Soft Intelligent Learning Techniques for Pattern Analysis

Thesis submitted by
Jnanendra Prasad Sarkar

DOCTOR OF PHILOSOPHY (Engineering)

Department of Computer Science and Engineering,
Faculty Council of Engineering & Technology,
Jadavpur University
Kolkata, India
2023

JADAVPUR UNIVERSITY

KOLKATA-700032, INDIA

INDEX NO. 118/17/E

1. Title of the Thesis:

   Soft Intelligent Learning Techniques for Pattern Analysis

2. Name, Designation & Institution of the Supervisor/s:

   (a) Dr. Ujjwal Maulik
   Professor
   Department of Computer Science and Engineering
   Jadavpur University, Kolkata-700032, India

   (b) Dr. Anasua Sarkar
   Asst. Professor
   Department of Computer Science and Engineering
   Jadavpur University, Kolkata-700032, India

   (c) Dr. Indrajit Saha
   Asst. Professor
   Department of Computer Science and Engineering
   National Institute of Technical Teachers' Training & Research, Kolkata-700106, India

List of Publications

Papers in Journals

1. J.P. Sarkar, I. Saha, S. Chakraborty, and U. Maulik, "Machine learning integrated Credibilistic semi supervised clustering for categorical data", Applied Soft Computing, vol. 86, p. 105871, 2020

2. I. Saha, J.P. Sarkar, and U Maulik, "Integrated rough fuzzy clustering for categorical data analysis", Fuzzy Sets and Systems, vol. 361, pp. 1–32, 2019

3. J.P. Sarkar, I. Saha, and U. Maulik, "Rough possibilistic type-2 fuzzy C-means clustering for MR brain image segmentation", Applied Soft Computing, vol. 46, pp. 527–536, 2016

4. I. Saha, J.P. Sarkar, and U. Maulik, "Ensemble based rough fuzzy clustering for categorical data", Knowledge Based Systems, vol. 77, pp. 114–127, 2015

Papers in Conference Proceedings

1. I. Saha, S. Rakshit, M. Denkiewicz, J.P. Sarkar, D. Maity, U. Maulik, and D. Plewczynski, “Survival Analysis with the Integration of RNA-Seq and Clinical Data to Identify Breast Cancer Subtype Specific Genes”, 8th International Conference on Pattern Recognition and Machine Intelligence, 2019

2. J.P. Sarkar, I. Saha, and U. Maulik, “Improved Fuzzy Clustering using Ensemble based Differential Evolution for Remote Sensing Image”, IEEE Region 10 Conference (TENCON), 2019

3. J. P. Sarkar, I. Saha, S. Rakshit, M. Pal, M. Wlasnowolski, A. Sarkar, U. Maulik, and D. Plewczynski, “A New Evolutionary Rough Fuzzy Integrated Machine Learning Technique for MicroRNA Selection using Next-Generation Sequencing Data of Breast Cancer”, GECCO’19: Genetic and Evolutionary Computation Conference Companion, 2019

4. J.P. Sarkar, I. Saha, A. Sarkar, and U. Maulik, “Improving Modified Differential Evolution for Fuzzy Clustering”, HIS 2017: 17th International Conference on Hybrid Intelligent Systems, 2017

5. J.P. Sarkar, I. Saha, and U. Maulik, “A new SVM integrated rough type-II fuzzy clustering technique”, 9th International Conference on Industrial and Information Systems (ICIIS), 2014

List of Presentations in National/International Conference:

1. I. Saha, S. Rakshit, M. Denkiewicz, J.P. Sarkar, D. Maity, U. Maulik, and D. Plewczynski, “Survival Analysis with the Integration of RNA-Seq and Clinical Data to Identify Breast Cancer Subtype Specific Genes”, 8th International Conference on Pattern Recognition and Machine Intelligence, 2019

2. J.P. Sarkar, I. Saha, A. Sarkar, and U. Maulik, “Improving Modified Differential Evolution for Fuzzy Clustering”, HIS 2017: 17th International Conference on Hybrid Intelligent Systems, 2017

STATEMENT OF ORIGINALITY

I Jnanendra Prasad Sarkar registered on 03/03/2017 do hereby declare that this thesis entitled "Soft Intelligent Learning Techniques for Pattern Analysis" contains a literature survey and original research work done by the undersigned candidate as part of Doctoral studies.

All information in this thesis has been obtained and presented in accordance with existing academic rules and ethical conduct. I declare that, as required by these rules and conduct, I have fully cited and referred all materials and results that are not original to this work.

I also declare that I have checked this thesis as per the "Policy on Anti Plagiarism, Jadavpur University, 2019", and the level of similarity as checked by iThenticate software is 04%.

Jnanendra Prasad Sarkar

Signature of the Candidate:

Date: 03/10/2023

Certified by Supervisor:
(Signature with date,seal)

Ujjwal Maulik 03/10/23

1. Dr. Ujjwal Maulik

Professor
Computer Sc. & Engg. Department
Jadavpur University
Kolkata-700032

Anasua Sarkar 03/10/23

2. Dr. Anasua Sarkar

ANASUA SARKAR
Assistant Professor
Computer Science & Engineering Department
Jadavpur University

Indrajit Saha 03/10/2023

3. Dr. Indrajit Saha

Dr. Indrajit Saha
Assistant Professor, Computer Science & Engineering
NATIONAL INSTITUTE OF TECHNICAL TEACHERS' TRAINING & RESEARCH
(Established by the Ministry of Education, Govt. of India)
Block-FC, Sector-III, Salt Lake City, Kolkata-700106

CERTIFICATE FROM THE SUPERVISOR

This is to certify that the thesis entitled **Soft Intelligent Learning Techniques for Pattern Analysis** submitted by Jnanendra Prasad Sarkar, who got his name registered on 03.03.2017 for the award of Ph.D. (Engineering) degree of Jadavpur University, is absolutely based upon his own work under the supervision of Prof. Dr. Ujjwal Maulik, Department of Computer Science and Engineering, Jadavpur University, Kolkata-700032, India, Dr. Anasua Sarkar, Department of Computer Science and Engineering, Jadavpur University, Kolkata-700032, India, Dr. Indrajit Saha, Department of Computer Science and Engineering, National Institute of Technical Teachers' Training & Research, Kolkata-700106, India, and that neither his thesis nor any part of the thesis has been submitted for any degree/diploma or any other academic award anywhere before.

Ujjwal Maulik 03/10/23

....

(Dr. Ujjwal Maulik)
Signature of the Supervisor
and date with Office Seal

Professor
Computer Sc. & Engg. Department
Jadavpur University
Kolkata-700032

Anasua Sarkar 03/10/23

....

(Dr. Anasua Sarkar)
Signature of the Supervisor
and date with Office Seal

ANASUA SARKAR
Assistant Professor
Computer Science & Engineering Department
Jadavpur University

Indrajit Saha 03/10/2023

....

(Dr. Indrajit Saha)
Signature of the Supervisor
and date with Office Seal

Dr. Indrajit Saha
Assistant Professor, Computer Science & Engineering
NATIONAL INSTITUTE OF TECHNICAL TEACHERS'
TRAINING & RESEARCH
(Established by the Ministry of Education, Govt. of India)
Block-FC, Sector-III, Salt Lake City, Kolkata-700106

Dedication

To my family and supervisor

Acknowledgements

Finally, it is time for me to acknowledge all those who inspired me, supported me and helped me to get to the place where I am today.

I take this opportunity to express a deep sense of gratitude to Prof. (Dr.) Ujjwal Maulik, (Dr.) Anasua Sarkar and (Dr.) Indrajit Saha for their supervision and invaluable co-operation. This thesis would not have been possible without constant inspiration and unbelievable support of them over the last few years.

I have had an amazing group of Lab-mates. Each of them deserves my gratitude: Ashmita Dey, Kakuli Mishra and Sagnik Sen. Working with them was a great experience that brought together several good and fruitful ideas. Thank you to all of you for the unselfish help, insights, feedback, and for making the science a collaborative effort. With that, a special thanks to Dr. Ranjib Biswas and Dr. Rajendra Ghosh. The Journey would remain incomplete without their suggestions, unconditional help, and encouragement.

