See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/337863410

Clustering Categorical Data: A Survey

Article *in* International Journal of Information Technology and Decision Making · December 2019

DOI: 10.1142/S0219622019300064

CITATIONS
0

READS
229

3 authors:

Sami Naouali

24 PUBLICATIONS **61** CITATIONS

SEE PROFILE

Semeh Ben Salem

Military Research Center MoD Tunisia

16 PUBLICATIONS **50** CITATIONS

SEE PROFILE

Zied Chtourou

Center for Military Research

58 PUBLICATIONS **509** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project

Smart Grid View project

Project

IBM WATSON services for Artificial Intelligence View project

All content following this page was uploaded by Semeh Ben Salem on 16 December 2019.

The user has requested enhancement of the downloaded file.

This copy corresponds to the manuscript accepted for publication in the International Journal of Information Technology & Decision Making. This version is uploaded by the authors and we suggest all persons interested in this research to get the final version directly from the web site of the editor World Scientific

CLUSTERING CATEGORICAL DATA: A SURVEY

SAMI NAOUALI

Virtual Reality and Information Technologies,
Military Academy of Fondouk Jedid, Nabeul, Tunisia
snaouali@gmail.com

SEMEH BEN SALEM

Polytechnic School of Tunisia,
La Marsa, Tunis B.P 743, Rue El Khawarizmi, 2078/Tunisia
semehbensalem0@gmail.com

ZIED CHTOUROU

Digital Research Center of Sfax
B.P. 275, Sakiet Ezzit,Sfax 3021, Tunisia
ziedchtourou@gmail.com

,

Clustering is a complex unsupervised method used to group most similar observations of a given dataset within the same cluster. To guarantee high efficiency, the clustering process should ensure high accuracy and low complexity. Many clustering methods were developed in various fields depending on the type of application and the data type considered. Categorical clustering considers segmenting a dataset with categorical data and was widely used in many real-world applications. Thus several methods were developed including hard, fuzzy and rough set based methods. In this survey, more than thirty categorical clustering algorithms were investigated. These methods were classified into hierarchical and partitional clustering methods and arranged in terms of their accuracy, precision and recall to identify the most prominent algorithms. Experimental results show that rough set based clustering methods provided better efficiency than hard and fuzzy methods. Besides, methods based on the initialization of the centroids also provided good results.

Keywords: unsupervised learning, categorical data clustering, rough set theory, fuzzy clustering, hard clustering.

1. Introduction

Clustering[1-4] permits identifying dense and sparse data agglomerations within a given dataset to highlight the overall distribution of the patterns and correlations among the attributes. Clustering is applied in various fields such as market research[5], social science and social networks[6], image processing[7], etc. The clustering process can be defined as follows[8]: "*given a representation of N objects, find K groups based on a measure of similarity such that the similarities between objects in the same group are high while the similarities between objects in different groups are low.*" Clustering can then considered as an optimization problem aiming to minimize (maximize) a measure of similarity (dissimilarity) between the observations within the dataset. Many clustering reviews were published in various fields[9-16], however, to our knowledge, no earlier surveys addressed the topic of categorical clustering. Since the publication of the *k*-modes in 1998, several variants based on the same paradigm were developed[17-30]. On the other hand, many real-world applications require uncertainty based models to arrange an observation into more than only one cluster such as the case of disease and weather forecast classification. Unfortunately, traditional hard clustering methods are unable to handle uncertainty in their process, which prevent them from being used in such critical tasks. To overcome this limitation, fuzzy methods such as the fuzzy *k*-modes were published since 1999[20,31-38]. Unfortunately, these methods suffer from stability issues and multiple runs are required to ensure their steadiness. To fix this limitation, the Rough Set Theory (RST)[39] was proposed to ensure both: stability and uncertainty[40-41]. Generally speaking, many of the proposed methods in this survey are enhancements of the original *k*-modes according to various perspectives: *Distance measure based methods*[31,42] *, initialization methods*[31,43-45], *Genetic Algorithms*[34,46], *the between-cluster information*[35,47], etc.

Clustering is a challenging field of research in which special requirements need to be addressed and can be summarized as follows:

- *Scalability*: Many clustering methods perform well on small datasets. However, dealing with a large database is also a requirement in many applications, especially in the context of Big Data. Highly scalable clustering algorithms are thus necessary.
- *Data type diversity:* several clustering algorithms can only deal with numerical data. However, in many real-world applications, other types of data (binary, categorical, ordinal, mixed) may exist and must be analyzed.
- *Shape clusters discovery:* most of the methods use Euclidean or Manhattan distance in their process that tends to find only clusters with a spherical convex shape. Unfortunately, this is not convenient since a cluster may have any shape and thus, it is important to develop algorithms that can detect clusters of arbitrary shape.
- *Input parameters:* in many applications, input parameters need to be added for clustering (number of desired clusters, fuzziness index, etc.). The final clusters may be quite sensitive to the input parameters that are often difficult to determine, especially for high-dimensional datasets.
- *Dealing with noisy data:* outliers are part of the dataset and are difficult to identify. Noisy data may have a negative impact on the quality of the clusters and thus developed methods should be able to handle these outliers and discard them from the clustering if necessary.
- *High dimensionality:* Data Warehouses are used to store datasets with several dimensions (attributes). Proposed clustering methods should consider the challenge of finding clusters of data objects in a high-dimensional space.

Since many categorical clustering algorithms were developed based on various theories, it is significant to make their inventory and spot the most effective ones to outline new research directives in the field. This survey represents a classification of the most common categorical clustering methods developed since the publication of the *k*-modes 20 years ago. It is tough to provide a complete list of all these techniques due to their diversity and the rapid growth of newly developed methods in the field.

The rest of this paper is organized as follows: Section 2 introduces the clustering paradigm and the motivations behind considering such a topic. In sections 3, 4 and 5, we focus on the challenge of classifying three types of categorical clustering algorithms: hard, fuzzy and rough set respectively either into partitional or hierarchical methods and a detailed overview of the most recent techniques for each category is provided. In section 6, we present the conducted experiments and their interpretation. Finally, a discussion and a conclusion are given based on the obtained results.

2. Categorical clustering and motivations

2.1. *Categorical Information System*

A categorical information system is a quadruple $IS= (U, A,V, f)$ where:

- $U= \{obs_1, obs_2, . . ., obs_N\}$ is a non-empty set of N observations, called the *universe*;
- $A= \{a_1, a_2, . . ., a_d\}$ is a non-empty set of $d=|A|$ categorical attributes;
- V is the union of attribute domains, *i.e* $V= \bigcup_{j=1}^{d} V_{a_j}$ where $V_{a_j}=\{a_j^{(1)},a_j^{(2)},\ldots,a_j^{(n_j)}\}$ is the value domain of the categorical attribute a_j and is finite and unordered, e.g., for any $1\leq p\leq q\leq n_j$, either $a_j^{(p)} = a_j^{(q)}$ or $a_j^{(p)} \neq a_j^{(q)}$. Here n_j is the number of categories (modalities) of attribute a_j for $1\leq j\leq d$;
- $f : RxA\rightarrow V$ is a function, $f\,(x_i,a_j)\in V_{a_j}$ for $1\leq i\leq N$ and $R= V_{a_1} x V_{a_1} x \ldots x V_{a_d}$.

An information system is presented as follows:

Table 1: An information system representation.

U	a_1	a_2	…	$a_{\|A\|}$
obs_1	$f(obs_1, a_1)$	$f(obs_1, a_2)$	⋮	$f(obs_1, a_{\|A\|})$
obs_2	$f(obs_2, a_1)$	$f(obs_2, a_2)$	⋮	$f(obs_2, a_{\|A\|})$
⋮	⋮	⋮	⋮	⋮
obs_N	$f(obs_N, a_1)$	$f(obs_N, a_2)$	…	$f(obs_N, a_{\|A\|})$

According to Table 1, each obs_i is represented by $|A|=d$ attributes having values defined by f and described by a vector: $obs_i=(f(obs_i, a_1), f(obs_i, a_2), f(obs_i, a_3), \ldots , f(obs_i, a_{|A|}))$ for $i=1,2,3, \ldots,N$. In an information table, two distinct observations may have the same attributes, which is *not permissible* in relational databases. Thus, information systems permit generalizing relational databases.

Clustering algorithms can be divided into hierarchical and partitional methods. In hierarchical[74,20,49] clustering, a hierarchical structure is iteratively constructed and nested clusters are progressively identified either in an agglomerative[26,95,38] or divisive[98,99,75,97] way: for agglomerative approaches (top-up), each object is initially arranged in an independent cluster, then most similar pairs of the resulting clusters are merged. For divisive approaches (top-down), all the observations are initially put in one cluster that is recursively divided into smaller ones.

As an application, hierarchical clustering was used for several dog species partitioning[48] where more than 48,000 dog categories were given based on their genetic processes of evolution, as shown in Figure 1. Many groups of dogs were then identified, such as wolves, toy dogs, sight hounds, etc. and some correspondences were also created between them to highlight potential similitudes.

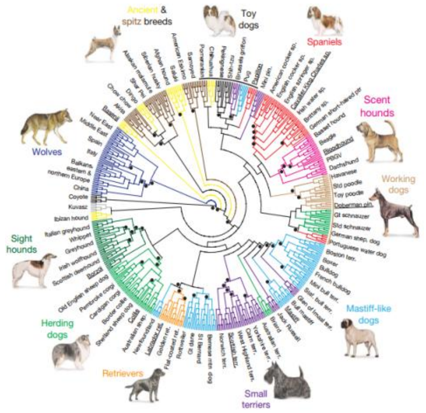

Figure 1: Clustering dog species using hierarchical methods

In addition to their simplicity and ease of implementation, hierarchical clustering methods provide a structure that is more informative than the unstructured set of flat clusters returned by partitional methods. Therefore, it is easier to decide on the number of clusters by directly looking at the dendrogram. Unfortunately, the time complexity and storage requirements of these methods depict their unsuitability for large datasets. Besides, these methods are also very sensitive to outliers. In addition, the user is not required to specify the number of initial clusters. They are also useful if the underlying application has a taxonomy.

In, partitional clustering, the process starts by initially selecting k observations of the datasets as centroids (cluster centers). The algorithm then attempts to iteratively search for k-partitions of the initial dataset. The main advantage of partitional clustering methods is their relative low complexity and high accuracy when compared to hierarchical clustering algorithms. Unfortunately, these methods have some limitations including their unsuitability for nonconvex data, sensitivity to outliers, convergence to a local optimal solutions, the final resulting clusters depend on the initial number of clusters K which is a user-specified parameter and the selection of the initial centroids.

As mentioned, partitional and hierarchical clustering algorithms fall under hard clustering. The main difference between these two types of clustering is the resulting structure of the clustered data set. Whereas partitional clustering results in a structure consisting of discrete partitions, hierarchical clustering results in a tree-like, nested structure.

The clustering process is performed using a distance metric, such as the Euclidean (L_2) and Manhattan (L_1)[28] distances proposed for numeric clustering and the simple matching dissimilarity measure (Hamming distance), proposed in the k-modes[14,15,49-52] algorithms, for categorical clustering. The Hamming distance often results in clusters with weak intra-similarity[53] and thus lower accuracy will be obtained. Consequently, the distance was modified in many variants of the *k*-modes by taking into account the frequency of the attributes in the mode[30,42] in the cluster. Historically, one of the most intuitive methods used for categorical clustering was to convert the categorical attributes into binary values[16] using 0 or 1 to indicate whether the corresponding modality is absent or present in the observation. However, this method was not efficient due to the high dimensionality of the datasets to be generated[54,55].

2.2. The motivation for considering categorical clustering

Categorical clustering has found great applications in various fields including:

- **Healthcare and Medical Sciences:** to evaluate and localize zones of diseases, also known as *cluster alarms*. Detecting these infected regions permits correctly assigning adequate resources for health control and prevention[56,57,58].
- **Computer Science:** to analyze data streams in web applications (online social networks, blogs, wikis, etc.) which would permit group segmentation and target marketing for electronic commerce[59,60]. In cyber-security, clustering can detect intrusions and identify anomalies and malicious content and thus clusters of normal and malicious network traffic can be categorized[61-63].
- **Business, marketing and finance:** to categorize the world's top tourist destinations based on the growth of the main tourism indicators[64] which permits identifying highly desirable destinations. In another application, customer churn is a critical and challenging problem that can affect the telecommunication industry. In fact, retaining already existing customers will generate lower financial impacts than acquiring new ones[65].
- **Social sciences:** Intelligent crime analysis permits identifying regions with high criminology and unlawful activities rates[66]. In[67], the traveling activity of more than 30,000 individuals in Chicago is analyzed. The analysis explored the inherent daily activity structure of the individuals, the variation of their daily activities, and identifies clusters of individual behaviors and their socio-demographic information**.**

Many categorical data repositories in various fields are freely available online such as:

- Nature[*] that provides a huge data archive in many disciplines including biological, health, chemistry, chemical biology, earth, environment, space sciences, etc.
- The National Aeronautics and Space Administration[†] also provides an Astrophysics Data System (ADS) which is a Digital Library portal for researchers in Astronomy and Physics containing more than 14.3 million records covering publications in astronomy and astrophysics, physics.
- The UCI Machine Learning Repository[‡] contains more than 468 datasets with various types and application fields such as life sciences, business, physical sciences, etc. Many categorical datasets stored in this repository were widely used in the surveyed methods given in this study such as the Soybean Small[19-22,31-35,41-45,98,99], Breast Cancer Wisconsin[22,35,43-46,73,98], Zoo[31,32,41-46,75,98,99], Lung Cancer[35,44,45],

[*] https://www.nature.com/sdata/policies/repositories

[†] NASA: https://www.nasa.gov/

[‡] https://archive.ics.uci.edu/ml/index.php

Mushroom[22,35,41-46,73,75,98,99], Congressional Vote[34,42-46,73,75], dermatology[22,35,44], Heart disease[22,35], Credit approval[19,31,35,98], and Letter recognition[35].

This survey represents a good support for researchers interested in the field of categorical clustering in order to identify new research directives. The main contributions of this paper are as follows:

- A classification of more than thirty categorical clustering methods corresponding to the most popular methods into partitional and hierarchical methods.
- These methods are evaluated using three evaluation metrics: accuracy, precision and recall.
- The algorithms are classified into three clustering families: *hard methods*, *fuzzy methods* and *rough set based methods*.

3. Hard Clustering Algorithms

3.1. Partitional categorical clustering methods

3.1.1 The k-modes and its improved distance metrics

The *k*-modes[19] was proposed in 1998 as an extension of the *k*-means published since 1965. It permits avoiding the numeric limitation of the *k*-means according to the following three features:

(i) using a simple matching dissimilarity measure between obs_j and cen_j as defined in Table 2 ;

(ii) Using a frequency-based method to update the modes in each iteration in order to minimize the cost function.

(iii) Replacing the means by modes to compute the centroids of the clusters in every iteration. The mode represents the most frequent categorical value in an attribute;

A mode of a categorical dataset U described by d attributes is a vector $Q = [q_1, q_2, \ldots, q_d]$ that minimizes the following quantity (according to the definition of Huang[19]):

$$\mathrm{L}(U,Q)=\sum_{i=1}^{N} D\,(obs_i,Q) \tag{3.1}$$

$Q=\{cen_1, cen_2, \ldots, cen_K\}$ represents the mode of the cluster and $W=[w_{ji}]$ is a {0,1} matrix that represents the current membership of an observation.

To improve the efficiency of the *k*-modes, He (2005)[42] and Ng (2007)[23] modified the simple matching dissimilarity measure by incorporating the relative frequency of the attributes in the distance metric as given in Table 2 in order to recognize clusters with weak intra-similarity.

