categories $a_i$ and $a_j$ of A, $a_j \neq a_i$. The category mean rawness for the category $a_i$ against the category $a_j$, denoted by $Rough_{a_j}(a_i)$, is defined as follows (Parmar et al., 2007):

$$Rough_{a_j}(a_i) = \frac{\sum_{X \in U/Ind\{a_i\}} \rho_{a_j}(X)}{|U/Ind\{a_i\}|} \quad (6)$$

Which $|U/Ind\{a_i\}|$ is the equivalent class in the partition $U/Ind\{a_i\}$, $\rho_{a_j}(X)$ is the roughness of each class X in $U/Ind\{a_i\}$ for category $a_j$ is determined with formula 3, specifically:

$$\rho_{a_j}(X) = 1 - \alpha_{a_j}(X) = 1 - \frac{\left|\underline{a_j X}\right|}{\left|\overline{a_j X}\right|}.$$

The $Rough_{a_j}(a_i)$ smaller the value, the higher the similarity degree of the category $a_j$ among the objects generated by $a_i$ in each class.

**Definition 10:** Min-Mean-Roughness: Given the clustered data set in the form of information system $S = (U, A, V, f)$, category $a_i \in A$. The minimum mean-roughness of the category $a_i$ for each category $a_j \in A$, $a_j \neq a_i$, denoted by $MR(a_i)$, is determined by (Parmar et al., 2007):

$$MR(a_i) = \min_{a_j \in A \wedge a_j \neq a_i} \left\{Rough_{a_j}(a_i)\right\} \quad (7)$$

**Definition 11:** Min-Min-Mean-Roughness: Given the clustered data set in the form of an information system $S = (U, A, V, f)$. The minimum value of the mean minimum roughness, denoted MMR, is defined as follows (Bello and Falcon, 2017):

$$MMR = \min_{a_i \in A}\{MR(a_i)\}. \quad (8)$$

In each iteration, the MMR algorithm chooses the category $a \in A$ for the smallest MR as the partition category, specifically,

$$a = \text{argmin}_{a_j \in A}\{MR(a_j)\}$$

After the partition category, a is determined, X is then dichotomized as follows:

- Identify partition of X on a by solving

$X/Ind(a) = \{X_1, \dots, X_s\}$,

- For each equivalent class $X_r$, sum the roughness for each category $a_j \in A, a_j \neq a$ by:

$$A\rho(X_r) = \sum_{a_j \in A \wedge a_j \neq a} \rho_{a_j}(X_r)$$

- Take the class with the smallest value $A\rho$ as a cluster (leaf) and the union of the remaining classes as a node for bifurcation in the next step.

Though MMR is considered as one of the successful and pioneering RST clustering algorithms, it still has certain shortcomings as mentioned in the previous section; specifically, (1) MMR tends to choose the category with fewer values (Qin et al., 2012), and (2) MMR chooses a leaf node that has more objects to split further, thus producing undesirable clustering results. To overcome the above limitations, it can be further improved as follows.

### 3.2. Improved algorithm IMMR

To overcome the first limitation, at each step of the iterative process, before performing the computations to determine the best dichotomous category, we need to remove all the single-valued categories, i.e., remove all categories for the node partition to be split consisting of only a single class. And, to deal with the second one, we need to determine which node to be further dichotomized by considering the sum of the entropy of all the categories on each node as presented in Definition 8. The following example is used as an illustration.

**Example:** Consider the information system in Table 1 and we need to group these into 3 clusters (k=3).

**Table 1:** Information systems

| U | a1 | a2 |
|---|---|---|
| 1 | Medium | F |
| 2 | Small | F |
| 3 | Small | T |
| 4 | Small | T |
| 5 | Small | T |
| 6 | Big | T |

At the first step, both MMR and IMMR algorithms take all internal objects U as nodes to be dichotomized, and determine the best partition category as the one that gives the smallest MR value (Definition 11); we have:

$U/a1 = \{\{1\},\{2,3,4,5\},\{6\}\}$; $U/a2 = \{\{1,2\},\{3,4,5,6\}\}$;
$Rough_{a2}(a1) = 1, MR(a1) = 1,\ MMR(a1) = 1$;
$Rough_{a1}(a2) = 4/5, MR(a2) = 4/5,\ MMR(a2) = 4/5$.