I would also like to thank all of my colleagues from Computer Science and Engineering Department, Jadavpur University for providing me a friendly research environment.

Finally, most important of all, I would like to dedicate the thesis to my father Late Mr. Gopal Ch. Sarkar, my mother Mrs. Mira Sarkar, my daughter Miss. Ayanna Sarkar, and my wife Mrs. Kakali Sarkar to honor their love, patience, encouragement, and support during my research. I would also express my appreciation to my brother Mr. Rajendra Pr. Sarkar, my close friends and my office colleague for their unwavering supports and love.

Jnanendra Prasad Sarkar

Date: 03/10/2023

(Jnanendra Prasad Sarkar)

Abstract

Data mining is the process of grouping a set of data into groups that exhibit homogeneity. The increasing availability and use of data from diverse digital sources across several fields, such as telecommunications, geographic information systems (GIS), and commercial enterprises, has garnered significant interest. Categorical data clustering, which pertains to the grouping of data items characterized by non-numerical attributes, has garnered significant interest in recent times. Real-world datasets can exhibit overlapping characteristics and inherent ambiguities. Fuzzy set theory is widely used as a prominent approach for addressing uncertainty, but encountering challenges in environments characterized by noise. The conventional type-1 fuzzy concept is limited in its ability to accommodate a wide range of intrinsic uncertainty. As a result, the type-2 fuzzy set theory has been extensively explored in many scholarly works. Fuzziness, in the context of pattern recognition, is an inherent and basic source of uncertainty. The main membership value of the type-1 fuzzy concept is also characterized by ambiguity. The main membership pertains to the stochastic nature of the feature space, whereas the secondary membership of type-2 fuzzy characterizes the degree of fuzziness associated with the primary membership value.

In addition, there have been some novel methodologies developed to address the issue of managing uncertainty in the clustering process. In order to address the constraints inherent in fuzzy-based clustering algorithms and its derivatives, other approaches such as possibilistic and rough set-based clustering have been proposed. Nevertheless, a significant portion of clustering algorithms exhibit limitations in properly addressing vagueness, ambiguity, and indiscernibility, or were primarily developed with a focus on numerical data. In environments characterized by high levels of noise, probabilistic approaches provide superior performance compared to fuzzy clustering. However, it is important to note that probabilistic techniques are not without limitations, since they are susceptible to the coincident issue when dealing with clusters that are in close proximity to each other. Rough set based clustering, via its use of lower and higher approximations, demonstrates the ability to effectively handle indiscernibility. However, it is important to note that its primary purpose does not extend to addressing coincident difficulties. The credibilistic theory seems to be beneficial in addressing concurrent challenges associated with the concept of self-duality.

Currently known clustering algorithms have two primary drawbacks. First, the bulk of these creations were formulated based on one or two mathematical concepts. Given the existence of several challenges such as overlapping partition, ambiguity, vagueness, indiscernibility, and coincident clustering difficulty, it is evident that a single solution cannot effectively address all of these issues. Second, it is worth noting that although there has been some focus on clustering categorical data with inherent complexity, the majority of existing approaches have mostly been developed for numerical data.

To address the aforementioned challenges, the current thesis proposes soft intelligent learning techniques that are comprehensively elaborated upon in Chapters 2 to 5. To accomplish this, a preliminary ensemble-based rough fuzzy clustering method is proposed for handling categorical data. The experiment is thereafter conducted using six synthetic and four real data sets. This approach uses both the rough set and fuzzy set principles. The proposed approach involves the integration of possibilistic and type-2 fuzzy concepts, resulting in the development of an extended clustering algorithm. Rough sets are capable of addressing uncertainty and ambiguity by using the notion of lower and upper approximation. On the other hand, possibilistic ideas are designed to tackle challenges associated with noisy data and outliers. In contrast, the use of the type-2 fuzzy set methodology enables the handling of uncertainty and unpredictability within the clustering process.

Due to the random selection of starting cluster modes, the rough fuzzy-based clustering technique is susceptible to the problem of local optima. As a result, the use of multi-phase learning is employed in order to develop an integrated clustering methodology. To enhance clustering performance in this regard, two approaches have been developed: Simulated Annealing-based Rough Fuzzy *K*-Modes and Genetic Algorithm-based Rough Fuzzy *K*-Modes. The process involves seeing clustering as a fundamental problem of optimisation. Each of these clustering algorithms yields a cluster characterized by a certain arrangement of central and peripheral points. Subsequently, in order to enhance the clustering outcomes, Random Forest is used to allocate peripheral points to certain crisp clusters for each example, with the central points serving as the training set. Additionally, the dissimilar cardinality of the training and testing sets produced by each clustering technique has led to the proposition of a comprehensive approach known as Integrated Rough Fuzzy Clustering using Random Forest. In this approach, the roughness measure is determined by utilizing the outcomes of the three aforementioned clustering techniques. This measure is used to delineate three distinct sets, often known as *best central points*, *semi-best central points* and *pure peripheral points*. Subsequently, using a multi-phase learning approach, the most optimal central points are used for the purpose of categorizing the semi-optimal central points. The Random Forest algorithm uses the aforementioned features to classify outlying areas that have pure peripheral points. The effectiveness of the recommended methodology is shown by quantitative, visual, and statistical analysis of experimental outcomes on six synthetic datasets and five real-world datasets, in contrast to well-known state-of-the-art methods.

Two primary challenges often mentioned in pattern analysis in real-world scenarios are the scarcity of accurately labeled data and the effective handling of categorical data. Clustering techniques are used to categorize unlabeled data based on its uniformity. Semi-supervised approaches have the potential to be advantageous in this particular scenario. Nevertheless, the resolution of coincident issues remains challenging when using possibilistic measure, rough set theory, and type 2 fuzzy measure. A novel strategy has been developed to address the challenges associated with categorizing categorical data, as previously mentioned. This approach uses a semi-supervised clustering technique that incorporates credibilistic measure and integrates machine learning methods. The use of a credibilistic measure assists in identifying homogeneity in this scenario by addressing the problem of coincident clustering and accurately identifying the points that belong to certain clusters. The dataset that has been clustered is then used to construct a supervised model, which aims to categorize new data that is either unlabeled or uncertain . This classification process is carried out using a semi-supervised technique. Furthermore, apart from its enhanced capability in managing unannotated data, this approach also demonstrates higher performance when dealing with ambiguous or uncertain data instances, when the credibilistic measure is same across several classes. The performance of the recommended methodology is shown via the usage of eight synthetic datasets and four real-world datasets. These datasets are used to conduct quantitative, visual, and statistical comparisons between the proposed approach and generally recognized state-of-the-art techniques.

In conclusion, the thesis provides an overview of the results obtained from each proposed approach, while also acknowledging any limitations encountered throughout the research process. Furthermore, the thesis suggests prospective avenues for further study by building upon the notions put forward.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

The proliferation of digitization in recent years has resulted in the generation of substantial volumes of data from diverse electronic sources across numerous fields, such as telecommunications, geographic information systems (GIS), and commercial business sectors. The datasets exhibit a diverse range of sizes and kinds, and are characterised by their substantial magnitude. Therefore, researchers have become interested in utilising computational intelligence and advanced data mining techniques, such as machine learning algorithms and statistical analytical models, to extract valuable insights from large datasets and develop sophisticated predictive analytical methods across different domains. The approaches used in this study include evolutionary computing [2], artificial neural network inspired by the human brain system [3], fuzzy and rough sets theory for managing ambiguity and overlaps [4], Random forest [5], and Support Vector Machine [6].