Table 2: Objective function and dissimilarity measures for hard categorical clustering methods.

Objective function	$\mathrm{P}(W,Q)=\sum_{l=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{d}\omega_{il}\mathcal{D}(obs_{ij},cen_{lj})$	$\sum_{j=1}^{K} w_{ji}=1$ and $0<\sum_{i=1}^{N} w_{ji}<N$
The simple matching dissimilarity measure	$\mathcal{D}(cen_j,obs_j)=\sum_{j=1}^{d}\delta(cen_j,obs_j)$	$\delta(cen_j,obs_j)=\begin{cases}1, \text{ if } cen_j \neq obs_j\\ 0, \text{ if } cen_j = obs_j\end{cases}$

Ng's distance	$\delta_{Ng}\left(cen_{1\le j\le d},obs_j\right)=\begin{cases}1, \text{ if } cen_j\neq obs_j \\ 1-\dfrac{\lvert c_{ljr}\rvert}{\lvert c_l\rvert} & otherwise\end{cases}$	$\lvert C_l\rvert=\lvert\{i\mid\omega_{li}=1\}\rvert,\ 1\le i\le N$ is the number of observations in the l^{th} cluster$1\le l\le K$ $\lvert C_{ljr}\rvert=\left\lvert\left\{\omega_{ls}\mid z_{lj}=x_{sj}=a_j^{(r)},\omega_{ls}=1\right\}\right\rvert$ is the cardinality of the observations with category $a_j^{(r)}$ of the j^{th} attribute in the l^{th} cluster.
He's distance measure	$\delta_{He}\left(cen_j,obs_j\right)=\begin{cases}1-f_r\left(A_j=cen_j\mid C_l\right), \text{ if } obs_j=cen_j \\ 1, \text{ otherwise}\end{cases}$	f_r $(A_j= cen_j \mid C_l)$ is the frequency of the j^{th} modality in the l^{th} cluster of the mode denoted cen_j

However, according to the experiments conducted on the Ng's distance metric[23], it was shown that although the proposed algorithm is scalable, *i.e*, the computational time increase linearly with respect to either the number of attributes, categories, cluster or objects, the Ng's *k*-modes requires more computational time than the original *k*-modes due to the additional arithmetic computations required.

3.1.2 Initialization methods for the k-modes

The *k*-modes is based on randomly selecting the initial centroids during the first step of the clustering process, which can affect the quality of the final results. Performing several runs of the algorithm with various sets of initial centroids is then necessary to ensure steadiness and detect the most convenient centroids. To avoid this limitation, many centroid initialization methods were proposed[68-71,43,44]. Although their efficiency, these methods suffer greatly from their high quadratic time complexity.

The IBD[43] is an Initialization method Based on the Density that uses the average density of an attribute in a given observation. The density of a given observation is defined as follows:

$$\text{Dens}(obs_i)=\frac{\sum_{a\in A} Dens_a(obs_i)}{\lvert A\rvert}=\frac{\sum_{a\in A}\lvert\{y\in U\mid f(obs_i,a)=f(y,a)\}\rvert}{\lvert A\rvert\lvert U\rvert} \qquad (3.2)$$

where

$$\frac{1}{\lvert U\rvert}\le Dens(obs_i)\le 1$$

if $\lvert\{y\in U\mid f(obs_i,a)=f(y,a)\}\rvert=1$ then $Dens(obs_i)=\frac{1}{\lvert U\rvert}$ and if $\lvert\{y\in U\mid f(obs_i,a)=f(y,a)\}\rvert=\lvert U\rvert$ then $Dens(obs_i)$=1. In the universe, a high-density $Dens(obs_i)$ corresponds to a high number of objects located around $obs_{i,}$ which means more chance for obs_i to be considered as a centroid.

The Cluster Center Initialization *k*-modes (CCI*k*-M)[44] is also an initialization method that uses different attribute values according to one of the following methods:

- *The Vanilla method*: all the attributes are considered during the clustering process;
- *Prominent attributes*: only a few numbers of attributes (<*K)* will possess a higher discriminatory power and will play a significant role in determining the initial centroids**.**
- *Significant attributes*: the most important step is to find the distance between any two categorical values of an attribute. This distance is computed as a function of their overall distribution and co-occurrence with other attributes.

It is also possible to use outlier detection methods as an initialization technique[45]. This permits initially identifying the outliers and discard them from being selected as initial centroids. Two methods were developed in this context: *Ini_Distance* using the distance-based outlier detection technique, and the *Ini_Entropy* using the partition entropy-based outlier detection technique within the framework of rough sets.

The main advantage of these initialization methods is their ability to avoid the random selection of the modes in the first step of the algorithm. Thus, more stable and accurate results can be obtained and multiple runs with multiple initial modes are no longer required to find the most suitable partitions. For the IBD[43], the time complexity is of $O(NdK^2)$, which is linear with respect to the number of data objects N and number and dimensions d. Thus, the method can be used effectively to cluster large categorical and high dimensional datasets. Unfortunately, the IBD may not be as scalable as required if the number of clusters to be identified in the dataset is considerable. Besides, in order to calculate the density of a point, the algorithm calculates the summary of all the other points. Hence, there is information loss that may lead to improper density calculation, which can affect the results[44]. The biggest advantage of the CCI*k*-M[44] is the worst-case log-linear time complexity of computation and fixed choice of initial cluster centers from dense, localized regions. For the outlier detection method[45], additional parameters are required including the degree of outlierness of each initial candidate center and the distances between initial candidate centers and all currently existing initial centers which can be considered as a limitation.

3.1.3 The G-ANMI

The Genetic Average Normalized Mutual Information (*G-ANMI*)[46] is based on the Genetic Algorithms (GAs) paradigm[72]**.** GAs have shown their great efficiency when dealing with optimization problems. In GAs, potential solutions are encoded in the form of strings called chromosomes. Initially, a population composed of P chromosomes is generated. This population will then evolve over various steps to create multiple generations. During each step, three genetic operators, *i.e.* selection, crossover and mutation, can be applied to produce new individuals potentially corresponding to more accurate solutions. Each chromosome will be evaluated using a fitness value represented by the ANMI[73]. Chromosomes with the highest fitness value will be included in the next generation. The use of GAs permits overcoming the optimal local solution generated when using the *k*-modes.

Unfortunately, GAs based clustering methods require additional parameters including the number of initial chromosomes in each population, the crossover rate, the mutation rate, the random seed, etc. Besides, the computation of the fitness value is very time-consuming since a whole scan of the dataset is required.

3.1.4 Integrating the between-cluster similarity

In the original *k*-modes, the centroids are updated considering only the within-cluster information, *i.e.*, the within-cluster compactness. The between-cluster information, *i.e.*, the between-cluster separation, is not considered, which may produce weak clustering results. The Between Cluster *k*-Modes (BC*k*-M)[47] was proposed to minimize the within-cluster dispersion and enhance the between-cluster separation by adding the following term to the objective function of the *k*-modes:

$$\mathrm{B}(W,Z)=\sum_{l=1}^{K}\sum_{i=1}^{N}\omega_{li}S(cen_l) \tag{3.3}$$

$S(cen_l)$ denotes the similarity between the l^{th} cluster and other clusters defined as follows:

$$\mathrm{S}(cen_l)=\frac{1}{N}\sum_{i=1}^{N}s(cen_l,obs_i)=\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{d}\phi_0^{a_j}(cen_l,obs_h)=\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{d}1\text{-}\delta_0^{a_j}(cen_l,obs_h) \tag{3.4}$$

$\delta_0^{a_j}(cen_l,obs_h)$ represents the similitude between obs_h and cen_l.

The modified objective function then becomes as follows:

$$\mathrm{F}(W,Z,\gamma)=\sum_{l=1}^{k}\sum_{i=1}^{N}\omega_{li}d(cen_l,obs_i)+\gamma\sum_{l=1}^{k}\sum_{i=1}^{N}\omega_{li}S(cen_l) \tag{3.5}$$

γ is used to maintain a balance between the effect of the within-cluster information and that of the between-cluster information.

The main limitations of the BC*k*-M[47] are the value of the parameter γ that should not be too large so that the between-cluster similarity term would not dominate the clustering process. Therefore, it is suggested to have a value of γ such that $0<\gamma<1$. The appropriate setting of γ depends on the domain knowledge of the dataset, it is difficult to choose a suitable value directly. On the other hand, the experiments show that the BC*k*-M requires more computational times than the original *k*-modes, which is an expected outcome since it requires the more additional arithmetical operations of the between-cluster information. Moreover, the BC*k*-M is scalable, *i.e.*, the computational times increase linearly with respect to either the number of objects, attributes or clusters. Therefore, it can cluster large categorical data efficiently.

3.2. *Hierarchical categorical clustering methods*

In this section, four hierarchical categorical clustering algorithms are given: ROCK, SQEEZER, MGR and DHCC.

ROCK[26] is an agglomerative method that uses the concept of links to merge two clusters. During the clustering process, clusters with a higher number of links are merged. The main features of ROCK are as follows:

- *Random sampling:* it is used to deal with large datasets and permits reducing the main memory resources to be allocated, which may have significant improvements in the execution time.
- *Handling outliers:* outliers correspond to observations that are relatively isolated from the rest of the points. ROCK efficient when it comes to predicting the existence of such type of data and thus can be deleted.

Squeezer[49] is also a categorical clustering methods bas on hierarchies that permits: (*i*) handling outliers, (*ii*) clustering data streams by reading each observation in sequence, (*iii*) performing the clustering in only one scan which makes it very efficient for disk-resident datasets. Considering a set of *d* categorical attributes with domains D_1, D_2, …, D_d and *TID* be the set of unique IDs of all tuples. For each $tid \in TID$, the attribute value of A_i of the corresponding tuple is represented as $val(tid, A_i)$. The similarity between an observation and a cluster is defined as follows:

$$\mathrm{sim}(tid,\ C)=\sum_{i=1}^{m}\left(\frac{sup(a_i)}{\sum_j sup(a_j)}\right) \tag{3.6}$$

$tid.A_i{=}a_i$ and $a_j \in VAL_i(C)$

Given a Cluster C, and $a_i{\in}D_i$, the support of a_i in C with respect to A_i is defined as $sup(a_i){=}|\{tid|tid.A_i{=}A_i\}|$. Besides, the *d-Squeezer*, which is a variant of the *squeezer*, was also proposed to handle larger datasets.

In the Mean Gain Ratio (MGR)[74], the partitions are derived from the attributes instead of the observations. Each attribute is evaluated according to its ability to share the most information with the partitions defined by other attributes. Then, the equivalence class of that attribute is generated to output a cluster. These two steps are repeated until all the objects are assigned to a cluster. Given an attribute a_i and its corresponding partition on the universe U defined by $U/a_i{=}\{obs_1, \ldots, obs_j, \ldots, obs_h\}$ where h represents the number of observations contained in that partition. The concepts of information gain and gain ratio are used to measure the similarity between that partition and the partitions defined by all the other attributes. The main advantage of this algorithm is the possibility to either specify or not the number of clusters while few existing clustering methods verify this condition.

The DHCC[75] (Divisive Hierarchical Clustering Algorithm for Categorical data) was proposed using the Multiple Correspondence Analysis and a procedure based on the Chi-square distance as an objective function to initialize and refine the splitting of clusters defined as follows:

$$\mathrm{S}CE=\sum_{k=1}^{K}\sum_{Z_i \in C_k} d_{chi}(obs,Cen_k) \tag{3.7}$$

$d_{Chi}(obs_i,\ Cen_k)$ is the Chi-square distance between object obs_i and centroid Cen_k, which is defined as follows:

$$d_{chi}(Z_i,\ C_k)=\sum_{j}\frac{\left(obs_{ij}\text{-}\overline{cen}_{kj}\right)^2}{\overline{cen}_{kj}} \tag{3.8}$$

$\overline{cen}_{kj}$ $(1{\leq}j{\leq}J)$ is the j^{th} element of the centroid cen_k, and object obs_i is in cluster C_k.

The main limitation of the previous methods is their inability to handle uncertainty inherent in data especially when the considered application may contain observations that can be assigned to more than one cluster simultaneously such as weather forecast and disease prediction. In fact, one of the most difficult problems in clustering is the classification of boundary data, that is, data located in the outer block of each cluster. Such data are more likely to be either misclassified to an incorrect cluster or assigned the same distance values to two neighboring clusters. This issue is not considered in hard clustering but was fixed in fuzzy clustering.

4. Fuzzy-Based clustering

In fuzzy clustering, each observation is assigned a belonging label value in the interval [0,1] instead of the two values 0 or 1 used in hard clustering[76]. This is particularly useful when the boundaries among the clusters are not well separated and ambiguous.

4.1. *Partitional fuzzy clustering*

4.1.1 The Fuzzy k-modes (FkM)

The Fuzzy *k*-modes[20] is a generalized version of the *k*-modes that incorporates fuzzy sets in the clustering process. The fuzzy objective function then becomes:

$$F_c(W,Z)=\sum_{j=1}^{K}\sum_{i=1}^{N} w_{ji}^{\alpha} d\left(cen_j,obs_i\right) \tag{4.1}$$

α is called the fuzziness index allows observation to have membership functions to all clusters and d is the simple dissimilarity measure as defined in the *k*-modes.

Although its efficiency, the fuzzy *k*-modes has the same drawbacks of the *k*-modes: (*i*) the initialization problem of the centroids, (*ii*) the final local solution obtained and (*iii*) the adjusting of an additional control parameter of the membership fuzziness.

4.1.2 Fuzzy k-Modes with Fuzzy Centroids (Fk-MFC)

Although the fuzzy *k*-modes provided efficient clustering results, using a hard centroid assignment reduces its precision and ability to classify boundary data. In order to avoid this limitation, the fuzzy *k*-Modes with Fuzzy Centroids (F*k*-MFC)[31] was proposed. The fuzzy theory was applied in both: the clustering paradigm and the assignment of the initial centroids. Considering a dataset U composed of N observations and described by d categorical attributes A_l ($1 \leq l \leq d$) with domain values denoted by $DOM(A_l)=\left\{a_l^{(1)},a_l^{(2)},\ldots,a_l^{(n_l)}\right\}$ where n_l represents the number of modalities of the attribute A_l. Each attribute of a fuzzy centroid $\widetilde{V}$ has a fuzzy category value defined as $\widetilde{V}=[\tilde{v}_1,\ldots,\tilde{v}_l,\ldots,\tilde{v}_d]$ where:

$$\begin{gathered}\tilde{v}_l=\frac{a_l^{(1)}}{\omega_l^{(1)}}+\frac{a_l^{(2)}}{\omega_l^{(2)}}+\ldots+\frac{a_l^{(t)}}{\omega_l^{(t)}}+\ldots+\frac{a_l^{(n_l)}}{\omega_l^{(n_l)}}\\ 0 \leq \omega_l^{(t)} \leq 1 \text{ for } 1 \leq t \leq n_l\\ \textstyle\sum_{t=1}^{n_l}\omega_l^{(t)}=1,\ 1 \leq l \leq d.\end{gathered} \tag{4.2}$$

The distance measure proposed for the *Fk-MFC* is defined as follows:

$$\mathrm{d}\left(\widetilde{V},obs\right)=\sum_{l=1}^{d}\delta(\tilde{v}_l,obs_l)=\sum_{l=1}^{d}\sum_{t=1}^{n_l}\tau\left(a_l^{(t)},obs_l\right) \tag{4.3}$$

where

$$\tau\left(a_l^{(t)},obs_l\right)=\begin{cases}0,\ a_l^{(t)}=obs_l\\ \omega_l^{(t)},a_l^{(t)}\neq obs_l\end{cases} \tag{4.4}$$

$$\omega_l^{(t)}=\sum_{j=1}^{N}\gamma\left(x_{jl}\right) \tag{4.5}$$

$$\gamma\left(x_{jl}\right)=\begin{cases}\mu_{ij}^{\alpha},\ a_l^{(t)}=x_{jl}\\ 0,\ a_l^{(t)}\neq x_{jl}\end{cases} \tag{4.6}$$

μ_{ij}^{α} is the membership degree of observation obs_j to the i^{th} cluster and α is a parameter that controls the fuzziness of membership of each observation.