Since $MMR(a_2) < MMR(a_1)$, category a2 is taken by both algorithms as the first partition category. Thus, two nodes $X1 = \{u \in U | a2 = F\} = \{1,2\}$ and $X2 = \{u \in U | a2 = T\} = \{3,4,5,6\}$ are created. Then, because the leaf node MMR algorithm has more objects to further dichotomized, the next bifurcation node chosen by the MMR is $X2 = \{3,4,5,6\}$.

In the second step, the selected partition category is a1. The final clustering result of MMR can be represented as a tree in Fig. 1.

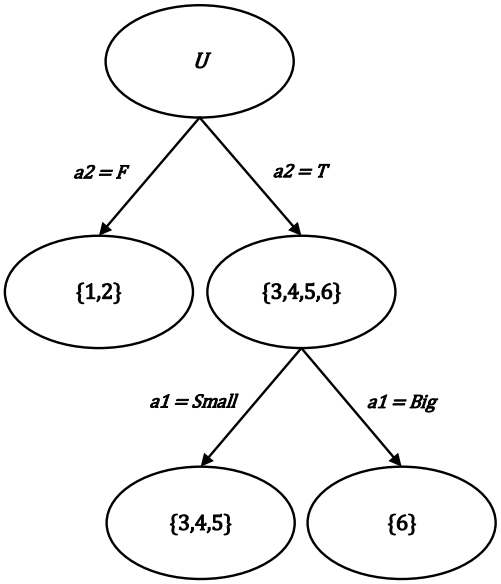


**Fig. 1:** Clustering result by MMR algorithm

Now, we determine the bifurcation node in the second step by IMMR, i.e., we consider the sum of the entropy of all categories on the nodes, denoted by TENT:

$$\text{TENT}(X1) = entropy(tt1 \mid X1) + entropy(tt2 \mid X1)$$
$$= -\left(\frac{1}{2} \times log_2 \frac{1}{2} + \frac{1}{2} \times log_2 \frac{1}{2}\right) - (1 \times log_2 1) = 1.$$
$$\text{TENT}(X2) = entropy(tt1 \mid X2) + entropy(tt2 \mid X2)$$
$$= -\left(\frac{3}{4} \times log_2 \frac{3}{4} + \frac{1}{4} \times log_2 \frac{1}{4}\right) - (1 \times log_2 1) = 0.8113.$$

The node $X1 = \{1,2\}$ has a greater value of TENT, so it is selected as the node to be dichotomized. Thus, dichotomizing X1 according to a1 gives the final clustering result presented in the form of a tree in Fig. 2.

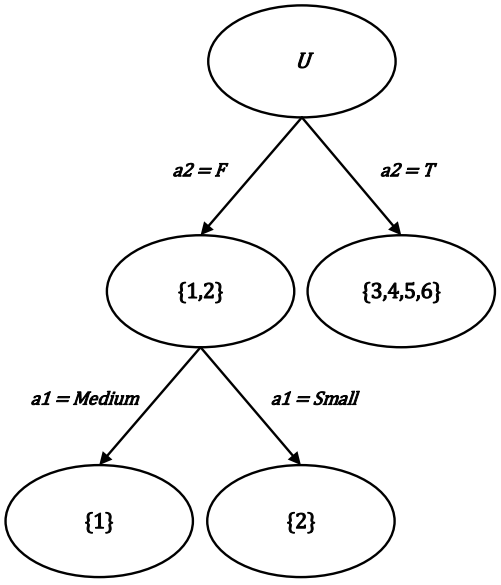


**Fig. 2:** Clustering result by IMMR algorithm

Let's evaluate the performance of the two clustering algorithms according to Definition 8.