Pattern analysis is a crucial technique used to extract information and identify patterns within datasets. This method has been extensively studied and documented in the academic literature [7–11]. It encompasses a range of mathematical methodologies [12]. This technique involves the allocation of data items to either known classes, referred to as supervised learning, or unknown classes, known as unsupervised learning [13]. The primary technique used in unsupervised learning is known as clustering [14]. This approach involves the partitioning of a dataset into distinct groups, depending on the shared characteristics shown by its patterns. Segmenting data in various data sets may sometimes be a complex task, as there is often a lack of distinct criteria for dividing the data. Consequently, clustering becomes a formidable process in such scenarios. The clustering of categorical data is a greater challenge due to the absence of a consistent natural ordering among the attribute domain components. While several clustering approaches exist, the majority of them are designed for numerical data, leaving categorical data clustering relatively understudied.

Furthermore, it is important to note that real-world datasets include a variety of data types, including categorical, numerical, and mixed data. These datasets often exhibit overlapping characteristics, as well as inherent uncertainty and vagueness. The assessment of similarity and dissimilarity has significant importance in the domains of clustering and classification. Certain measures rely on mutual correlation, whilst others are grounded on Euclidean geometry. In this context, a plethora of algorithms have been introduced in the last several decades to tackle a wide range of complex difficulties. The bulk of the approaches that have been published in the literature focus on the analysis of numerical data via the use of diverse similarity and dissimilarity criteria. Moreover, a significant proportion of methodologies exhibit certain constraints when dealing with real-world data as a result of inherent ambiguity and uncertainty. Nevertheless, the study of categorical data has been quite restricted.

The aforementioned facts have provided motivation to conduct a more comprehensive analysis of clustering, get a thorough understanding of the many strategies now in use and their respective limitations, and propose the implementation of integrated and ensemble-based soft intelligent learning techniques for clustering. These proposed techniques aim to mitigate the constraints inherent in existing methods. The strategies that have been developed also demonstrate the ability to effectively address challenges related to vagueness, uncertainty and indiscernibility, especially when dealing with categorical data.

1.2 Preliminaries

This section provides an overview of pattern analysis and soft computing approaches, which are considered fundamental prerequisites for the topic at hand.

1.2.1 Pattern Analysis

Pattern analysis [15] is an essential technique used in advanced analytical applications, including consumer segmentation [16], medical systems [17], mobile performance marketing [18] and several other domains to extract valuable information and patterns from datasets. The goal of pattern analysis is to find pattern and accordingly put them into known or unknown groups. The first one is called supervised classification, while the second one is called unsupervised learning. Supervised classification is used mostly as a supervised learning technique and clustering for unsupervised learning technique.

Classification

Classification strategies aim to allocate each data point inside the whole of the feature space to one of the K possible classes. Classifiers are often developed using annotated data, hence leading to the designation of such issues as supervised classification in some instances. Several widely used classification techniques include the Support Vector Machine (SVM) [19–22], Artificial neural network (ANN) [23], K Nearest Neighbour (KNN) [24, 25], Decision Tree (DT) [26], Random Forest (RF) [27], Naive Bayes(NB) [28] etc. The use of training and test data is crucial in the development of a supervised classification model. In the realm of machine learning, training data is often comprised of labeled data that is sufficiently extensive to include occurrences of all the labels. Typically, the training data is used to train the supervised classifier, enabling it to provide prediction outcomes for unlabeled data. In order to enhance the classifier's ability to handle unfamiliar data, it is crucial to assess the accuracy of its predictions. Consequently, an alternative collection of annotated data is used for the purpose of evaluating the classifier, often referred to as test data. The classifier is not trained using any test data. In addition, it is important for the test data to accurately reflect the characteristics of the original dataset and possess a sufficient size in order to provide meaningful predictions.

Support Vector Machine:
The Support Vector Machine (SVM) is a supervised classification technique that use a data-driven approach to address classification tasks. In the case of a high number of characteristics, it has been observed that the prediction error is less [29]. Support Vector Machines (SVMs) are specifically developed to optimize the margin between two classes, hence facilitating the generalization of unseen data by the trained model. In a feature space of dimension d, the Support Vector Machine (SVM) algorithm generates a hyperplane that maximally separates the two classes of data points in the d-dimensional space. Two parallel hyperplanes are formed on either side of the splitting hyperplane, which are positioned next to the two classes of the data points. An optimal separation is attained by the hyperplane that maximizes the distance between the adjacent data points of each class. A greater separation between the two parallel hyperplanes is indicative of a superior generalization error of the SVM classifier. From a geometric perspective, the support vectors may be defined as the data points that are in close proximity to the splitting hyperplane. Kernel functions are a kind of mathematical function that takes an input space and transforms it into a space with a higher dimension. There are several types of kernel functions, including linear, radial basis functions (RBF), sigmoidal, polynomial etc. The SVM classifier is primarily intended for binary classification tasks. Multi-class issues may be addressed by using a series of one-against-all two class Support Vector Machines (SVMs). Support Vector Machines (SVMs) are extensively used in several disciplines, including Bioinformatics, with the objective of reducing mistakes.

Artificial neural network:
The artificial neural network (ANN) is a versatile and potent technique within the field of machine learning. In the fields of topology and function, Artificial Neural Networks (ANNs) serve as a computational model that shares similarities with the structure and functioning of the human brain. The transmission of signals occurs between input and output nodes. The input signals undergo a process of weighting prior to reaching the output nodes, with each signal being assigned a weight that corresponds to its relative significance. Subsequently, the collective signal undergoes processing by an activation function.

K **Nearest Neighbour:**
The *K* Nearest Neighbour (*K*-NN) algorithm is a fundamental approach in the field of machine learning. The classification of an item in this method is determined by a neighbouring object that receives the majority of votes. The item is thereafter allocated to the class that is most prevalent among its *K* nearest neighbours (KNN), where *K* is a positive integer that is normally of modest magnitude. In the case when *K* is equal to 1, the object is straightforwardly allocated to the class of its closest neighbour.

Decision Trees:
Decision trees, also known as DTs, are created by the analysis of a collection of training samples in which the class labels are already known. Subsequently, these techniques are used to categorise instances that have not been seen before. When decision trees are trained with high-quality data, they have the potential to provide very accurate predictions. The decision tree algorithm is used to classify data instances by iteratively asking a sequence of inquiries about the attributes linked to these instances. Every question is encapsulated inside a node, and each internal node has many child nodes, each corresponding to a potential solution to the question. The questions are organised in a hierarchical structure, shown as a tree.

Random Forests:
Random forests (RF) are constructed by using decision trees as its fundamental building blocks. These decision trees consist of a series of binary choices, which are determined by the optimal values of model parameters. These decisions aim to effectively partition the data into their appropriate classifications. A random forest is created by building a significant number of decision trees, where each tree uses a random subset of model parameters to determine each split in the tree. The eventual classification of a particular data set is determined by the mode of the classifications obtained from all the distinct decision trees in the random forest.

Naive Bayes:
Naive Bayes (NB) classification is a probabilistic classification approach that is based on Bayes' theorem. It assumes that the characteristics used for classification are independent of each other. Bayes' theorem is a mathematical principle that enables the prediction of an event by using past knowledge and current information. The forecast has been revised in light of the growing body of information.

Clustering

While data are not labeled, the classification problem is referred to unsupervised learning and clustering [30–35] is a crucial technique for unsupervised learning with applications in a wide range of domains. As a basic composition of pattern analysis, clustering plays an important role. In clustering, a set of patterns which are normally a vector in a multidimensional space are grouped into clusters in such a way that patterns in the same cluster are similar in some sense and patterns in different clusters are dissimilar in the same sense. Either a model-based approach or a distance-based method is used for clustering. In distance-based technique, one must establish a similarity or proximity metric on the basis of which cluster assignments are made in order to divide a data set. On the other hand, model-based approaches use the assumption that data is produced by a finite mixture model, and as a result, various model parameters are estimated from data in order to group data into various clusters based on the posterior probability. Broadly clustering algorithms are categorized into three categories namely hierarchical, partitional and density-based. Most popular and widely used algorithms are briefly described below.