In the F*k*-MFC, 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, in addition to the membership degree μ_{ij}^{α} required in all fuzzy methods, the F*k*-MFC requires specifying a random membership value $\omega_l^{(t)}$ for each initial centroid. The selection of these initial centroids is always random, which leads to unstable results. Besides, in the experiments[31], it was demonstrated that the execution time of the F*k*-MFC is faster than that of the fuzzy *k*-modes due to the fewer iterations required for the convergence of the F*k*-MFC.

4.1.3 The Fuzzy k-partitions, modified fuzzy k-partitions and fuzzy genetic k-modes

The fuzzy *k*-partitions (F*k*P)[32] was proposed based on the likelihood function of multivariate multinomial distributions and a modified version was given[33]. Partitioning $P = \{P_1, \ldots, P_K\}$ of $\mathcal{D}$ into K classes can be represented using mutually disjoint sets $P_1, \ldots, P_K$ such that $P_1 \cup \cdots \cup P_K = \mathcal{D}$ or equivalently using the indicator functions $z_1, \ldots, z_K$ such that $z_k(obs) = 1$ if $obs \in P_k$ and $z_k(obs) = 0$ otherwise, for all the observations in $\mathcal{D}$ and all $k = 1, \ldots, K$. Considering a set of observations $obs_1, \ldots, obs_N$ as a random sample of size N from a distribution $f(obs;\lambda)$, whereλ represents the probability of a response l for the j^{th} attribute by the i^{th} individual with the k^{th} extreme profile is λ_{kjl}, *i.e.* $P(obs_{ijl}=1|obs_i$ in k class$) = \lambda_{kjl}$ and f represents the likelihood function (for more details on this algorithm refer to the reference).

In[33] a Modified version of the Fuzzy *k*-Partition based on an INDiscernibility relation (MF*k*-PIND) was proposed using the indiscernibility relation that induces an approximation space constructed by equivalence classes of indiscernible objects. Although the fuzzy *k*-modes are very effective for categorical clustering, the final partitions represent a locally optimal solution instead of a global one, which is a significant limitation. To overcome this restriction, a hybrid Genetic Fuzzy *k*-Modes (GF*k*-M) algorithm was proposed[34] integrating both, the genetic algorithm and the fuzzy *k*-modes and preserving the same objective function than the Fuzzy *k*-modes.

The GF*k*-M requires multiple inputs including the number of clusters K, the initial dataset with N observations, the maximum number of generations G_{max}, the weighting component or the fuzziness index $\alpha > 1$, the particular parameter $\beta \in [0,1]$, the mutation probability P_m … which may be confusing. But also permits converging to a globally optimal solution instead of a local one.

4.1.4 Fuzzy between-cluster information algorithm

A new fuzzy clustering algorithm (NF*k*M) was proposed[35] by adding the between-cluster information to the objective function of the fuzzy *k*-modes. The objective function then becomes as follows:

$$F_n(W,Z,\gamma) = F(W,Z) + \gamma B(W,Z) \tag{4.7}$$

Where

$$\mathrm{F}(W,Z) = \sum_{l=1}^{k} \sum_{i=1}^{n} \omega_{li}^{\alpha}\, d(cen_l, obs_i) \tag{4.8}$$

And

$$\mathrm{B}(W,Z) = \sum_{l=1}^{k}\sum_{i=1}^{N}\omega_{li}^{\alpha}\frac{1}{N}\sum_{p=1}^{N}s\left(cen_l,obs_p\right) \tag{4.9}$$

γ is used to maintain a balance between the effect of the within-cluster information and the between-cluster information.

In[36], a fuzzy Set-Valued-*k*-modes was proposed using the Jaccard coefficient, defined as follows, as a dissimilarity measure.

$$\mathcal{D}\left(obs_i,obs_j\right) = \sum_{s=1}^{d}\delta'\left(obs_{is},obs_{js}\right) = \sum_{s=1}^{d}1-\frac{\left|obs_{is}\cap obs_{js}\right|}{\left|obs_{is}\cup obs_{js}\right|} \tag{4.10}$$

Since the fuzzy SV-*k*-modes extends the fuzzy *k*-modes with set-valued attributes, the same objective function defined for the fuzzy *k*-modes is also used for the new algorithm. The main advantage of the fuzzy Set-Valued-*k*-modes that was not discussed in previous methods is its ability to deal with single-valued and set-valued categorical attributes together which is desired when an object of the dataset may take multiple values in some attributes.

4.2. *Fuzzy hierarchical clustering methods*

In this section, two fuzzy hierarchical clustering methods are detailed: the FHC[37] and the SS-FCC[38].

FHC[37] is based on fuzzy graph connectedness used for high-dimensional and mixed datasets which a great advantage. It first partitions the dataset into several sub-clusters generating a fuzzy graph based on the fuzzy-connectedness degree. The algorithm can deal with numeric datasets using the Euclidean distance or categorical datasets using the Jaccard coefficient.

The SS-FCC[38] is a semi-supervised fuzzy co-clustering algorithm used to categorize large web documents and enable the analysis of large collections of textual data. Document clustering permits automatically organizing text documents into meaningful groups of coherent topics. In this case, although a single document may span multiple topics, the algorithm is bale to perfectly deal with this constraint which is a great benefit. On the other hand, hard clustering approaches using binary memberships are not adequate for such applications and fuzzy methods have been explored in this context and provided good results. In the clustering process, some prior domain knowledge of the dataset is incorporated in the form of user's pairwise constraints in order to increase the clustering accuracy and reduce the sensitivity to fuzzifier parameters. The problem will then be considered as maximizing a competitive agglomeration cost function with fuzzy terms, taking into account the provided domain knowledge. The words that co-occur together in the documents tend to be linked with comparable concepts and are represented using a word-document vector to formulate the clustering process and build the objective function. Although the SS-FCC is a hierarchical method, its complexity is *O(NdK)* which allows it to be used for clustering large categorical datasets either in terms of the number of observations or dimensions or even for a high number of classes. The most prominent limitation of this fuzzy method is the selection of the value of the fuzziness index that is still problematic.

5. Rough set based clustering

5.1. *Basic concepts of rough set theory*

The Rough Set Theory (RST) was largely used in Machine Learning such as dimensionality reduction[77], classification[78-80], extraction rules[81], anomaly detection[82] predictive analysis[83]**,** regression[84], etc. Although fuzzy clustering permits handling uncertainty in the clustering process, the appropriate value of the fuzziness parameter is problematic. The RST[85,86] was proposed to overcome this limitation since no additional parameters are required as a user-specified values. Hard clustering aims to produce non-overlapping clusters where the objects belong to only one cluster. However, in some real-world applications such as weather forecasting and disease diagnosis, observations may depict different patterns and thus should be put in multiple distinct clusters which can be considered by the RST efficiently[87].

In order to better illustrate the use of the RST, an example of churn prediction in a Telco company is provided in the following Table 3[88-91]. The dataset is composed of six categorical observations (obs_1 → obs_6) described by four attributes (a_1 → a_4). The decision attribute is given by: *y* for **churn** and *n* for **no churn**.

Table 3: Churn prediction in a Telco company.

Objects	a_1	a_2	a_3	a_4	**decision**
obs_1	a	c	b	a	y
obs_2	b	a	b	c	n
obs_3	a	b	b	b	n
obs_4	b	a	b	c	y
obs_5	a	b	b	c	y
obs_6	a	c	b	a	n

According to the decision attribute, two sets can be identified $S=\{obs_i|$ decision $=y\}$ and $S'=\{obs_i|$ decision $=n\}$. Although $\{obs_1,obs_6\}$ and $\{obs_2, obs_4\}$ are indiscernible (similar), their corresponding decision attributes are different and thus they may either belong to *S* or *S'*. When using hard or fuzzy clustering, these two cases will be put in the same group. obs_3 strictly belongs to the decision set *n* and obs_5 to the decision set *y*. By applying the RST, obs_1, obs_2, obs_4 and obs_6 will be part of the upper approximations of both *y* and *n* decision sets, obs_3 will be part of the lower approximation of the decision set *n* and obs_5 will be part of decision set *y*.

The main concepts that guide the development of the RST are the indiscernibility relation, the lower and the upper approximation. These concepts are detailed as follows:

The indiscernibility relation.

For two observations obs_i and obs_j described by a set of attributes *A*, let $B \subseteq A$, obs_i and obs_j are said to be *B*-indiscernible if and only if $f(obs_{i,} a)= f(obs_{j,} a)$ for each $a \in B$.

The lower approximation.

The lower approximation of a subset of observations $X \subseteq U$ and attributes $B \subseteq A$ denoted $B_*(X)$ or $\underline{B}$(X) contains all the objects that surely belong to *U* and is defined as follows:

$$B_*(X)=\bigcup_{obs \in U}\{B(obs):\ B(obs)\subseteq X\}$$

The upper approximation.

The upper approximation of a subset of observations $X \subseteq U$ and attributes $B \subseteq A$ denoted B*(X), or $\overline{B}(X)$ contains all the objects that possibly belong to U defined as follows:

$$B^{*}(X)= \bigcup_{obs \in U} \{B(obs): B(obs) \cap X \neq \emptyset \}$$

The boundary region.

The boundary region with respect to B is defined as follows:

$$BN_B(X)=B^{*}(X) - B_{*}(X) .$$

The accuracy of roughness.

The accuracy of approximation (accuracy of roughness) of any subset X of U with respect to $B \subseteq A$, denoted $\alpha_B(X)$ is measured according to the following equation:

$$\alpha_B(X)= \frac{\left|\underline{B(X)}\right|}{\left|\overline{B(X)}\right|}$$

In Figure 2, the example provided in Table 3 is reconsidered and the upper, lower approximations and boundary region previously defined are given as follows:

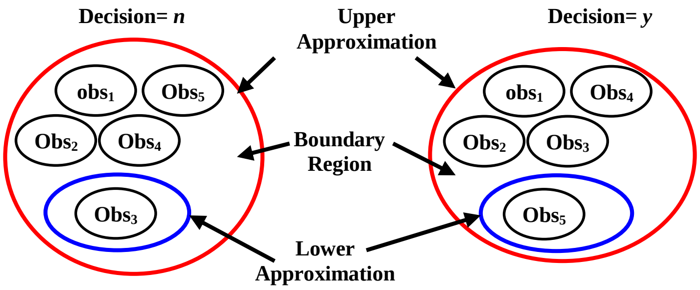

Figure 2: Upper, lower approximations and the boundary region.

As detailed in the previous section 4, the first attempts provided to handle uncertainty in the clustering process were based on fuzzy set theory. However, these methods have a significant restriction related to their stability that limited their use in many real-world applications. In fact, they require considerable computation costs and experiments to adjust the fuzziness index. Rough Set based clustering methods such as the MMR, MMeR, R*k*M, SDR, etc have no such problems which motivated their fast development and implementation.

5.2. *Clustering using upper and lower approximations*

In rough clustering, observations that are close to a particular cluster C_1 and far away from all other clusters will be assigned to the lower approximation of C_1. Each observation that has close distances to at least two nearest cluster centers will be assigned to their upper approximations. In this type of clustering, three methods will be investigated: The Rough *k*-modes (R*k*M)[92], the Entropy-based rough *k*-modes (ER-*k*-modes)[93] and the Rough Set based Fuzzy *k*-Modes (RSF*k*M)[94].

In the Rough *k*-modes (R*k*M)[92], the distance measure used to compute the dissimilarity is given as follows:

$$\mathrm{D}\left(cen_j,\, obs_i\right) = \sum_{l=1}^{d} \emptyset\left(cen_{jl}, obs_{il}\right) \tag{5.1}$$

Where

$$\emptyset\left(cen_{jl}, obs_{il}\right) = \begin{cases} 1 \text{ if } cen_{jl} \neq obs_{il} \\ 1 - \dfrac{\left|C_{jl}\right|}{\left|C_j\right|} \text{ otherwise} \end{cases} \tag{5.2}$$

$|C_{jl}|$ is the number of objects with category value obs_{il} for the l^{th} attribute in the j^{th} cluster.

Let's consider $T = \{l : d\ (cen_l,\, obs_i)/d\ (cen_j,\, obs_i) \leq \varepsilon \text{ and } l \neq j\}$ as the set of centroids that are close to a given observation. If $T \neq \emptyset$ then the observation would be assigned to the lower and upper approximation of that cluster else the observation will be assigned only to the upper approximation of the cluster.

The second step of the algorithm consists of updating the centroids in order to identify the new modes. Thus, the cluster-wise attribute value frequencies $freq_j^{low}(x_{il})$ and $freq_j^{up}(x_{il})$ representing the most frequent modalities of the upper and lower approximations thus identified are counted. The new mode cen_j^* corresponds to the observation with the highest density $max_{obs_i \in C_j} density_j(obs_i)$.

In Entropy-based Rough k-modes (ERkM)[93], the dissimilarity measure used the information entropy defined as follows:

$$\mathrm{E}(\mathcal{D}) = -\sum_{i=1}^{K} p_i log_2 p_i = -\sum_{i=1}^{K} \frac{n_i}{N} log_2 \frac{n_i}{N} \tag{5.3}$$

p_{i} is the probability that an object belongs to the i^{th} cluster C_i and $|C_i|=n_i$

The Rough Set based Fuzzy k-Modes (RSFkM)[94] is a hybrid method that uses fuzzy sets (membership function) and rough sets concepts (lower and upper approximations). In the clustering process, each partition is represented by a set of three parameters: a cluster mode, a crisp lower approximation and a fuzzy boundary.

Let μ_{li} be the highest and μ_{hi} is the second-highest fuzzy membership value of observation obs_i among all clusters where $1 \leq l,\ h \leq K \text{ and } h \neq l$, the threshold value, ∂ is defined as the median of $(\mu_{li} - \mu_{hi})$, $\forall i = 1, 2, \ldots, N$. If $(\mu_{li} - \mu_{hi}) > \partial$ then obs_i is assigned to both $\overline{B}(X_l)$ and $\underline{B}(X_l)$. Otherwise, it belongs to the upper approximation of many clusters: $\overline{B}(X_l)$ and $\overline{B}(X_h)$.

5.3. *Rough hierarchical Set clustering: decision attribute*

5.3.1 The RAHCA

In 2006, a Rough Set-Based Agglomeration Hierarchy Clustering Algorithm (RAHCA)[95] was proposed where a decision table is generated based on introducing a class attribute. The Euclidean distance was used as the similarity measure. During the clustering process, the clusters are merged iteratively based on the clustering level until the number of clusters K or the aggregate degree threshold λ is met.