With k=3, MMR results in clusters {{1,2}, {3,4,5}, {6}}. The total entropy of the category of each cluster is:

$$TENT(\{1,2\}) = 1,$$
$$TENT(\{3,4,5\}) = -\big((1 \times log_2 1) + (1 \times log_2 1)\big) = 0,$$
$$TENT(\{6\}) = -\big((1 \times log_2 1) + (1 \times log_2 1)\big) = 0.$$

The WSTENT-weighted sum of the entropies of the 3 clusters is determined by:

$$WSTENT(\{1,2\},\{3,4,5\},\{6\}) = \frac{1}{3}TENT(\{1,2\}) + \frac{1}{2} + TENT(\{3,4,5\}) + \frac{1}{6}TENT(\{6\}) = \frac{1}{3}.$$

Three clusters by IMMR as shown in Fig. 2 is {{3,4,5,6}, {1}, {2}}. The total entropy of the category of each cluster and the weighted sum of the entropies of the 3 clusters are respectively determined by:

$$TENT\{1\} = 0, TENT\{2\} = 0, TENT\{3,4,5,6\} = 0.3113$$
$$WSTENT(\{1\},\{2\},\{3,4,5,6\}) = \frac{1}{6}TENT\{1\} + \frac{1}{6}TENT\{2\} + \frac{4}{6}TENT\{3,4,5,6\} = 0.2075.$$

Based on the values of $WSTENT$, we can conclude that the clustering of IMMR is better than that of MMR.

Generally, the proposed IMMR algorithm can be coded as the following:

*Procedure IMMR(U,k)*
*Begin*
*set current number of cluster* $CNC = 1$ *//number of existing clusters*
*set* $CNode = U$*//CNnode denotes numbers of nodes to be dichotomized*
*while* $(CNC < k)$ *do*
*B=A*
*for each* $a_i \in A$
*Partition* $CNode/Ind\{a_i\}$
*If* $|CNode/Ind\{a_i\}| = 1$*//every instance in CNode has the same value of* $a_i$
*B=B-*$a_i$*// remove category* $a_i$
*end if*
*end*
*//Proceed to partition the node CNode*
*For each* $a_i \in B$
*Calculate* $CNode/Ind\{a_i\}$
*end*
*For each* $a_j \in B$
*Calculate* $Rough_{a_j}(a_i)$
*end*
*Determine* $MR(a_i) = Min_{a_j \in A}\{Rough_{a_j}(a_i)\}$
*Determine category* $a \in B$ *satisfying* $a = argmin_{a_j \in A}\{MR(a_j)\}$
*Partition based on the category a*
*CNC=CNC+1*
*If CNC<k then*
*CNode=leaf node with the smallest total entropy of the categories*
*end if*
*end while*
*end*

Assuming that the given data set has n objects, m categories, k is an assigned number of clusters and l is the maximum value of the possible category domains, then, to group objects into k clusters, the MMR algorithm needs to perform k-1 iterations. At each iteration, the time to find the partition of the categories is mn, the time to calculate the mean roughness is m2l, the time to calculate the MR and MMR is 2m, the time to compute the entropy of the categories is m. Therefore, the time complexity of IMMR is polynomial and is determined with O(knm+km2l).

### 3.3. Performance evaluation

Evaluation of clustering quality is often a difficult and subjective task (Ienco et al., 2012; Parmar et al., 2007). In this paper, we use the index called Overall purity proposed by Parmar et al. (2007) as it is an external criterion, a simple and easily accepted evaluation criterion (Ienco et al., 2012). It evaluates the clustering quality of a clustering result against the actual data set, where each object is preceded by a particular class label. Using information about the actual class labels and information about the cluster labels to which the objects are clustered by the algorithm, it evaluates how well the clustering results match with the initially given classes.

Assuming that a data set includes an actual object that needs to be classified into k classes {c1, c2, …, ck} and k clusters {ω1, ω2, …, ωk}. Let ni denote the number of objects that have been grouped into the cluster ωi, nij denotes the number of objects belonging to the cluster ωi with class labels cj in the set of known class labels.

Cluster purity $\omega_i$ is defined as the ratio between the number of objects $\omega_i$ in the dominant class label and the number of objects $n_i$:

$$Purity(\omega_i) = \frac{1}{n_i} max_j(n_{ij})$$

Overall purity is defined as the proportion of properly classified objects among all the objects present in the data set, i.e.,

$$Overallpurity = \frac{\sum_{i=1}^{k} max_j(n_{ij})}{n}$$

The Overall purity has a range of [0,1]. The higher the Overall purity, the better the quality of the clustering result. A perfect clustering gives an Overall purity value of 1. The Overall purity increases as the number of clusters increases. In particular, the Overall purity is 1 if each cluster consists of only one object.