Hierarchical Clustering:
Hierarchical clustering [36] is a technique that organises variables into a hierarchical tree structure, with the length of branches indicating the level of similarity among the variables. The objective is often accomplished by using an agglomerative methodology, in which nodes are combined and organised into a hierarchical tree structure known as a dendrogram. The procedure entails the use of several connection techniques, such as single, complete, and average linkage. Different linking techniques use unique algorithms to compute the distances between nodes and merged clusters. The **single linkage** approach involves merging groups based on the smallest distance between two data points belonging to separate categories. In contrast, the **complete linkage** method is characterised by the merging of groups through the consideration of the maximum distance between two data points originating from distinct groups. Conversely, the **average linkage** method

entails the merging of groups by taking into account the average distance between all data points within one group and all data points within the other group. In the realm of hierarchical clustering, several notable algorithms include BIRCH [37], CURE [38], and ROCK [39].

Partitional Clustering:

Partitional clustering [36] refers to the process of partitioning a set of data points into distinct and non-overlapping groups, commonly known as clusters. In this method, each data point is assigned to a single group, ensuring that there is no overlap between clusters. There are several partitional clustering algorithms, some of which are referred to as hard clustering and others as soft clustering. In the context of hard clustering, each data point is assigned exclusively to one specific group. For instance, the algorithms known as *K*-Means [40], *K*-Medoids [41], and *K*-Modes [42] are extensively employed in various domains. Conversely, soft clustering algorithms allow for the possibility that a data point can be assigned to multiple clusters simultaneously. Fuzzy-based algorithms and their variants serve as examples of soft clustering algorithms. This section provides a brief discussion of the two most popular algorithms, namely *K*-Means and *K*-Medoids. Additionally, an overview of fuzzy-based soft clustering can be found in section 1.2.2.

The *K***-Means** algorithm is widely recognized as the most commonly employed clustering technique for numerical data. The primary aim of this algorithm is to minimize the objective function as stated below:

$$\mathcal{J}(K) = \sum_{i=1}^{n} \sum_{k=1}^{K} u_{ik} D(c_k, x_i) \tag{1.1}$$

Subject to:

$$\sum_{k=1}^{K} u_{ik} = 1, \;\; 1 \leq \; i \leq \; n$$
$$u_{ik} \in \{0, 1\}, \;\; 1 \leq \; i \; \leq \; n, \;\; 1 \leq \; k \; \leq \; K$$

Here, the variable W is represented as a partition matrix, denoted by $[u_{ik}]$ of size $n \times K$, where each element u_{ik} is equal to 1 if the data point x_i belongs to the cluster k, and 0 otherwise. *D(.)* denotes a distance measure between ith point x_i and kth center, c_k, whereas n and K represent the number of points and number of clusters respectively. The cluster center is mathematically computed as follows.

$$c_k = \frac{\sum_{i=1}^{n} u_{ik} x_i}{\sum_{i=1}^{n} u_{ik}}, \quad 1 \leq k \leq K \tag{1.2}$$

The steps of *K*-Means algorithm is as follows.

Algorithm 1 Steps of *K*-Means

```
Input:
    $X$: dataset
    $\epsilon$, threshold value which is very small real value between [0,1]
    $K$, number of clusters

Select $K$ random points from dataset to initiate $K$ cluster centers
repeat
    Compute distance of all the points from all the cluster centers
    Assign the data points to their respective closest cluster based on the disctance between the data point and respective cluster center
    Compute mean of each cluster as new cluster center
    Compute value of objective function using Equation 1.1
until $|Current\ \mathcal{J}(K) - Previous\ \mathcal{J}(K)| \leq \epsilon$
```

The *K*-**Modes** technique, as proposed by Huang [42], was developed based on the *K*-Means [40] paradigm for clustering categorical data . There are two fundamental distinctions between the *K*-Modes and the *K*-Means algorithm. Firstly, the assessment of dissimilarity of attribute values for categorical data. Secondly, instead of computing the mean of the cluster, it calculates it by considering the frequency of the attribute values. The objective function that is minimised when using the *K*-Modes algorithm is as follows:

$$\mathcal{H}(K) = \sum_{l=1}^{K} \sum_{x_i \in V_l} D(v_l, x_i) \tag{1.3}$$

Here, *D(.)* denotes a dissimilarity measure. The mode of a group of points/objects, v_l is a point/object (that does not necessarily belong to V_l) whose jth attribute value is calculated as the jth attribute's most frequent value across all of the points/objects in V_l. If there are multiple most frequent values, one of them is randomly selected. Therefore, if X is set of categorical objects that are defined by m categorical attributes and *D(.)* is dissimilarity measure, then mode of lth cluster ($V_l \subseteq X$) can be mathematically defined as a vector $v_l = [v_l 1, v_l 2, \ldots, v_l m]$, such that the following function is minimized [42].

$$\mathcal{D}(V_l, v_l) = \sum_{x_i \in V_l} D(v_l, x_i) \tag{1.4}$$

Up until $\mathcal{H}(K)$ stops changing, the algorithm iterates. A points/object is assigned to a cluster that has minimum dissimilarity with the points/object. The algorithm continues until objective function values is converged to a optimal value.

The K-**Medoids** (KMdd) clustering, is another variation of the K-Means with the goal of reducing the $\mathcal{Z}$(K) within cluster variance.

$$\mathcal{Z}(K) = \sum_{i=1}^{K} \sum_{x_i \in V_l} D(\hat{v}_l, x_i) \tag{1.5}$$

Here $\hat{v}_l$ is the medoid of cluster V_l. It uses cluster mediod as opposed to cluster mode, which is the main distinction from KMd. The most centrally situated point inside the cluster, or the location from which the total distances to the other points of the cluster are shortest, is referred to as a cluster medoid. Mathematically, the cluster mediod ($\hat{v}_l$) of cluster V_l is defined as follows.

$$\hat{v}_l = \underset{y \in V_l}{\operatorname{argmin}} \sum_{x_i \in V_l} D(y, x_i), \quad 1 \leq i \leq n \tag{1.6}$$

Density-based Clustering:

This particular algorithm [36] utilizes the density of data points within the data space in order to generate clusters. Clusters are formed to separate regions with higher density, while partitions are utilized for regions with very low density. This serves as a safeguard against outliers or noisy data. The process commences by considering unvisited arbitrary data-points and examining their surrounding vicinity. A cluster is formed only if there exists an adequate number of points within a specific distance, 'ϵ'. If not, the data point is labeled an outlier [43]. The aforementioned procedure is executed in an iterative manner for each collection of points that have not yet been visited.

DBSCAN [44] is the most common density-based clustering method. Utilising the idea of density reachability [36], the cluster is defined. If a point b is within a specific distance, *delta*, from another point a, it is said to be directly density-reachable from that other point. If a is surrounded by enough points in its *delta* neighbourhood that a and b may be thought of as a part of a cluster, then b is a part of a's *delta* neighbourhood.Furthermore, the term "indirectly density reachable" is used to describe the relationship between two points, a and b, if there exists a sequence of points $a_1, a_2, \ldots, a_p$ where $a_1 = a$ and $a_p = b$, and each a_{i+1} is directly density reachable from a_i. δ and minimum number of points (*mpts*) required to form a cluster are most important parameters of DBSCAN. A random starting data point that has not yet been visited serves as the cluster's initial starting point. Subsequently, the δ neighbourhood of these points is computed. The cluster is created if the neighbourhood of *delta* contains at least *mpts*. Otherwise, the points are referred to as noise. However, these points may be grouped into another cluster at a later stage if the point is found to be within the δ neighbourhood of another point.

Another two commonly employed density-based clustering techniques are Ordering Points To Identify the Clustering Structure (OPTICS) [45] and Generalised DBSCAN (GDBSCAN) [44]. GDBSCAN is often regarded as a generalised iteration of DBSCAN, whereas OPTICS enhances the foundational DBSCAN algorithm. Depending on their spatial and non-spatial attributes, it can cluster both point objects and spatially extended objects. On the other hand, the OPTICS method improves upon the DBSCAN algorithm by using an enhanced ordering of the data points to effectively address variations in local density.