5.3.2 The Min Min Roughness: MMR

The MMR[41] is a divisive categorical method proposed to handle outliers and cluster large datasets. A categorical attribute $a_i \in A$ may have n_j different modalities noted $a_i^{n_j}$. For a subset of observations X ($a_i{=}a_i^{n_j}$) of the initial dataset, $V(a_i)$ refers to the domain of the a_i values. The roughness $R_{a_j}(X)$ of X with respect to $\{a_j\}$ is defined as follows:

$$R_{a_j}(X|a_i{=}\alpha){=}1 - \frac{\left|\underline{X_{a_j}}(a_i{=}\alpha)\right|}{\left|\overline{X_{a_j}}(a_i{=}\alpha)\right|} \tag{5.4}$$

$\underline{X_{a_j}}(a_i{=}\alpha)$ is the lower approximation and $\overline{X_{a_j}}(a_i{=}\alpha)$ is the upper approximation with respect to $\{a_j\}$. The MMR proposes a new data similarity measure called mean roughness on attribute a_i with respect to $\{a_j\}$ and defined as follows:

$$Rough_{a_j}(a_i){=}\frac{\sum_{k=1}^{|V(a_i)|} R_{a_j}(X|a_i{=}\alpha_k)}{|V(a_i)|} \tag{5.5}$$

a_i, $a_j \in A$ and $a_i \neq a_j$.

The MR (min-roughness) of attribute a_i is defined as follows:

$$\mathrm{M}R(a_i){=}Min\left(Rough_{a_1}(a_i),\ldots, Rough_{a_j}(a_i), \ldots\right), \\ \text{where } a_i,\ a_j \in A,\ a_i \neq a_j,\ 1 \leq i,j \leq N. \tag{5.6}$$

The MMR is then defined as the minimum of the Min-Roughness of the d attributes:

$$MMR{=}Min(MR(a_1),\ \ldots,\ MR(a_i),\ \ldots) \tag{5.7}$$

The main advantages of the MMR are as follows: (1) ability to handle uncertainty; (2) more stable results can be obtained; (3) capability to handle large datasets. The clustering complexity of the MMR is $O(Kd(N+dl))$ where l is the maximum number of values in the attribute domains. Thus, the MMR is not appropriate for high dimensional datasets, however, it is well adapted for clustering large datasets either with a high number of observations or number of clusters.

The MMR selects one splitting attribute among a list of candidate ones using the minimum of the Min–Roughness. It is a top-down hierarchical method and takes the number of initial clusters as an input. Unfortunately, the roughness cannot reflect the discernibility power to the boundary objects. The MMR is very robust since it enables users to obtain stable results by only one input and is able to handle large datasets[98].

5.3.3 The MDA

In the Maximum Dependency Attributes (MDA)[96], the notion of attributes dependency is expressed using a factor called degree k of dependency. For a categorical information system and D and C any subsets of attributes of A. Dependency attribute D on C in a degree k in [0,1] for a subset X of U, is denoted by $C \underset{k}{\Rightarrow} D$. The degree k is defined as follows:

$$\mathrm{k}{=}\frac{\sum_{x \in U/D}\left|\underline{C}(X)\right|}{|U|} \tag{5.8}$$

A clustering attribute is selected based on the maximum degree of k and the corresponding equivalence classes. Then the dependency degree of each attribute is determined to use the following equation:

$$\mathrm{D}\left(\underline{R}(X),\overline{R}(X)\right){=}1 - \frac{\left|\underline{R}(X) \cap \overline{R}(X)\right|}{\left|\underline{R}(X) \cup \overline{R}(X)\right|}{=}1 - \frac{\left|\underline{R}(X)\right|}{\left|\overline{R}(X)\right|}{=}1 \text{-} \alpha_R(X) \tag{5.9}$$

α_R represents the accuracy of roughness or accuracy of the approximation. The MDA then selects the maximum dependency degree that is the most accurate (higher of the accuracy of approximation) for selecting the clustering attribute. The total roughness of an attribute a_i with respect to an attribute a_j, where $i \neq j$, denoted TR(a_i), is given by:

$$\mathrm{T}R(a_i)=\frac{\sum_{j=1}^{|A|} Rough_{a_j}(a_i)}{|A|\text{-}1} \tag{5.10}$$

The overall complexity of this method is $O(N(N - 1) + Nd)$, which means that the MDA is not appropriate for large datasets with a high number of observations. On the other hand, the MDA can perform well for high dimensional datasets, which is interesting.

5.3.4 The SSDR and MTMDP

The SSDR (Standard deviation of Standard Deviation Roughness)[97] is a hierarchical divisive clustering method used for mixed datasets. The SSDR is based on computing the Standard Deviation Roughness (SDR) defined as follows for each categorical attribute:

$$\mathrm{SDR}(a_i=\alpha)=\sqrt{\frac{1}{N\text{-}1}\sum_{i=1}^{N\text{-}1}\Big(R_{a_i}(X/a_i=\alpha)\text{-}MeR(a_i=\alpha)\Big)^2} \tag{5.11}$$

The mean roughness of a given modality α of an attribute a_i is defined as follows:

$$MeR(a_i=\alpha)=\frac{\sum_{\substack{j=1\\ j\neq i}}^{n} R_{a_j}(X/a_i=\alpha)}{(N\text{-}1)} \tag{5.12}$$

R_{a_j} is the roughness of an attribute a_j as defined in Equation 5.4. The Minimum Standard Deviation Roughness (MSDR) is selected and corresponds to the splitting attribute defined in that step:

$$MSDR = \{\min \{SD\ (a_i\text{=}\alpha_1),\ldots.SD\ (a_i = \alpha_{k_j})\} \tag{5.13}$$

k_j is the number of equivalence classes in the domains of the attribute a_i.

The parent cluster node is then split according to the modalities defined by the splitting attribute into the appropriate number of leaves and the process continues until reaching clusters with only one observation. SSDR is also able to handle hybrid datasets which is a great advantage in many real world applications.

The MTMDP (Maximum Total Mean Distribution Precision)[98] uses the Mean Distribution Precision (MDP) in order to identify the splitting attribute. As compared to the MMR, using only the concept of roughness to determine the clustering attribute from all candidate attributes cannot reflect the discernibility power to the boundary objects. Thus, the MTMDP is more appropriate and concise since it uses the concept of distribution approximation precision. For a *CIS*, consider $V(a_i)$ as the set of modalities of a_i, the MDP of a_i with respect to a_j ($a_j \neq a_i$) is computed as follows:

$$MDP_{a_j}(a_i)=\frac{\sum_{X\in U/\{a_i\}} r^{d}_{\{a_j\}}(X)}{|V(a_i)|} \tag{5.14}$$

$|V(a_i)|$ is the number of distinct values of attribute a_i. The TMDP of attribute a_i is given by:

$$\mathrm{T}MDP(a_i)=\frac{\sum_{A_j\in A\wedge a_i\neq a_i} MDP_{a_j}(a_i)}{|A|\text{-}1} \tag{5.15}$$

The greater the $MDP_{a_j}(a_i)$ is, the smaller the coupling between the equivalence classes of $U/Ind\{a_i\}$ is. The MTMDP has a complexity of $O(KNd^2)$ which indicates the possibility to use it for large datasets with a high number of observations and classes. However, the method will provide inaccurate results when used for high-dimensional datasets since it is based on an attribute splitting method.

The main advantages of the MTMDP can be summarized as follows: it searches the clustering attribute by taking into account the mean distribution precision of all attributes, which is better than the MR (Min–Roughness) criterion. Then, it determines the further clustering node by considering the cohesion degree of all nodes, which is a more reasonable method compared with the method used in the MMR. For a top-down hierarchical clustering algorithm, it is crucial to determine which leaf node is selected for further splitting. The MTMDP is capable of handling uncertainty and does not depend on initial values and the input order of the observations. Thus, more stable clustering results can be obtained.

5.3.5 The MMeMeR

The Min-Mean-Mean-Roughness (MMeMeR)**[99]** is a divisive clustering method that can deal with heterogeneous datasets (categorical and numeric). The MMeMeR starts with a unique cluster containing all the observations that will be iteratively divided until reaching a specific stop criterion, which usually corresponds to the number of desired clusters. In this process, the entire dataset is considered and the domain of all the equivalent classes for each attribute a_i are computed, then the roughness of each remaining attribute a_i $(j\neq i)$ is calculated. The Mean Roughness of the attribute a_i is then computed. Finally, the process ends by selecting the minimum mean roughness of a_i, which will correspond to our splitting attribute.

$$\mathrm{M}eMeR(a_i)=\frac{\{MeR(a_i=\alpha)+\ldots+MeR(a_i=\delta)\}}{|V_{a_i}|} \tag{5.16}$$

$$\mathrm{M}MeMeR(a_i)=min\{MeR(a_i=\alpha)+\ldots+MeR(a_i=\delta)\} \tag{5.17}$$

MeR corresponds to the mean roughness as defined in the Equation 5.12 and $|V(a_i)|$ is the number of distinct values of attribute a_i

The main advantage of the MMeMeR is its ability to deal with heterogeneous datasets not only categorical ones such as SDR and SSDR. Besides, the algorithm consider the min-mean instead of only selecting the mean which would permit obtaining more accurate results.

6. Evaluating the performance of a categorical clustering method

In the previous sections, three classes of categorical clustering methods were provided:

- Hard clustering methods (section 3) including partitional methods such as *k*-modes[19], He's *k*-modes[42], Ng's *k*-modes[23], IBD[43], CCI*k*-M[44], G-ANMI[46] and BC*k*-M[47], and hierarchical methods including ROCK[26], Squeezer[49], MGR[74] and DHCC[75].
- Fuzzy methods (section 4) including Fuzzy *k*-modes[20], F*k*-MFC[31], F*k*P[32], MF*k*-PIND[33], GF*k*-M[34] and NF*k*M[35] as partitional methods and FHC[37] and SS-FCC[38] as hierarchical methods.

- Rough Set based algorithms (section 5) including partitional methods such as R*k*M[92], ER-*k*-modes[93] and RSF*k*M[94] and hierarchical methods including RAHCA[95], MMR[41], MDA[96], SSDR[97], MTMDP[98] and MMeMeR[99].

6.1. *Clustering performance*

To better draw an understanding overview of the previous algorithms, a comparison of their clustering performance is provided in section 6.3 using three validation indices (accuracy, precision and recall) and some experimental datasets (Soybean[§], Zoo[**], Mushroom[††] and Breast Cancer[‡‡]). The evaluation of the proposed methods permits comparing their scalability, efficiency and complexity. These parameters are key features for any clustering method and are defined as follows:

- The *Scalability* characterizes the capability of an algorithm to handle a growing amount of input data and how it would potentially react to accommodate that requirement. Thus, the algorithm is said to *scale* if it is able to avoid failure when applied to larger data processing scenarios than initially predicted.
- An algorithm is considered *efficient* if its resource consumption is acceptable considering the provided resources, *i.e*, it should run in a reasonable time. The efficiency is generally measured using time constraints, that determines how long does the algorithm take to achieve the required computations and space that identifies how much working memory (typically RAM) is needed.
- *The complexity* of an algorithm is a function $f\ (N)$ which measures the required time and space for its execution in terms of the input size N or other parameters such as the number of clusters. The complexity of the surveyed methods is given in Annex II in order to assess the efficiency of each method in terms of the number of objects N, dimensions d and clusters K or other specific parameters.

6.2. *Evaluation metrics*

In this study, more than thirty categorical clustering methods were surveyed. It is necessary to compare their efficiency and identify the most accurate ones using adequate validation metrics[100]. Two types of evaluation measures can be used in this context:

- External measures[101] such as purity[19,20,23,31,35,41-46,73,98,99], Precision[23,35,43-45], Recall[23,35,43-45], Corrected Rand Measure[34,22], match metric[44]**,** Normalized Mutual Information[75,98], etc. They use external information (ground truth), initially not provided in the dataset and not used in the clustering process such as class labels. These classes are generally created by expert humans to ensure highest intra-cluster compactness and inter-cluster separation.
- Internal measures[102] such as Davies–Bouldin index[8,33], Dunn index[8,33], silhouette coefficient[8], etc. They only rely on the initial dataset and thus a good value reported by this method does not necessarily correspond to the best clustering method.

[§] https://archive.ics.uci.edu/ml/datasets/Soybean+(Large)
[**] http://archive.ics.uci.edu/ml/datasets/zoo
[††] https://archive.ics.uci.edu/ml/datasets/mushroom
[‡‡] http://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+%28diagnostic%29

In practice, for many real-world applications, class labels are often not available and thus using internal evaluation measures is the only option left to estimate the clustering efficiency. Unfortunately, in this case, no ground truth knowledge can be used to validate the final results. Clustering, that can be considered as an "unsupervised classification" method, is applied to a set of experimental datasets that have predefined class labels inherited in them. Thus, in the clustering process, more knowledge, such as the number of classes within each dataset, is provided. In this case, it is possible to use external measures to evaluate their efficiency instead of internal measures. A high value of these measures depicts a high number of observations correctly assigned and thus better efficiency.

Accuracy

It counts the number of most frequent objects with the same label in a given cluster. Unfortunately, using the accuracy does not penalize partitions with a high number of clusters: in fact, the purity of 1 is reachable if each cluster is composed of only one object of the dataset. The purity is a positive measure with values between 0 and 1 and is defined as follows:

$$\mathrm{AC}=\sum_{i=1}^{K}\frac{a_i}{N}$$

N is the number of total objects in the dataset.

a_i is the number of correctly classified observations in cluster C_i.

Precision and Recall

Precision represents the fraction of relevant instances among the retrieved instances, while recall corresponds to the fraction of relevant instances that have been retrieved over the total number of relevant instances. The precision and recall are respectively defined as follows:

$$PR=\frac{\sum_{i=1}^{K}\frac{a_i}{a_i+c_i}}{K} \qquad \text{and} \qquad RE=\frac{\sum_{i=1}^{K}\frac{a_i}{a_i+d_i}}{K}$$

a_i is the number of objects that are correctly assigned to the i^{th} cluster as defined in the accuracy,

c_i is the number of objects that were erroneously assigned to the i^{th} cluster

d_i is the number of objects that should have been assigned to *the* i^{th} cluster but were not.

Adjusted Rand Measure

The Adjusted Rand Measure is an external index that evaluates the similarity between the final clusters generated by the clustering process. It is defined as follows:

$$\gamma=\frac{\binom{N}{2}\sum_{i=1}^{K_1}\sum_{j=1}^{K_2}\binom{n_{ij}}{2}-\sum_{i=1}^{K_1}\binom{|C_i|}{2}\sum_{j=1}^{K_2}\binom{|C_j'|}{2}}{\frac{1}{2}\binom{N}{2}\left[\sum_{i=1}^{K_1}\binom{|C_i|}{2}+\sum_{j=1}^{K_2}\binom{|C_j'|}{2}\right]-\sum_{i=1}^{K_1}\binom{|C_i|}{2}\sum_{j=1}^{K_2}\binom{|C_j'|}{2}}$$

$\wp=\{C_1,C_2,\dots,C_{K_1}\}$ and $\wp'=\{C'_1,C'_2,\dots,C'_{K_2}\}$ are two clustering of the dataset

n_{ij} represents the number of objects of the i^{th} class and j^{th} cluster,

n_i indicates the number of objects in a priori class i

n_j indicates the number of objects in cluster j

N is the total number of objects in the dataset.

Davis and Bouldin index

Davies Bouldin index[33] is an internal evaluation measure. It attempts to minimize the distance between each cluster and the most one similar to it. This metric is given as follows**:**

$$\mathrm{D}B=\frac{1}{K}\sum_{k=1}^{K}\max_{k\neq k'}\left(\frac{\sigma_k+\sigma_{k'}}{d(C_k+C_{k'})}\right)$$

K is the number of clusters,

σ_k is the average distance of all elements in the k^{th} cluster

$d(C_k+C_{k'})$ is the hamming distance between clusters C_k and $C_k{}'$ that will be computed between their centroids. This index can also be used to determine the optimal number of clusters K[104] of the dataset.