To calculate the Overall purity, we first create a confusion matrix as shown in Table 2, by browsing through each phrase ωi and count how many objects belong to each class cj. Then, from each row for each cluster ωi, we select the maximum value, sum them together and finally get the total divided by the number of all objects in the data set.

**Table 2:** Confusion matrix

| | $c_1$ | $c_2$ | ... | $c_k$ | |
|---|---|---|---|---|---|
| $\omega_1$ | $n_{11}$ | $n_{12}$ | ... | $n_{1k}$ | $n_1$ |
| $\omega_2$ | $n_{21}$ | $n_{22}$ | | $n_{2k}$ | $n_2$ |
| . | . | . | . | . | . |
| . | . | . | . | . | . |
| . | . | . | . | . | . |
| $\omega_k$ | $n_{k1}$ | $n_{k2}$ | | $n_{kk}$ | $n_k$ |

## 4. Experimental results

### 4.1. Computational environment

All necessary experimental calculations were performed on an Intel computer with Intel core 2, Quad@2.4 GHz, 2GB RAM, 160GB HDD. IMMR and MMR algorithms are developed in the R environment with the support of the RoughSets package.

### 4.2. Experimental data sets of calculation results

We conducted an IMMR test with real datasets, including Zoo, Mushroom, and Car Evaluation taken from the UCI machine learning dataset (UCI, 2013) and compared the clustering results obtained against the results given by MMR. Information about these datasets and the calculation results is as follows:

***Zoo dataset***

The Zoo dataset contains 101 objects; each object belongs to an animal species, described by 18 taxonomic categories. Subjects were pre-classified into seven classes (mammals, birds, etc.). Since each animal belongs to one of the seven classes, Parmar et al. (2007) tested for MMR the number of clusters to collect $k = 7$. The clustering results given by MMR on the Zoo dataset are summarized in Table 3.

Out of 101 objects, there are 3+39+1+13+10+6 +20=92 clustering objects with majority class labels, so the Overall purity of the clustering result is given by the MMR algorithm is 92/101=91%.

Also with the Zoo dataset, the clustering result with our proposed IMMR is shown in Table 4.

**Table 3:** Results of clustering by MMR for Zoo dataset

| Clusters found | Allocate objects in classes | | | | | | | Purity |
|---|---|---|---|---|---|---|---|---|
| | Class 1 | Class 2 | Class 3 | Class 4 | Class 5 | Class 6 | Class 7 | |
| Cluster 1 | 0 | 0 | 3 | 0 | 3 | 0 | 0 | 0.50 |
| Cluster 2 | 39 | 0 | 0 | 0 | 0 | 0 | 0 | 1.00 |
| Cluster 3 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0.50 |
| Cluster 4 | 0 | 0 | 1 | 13 | 0 | 0 | 0 | 0.93 |
| Cluster 5 | 0 | 0 | 0 | 0 | 0 | 2 | 10 | 0.83 |
| Cluster 6 | 2 | 0 | 0 | 0 | 0 | 6 | 0 | 0.75 |
| Cluster 7 | 20 | 0 | 0 | 0 | 0 | 0 | 0 | 1.00 |

**Table 4:** Results of clustering by IMMR for Zoo dataset

| Clusters found | Allocate objects in classes | | | | | | | Purity |
|---|---|---|---|---|---|---|---|---|
| | Class 1 | Class 2 | Class 3 | Class 4 | Class 5 | Class 6 | Class 7 | |
| Cluster 1 | 41 | 0 | 0 | 0 | 0 | 0 | 0 | 1.00 |
| Cluster 2 | 0 | 20 | 0 | 0 | 0 | 0 | 0 | 1.00 |
| Cluster 3 | 0 | 0 | 0 | 0 | 0 | 8 | 3 | 0.73 |
| Cluster 4 | 0 | 0 | 0 | 0 | 0 | 0 | 7 | 1.00 |
| Cluster 5 | 0 | 0 | 1 | 13 | 0 | 0 | 0 | 0.93 |
| Cluster 6 | 0 | 0 | 3 | 0 | 4 | 0 | 0 | 0.57 |
| Cluster 7 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1.00 |

With IMMR, out of 101 objects, there are 41+20+8+7+13+4+1=94 objects clustered with majority class labels, so the overall purity of the clustering results given by IMMR is 94/101=93%. Thus, the Overall purity by IMMR is 2% higher than that by MMR.