Feature Selection

The technique of optimising dimensionality is widely recognised as a crucial aspect in the field of data mining. This procedure involves the use of feature extraction and feature selection techniques. In the context of pattern analysis, the determination of the relevance or irrelevance of traits is contingent upon their impact on processing performance. Features may also demonstrate redundancy and possess different degrees of discriminative or predictive effectiveness. Feature selection is the process of selecting a subset of features from a larger set in a dataset, with the aim of optimising processing and criteria according to specified objectives. Optimum feature selection is a procedural approach that aims to fulfil a processing target by reducing a designated feature selection criteria. It is crucial to acknowledge that this technique is not inherently exclusive.

Various feature selection approaches have been presented over the years. The process of feature selection may be categorized into two main approaches: supervised and unsupervised. Filter, wrapper, embedding, and ensemble techniques represent distinct categories of supervised feature selection approaches. **Filter** approaches evaluate the discriminatory capability of features just by considering the inherent characteristics of the data. Typically, these algorithms use a relevance scoring mechanism and employ a threshold strategy to identify the most optimal features. **Wrapper** approaches aim to identify the subset of characteristics that exhibit the highest level of discrimination by minimising the prediction error associated with a certain classifier. The efficacy of these approaches is contingent upon the specific classifier used, and they have garnered significant criticism due to their substantial computing requirements. Furthermore, the use of an alternative classifier for prediction does not ensure the provision of an ideal solution. **Embedded** approaches may be classified as a distinct category of methods due to their ability to facilitate interactions with the learning algorithm. However, it is important to note that these techniques exhibit a shorter computing time compared to wrapper methods. **Ensemble** approaches may be classified as a very recent category of methodologies. Various approaches have been suggested as a means to address the instability problems that arise in many feature selection methods when encountering minor perturbations in the training dataset. The methodologies used in these approaches are based upon distinct subsampling processes. The feature selection process is executed on many subsamples, and the resulting features are consolidated into a more robust subset.

Principal Component Analysis (PCA) is well recognized as a prominent unsupervised technique for feature selection [46]. This approach involves the generation of principal components from the given dataset. The process involves assessing the connection between characteristics in order to discover the principal components that have the most significance. The methodology being referred to is a data reduction method that enables the reduction in size of data sets including several interconnected characteristics, hence allowing the representation of the existing data with a reduced number of variables [47]. The use of variable correlation in this context yields characteristics that enhance the algorithm's performance via the mitigation of time consumption and overfitting tendencies of the model. Principal Component Analysis (PCA) does not impose limitations on the use of data, hence making it agnostic to any given dataset.

1.2.2 Soft Computing

Soft computing is a approach [48] that operates in harmony to provide adaptable information processing skills for effectively managing complex and uncertain conditions encountered in real-world scenarios. In contrast to traditional computing methods, soft computing approaches are specifically designed to address the challenges posed by partial truth, uncertainty, and approximation in order to tackle complicated problem domains. The fundamental premise entails the development of computational techniques that provide a satisfactory answer at a minimal expense by pursuing an approximate resolution to an issue that is either accurately or poorly defined. In the field of data mining, it is frequently unfeasible to anticipate the ideal or precise answer. Furthermore, the efficacy of mining algorithms is contingent upon their ability to provide high-quality answers within a reasonable timeframe. Consequently, it is often seen that the criteria for a data mining algorithm align with the fundamental tenets of soft computing, thus rendering the use of soft computing in data mining both inherent and suitable. The clustering algorithms used in soft computing mostly rely on fuzzy set theory and its variations, as well as genetic algorithms, among others. In a similar vein, artificial neural networks are widely used in the domain of soft computing categorization algorithms. In contemporary research, additional mathematical ideas such as possibilistic theory, rough set theory, and credibilistic measure have been used to further the development of diverse soft computing clustering algorithms.

The accompanying paragraph provides a comprehensive description of clustering algorithms based on fuzzy-based soft computing. Accompanying chapters will discuss soft computing clustering strategies based on possibilistic, rough set, and credibilistic measures.

Fuzzy C-Means

The development of fuzzy set theory was motivated by the need to effectively address uncertainties that arise from imprecise, incomplete, overlapping patterns inside diverse problem solving systems. This method has been formulated on the recognition that an item might possess membership in many classes, exhibiting various degrees of class affiliation. Uncertainty may arise due to insufficient or unclear input data, imprecise issue definitions, overlapping borders between classes or regions, and the lack of clarity in defining or extracting characteristics and their relationships. The Fuzzy C-Means (FCM) [49] technique is widely recognized as the most common soft computing approach for partitioning, since it introduces a fuzzy variant of the standard Hard C-Means (HCM) or K-Means algorithm. In order to minimize the objective function as defined in Equation 1.7, FCM employs the fuzzy set theory technique to partition the data set $X = \{x_i \mid 1 \leq i \leq n\}$ into K clusters.

$$J_{\mathcal{FCM}} = \sum_{i=1}^{n} \sum_{l=1}^{K} \mu_{li}^{\eta_1} D(c_l, x_i). \tag{1.7}$$

Here, the function $D(c_l, x_i)$ quantifies the distance between the point x_i and the cluster centre c_l. η_1 represents the weighting coefficient, whereas μ_{li} refers the fuzzy membership value or the degree of belongingness of the ith point to the lth cluster. The initial cluster centres in the Fuzzy C-Means (FCM) algorithm are selected randomly from a set of K potential cluster centres. Subsequently, the algorithm proceeds to iteratively update the membership values (μ_{li}) until convergence, where convergence is defined as the absence of any further change in the objective value. The recalculation of cluster centres occurs at each iteration in order to minimise the objective function, with the specific values of membership taken into consideration. Ultimately, each point is allocated to the cluster that exhibits the greatest degree of membership. The computation of membership value and center is further upon in Chapter 3

Fuzzy *K*-Modes

The Fuzzy K-Modes (FKMd) method, proposed by Huang et al. [50], is a categorical domain extension of the Fuzzy C-Means (FCM) algorithm. It can be considered as fuzzy variant of the K-Modes. The procedures involved in FKMd, including the calculation of the objective function and membership values, exhibit similarities to those of FCM. However, the computation of the cluster mode in FKMd differs from that of FCM. Instead than determining the cluster's mean by traditional computation methods, it instead calculates the mean by taking into account the frequency of the attribute values.

Type-2 Fuzzy Set

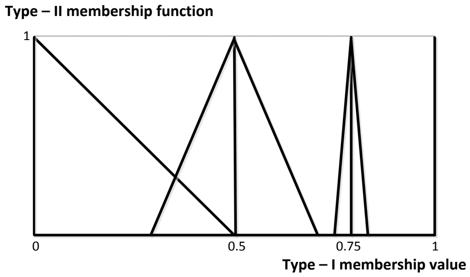


Figure 1.1: Type-2 membership functions for type-1 membership values as explained by Rhee *et al.* in [1]

The Fuzzy C-Means (FCM) algorithm is specifically designed using the principles of type-1 fuzzy set theory. Nevertheless, the theory of type-1 fuzzy sets is subject to certain constraints in terms of the degrees of freedom. Consequently, Professor Zadeh expanded upon the notion by proposing type-2 fuzzy set theory, so including the idea of type-1 fuzzy set. Type-1 fuzzy sets are characterised by membership values that consist of real numbers within the interval [0, 1]. In contrast, type-2 fuzzy set theory introduces a main membership value that has inherent fuzziness. Figure 1.1 illustrates the increase in uncertainty of type-2 membership values as the type-1 membership value approaches 0. In very exceptional circumstances, the value of the type-2 membership for the type-1 membership is 1. Following this, Rhree and colleagues proposed the development of type-2 Fuzzy C-Means (T2FCM) clustering technique [1].