Dunn's validation index

Dunn's Validity Index[33] is an internal validation measure that identifies the most compact and well-separated clusters. This measure is defined as follows:

$$\mathrm{D}n=\min_{1\leq k\leq K}\left(\min_{k+1\leq k'\leq k}\left(\frac{d(C_k,C_{k'})}{\max\limits_{1\leq n\leq k} d^{'}(n)}\right)\right)$$

$d(C_k,C_{k'})$ represents the inter-cluster distance between cluster k and cluster k' and $d^{'}(n)$ is the intracluster distance of the n^{th} cluster.

MCDM[§§] methods

MCDM involves making decisions based on considering multiple criteria (or objectives). This task is usually complex and difficult since it requires expert judgment and specialized techniques. Due to the fact that cluster validation may involve multiple criteria, it can be considered as an MCDM problem[105]. In[105], three MCDM methods were proposed:

- TOPSIS[***]: is based on the fact that the considered criteria should ensure the shortest geometric distance from the positive ideal solution (PIS) and the longest geometric distance from the negative ideal solution (NIS).
- DEA[†††]: DEA considers the entropy and time as input parameters and the Dunn's index, rand index, purity, silhouette, Jaccard coefficient, true positive rate, true negative rate, precision, and F-measure as output components. Thus a final score is given to assess how efficient is a clustering method based on these components.
- VIKOR[‡‡‡]: this method ranks alternatives in the presence of conflicting criteria by introducing the multicriteria ranking index, which is based on the particular measure of closeness to the ideal alternative.

[§§] Multiple Criteria Decision Making

[***] Technique for Order Preference by Similarity to Ideal Solution

[†††] Data Envelopment Analysis

[‡‡‡] Serbian abbreviation for the VlseKriterijumska Optimizacija I Kompromisno Resenje (means Multi-criteria Optimization and Compromise Solution)

In the experiments presented in the paper, it was shown that no algorithm can achieve the best performance on all measurements for any data set and it is necessary to utilize more than one single performance measure to evaluate clustering algorithms.

6.3. *Experimentations*

In this section, we report the experiments conducted using the clustering methods discussed in the previous sections. Three evaluation metrics are used: the accuracy, purity and recall. For each evaluation metric, four datasets are used: the Soybean, the Zoo, the Mushroom and the Breast Cancer. The best top five clustering methods that provided the highest values of the evaluations measures are spotted and given in separate summary tables. This consideration permits only focusing on the most remarkable class of methods that provide the best experimental results.

The following Figures 3 (a-d) reports the accuracy computed for the clustering methods for the four experimental datasets (Soybean, Zoo, Mushroom and Breast Cancer):

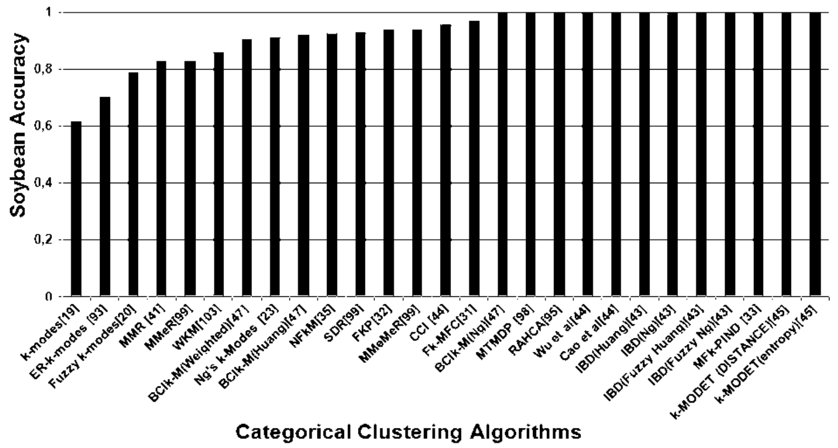

Figure 3(a): Accuracy computed for the Soybean dataset.

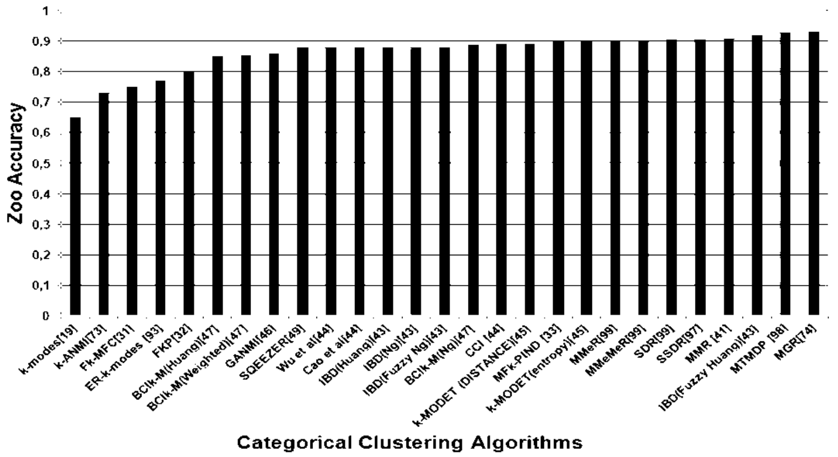

Figure 3(b): Accuracy computed for the Zoo dataset.

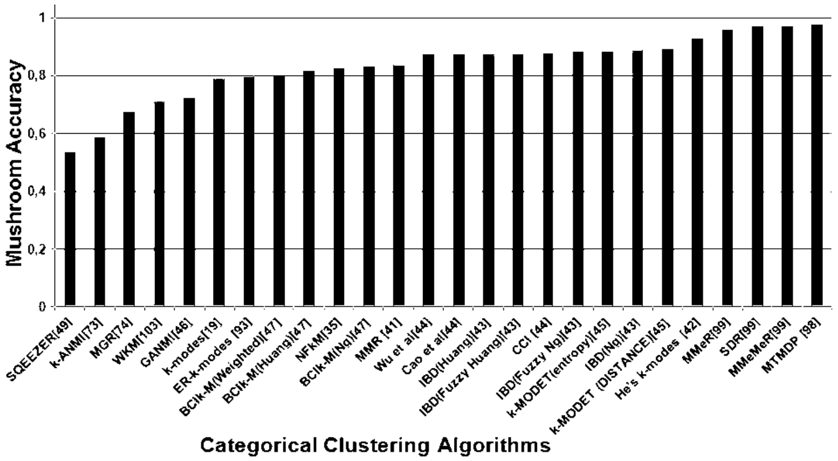

Figure 3(c): Accuracy computed for the Mushroom dataset.

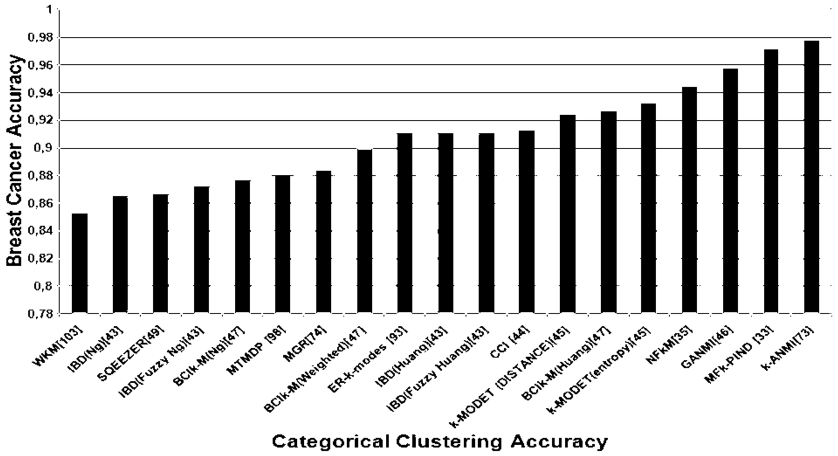

Figure 3(d): Accuracy computed for the Breast cancer dataset.

In Table 4, we report the list of the top five algorithms given by class for each dataset. The results are provided in terms of the accuracies computed.

Table 4: Top categorical clustering algorithms using accuracy.

	Hard Clustering	**Fuzzy Clustering**	**Rough Set Clustering**	
Soybean	*k*-MODET(entropy), *k*-MODET (distance)[45]	IBD (Fuzzy Ng)[43], IBD (Fuzzy Huang)[43]	MF*k*-PIND[33]	
Zoo	MGR[74]	IBD (Fuzzy Huang)[43]	MTMDP[98], MMR[41], SSDR[97]	
Mushroom	He's *k*-modes[42]	-	MTMDP[98], MMeMeR[99], SDR[99], MMeR[99]	
Breast Cancer	*k*-ANMI[73], GANMI[46], *k*-MODET(entropy)[45]	NF*k*M[35]	MF*k*-PIND[33]	
Number of cases	**7**	**4**	**9**	**20**

According to the experiments provided in Table 4, 9 RST based clustering algorithms provided the best values of the accuracy out of 20 (45%) as shown in column 4 of the table. Hard clustering methods come in the second position (35%) and finally fuzzy methods with only 4 positions. Besides, the computed values of the accuracy for these methods are considerably high as given in the following:

- For RST based methods, seven algorithms provided the best clustering results: MF*k*-PIND, SDR, SSDR, MMeR, MMeMeR, MTMDP and MMR. The accuracy varies from 90.7% (SSDR) to 100% (MFk-PIND). The MFk-PIND and MTMDP registered the best accuracy for two datasets: Soybean (100%), Breast Cancer (97.17%) and Zoo (93%), Mushroom (98%) respectively. The MMR was spotted in only one case for the Zoo dataset (91%), although it also registered high accuracy for two datasets (Soybean: 83% and Mushroom: 84%). The SDR was also spotted best in one case for the Mushroom dataset (97.2%), and also provided good results for the other datasets: 93% (Soybean) and 90.7% (Zoo).
- For hard clustering methods, seven algorithms provided the best accuracies. The *k*-MODET, with its two variants (entropy and distance) provided best accuracies in two cases: 100% (Soybean) and 93.28% (Breast Cancer). It also provided good results with the Zoo (89.11% and 90.1%) and Mushroom (88.76% and 89.41%) datasets. The MGR provided the accuracy of 93.1% for the Zoo dataset. Although the *k*-ANMI and GANMI provided good results for the Breast Cancer (97.8% and 95.8%), they provided lower accuracy with the Zoo and Mushroom datasets ranging from 58.7% to 86%.
- For fuzzy clustering, the IBD, in its two versions: using either the Huang or the Ng's metrics, provided good accuracies for two datasets: Soybean (100%) and Zoo (92.08%). In the other datasets, the values of the accuracy of this algorithm range from 87.27% to 100%.

In all cases, the most interesting values for the accuracy provided in this first set of experiments correspond to the RST based clustering methods followed by hard clustering and then fuzzy clustering methods.

In the second step of the experiment, the precision is used as an evaluation metric and the results reported in Figures 4(a-d).

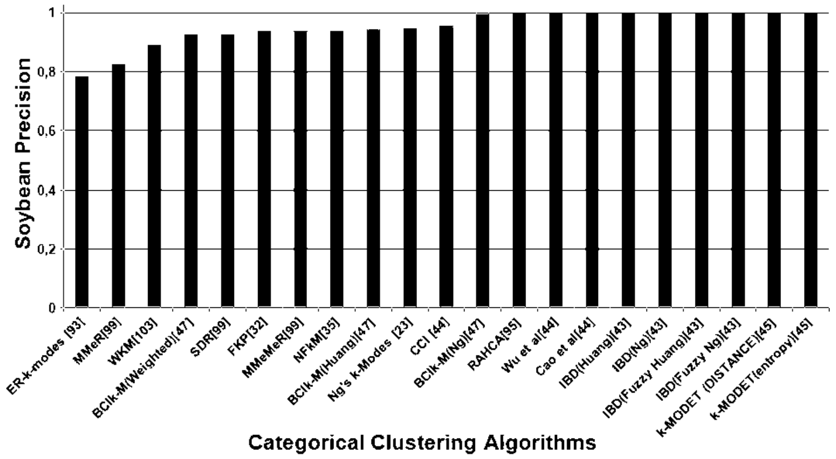

Figure 4(a): Precision computed for the Soybean dataset.

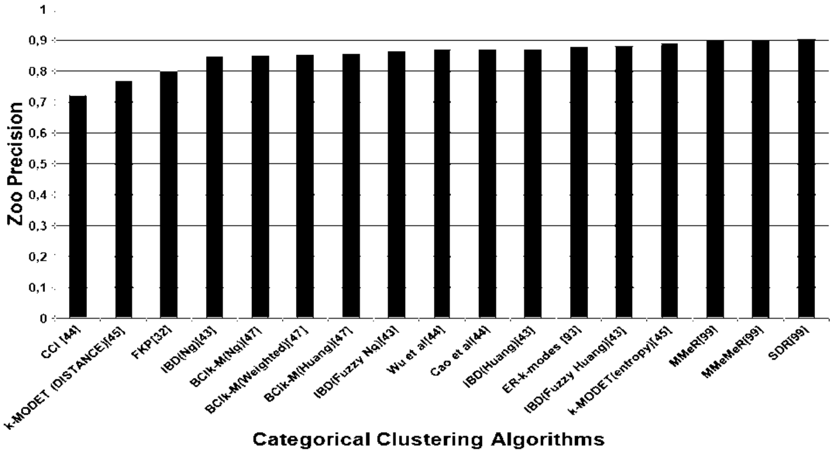

Figure 4(b): Precision computed for the Zoo dataset.

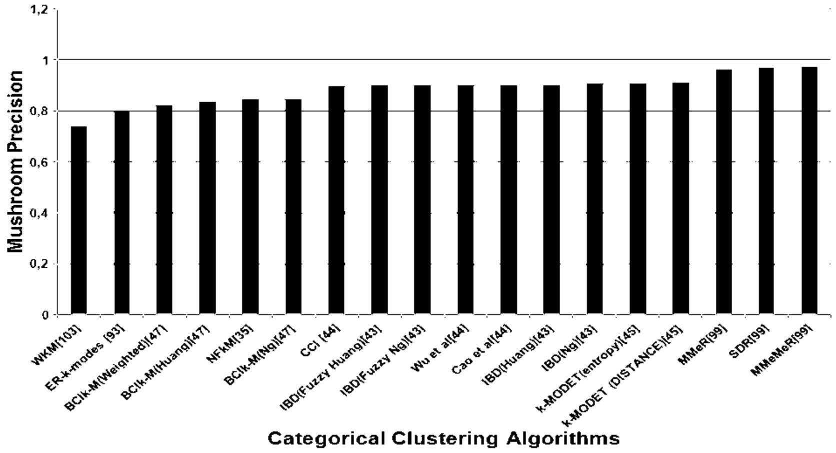

Figure 4(c): Precision computed for the Mushroom dataset.

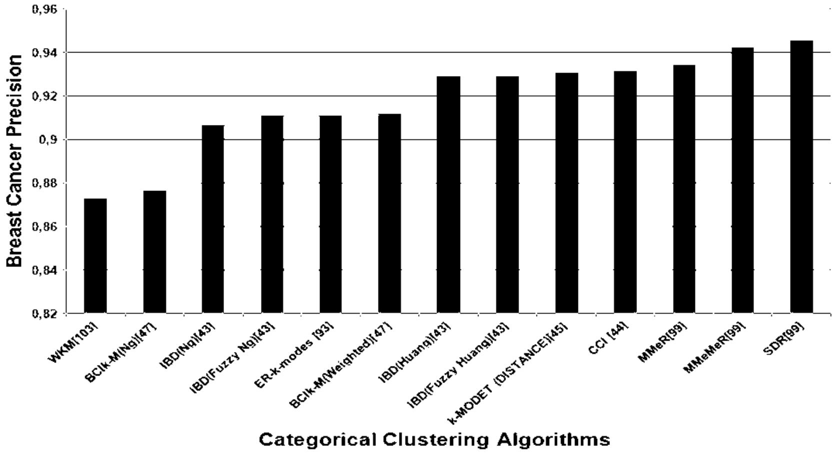

Figure 4(d): Precision computed for the Breast cancer dataset

In Table 5, we report the best algorithms that provided the highest values of precision. We also notice that RST based clustering methods and hard clustering methods also provided the highest values of the precision.