***Mushrooms dataset***

The Mushroom dataset contains 8124 objects, where each object contains information about a mushroom. Mushroom has 22 taxonomic properties, each corresponding to a physical characteristic of the fungus. Each subject belonged to one of two types of mushrooms: Edible (4208 subjects) and poisonous (3916 subjects). Parmar et al. (2007) tested the MMR algorithm on the Mushroom dataset with 20 clusters (k=20). Their test resulted in an overall purity of 84%. Table 5 briefly shows the clustering results by our proposed IMMR.

Out of 8124 objects, there are 7386 objects belonging to the majority class label. Therefore, the Overall purity of clustering by IMMR is 7386/8124=91%, indicating that the Overall purity by IMMR is 7% higher than that by MMR.

***Car evaluation dataset***

The Car evaluation dataset has 1728 objects. Each object is described by 6 categorical categories and can belong to four classes: unacc (1210 objects), acc (384 objects), good (69 objects, and v-good (65 objects). The MMR algorithm results in an overall purity of 70%, whereas our proposed IMMR results in an overall purity of 72% as shown in Table 6.

The experimental results on the above actual data sets show that the IMMR algorithm gives better clustering results than the MMR algorithm.

## 5. Conclusion

Most algorithms clustering categorical data fail to handle the uncertainty in the data sets. To overcome such shortcomings, we propose an improved version of the MMR algorithm by removing all the single-valued categories before clustering and considering

the sum of the entropy of all the categories on each node to determine which node needs further dichotomized. The experimental results with actual datasets show that our proposed algorithm IMMR gives better clustering results than the MMR algorithm, indicating that IMMR can be used successfully in the clustering of categorical data.

**Table 5:** IMMR clustering results for the Mushroom dataset

| Clusters found | Allocate instances in classes | | Purity |
|---|---|---|---|
| | Class 1 | Class 2 | |
| Cluster 1 | 192 | 18 | 0.9143 |
| Cluster 2 | 0 | 8 | 1.000 |
| Cluster 3 | 0 | 18 | 1.000 |
| Cluster 4 | 528 | 72 | 0.8800 |
| Cluster 5 | 2528 | 552 | 0.8210 |
| Cluster 6 | 816 | 96 | 0.8947 |
| Cluster 7 | 0 | 1296 | 1.000 |
| Cluster 8 | 0 | 1728 | 1.000 |
| Cluster 9 | 0 | 32 | 1.000 |
| Cluster 10 | 0 | 96 | 1.000 |
| Cluster 11 | 48 | 0 | 1.000 |
| Cluster 12 | 32 | 0 | 1.000 |
| Cluster 13 | 32 | 0 | 1.000 |
| Cluster 14 | 16 | 0 | 1.000 |
| Cluster 15 | 4 | 0 | 1.000 |
| Cluster 16 | 4 | 0 | 1.000 |
| Cluster 17 | 4 | 0 | 1.000 |
| Cluster 18 | 2 | 0 | 1.000 |
| Cluster 19 | 1 | 0 | 1.000 |
| Cluster 20 | 0 | 1 | 1.000 |

**Table 6:** IMMR clustering results for the Car Evaluation dataset

| Clusters found | Allocate objects in classes | | | | Purity |
|---|---|---|---|---|---|
| | Class 1 | Class 2 | Class 3 | Class 4 | |
| Cluster 1 | 0 | 0 | 576 | 0 | 1.00 |
| Cluster 2 | 198 | 36 | 312 | 30 | 0.54 |
| Cluster 3 | 45 | 9 | 138 | 0 | 0.72 |
| Cluster 4 | 141 | 24 | 219 | 0 | 0.66 |