Evolutionary Approach

Evolutionary strategies [51] refer to stochastic and versatile approaches utilized for the resolution of optimization issues. Given that the clustering problem may be formulated as an optimization problem [52], it is plausible to consider the suitability of evolutionary techniques in this context. The goal is to create a globally optimum clustering by combining evolutionary operators with a population of clustering structures. The process of candidate clustering involves the representation of clusters as chromosomes. The evolutionary operators that are frequently employed in evolutionary algorithms are selection, recombination or crossover, and mutation. The evaluation of a fitness function on a chromosome influences the probability of its survival in the subsequent generation.

Genetic Algorithm:

The genetic algorithm (GA) [53] is the prevailing evolutionary approach employed in clustering problems. In essence, a fitness rating is assigned to the structure of each cluster. A greater fitness score signifies an enhanced cluster structure. The inverse of squared error value is a viable fitness function. Cluster structures that exhibit a less squared error will possess a higher fitness value.

One straightforward method for representing structures is through the use of strings with a length equivalent to the number of instances (m) available in the provided set. The ith element of the sequence represents the cluster to which the ith instance is assigned. As a result, it is possible for each entry to possess values ranging from 1 to K. The utilization of this particular representation enables clustering problem to get transformed into a permutation problem.

In the context of genetic algorithm (GAs), the selection operator serves the purpose of propagating solutions from the present generation to the subsequent generation, taking into consideration their fitness levels. The process of selection utilizes a probabilistic approach wherein solutions with greater fitness are more likely to be replicated. The crossover operator receives a pair of chromosomes, referred to as parents, as its input and generates a new pair of chromosomes, known as children or offspring, as its output. This method facilitates the exploration of the solution search space. Mutation is employed as a means to prevent the algorithm from being stuck in a local optimum. Offspring are produced by the reproductive process of parents, wherein they undergo a series of selection, crossover, and mutation rounds until their fitness value converges or a certain number of iterations is attained.

Simulated Annealing:

Another general-purpose stochastic search technique that can be used for clustering is simulated annealing (SA) [54, 55], which is a sequential stochastic search technique designed to avoid local optima. This is accomplished by accepting with some probability a new solution for the next iteration of lower quality (as measured by the criterion function). The probability of acceptance is governed by a critical parameter called the temperature (by analogy with annealing in metals), which is typically specified in terms of a starting (first iteration) and final temperature value. Selim and Al-Sultan (1991) studied the effects of control parameters on the performance of the algorithm. SA is statistically guaranteed to find the global optimal solution. The SA algorithm can be slow in reaching the optimal solution, because optimal results require the temperature to be decreased very slowly from iteration to iteration.

Particle Swarm Optimization:
Particle Swarm Optimization (PSO) [56] is also used as a global optimizer [57] to achieve the optimal solution. PSO works with a population of candidate solution called Swarm where candidate solutions are represented as particles ($\mathcal{P}_j$, where $j = 1, 2, \ldots, \mathcal{N}_{par}$ and $\mathcal{N}_{par}$ is number of particles). Element of each such particle is composed of position and length ($\mathcal{L}$). The movement of a particle is tracked by updating velocity ($\mathcal{V}_j$) and position as defined in Equation 1.8.

$$\mathcal{V}_j^{(t+1)} = \alpha \times \mathcal{V}_j^{(t)} + \beta_1 \times (\mathcal{P}_{l_{best}}^{(t)} - \mathcal{P}_j^{(t)}) + \beta_2 \times (\mathcal{P}_{g_{best}}^{(t)} - \mathcal{P}_j^{(t)}) \tag{1.8}$$

$$\mathcal{P}_j^{(t+1)} = \mathcal{P}_j^{(t)} + \mathcal{V}_j^{(t+1)} \tag{1.9}$$

Where, t is time of different iterations, α is the inertia weight $\in [0.5, 1]$, β_1 is cognitive constant and β_2 is social constant. Moreover, $\mathcal{P}_{l_{best}}$ and $\mathcal{P}_{g_{best}}$ represent local best particle of current iteration and global best particle till current iteration respectively. PSO algorithm terminates after fix number of iterations.

In *InitialPopulation* step, a particle is prepared after random selection of elements from pre-processed dataset. The encoded particle is then used to compute fitness using an appropriate objective function. The fitness value ranges from 0 to 100 where, higher value denotes better result. Based on fitness value, local and global best particles are identified to update the *Velocity*. Thereafter, new position of the particle is computed using updated velocity. Finally, the algorithm gets terminated after a fix number of iterations producing the optimal feature set.

1.3 Related Work

The issue of clustering is one that has been around for quite some time. Due to the exponential growth of data in the digital age, it is more important than ever to logically classify related items in order to avoid data confusion and make it easier for researchers to identify the logic at work. Clustering techniques are currently employed extensively in a variety of domains [58–61], including pattern recognition [62, 63], computer vision [64–67], and data mining [68, 69]. The majority of researchers from these various disciplines have approached the data clustering problem in a variety of ways by proposing a large number of algorithms [57,70–87], but they have mainly focused on numerical data that has a geometrical shape and a clear definition of distance, while categorical data clustering has received very little attention. Several methods [35,39,41,42,50,88–117] for clustering categorical data have also been introduced in recent years due to the rising demand in numerous applications, including market basket analysis and consumer databases. Ralambondrainy [118] developed a method to treat the binary attributes as numeric in the K-means algorithm and to convert multiple category characteristics into binary attributes using 0 and 1 to denote either a category absence or presence. The Expectation-Maximization (EM) algorithm is a partitional clustering technique presented by Dempster et al. in [119]. For each cluster, EM initially randomly assigns a distinct probability to each class or category. In order to maximise the likelihood of the data given the required number of clusters, these probabilities are then successively adjusted. Each observation has a certain likelihood of belonging to each cluster since the EM algorithm calculates the categorization probabilities. Based on the observation with the highest classification probability is actually assigned to a cluster. A locally optimal solution is reached by EM after a significant number of iterations. An association rule hyper-graph-based clustering technique is proposed by Han et al. [120] to group related products in a market database. The relatedness model is a hyper-graph. The strategy is directed at binary transactional data. It presumes that the item sets that comprise clusters are distinct from one another and do not overlap. This presumption might not hold true in reality, though, as transactions in various clusters might share a few common elements. Many clustering algorithms fall into the category of extending existing algorithms with a proximity measure for categorical data, which is one of the most popular approaches to the problem of categorical data clustering. Among them, the algorithms K-Modes (KMd) and Fuzzy K-Modes (FKMd) are frequently utilized. K-Modes approach has a $O\left(iKmn\right)$ computational complexity, where i is the number of iterations. The K-Medoids (KMdd) clustering, also known as partitioning around medoids (PAM), is another variation of the K-Means. In [121], Ng et al. revealed the computationally challenging results of their analysis of PAM. The worst case time complexity is determined to be $O\left(iK\left(n-K\right)^2\right)$, where i is the number of iterations. Both of the aforementioned methods cannot, however, manage overlapping partitions.