Table 5: Top categorical clustering algorithms using precision.

	Hard Clustering	Fuzzy Clustering	Rough Set Clustering	
Soybean	*k*-MODET(entropy)[45], *k*-MODET (distance)[45], IBD(Ng)[43]	IBD(Fuzzy Ng)[43], IBD(Fuzzy Huang)[43]	-	
Zoo	*k*-MODET(entropy)[45]	IBD(Fuzzy Huang)[43]	SDR[99], MMeMeR[99], MMeR[97]	
Mushroom	*k*-MODET (distance)[45], *k*-MODET(entropy)[45]	-	SDR[99], MMeMeR[99], MMeR[99]	
Breast Cancer	CCI[44],*k*-MODET (distance)[45]	-	SDR[99], MMeMeR[99], MMeR[99]	
Number of cases	**8**	**3**	**9**	**20**

According to Table 5, the following facts can be highlighted:

- For RST methods, SDR, MMeR and MMeMeR provided the best precision values: 90.7% (SDR) and 90.2% (MMeR, MMeMeR) using the Zoo dataset, 97.3% (MMeMeR), 97.2% (SDR) and 96.4% (MMeR) using the Mushroom dataset and 94.56% (SDR), 94.24% (MMeMeR) and 93.43% (MMeR) using the Breast Cancer dataset. Although for the Soybean these algorithms were not classified among the top five best methods, the values of the precision registered are 83% (MMeR), 94.25% (MMeMeR) and 93% (SDR) which can also be considered as good clustering results.
- For hard clustering methods, the *k*-MODET with its two variants, was spotted in six positions out of 8 total positions for the four datasets. Its precision ranges from 89.06% to 100%: 100% for the Soybean dataset, 89.06% for the Zoo dataset, 91.38% and 90.95% for the Mushroom and 93.09% for the Breast Cancer. The IBD also provided a precision of 100% (Soybean), 87.02% and 84.94% (Zoo), 90.83% and

90.19% (Mushroom) and 92.92% and 90.69% (Breast). The CCI also provided good precision with 93.18% (Breast Cancer), 89.75% (Mushroom), 95.83% (Soybean) and 72.24% (Zoo).

- For Fuzzy clustering, IBD provided the highest precision: from 100% (Soybean) to 88.19% and 86.48% (Zoo). Although the algorithm was not spotted among top five clustering methods for the Mushroom, it provided high precision of 90.13% (Huang) and 90.15% (Ng). For the Breast Cancer, the IBD provided a precision of 91.1% (Ng) and 92.92% (Huang), which can also be considered as good results.

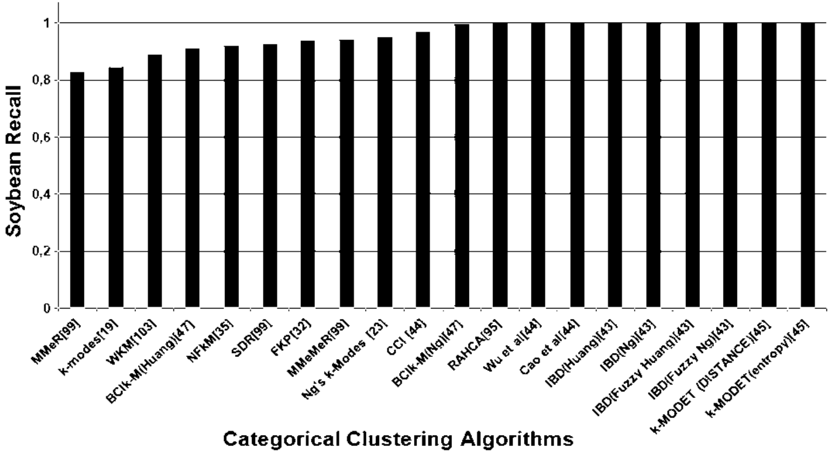

Figure 5(a): Recall computed for the Soybean dataset.

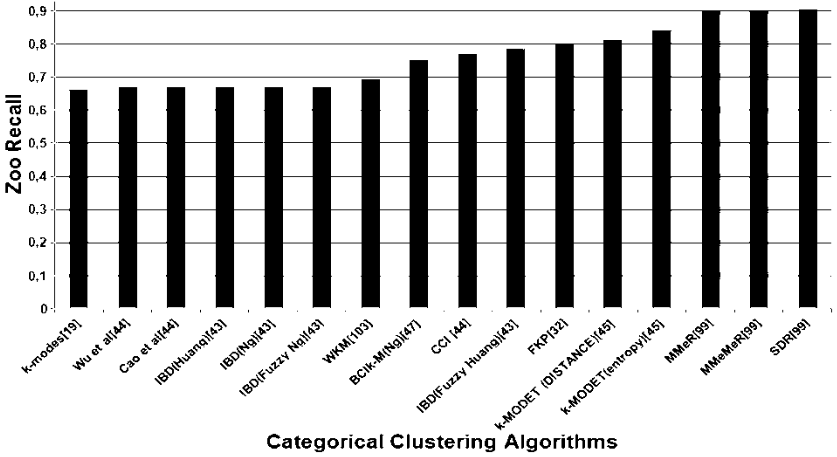

Figure 5(b): Recall computed for the Zoo dataset.

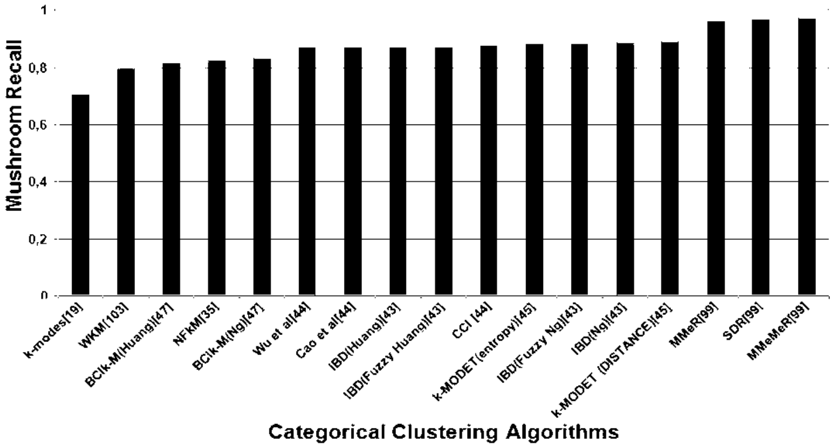

Figure 5(c): Recall computed for the Mushroom dataset.

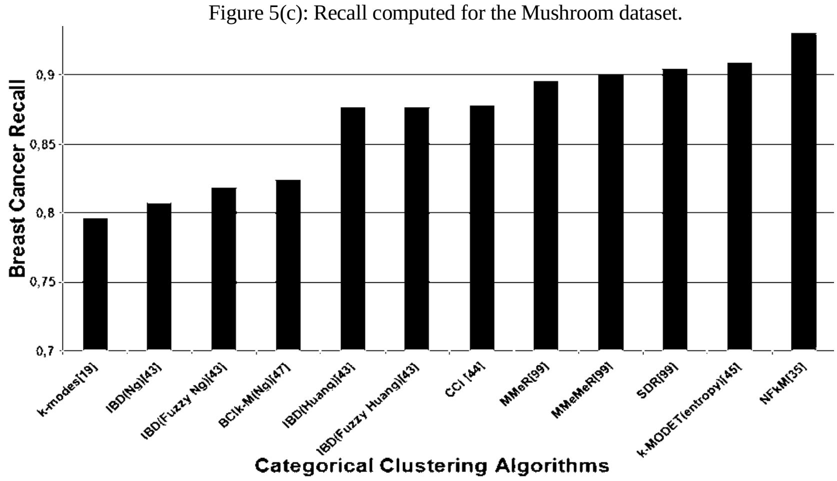

Figure 5(d): Recall computed for the Breast Cancer dataset.

In Table 6, the algorithms that registered the highest values of a recall are reported. It is again obvious that RST based techniques and hard clustering methods provided the highest values of recall for the conducted experiments.

Table 6: Top categorical clustering algorithms using the recall.

	Hard Clustering	Fuzzy Clustering	Rough Set Clustering	
Soybean	*k*-MODET(entropy)[45], *k*-MODET (distance)[45], IBD(Ng)[43]	IBD(Fuzzy Ng)[43], IBD(Fuzzy Huang)[43]	-	
Zoo	*k*-MODET(entropy)[45], *k*-MODET (distance)[45]	-	SDR[99], MMeMeR[99], MMeR[99]	
Mushroom	*k*-MODET (distance)[45], IBD(Ng)[43]	-	SDR[99], MMeMeR[99], MMeR[99]	
Breast Cancer	*k*-MODET(entropy)[45]	NFkM[35]	SDR[99], MMeMeR[99], MMeR[99]	
Number of	**8**	**3**	**9**	**20**

positions

According to Table 6, the following facts can be reported:

- For RST based methods, 9 methods have provided best results for three datasets including SDR (90.5% → 97.2%), MMeMeR (90.09% → 97.3%) and MMeR (89.59% → 96.4%). For the Soybean dataset, the values of the recall computed for the three algorithms were also important: SDR (93%), MMeMeR (94.25%) and MMeR (83%).
- For hard clustering, the *k*-MODET with its two variants has registered 6 best positions in the experiments for all the datasets with recall values ranging from 81.46% to 100%. The IBD comes second in two positions with 88.87% for the Mushroom dataset and 100% for the Soybean dataset. For the Breast Cancer, the IBD registered 80.79% and 87.73% recall values, which are considered good results.
- For fuzzy clustering, the IBD in its two versions provided good results with 100% when using the Soybean dataset. For the other datasets, the values of the recall were 67.14% and 78.57% for the Zoo dataset, 88.39% and 87.09% for the Mushroom dataset, 81.83% and 87.73% for the Breast Cancer dataset which can be considered as high values of recall.

7. Discussion

According to the previous results, RST based clustering methods provided the best clustering results: they were spotted in 27 cases (out of 60) which corresponds to an average of 45%. Hard clustering methods come in the second position with 23 positions classified among the top methods (38.33%) and finally, fuzzy methods with only 10 positions (16.66%).

On the other hand, for RST methods, the best algorithms were MMeMeR[99], SDR[99], MMeR[99] in seven experiments each, MTMDP[98] in two experiments, and MMR[41], SSDR[97] and MFk-PIND[33] in only one experiment. In all cases, even though these methods provided best results in only one or two cases, the evaluation values computed were high as discussed above.

For hard clustering methods, initialization based techniques also provided good results, including:

- The initialization *k*-Modes Outlier detection[45] algorithm with its two variants;
- The initialization Based on Density[43] using either the Huang or the Ng distance measures (IBD (Huang) and IBD (Ng)).
- The Cluster Center Initialization *k*-Modes (CCI*k*-M)[44].

For fuzzy clustering, initialization methods also provided good results when using the Initialization Based Density[43] (Ng or the Huang distance measures) in 8 cases.

The surveyed methods in this paper have the advantage to process large categorical and high dimensional datasets and thus can be effectively applied in the context of Big Data. This outcome is easily derived from their complexity analysis provided in Annex II. Hard categorical methods, discussed in section 3, are unfortunately not able to handle uncertainty in the clustering process, which is a highly desirable requirement in many real-world applications. On the other hand, these methods tend to generate local optimal solutions instead of global ones unless using more advanced techniques to handle this issue such as Genetic Algorithms[34,46,72]. Besides, the final results depend on the initialization of the modes and the processing order of objects in the datasets, which requires using initialization methods[43-45,68,70,71]. Thus, these methods are very sensitive to the initial cluster centers. Usually, these methods are run with different initial guesses of clusters centers and the results are compared in order to determine the best clustering results. Moreover, in hard clustering, each object is assigned to only one cluster, and all of the objects have the same degree of confidence, which limits handling boundary data. The number of clusters *K* is also problematic in partitional clustering either for hard, fuzzy or rough set based methods. In fact, this parameter needs to be determined in advance as an input value. In a real dataset, *K* is usually unknown. In practice, different values of *K* are tried and cluster validation techniques are used to measure the clustering results and determine the best value of *K*. For hierarchical methods (agglomerative[26,38,95] or divisive[37,41,74,75,96-99]), nested clusters are successively formed and merged. The main advantage of such methods is that clustering is not influenced by initialization and local minima and the number of clusters is not required initially. Using fuzzy and rough set methods permit avoiding stability issues related to hard clustering methods and introduce the possibility to obtain overlapping clusters which increase the accuracy of these methods. However, although fuzzy methods permit handling uncertainty, they are still influenced by the initialization of the modes and the processing order of the objects in the datasets. Furthermore, these methods need to adjust the membership fuzziness parameter to obtain better solutions. In many real-world applications, the optimal value of the fuzziness parameter is selected on the basis of the decision makers' previous knowledge of the domain and their intuition or the proposed criteria[98]. This issue was fixed with rough set based clustering methods where no additional heuristics are required such as thresholds or expert knowledge in a particular domain.

8. Conclusion

Clustering permits recovering initially hidden patterns and has shown high efficiency when applied for substantial structured or unstructured datasets for a wide range of applications. In this survey paper, most recent categorical clustering algorithms were investigated according to three active research theories: hard sets, fuzzy sets and rough sets and arranged into either partitional or hierarchical techniques. The algorithms were compared and classified according to three evaluation metrics. Experimental results demonstrate that rough set based clustering provided the best clustering results followed by hard methods based initialization techniques and then fuzzy clustering methods based on initialization methods of the initial centroids.

ANNEX I: EXPERIMENTAL PERSPECTIVES OF THE SURVEYED CLUSTERING METHODS.