## Compliance with ethical standards

## Conflict of interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

## References


Bello R and Falcon R (2017). Rough sets in machine learning: A review. In: Wang G, Skowron A, Yao Y, Ślęzak D, and Polkowski L (Eds.), Thriving rough sets: 87-118. Volume 708, Springer, Cham, Switzerland. https://doi.org/10.1007/978-3-319-54966-8_5

Cao F, Liang J, and Bai L (2009). A new initialization method for categorical data clustering. Expert Systems with Applications, 36(7): 10223-10228. https://doi.org/10.1016/j.eswa.2009.01.060

Gibson D, Kleinberg J, and Raghavan P (2000). Clustering categorical data: An approach based on dynamical systems. The VLDB Journal, 8(3): 222-236. https://doi.org/10.1007/s007780050005

Guha S, Rastogi R, and Shim K (2000). ROCK: A robust clustering algorithm for categorical attributes. Information Systems, 25(5): 345-366. https://doi.org/10.1016/S0306-4379(00)00022-3

Han J and Kamber M. (2006). Data mining: Concepts and techniques. 2nd Edition, Morgan Kaufmann Publishers, Burlington, USA.

Hassanein WA and Elmelegy AA (2014). Clustering algorithms for categorical data using concepts of significance and dependence of attributes. European Scientific Journal, 10: 381-400.

Herawan T, Deris MM, and Abawajy JH (2010). A rough set approach for selecting clustering attribute. Knowledge-Based Systems, 23(3): 220-231. https://doi.org/10.1016/j.knosys.2009.12.003

Huang Z (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Mining and Knowledge Discovery, 2(3): 283-304. https://doi.org/10.1023/A:1009769707641

Ienco D, Pensa RG, and Meo R (2012). From context to distance: Learning dissimilarity for categorical data clustering. ACM Transactions on Knowledge Discovery from Data (TKDD), 6(1): 1-25. https://doi.org/10.1145/2133360.2133361

Jensen R and Shen Q (2008). New approaches to fuzzy-rough feature selection. IEEE Transactions on Fuzzy Systems, 17(4): 824-838. https://doi.org/10.1109/TFUZZ.2008.924209

Jyoti D. (2013). Clustering categorical data using rough sets: A review. International Journal of Advanced Research in IT and Engineering, 2(12): 30-37.

Khandelwal G and Sharma R (2015). A simple yet fast clustering approach for categorical data. International Journal of Computer Applications, 120: 25-30. https://doi.org/10.5120/21321-4341

Kim DW, Lee KH, and Lee D (2004). Fuzzy clustering of categorical data using fuzzy centroids. Pattern Recognition Letters, 25(11): 1263-1271. https://doi.org/10.1016/j.patrec.2004.04.004

Mazlack LJ, He A, and Zhu Y (2000). A rough set approach in choosing partitioning attributes. In the 13th ISCA International Conference on Parallel and Distributed Computing Systems, Las Vegas, USA: 1-6.

McCaffrey J (2013). Data clustering using entropy minimization. Visual Studio Magazine, California, USA.

Mesakar SS and Chaudhari MS (2012). Review paper on data clustering of categorical data. International Journal of Engineering Research and Technology, 1(10): 1-18.

Parmar D, Wu T, and Blackhurst J (2007). MMR: An algorithm for clustering categorical data using rough set theory. Data and Knowledge Engineering, 63(3): 879-893. https://doi.org/10.1016/j.datak.2007.05.005

Pawlak Z (1991). Rough sets: Theoretical aspects of reasoning about data. Springer Science and Business Media, Berlin, Germany.

Qin H, Ma X, Zain JM, and Herawan T (2012). A novel soft set approach in selecting clustering attribute. Knowledge-Based Systems, 36: 139-145. https://doi.org/10.1016/j.knosys.2012.06.001

UCI (2013). Machine learning databases. Available online at: https://archive.ics.uci.edu/ml/machine-learning-databases

Zhang Q, Xie Q, and Wang G (2016). A survey on rough set theory and its applications. CAAI Transactions on Intelligence Technology, 1(4): 323-333. https://doi.org/10.1016/j.trit.2016.11.001