The Fuzzy K-Modes (FKMd) [50] algorithm is an extension of the well-known Fuzzy C-Means [49] algorithm in the categorical domain. It is a fuzzy variant of the K-Modes. Computational complexity of FKMd is $O\left(Kn\left(2m+M\right)\right)$, where M is the total number of categorical values of all attributes. The potential for becoming caught in a local optimal solution is the primary drawback of KMd, KMdd, and FKMd. The derived membership of an item does not necessarily correspond to the object's actual belongingness, especially when the object is a noise or an outlier, which is another drawback of the fuzzy based method. Krishnapuram et al. [122, 123] presented a novel idea of possibilistic based clustering approach, where the possibilistic membership is defined based on possibilistic partition and represents the absolute distance between a clustering centre and an object, in order to alleviate the problem of fuzzy based method. As a result, outliers and noise are allocated with relatively low membership degrees, producing a robust anti-noise system. However, a significant coincident clustering issue results from the relaxing of the probabilistic condition for the possibilistic membership. In order to have the stability of the fuzzy method as well as to partially inherit the noise immunity of the possibilistic approach, Pal et al. [87,124] presented a hybrid clustering algorithm based on possibilistic partition and fuzzy partition to tackle the coincident clustering problem. To increase anti-noise robustness, Krishnan et al. [125] added additional weighted values to the objective function of the hybrid method. Szilagyi et al. [79] as another advancement that addresses the coincident clustering issue by incorporating fuzzy memberships and probabilistic memberships into the objective function by multiplication. Recently, Bose et al. [126] and Truong et al. [127] proposed a hybrid idea to further improve the system. By replacing euclidean distance with mahalanobis distance, a smaller number of academics [128–131] have attempted to expand the hybrid approach that combines probabilistic and fuzzy thinking. When dealing with multi-class datasets with high noise injection, the proposed approaches were still plagued by the issues of noise sensitivity and partially coincident clustering. Furthermore, these techniques take a lot of time and struggle to handle non-spherical datasets with noise injection. The most recent clustering algorithm, DS-PFGK (Double-suppressed Possibilistic Fuzzy Gustafson-Kessel clustering) [75], which is a hybrid method of Possibilistic, Fuzzy, and Gustafson-Kessel clustering approach, claims an improvement for clustering results for ellipsoidal or spherical datasets with strong component correlation. reduce centre deviations and identify the majority of noise. However, because DS-PFGK is a partition clustering approach and always assumes balanced sample sizes, it only performs effectively on datasets with balanced sample sizes. Additionally, for datasets with noticeably unbalanced sample sizes, it has a propensity to produce substantial centre deviations of small goals. The phenomenon known as the "curse of dimensionality" causes the clustering performance to decline as all dataset features are used in the clustering.

Ganti et al. [91] describes the development of CACTUS (Clustering Categorical Data Using Summaries), a subspace-based clustering method. Summarization, clustering, and validation are the three stages of the process, which forms a cluster for categorical data by generalising it as numerical data. In order to identify clusters, CACTUS uses both inter- and intra-attribute summaries. However, there hasn't been any information reported on how to apply this method for clustering broad data sets. Gibson et al. [92] describes STIRR (Sieving Through Iterated Relational Reinforcement), an iterative approach based on non-linear dynamical systems. It is possible to translate the methodology employed in [92] to a specific class of non-linear systems. The categorical databases can be clustered if the dynamic system converges. To find sets of closely related attribute values, STIRR needs to go through a non-trivial post-processing step. Furthermore, the STIRR method has difficulty detecting certain classes of clusters. Furthermore, according to Zhang et al. [132], STIRR cannot ensure convergence. The bottom-up clustering technique ROCK (Robust Clustering using Links) uses a novel "link" based distance metric in addition to an agglomerative hierarchical clustering approach. It utilizes the Jaccard coefficient-based [39] similarity function, which is defined by the quantity of neighbours that are shared . If two data objects share more neighbours, they are more similar to one another. The bottom up hierarchical approach clusters a randomly sampled data set before partitioning the full data set based on these clusters since its time complexity is quadratic. However, ROCK is quite sensitive to the threshold value, which has been addressed and improved in [107]. The top-down clustering technique COBWEB [133], on the other hand, builds a classification tree to store cluster information. The classification tree of COBWEB is not height-balanced for skewed input data, which could result in increased time and space costs. An agglomerative hierarchical method called BIRCH [37], works well when clusters are identical in size and have either convex or spherical geometries. However, it is influenced by the data's input order, thus it may not work effectively when clusters have irregular or non-spherical geometries or various sizes. In this regard, the CURE [38] agglomerative hierarchical algorithm can recognise non-spherical structures in big databases of various sizes. In order to process huge databases, it combines partitioning and random sampling. The effectiveness of the random sampler, however, has an impact. Another threshold-based one-pass categorical data clustering technique that is appropriate for dividing data streams is Squeezer [94]. The Tabu Search based Fuzzy K-Modes (TSFKMd) had been developed in [134]. An entropy-based approach for categorical clustering is COOLCAT [93]. From a sample of the full dataset, it first identifies a group of clusters that are suitable. Then, it chooses an appropriate cluster for the remaining objects. This algorithm's primary flaw is that the quality of the clustering is significantly impacted by the sequence in which the processing objects are applied. The iterative clustering technique CLOPE [95] makes advantage of the height-to-width ratio of the cluster histogram. The categorical clusters of [96] are found using a genetic method, which is based on the idea of generalised conditional entropy. By extending the conventional self-organizing map (SOM), He et al. [105] presented the TCSOM technique for clustering binary data. The scalable hierarchical category clustering algorithm LIMBO, which was first proposed in [102] expands on the Information Bottleneck framework. By utilising the idea of a correlated-force ensemble, Chen et al. [99] developed the CORE method. Later in 2005, the hybrid of categorical data clustering (CDC) and cluster ensemble (CE), ccdByEnsemble [106], is presented. The properties of symbolic objects are used to compute categorical values in CDC, which is a specific case of symbolic data clustering [88–90,97,103]. However, agglomerative methods [32] are typically used when grouping symbolic data. The main difference between the agglomerative algorithms is the linkage measure, also known as *single*, *average* and *complete* linkage.

The aforementioned techniques don't perform equally well for overlapping partitions without much care for the underlying dataset's vagueness, uncertainty, and indiscernibility. These are significant concerns for numerous real-world applications where lag commonly occurs at the sharp boundary between clusters. Since the invention of Rough Set Theory [4], developing methods to deal with vagueness, uncertainty, and indiscernibility in the dataset has been highly appealing. It can be viewed as a productive mathematical instrument that can be vital in the extraction of valuable characteristics, the simplification of information processing, the research of expression learning, and the discovery of imprecise and ambiguous information. The basic tenet of the rough set based clustering approach is to allocate objects to the lower and upper approximations of a set and to separate discernible from indiscernible objects. RST has currently been utilized successfully in the disciplines of machine learning, decision analysis, process control, approximative reasoning, pattern recognition, data mining, and other intelligent information processing.

This fact also inspired Parmar et al. to offer a clustering method based on Min-Min-Roughness (MMR) [135]. A new stopping criterion was introduced by the MMR algorithm. MMR calculates the distance between the objects falling under each leaf node rather than selecting the subset of the dataset with the most objects. At the following iteration, the leaf node with the greatest gap between the items is chosen for splitting. When the number of clusters is known a priori, the MMR method can handle uncertainty and provides stable clusters. However, MMR does not eliminate any outliers that might have an impact on the clusters' size. Worst case time complexity of MMR is $O\left(Kmn + Km^2M\right)$, where M is the total number of categorical domain values over all attributes. Recently, a mutual information-based CDC approach known as k-ANMI [110], which is remarkably similar to the K-Means algorithm, was created. This technique, though, is equally susceptible to local optimal solution trapping. Therefore, Average Normalised Mutual Information (G-ANMI) based on Genetic Algorithms is provided in [136]. On a set of predetermined partitions, the ANMI is calculated under the assumption that a good combined partition may share as much information as possible. Note that G-ANMI has already been found to outperform k-ANMI [110], ccdByEnsemble [106], TCSOM [105], and Squeezer algorithms [94].

Contrary to Type-1 fuzzy set (T1FS), which is typically utilized in classic fuzzy set based clustering, Type-2 fuzzy sets (T2FS) offer an effective approach of handling uncertainties, including noisy observations. Several clustering methods have been proposed in recent decade that make use of the idea of Type-2 fuzzy sets, including general type-2 fuzzy clustering [80], interval Type-2 fuzzy clustering [81], kernelized interval Type-2 fuzzy clustering [82], interval type-2 fuzzy c-regression clustering [83], interval type-2 possibilistic c-means clustering [73, 84], interval type-2 relative entropy based fuzzy clustering [85], particle swarm optimization based interval Type-2 fuzzy clustering [57], interval-valued fuzzy set-based collaborative fuzzy clustering [86] etc. These T2FS-based algorithms have also been used effectively in other fields, including image processing [137,138], time series prediction [139], fire detection [140,141], etc.