N°	Algorithm	Acronym	scalability			complexity		heuristics	Datasets						Description of the algorithm	Several validation indexes	multiple runs	Fuzziness parameter	tests with previous methods
			N	*d*	*k*	Space	Time		Variety	Real	Synthetic	Large	High*d*	High*k*					
01	Min-Mean-Mean-Roughness[99]	MMeMeR (2017)							x	x			x		x				x(6)
02	fuzzy SV-*k*-modes algorithm[36]	FSV-*k*M (2017)	x	x	x		x	x	x	x	x	x	x		x	x			x(01)
03	entropy-based rough *k*-modes[93]	ER*k*M (2017)						x	x	x			x		x	x			x(02)
04	Modified Fuzzy *k*-Partition based INDiscernibility[33]	MF*k*-PIND (2016)	x			x			x	x					x	x	x	x	x(02)
05	initialization *k*-Modes using Outlier DETection[45]	*k*-MODET (2015)	x	x			x	x	x	x	x	x	x		x	x			x(04)
06	The Mean Gain Ratio[74]	MGR (2014)	x			x			x	x	x	x	x		x				x(5)
07	Maximum Total Mean Distribution Precision[98]	MTMDP (2014)	x	x	x		x		x	x		x	x	x	x	x			x(01)
08	Between Cluster Information *k*-Modes[47]	BCI*k*-M (2014)	x	x	x		x		x	x	x	x	x	x	x	x	x		x(03)
09	Semi-Supervised Fuzzy Co-Clustering[38]	SS-FCC (2013)						x	x	x		x	x		x	x			x(01)
10	Weighting k-modes[103]	W*k*M (2013)	x	x			x		x	x	x	x	x		x	x	x		x(02)
11	Fuzzy between-cluster information algorithm[35]	FBC (2013)					x	x	x	x		x	x		x	x	x	x	x(03)
12	Rough *k*-modes[92]	RKM (2013)							x	x	x	x	x		x				x(01)
13	Cluster Center Initialization *k*-Modes[44]	CCI*k*-M (2013)	x				x		x	x	x		x		x	x	x		x(03)
14	The Rough Set based Fuzzy *k*-Modes[94]	RSF*k*M (2012)						x	x	x	x		x		x	x	x	x	x05)
15	Divisive Hierarchical Clustering Categorical[75]	DHCC (2012)	x	x			x		x	x	x	x	x		x	x			x(04)
16	Mixed attribute Weighting *k*-modes[22]	MW*k*M (2011)	x	x	x		x	x	x	x	x	x	x		x	x	x		x(04)
17	Standard deviation of Standard Deviation Roughness[97]	SSDR(2011)								x					x				x(06)
18	Genetic Average Normalized Mutual Information[46]	G-ANMI (2010)	x					x	x	x			x		x		x		x(07)

19	The maximum dependency attributes[96]	MDA (2010)					x		x	x					x				x(02)
20	Genetic Fuzzy *k*-Modes[34]	GF*k*-M (2009)						x		x					x		x	x	
21	Initialization based density[43]	IBD (2009)	x			x			x	x			x		x	x			x(01)
22	fuzzy *k*-partitions[32]	F*k*P (2008)				x			x						x	x	x	x	x(02)
23	Min–Min–Roughness[41]	MMR (2007)							x	x			x	x	x				x(03)
24	Ng's modified *k*-modes[23]	Ng's *k*-modes (2007)	x	x	x					x	x	x			x	x	x		x(01)
25	Rough Set-Based Agglomeration Hierarchy Clustering Algorithm[95]	RAHCA (2006)				x	x		x	x	x		x		x		x		x(02)
26	Fuzzy Hierarchical graph connectedness[37]	FHC (2006)	x				x		x	x									x(01)
27	Improved *k*- modes with relative frequency[42]	He's *k*-modes (2005)							x	x			x	x	x		x		x(01)
28	Fuzzy *k*-modes with Fuzzy Centroids[31]	Fk-MFC (2004)	x			x	x	x	x	x		x			x		x	x	x(02)
29	fuzzy *k*-modes[20]	F*k*-modes (1999)								x	x	x					x		x(01)
30	*k*- modes[19]	*k*- modes (1998)	x		x				x	x	x	x		x	x		x		

ANNEX II: SUMMARY OF THE PROPOSED ALGORITHMS

N°	Algorithm	Acronym	Data type	Category of clustering	Type of clustering	complexity	Clustering algorithm for large parameters		
							N	*d*	*K*
01	Min-Mean-Mean-Roughness[99]	MMeMeR(2017)	hybrid	Rough	Hierarchical divisive	Not Communicated	-	-	-
02	fuzzy SV-*k*-modes algorithm[36]	FSV-*k*M(2017)	categorical	fuzzy	partitional	*O(NdtK× |V|)*, |*V*| is the maximum number of modalities in the attributes, *t* is the number of iterations required for the algorithm to converge	yes	yes	Yes
03	entropy-based rough *k*-modes[93]	ER*k*modes(2017)	categorical	Rough	partitional	Not Communicated	-	-	-
04	Modified Fuzzy *k*-Partition based INDiscernibility[33]	MFk-PIND(2016)	mixed	Fuzzy/Rough	partitional	*O(KM(N+1)t+Nd)*, M is the total number of modalities in the attributes.	yes	yes	Yes
05	initialization k-Modes using Outlier DETection[45]	*k*-MODET(2015)	categorical	hard	partitional	Ini_distance ($O(dN^2)$)	no	yes	yes
						Ini_entropy ($O(KdN+d^2N)$)	yes	no	yes
06	The Mean Gain Ratio[74]	MGR(2014)	categorical	hard	hierarchical	$O(Kd^2l + KdN)$ *l* is the maximum number of values in the attribute domains	yes	no	Yes
07	Maximum Total Mean Distribution Precision[98]	MTMDP(2014)	categorical	Rough	Hierarchical divisive	$O(KNd^2)$	yes	no	Yes
08	Between Cluster Information k-Modes[47]	BCI*k*-M(2014)	categorical	hard	partitional	$\mathrm{O}\left(N\sum_{j=1}^{d} n_j+NKd\sum_{e=1}^{O} t\right)$, n_j is the number of attribute values.	yes	yes	Yes
09	Semi-Supervised Fuzzy Co-Clustering[38]	SS-FCC(2013)	categorical	fuzzy	Hierarchical agglomerative	*O(NdKt)*	yes	yes	Yes
10	Weighting k-modes[103]	W*k*M(2013)	categorical	hard	partitional	O(*NdKt*)	yes	yes	yes

11	Fuzzy between-cluster information algorithm[35]	FBC(2013)	categorical	fuzzy	partitional	$\mathrm{O}\left(N\sum_{j=1}^{d} n_j + NKd\sum_{e=1}^{O} t\right)$	yes	yes	Yes
12	Rough k-modes[92]	RKM(2013)	categorical	Rough	partitional	*O(NKdt)*	*yes*	*yes*	*Yes*
13	Cluster Center Initialization k-Modes[44]	CCI*k*-M(2013)	categorical	hard	partitional	Using all/prominent attributes $O(Nd+tKd^2N+NlogN)$ Using significant attributes $O(Nd^2T^2+tKd^2N+NlogN)$ *t* is the number of required iterations to converge and T is the average number of distinct attribute values per attribute.	*yes*	*no*	*Yes*
14	The Rough Set based Fuzzy *k*-Modes[94]	RSF*k*M(2012)	categorical	Rough and fuzzy	partitional	Not Communicated	-	-	-
15	Divisive Hierarchical Clustering Categorical[75]	DHCC(2012)	boolean	hard	Hierarchical divisive	Not Communicated	-	-	-
16	Mixed attribute Weighting *k*-Modes[22]	MW*k*M(2011)	categorical	hard	Partitional	O(*NdKt*)	yes	yes	yes
17	Standard deviation of Standard Deviation Roughness[97]	SSDR(2011)	hybrid data	Rough	Hierarchical divisive	Not Communicated	-	-	-
18	The maximum dependency attributes[96]	MDA(2010)	categorical	Rough	Hierarchical divisive	$O(N(N - 1) + Nd)$	no	yes	Yes
19	Genetic Average Normalized Mutual Information[46]	G-ANMI(2010)	categorical	hard	partitional	Not Communicated	-	-	-
20	Genetic Fuzzy *k*-Modes[34]	GF*k*-M(2009)	categorical	fuzzy	partitional	Not Communicated	-	-	-
21	Initialization based density[43]	IBD(2009)	categorical	hard	partitional	O(NdK^2)	yes	yes	No
22	fuzzy *k*-partitions[32]	F*k*P(2008)	categorical	fuzzy	partitional	*O(2KNdt)*	yes	yes	Yes
23	Min–Min–Roughness[41]	MMR(2007)	categorical	Rough	Hierarchical divisive	$O(Kd(N+dl))$ *l* is the maximum number of values in the attribute domains	*yes*	*no*	*Yes*

24	Ng's modified *k*-modes[23]	Ng's *k*-modes(2007)	categorical	hard	partitional	$O(NKd)$	*yes*	*yes*	*Yes*
25	Rough Set-Based Agglomeration Hierarchy Clustering Algorithm[95]	RAHCA(2006)	categorical	Rough	Hierarchical agglomerative	$O(dN^2)$	no	yes	Yes
26	Fuzzy Hierarchical graph connectedness[37]	FHC(2006)	mixed	fuzzy	Hierarchical divisive	$O(2NK + NK^2 + K^2)$	yes	yes	No
27	Improved *k*- modes with relative frequency[42]	He's *k*-modes(2005)	categorical	hard	partitional	$O(TKN)$	*yes*	*yes*	*Yes*
28	Fuzzy *k*-Modes with Fuzzy Centroids[31]	F*k*-MFC(2004)	categorical	fuzzy	partitional	$O(d(Kmax(nl)+N)+KN$, where n_l is the number of category values of attribute Al.	*yes*	*yes*	*yes*
29	Sqeezer[49]	Sqeezer(2002)	categorical	hard	hierarchical	$O(NKd)$	*yes*	*yes*	*Yes*
30	Robust Clustering using linKs[26]	ROCK(2000)	categorical	hard	Hierarchical agglomerative	$O(N^2log(N)+N^2)$	no	yes	Yes
31	fuzzy *k*-modes[20]	F*k*-modes(1999)	categorical	fuzzy	hierarchical	$O(KN(d+M))$ $M = \sum_{j=1}^{m} n_j$is the number of all categories of all attributes and n_j is the number of categories in attribute *j*.	*yes*	*yes*	*Yes*
32	*k*- modes[19]	*k*- modes(1998)	categorical	hard	partitional	$O(TKN)$	*yes*	*yes*	*Yes*

References

1. Chen, CL Philip, and Chun-Yang Zhang. "Data-intensive applications, challenges, techniques and technologies: A survey on Big Data." *Information sciences* 275 (2014): 314-347.
2. Kimball, Ralph, and Joe Caserta. The data warehouse ETL toolkit: practical techniques for extracting, cleaning, conforming, and delivering data. John Wiley & Sons, 2011.
3. Xindong Wu, Xingquan Zhu, Gong-Qing Wu, Wei Ding, "Data Mining with big data." IEEE Transactions on Knowledge and Data EngineeringVolume: 26, Issue: 1, pp 97–107, 2014.
4. Adil Fahadet al "A Survey of Clustering Algorithms for Big Data: Taxonomy and Empirical Analysis." IEEE Transactions on Emerging Topics in Computing, Volume: 2, Issue: 3, pp 267–279, 2014.
5. Nanda, S. R., Biswajit Mahanty, and M. K. Tiwari. "Clustering Indian stock market data for portfolio management." *Expert Systems with Applications* 37.12 (2010): 8793-8798.
6. McAuley, Julian, and Jure Leskovec. "Discovering social circles in ego networks." *ACM Transactions on Knowledge Discovery from Data (TKDD)* 8.1 (2014): 4.
7. Ince, Turker. "Unsupervised classification of polarimetric SAR image with dynamic clustering: An image processing approach." *Advances in Engineering Software* 41.4 (2010): 636-646.
8. Dong Kuan Xu, Yingjie Tian, "*A comprehensive survey of clustering algorithms*." Annals of Data Science, Volume 2, Issue 2, pp 165-193, 2015.
9. Kelvin Sim, Vivekanand Gopalkrishnan, Arthur Zimek, Gao Cong, "*A survey on enhanced subspace clustering*.", Data Mining and Knowledge Discovery, Volume 26, Issue 2, pp 332-397, 2013.
10. Hai-Long Nguyen, Yew-Kwong Woon, Wee-Keong Ng, "A survey on data stream clustering and classification." Knowledge and Information Systems. Volume 45, Issue 3, pp 535–569, 2015.
11. Satyasai Jagannath Nanda, Ganapati Panda, "A survey on nature-inspired metaheuristic algorithms for partitional clustering." Swarm and Evolutionary Computation Volume 16, Pages 1-18, 2014.
12. X. Huang, Y. Ye, and H. Zhang, "Extensions of k-means-type algorithms: A new clustering framework by integrating intracluster compactness and intercluster separation," IEEE Trans. Neural Netw. Learn. Syst., vol. 25, no. 8, pp. 1433–1446, Aug. 2014.
13. J. Liang, L. Bai, C. Dang, and F. Cao, "The *K* -means-type algorithms versus imbalanced data distributions," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 4, pp. 728–745, Aug. 2012.
14. H.-L. Chen, K.-T. Chuang, and M.-S. Chen, "On data labeling for categorical clustering data," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 11, pp. 1458–1472, Nov. 2008.
15. J. Xie, B. Szymanski, and M. J. Zaki, "Learning dissimilarities for categorical symbols," in *Proc. 4th Workshop Feature Sel. Data Mining*, 2009, pp. 1058–1063.
16. R. H, "A conceptual version of the k-means algorithm," *Pattern Recognition Letters*, vol. 16, issue 11, p. 1147–1157, 1995.
17. M. Alamuri, B. R. Surampudi, and A. Negi, "A survey of distance/similarity measures for categorical data," in Proc. Int. Joint Conf. Neural Netw., 2014, pp. 1907–1914.
18. Z. Huang, "A fast clustering algorithm to cluster very large categorical data sets in data mining," in Proc. SIGMOG Workshop Res. Issues Data Mining Knowl. Discovery, 1997, pp. 1–8.
19. Z. Huang, "Extensions to the k-means algorithm for clustering large data sets with categorical values," Data Mining Knowl. Discovery, vol. 2, no. 3, pp. 283–304, 1998.
20. Z. Huang and M. K. Ng, "A fuzzy *k*-modes algorithm for clustering categorical data," *IEEE Transactions on Fuzzy Systems V*olume. 7, no. 4, pp. 446–452, Aug. 1999.

21. L. Bai, J. Liang, C. Dang, and F. Cao, “The impact of cluster representatives on the convergence of the K-modes type clustering,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1509–1522, Jun. 2013.
22. L. Bai, J. Liang, C. Dang, and F. Cao, “A novel attribute weighting algorithm for clustering high-dimensional categorical data,” *Pattern Recognit.*, vol. 44, no. 12, pp. 2843–2861, 2011.
23. Michael. K. Ng, Mark. J. Li, Joshua. Z. Huang, and Zengyou. He, “*On the impact of dissimilarity measure in k-modes clustering algorithm*,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 29, Issue. 3, pp. 503–507,2007.
24. M. K. Ng and L. Jing, “A new fuzzy k-modes clustering algorithm for categorical data,” *Int. J. Granular Comput., Rough Sets Intell. Syst.*, vol. 1, no. 1, pp. 105–119, 2009.
25. G. D, K. J et R. P, “Clustering categorical data: an approach based on dynamical systems.” *The Very Large Data Bases Journal*, vol. 8, issue 4, p. 222–236, 2000.
26. G. S, R. R et S. K, “*ROCK: a robust clustering algorithm for categorical attributes*.” Information Systems, vol. 25, issue 5, p. 345–366, 2000.
27. V. Ganti, J. Gehrke et R. Ramakrishnan, “CACTUS – clustering categorical data using summaries.” 5th *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1999.
28. S. Ben Salem, S. Naouali, Z. Chtourou “A fast and effective partitional clustering algorithm for large categorical datasets using a k -means based approach.” May 2018, Computers & Electrical Engineering 68:463-483.
29. S. Ben Salem, S. Naouali, Z. Chtourou , “A Computational Cost-Effective Clustering Algorithm in Multidimensional Space Using the Manhattan Metric: Application to the Global Terrorism Database” 19th International Conference on Machine Learning and Applications (ICMLA) 2017.
30. S. Ben Salem, S. Naouali, Z. Chtourou “Clustering Categorical Data Using the k-Means Algorithm and the Attribute's Relative Frequency Clustering Categorical Data Using the k-Means Algorithm and the Attribute's Relative Frequency” 19th International Conference on Machine Learning and Applications (ICMLA), 2017.
31. Dae-Won Kim, Kwang H. Lee, Doheon Lee, “Fuzzy clustering of categorical data using fuzzy centroids.” *Pattern Recognition Letters*, vol. 25, issue 11, p. 1263–1271, 2004.
32. M. Yang, Y. Chiang, C. Chen et C. Lai, “*A fuzzy k-partitions model for categorical data and its comparison to the GoM model.*” Fuzzy Sets Systems, Volume 159, Issue 4, pp. 390-405, 2008.
33. Iwan Tri Riyadi Yanto, Maizatul Akmar Ismail, Tutut Herawan, “*A modified Fuzzy k-Partition based on indiscernibility relation for categorical data clustering*.” Engineering Applications of Artificial Intelligence, vol. 53, p. 41–52, 2016.
34. G. Gang, J.Wu J et Yang. Z, “*A genetic fuzzy k-Modes algorithm for clustering categorical data*.” Expert Systems with Applications, Volume 36, pp. 1615-1620, 2009.
35. Liang Bai, Jiye Liang, Chuangyin Dang, Fuyuan Cao, “*A novel fuzzy clustering algorithm with between-cluster information for categorical data*.” Fuzzy Sets Systems, vol. 215, pp. 55–73, 2013.
36. Fuyuan Cao, Joshua Zhexue Huang and Jiye Liang, “*A fuzzy SV-k-modes algorithm for clustering categorical data with set-valued attributes*.” Applied Mathematics and Computation, Volume. 295, pp. 1–15, 2017.
37. Yihong Dong, Yueting Zhuang, Ken Chen, Xiaoying Tai, “*A hierarchical clustering algorithm based on fuzzy graph connectedness.*” Fuzzy Sets and Systems 157, pp 1760–1774, 2006.
38. Yang Yan, Lihui Chen, William-Chandra Tjhi, “*Semi-supervised fuzzy co-clustering algorithm for document categorization*.” Knowledge and Information Systems (2013) 34:55–74 DOI 10.1007/s10115-011-0454-9.
39. P. Z et S. A, “Rudiments of rough sets.” *Information Sciences: An International Journal* , Volume 177, Issue 1, p. 3–27, 2007.

40. J. M. L, A. He, Z. Y et C. S, “A rough set approach in choosing clustering attributes.” *Proceedings of the ISCA 13th, International Conference (CAINE-2000)*, 2000.
41. P. D, W. T et B. J, “*MMR: an algorithm for clustering categorical data using rough set theory*.” Data and Knowledge Engineering, vol. 63, p. 879–893, 2007.
42. Z. He, S. Deng, and X. Xu, “*Improving k-Modes Algorithm Considering Frequencies of Attribute Values in Mode.*” International Conference on Computational Intelligence and Security, pp. 157-162, 2005.
43. F. Cao, J Liang, L. Bai, “*A new initialization method for categorical data clustering*.”, Expert Systems and Applications Volume 36, Issue 7, pp 10223–10228, 2009.
44. S.S. Khan, A. Ahmad, “*Cluster center initialization algorithm for K-modes clustering*.”, Expert Systems with Applications Volume 40, Issue 18, pp 7444–7456, 2013.
45. Feng Jiang, Guozhu Liu, Junwei Du, Yuefei Sui, “*Initialization of k-modes clustering using outlier detection techniques.*” Information Sciences, Volume 332, pp 167-183, 2016.
46. S. Deng, Z. He et X. Xu, “*G-ANMI: A mutual information based genetic clustering algorithm for categorical data*.” Knowledge-Based Systems, vol. 23, pp. 144--149, 2010.
47. Liang Bai, Jiye Liang, “The k-modes type clustering plus between-cluster information for categorical data.” Neurocomputing 133 pp 111–121, 2014.
48. Bridgett M. vonHoldt et al., "*Genome-wide SNP and haplotype analyses reveal a rich history underlying dog domestication*", nature letters Vol 464 April 2010 |doi:10.1038/nature08837.
49. Z. He, X. Xu et Deng, “Squeezer: An efficient algorithm for clustering categorical data.” *Journal of Computer Science and Technology,* vol. 17, 2002.
50. H. Z, X. X et D. S, “Scalable algorithms for clustering large datasets with mixed type attributes.” *International Journal of Intelligent Systems,* vol. 20, Issue 10, p. 1077–1089, 2005.
51. T. GE, P. D, K. S, K. C et P. P, “Fuzzy clustering of categorical attributes and its use in analyzing cultural data.” *International Journal of Computational Intelligence, V*olume 1, Issue 2, pp. 147–151, 2004.
52. H. ZX, “Clustering large datasets with mixed numeric and categorical values.” Proceedings of the 1st Pacific Asia Knowledge Discovery and Data Mining Conference, World Scientific., Singapore, 1997.
53. H. CC, C. CL et S. YW, “Hierarchical clustering of mixed data based on distance hierarchy.” *Information Sciences,* vol. 177, Issue 2, pp. 474–492, 2007.
54. C. T, F. D, C. J, W. Y et J. C, “A robust and scalable clustering algorithm for mixed type attributes in large database environment.” *Proceedings of the 2001 International Conference on Knowledge Discovery and Data Mining (KDD01),* San Fran1cisco, 2001.
55. B. D, C. J et L. Y, “COOLCAT: an entropy-based algorithm for categorical clustering.” *Proceedings of the 7th International Conference on Information and Knowledge Management*, USA, 2002.
56. Lobelo, Felipe, et al. "Cardiorespiratory fitness and clustered cardiovascular disease risk in US adolescents." *Journal of Adolescent Health* 47.4 (2010): 352-359.
57. Azar, Ahmad Taher, Shaimaa Ahmed El-Said, and Aboul Ella Hassanien. "Fuzzy and hard clustering analysis for thyroid disease." *Computer methods and programs in biomedicine* 111.1 (2013): 1-16.
58. Paul, Razan, and Abu Sayed Md Latiful Hoque. "Clustering medical data to predict the likelihood of diseases." *Digital Information Management (ICDIM), 2010 Fifth International Conference on*. IEEE, 2010.
59. Aggarwal, Charu C., and S. Yu Philip. "On clustering massive text and categorical data streams." *Knowledge and information systems* 24.2 (2010): 171-196.
60. Chen, Jin-Yin, and Hui-Hao He. "A fast density-based data stream clustering algorithm with cluster centers self-determined for mixed data." *Information Sciences* 345 (2016): 271-293.

61. Buczak, Anna L., and Erhan Guven. "A survey of data mining and machine learning methods for cybersecurity intrusion detection." *IEEE Communications Surveys & Tutorials* 18.2 (2016): 1153-1176.
62. Singh, Raman, Harish Kumar, and R. K. Singla. "An intrusion detection system using network traffic profiling and online sequential extreme learning machine." *Expert Systems with Applications* 42.22 (2015): 8609-8624.
63. Alishahi, Mina Sheikh, Mohamed Mejri, and Nadia Tawbi. "Clustering spam emails into campaigns." *Information Systems Security and Privacy (ICISSP), 2015 International Conference on*. IEEE, 2015.
64. Claveria, Oscar, and Alessio Poluzzi. "Positioning and clustering of the world's top tourist destinations by means of dimensionality reduction techniques for categorical data." *Journal of Destination Marketing & Management* 6.1 (2017): 22-32.
65. Adnan Amin, Sajid Anwar, Awais Adnan, Muhammad Nawaz, Khalid Alawfi, Amir Hussain and Kaizhu Huang, Customer Churn Prediction in Telecommunication Sector using Rough Set Approach, *Neurocomputing,* http://dx.doi.org/10.1016/j.neucom.2016.12.009.
66. Phillips, Peter, and Ickjai Lee. "Crime analysis through spatial areal aggregated density patterns." *Geoinformatica* 15.1 (2011): 49-74.
67. Jiang, Shan, Joseph Ferreira, and Marta C. González. "Clustering daily patterns of human activities in the city." *Data Mining and Knowledge Discovery* 25.3 (2012): 478-510.
68. L. Bai, J.Y. Liang, C. Y. Dang, "*An initialization method to simultaneously find initial cluster centers and the number of clusters for clustering categorical data*." Knowledge-Based Systems Volume 24, Issue 6, pp 785-795, 2011.
69. T. Bai, C.A. Kulikowski, L.G. Gong, B. Yang, L. Huang, C.G. Zhou, "*A Global K-modes Algorithm for Clustering Categorical Data*.", Chinese Journal of Electronics, Volume 21, Issue 3, pp 460–465, 2012.
70. Wu, S., Jiang, Q., & Huang, J. Z. "*A new initialization method for clustering categorical data.*" In Proceedings of the 11th Pacific-Asia conference on advances in knowledge discovery and data mining PAKDD'07 (pp. 972–980). Berlin, Heidelberg: Springer-Verlag.
71. Khan, S. Ahmad, A, "*Cluster center initialization for categorical data using multiple attribute clustering*." In E.Mulle, T. Seid, S. Venkatasubramanian, workshop proceedings of the 3rd multicast workshop: discovering, summarizing and using multiple clustering, USA, 2012.
72. Gong, Yue-Jiao, et al. "*Genetic learning particle swarm optimization*." IEEE transactions on cybernetics 46.10 (2016): 2277-2290.
73. Z. He, X. Xu, S. Deng, "k-ANMI: a mutual information based clustering algorithm for categorical data.", Information Fusion 9 (2) (2008) 223–233.
74. Hongwu Qin, Xiuqin Ma, Tutut Herawan, Jasni Mohamad Zain, "*MGR: An information theory based hierarchical divisive clustering algorithm for categorical data*.", Knowledge-Based Systems, 2014.
75. Tengke Xiong, Shengrui Wang, André Mayers, Ernest Monga, "*DHCC: Divisive hierarchical clustering of categorical data*". Data Mining and Knowledge Discovery (2012) 24:103–135 DOI 10.1007/s10618-011-0221-2.
76. L.A. Zadeh, Fuzzy sets, Inform. and Control 8 (1965).
77. Qian, Yuhua, et al. "Positive approximation: an accelerator for attribute reduction in rough set theory." *Artificial Intelligence* 174.9-10 (2010): 597-618.
78. Amin, A., Anwar, S., Adnan, A., Khan, M. A., & Iqbal, Z. (2015). Classification of cyber attacks based on rough set theory 2015. 1st International Conference on AntiCybercrime, ICACC 2015.
79. Shi, Lei, et al. "Rough set and ensemble learning based semi-supervised algorithm for text classification." *Expert Systems with Applications* 38.5 (2011): 6300-6306.

80. Chen, Hui-Ling, et al. "A support vector machine classifier with rough set-based feature selection for breast cancer diagnosis." *Expert Systems with Applications* 38.7 (2011): 9014-9022.
81. Huang, Bing, et al. "Using a rough set model to extract rules in dominance-based interval-valued intuitionistic fuzzy information systems." *Information Sciences* 221 (2013): 215-229.
82. Shrivastava, Shailendra Kumar, and Preeti Jain. "Effective anomaly based intrusion detection using rough set theory and support vector machine." *International Journal of Computer Applications* 18.3 (2011): 35-41.
83. An, Shuang, et al. "Fuzzy rough regression with application to wind speed prediction." *Information Sciences* 282 (2014): 388-400.
84. Greco, Salvatore, Roman Słowiński, and Piotr Zielniewicz. "Putting dominance-based rough set approach and robust ordinal regression together." *Decision Support Systems* 54.2 (2013): 891-903.
85. Z. Qinghua, X. Qin et W. Guoyin, "A survey on rough set theory and its applications." *CAAI Transactions on Intelligence Technology,* vol. 1, pp. 323-333, 2016.
86. M. Quafafou, "α-RST: a generalization of rough set theory." *Information Sciences,* vol. 124, p. 301–316, 2000.
87. Lingras, Pawan, and Georg Peters. "Applying rough set concepts to clustering." *Rough Sets: Selected Methods and Applications in Management and Engineering*. Springer, London, 2012. 23-37.
88. Adnan Amin, Babar Shah, Asad Masood Khattak, Fernando Joaquim Lopes Moreira, Gohar Ali, Alvaro Rocha, Sajid Anwar, Cross-company customer churn prediction in telecommunication: A comparison of data transformation methods, International Journal of Information Management, 2018.
89. Amin, Adnan, et al. "Churn prediction in telecommunication industry using rough set approach." *New Trends in Computational Collective Intelligence*. Springer, Cham, 2015. 83-95.
90. Amin, Adnan, et al. "Customer churn prediction in the telecommunication sector using a rough set approach." *Neurocomputing* 237 (2017): 242-254.
91. Amin A., Rahim F., Ali I., Khan C., Anwar S. (2015) A Comparison of Two Oversampling Techniques (SMOTE vs MTDF) for Handling Class Imbalance Problem: A Case Study of Customer Churn Prediction. In: Rocha A., Correia A., Costanzo S., Reis L. (eds) New Contributions in Information Systems and Technologies. Advances in Intelligent Systems and Computing, vol 353. Springer, Cham.
92. N N R, Ranga Suri, Musti Narasimha Murty, Gopalasamy Athithan "A *Rough Clustering algorithm for mining outliers in categorical data*." 5th International Conference on Pattern Recognition and Machine Learning, 2013.
93. D. Qi, Y. You Long et L. Yang, "Rough k-modes Clustering Algorithm Based on Entropy." IAENG International Journal of Computer Science, vol. 44, issue 1, pp. 13-18, 2017.
94. I. Saha, J. Sarkar et M. P, "Rough Set Based Fuzzy k-Modes for Categorical Data." SEMCCO 2012. Lecture Notes in Computer Science, 2012.
95. Duo Chen, Du-Wu Cui, Chao-Xue Wang, Zhu-Rong Wang, "*A Rough Set-Based Hierarchical Clustering Algorithm for Categorical Data*.", International Journal of Information Technology, Vol.12, No.3, 2006.
96. Tutut Herawan, Mustafa Mat Deris, Jemal H. Abawajy, "*A rough set approach for selecting clustering attribute*." Knowledge-Based Systems 23, pp 220–231, 2010.
97. T. BK et G. Adhir, "*SSDR: An Algorithm for Clustering Categorical Data Using Rough Set Theory*." Advances in Applied Science Research, vol. 2, issue 3, pp. 314-326, 2011.
98. Min Li, Shaobo Deng, Lei Wang, Shengzhong Feng, Jianping Fan, "*Hierarchical clustering algorithm for categorical data using a probabilistic rough set model.*" Knowledge-Based Systems, 2014.

99. B.K. Tripathy, Akarsh Goyal, Rahul Chowdhury and Patra Anupam Sourav, “*MMeMeR: An Algorithm for Clustering Heterogeneous Data using Rough Set Theory.*" International Journal of Intelligent Systems and Applications, vol. 8, pp. 25-33, 2017.
100. Arbelaitz, Olatz, et al. "An extensive comparative study of cluster validity indices." *Pattern Recognition* 46.1 (2013): 243-256.
101. Günnemann, Stephan, et al. "External evaluation measures for subspace clustering." *Proceedings of the 20th ACM international conference on Information and knowledge management*. ACM, 2011.
102. Liu, Yanchi, et al. "Understanding of internal clustering validation measures." *Data Mining (ICDM), 2010 IEEE 10th International Conference on*. IEEE, 2010.
103. F.Y. Cao, J.Y. Liang, D.Y. Li, X.W. Zhao, A-weighting *k*-modes algorithm for subspace clustering of categorical data, Neurocomputing 108 (2013) 23–30.
104. Kryszczuk, Krzysztof, and Paul Hurley. "Estimation of the number of clusters using multiple clustering validity indices." *International Workshop on Multiple Classifier Systems*. Springer, Berlin, Heidelberg, 2010.
105. Gang Kou, Yi Peng, Guoxun Wang, Evaluation of clustering algorithms for financial risk analysis using MCDM methods, Information Sciences, Volume 275, 10 August 2014, Pages 1-12

View publication stats