Another mathematical notion that can be utilized to overcome some of the drawbacks of fuzzy and probabilistic based approaches, notably the coincident problem [142] of probabilistic clustering, is the credibilistic measure, first put forth by Liu et al. [143]. The concept of credibleistic clustering was then advanced by [144,145] and further developed in [146] by applying the alternate cluster estimation [147] principle. The membership function utilized in [148] was also used in the variant of credibilistic clustering from [146]. To the best of my knowledge, no work has been done to cluster categorical data using credibilistic measures. This is due to the fact that clustering numerical data is the main goal of all iterations of credibilistic clustering methods [144–146].

1.4 Scope of the Thesis

Detailed study of the literature pertaining to current research elucidates that a significant portion of the suggested methodologies often exhibit two primary limitations. The majority of these innovations were derived from one or two mathematical notions. The presence of many obstacles, including overlapping partition, ambiguity, vagueness, indiscernibility, and coincident clustering problem, indicates that a single solution is insufficient in efficiently resolving all of these concerns. Additionally, it is important to acknowledge that although there has been some attention given to the clustering of categorical data that has intrinsic complexity, the bulk of current methodologies have mostly been designed for numerical data. Given the complexity of the data, it is clear that each clustering approach has unique benefits and drawbacks. After careful examination of various works and inventions, it is evident that the utilisation of mathematical concepts such as fuzzy sets, type-2 fuzzifier, possibilistic approach, rough set, and credibilistic theories proves to be highly efficient in addressing complex real-life data scenarios. These approaches are particularly effective in situations where the structural topology of the data is absent and where inherent challenges within the data, such as overlapping, outliers, vagueness, uncertainty, and indiscernibility, need to be addressed. A few separate mathematical theories, or a restricted mix of these theories, have been employed in some studies to concentrate on the construction of algorithms. However, there are still difficulties in dealing with intrinsic limits when categorizing categorical data.

It is noted that fuzzy set theory can handle datasets with overlap, outliers, and uncertainty. Additionally, type-2 fuzzy set theory significantly expands on some limitations of type-1 fuzzy set theory. Furthermore, by giving an intuitive notion of an object's degrees of belongingness, possibilistic theory can enhance fuzzy clustering in noisy environments. Rough set theory, on the other hand, offers the ability to solve vagueness, uncertainty and indiscernibility by defining the dataset using the idea of *lower* and *upper approximations*. Credibilistic set theory is an additional mathematical tool for dealing with coincident problems that may arise when using a possibilistic strategy to cluster data.

Therefore, the contribution and objective of this thesis are to propose and develop a set of soft intelligent learning techniques as follows, which are characterized by their integrated and ensemble nature. The purpose of these strategies is to overcome the constraints of current clustering algorithms by improving their capacity to handle intrinsic complications, such as vagueness, ambiguity, and indiscernibility, in diverse datasets, including categorical data. The additional purpose is to reduce the likelihood of being caught in local optimum solutions and coincidental occurrences throughout the clustering process, with the aim of enhancing the overall quality of the clustering outputs.

- An ensemble-based rough fuzzy clustering approach that integrates the potential advantages of both rough set theory and fuzzy set theory in order to address challenges related to overlapping partition, vagueness, uncertainty, and indiscernibility in categorical data.

- A novel hybrid clustering technique that takes advantage of the benefits of type-2 fuzzy sets, rough sets, and probabilistic theory to provide superior clustering results in noisy environments, where the type-2 fuzzy set concept addresses uncertainty and randomness better. The approach is exemplified in the context of image segmentation application.

- A semi-supervised rough fuzzy clustering technique for categorical data based on evolutionary approach. The aim is to address the issue of local optima that might potentially arise in the existing rough fuzzy clustering methods.

- A semi-supervised clustering techniques for categorical data by leveraging credibilistic measure to better deal with the phenomena of clustering coincident problem in case of close clusters.

These novel techniques are discussed in Chapters 2 through Chapter 5.

Chapter 2 introduces the concept of Rough Fuzzy *K*-Modes (RFKMd), which is proposed as a method that harnesses the capabilities of both rough and fuzzy set theories. Moreover, the RFKMd is evaluated using a single dissimilarity metric. However, empirical evidence indicates that no one dissimilarity measure consistently produces optimum clustering results when applied to various categorical data sets. To address the aforementioned consideration and the inherent intricacy associated with the collecting of categorical data, an extension of RFKMd is proposed. This extension, referred to as Ensemble based Rough Fuzzy Clustering (ERFC), incorporates the use of several dissimilarity metrics. The experiment involves the use of six synthetic and four real categorical data sets. The evaluation process includes the assessment of several cluster validity indices and visual graphs. A statistical significance test was conducted to establish the superiority of the proposed techniques in comparison to existing algorithms.

The problem of pixel clustering inside the intensity space is widely acknowledged as a significant barrier in the classification of MR brain image segmentation into discrete homogeneous areas. The problem of automatically recognizing segments or groups of regions with significantly different sizes poses a considerable challenge. To address this problem, Chapter 3 has developed a hybrid clustering technique that combines the ideas of type-2 fuzzy set theory, probabilistic methodology, and rough set theory. The mechanism for implementing the initial rough and type-2 fuzzy based clustering algorithm, referred to as RT2FCM, has been elucidated. The use of probabilistic techniques has subsequently enabled the advancement of RT2FCM into RPT2FCM, with the objective of addressing the constraints inherent in traditional FCM. Furthermore, we have used type-2 fuzzy sets and rough set theories to efficiently address the inherent uncertainties, ambiguities, and indiscernibilities that exist within the data sets. The RPT2FCM method generates both crisp and rough points. Therefore, the initial data points are classified using the Random Forest algorithm, which has been trained with precise data points. The present study employs a collective approach known as RPT2FCM-RF, which is designed to improve the overall clustering results.

Chapter 4 presents an enhanced version of the rough fuzzy *K*-Modes clustering technique to better tackle the problem of indiscernibility and vagueness in categorical datasets. The Rough Fuzzy *K*-Modes clustering method exhibits a tendency to converge to a local optimum solution. To address this issue, two novel clustering methods, namely Simulated Annealing based Rough Fuzzy *K*-Modes and Genetic Algorithm based Rough Fuzzy *K*-Modes, have been proposed by integrating simulated annealing and genetic algorithms. These methods aim to enhance the performance of the original Rough Fuzzy *K*-Modes algorithm. These approaches have the capability to effectively handle clusters that consist of distinct central and peripheral points. The Random Forest classifier has been integrated independently to classify the peripheral points produced by these methods. The objectives include using the central points to instruct the classifier in order to classify the peripheral points accordingly. Furthermore, it has been noted that there are disparities in the outcomes derived from the use of the Rough Fuzzy *K*-Modes, Simulated Annealing-based Rough Fuzzy *K*-Modes, and Genetic Algorithm-based Rough Fuzzy *K*-Modes methodologies. Consequently, the cardinality of the sets including central and peripheral points has also undergone alteration in these techniques. A roughness measure has been calculated to facilitate the selection of the optimal set of central points from the three available sets. Subsequently, researchers have identified pure peripheral and semi-best central spots. Central points that exhibit relatively high quality are then classified based on the best central points. By using these classifications, the peripheral points are individually categorized using the Random Forest algorithm, resulting in improved clustering outcomes. The methodology has been designated as Integrated Rough Fuzzy Clustering using Random Forest.

Probabilistic theory was developed as a means to address the challenge of using fuzzy approaches in the presence of noise. However, in situations characterized by dense clusters, the issue of coincident clustering also poses challenges for the probabilistic method. Furthermore, the complexity of categorical data is heightened by the inherent characteristics that they include within their attributes. Moreover, locating enough and accurately annotated data is a significant challenge in real-world scenarios. Chapter 5 has presented CrKMd, a semi-supervised clustering algorithm that combines credibilistic measures with machine learning specifically designed for categorical data. This chapter of the thesis elucidates the mathematical aspects of the credibilistic measure, as well as explores the potential of certain traits, such as self-duality, to address the limitations of both fuzzy and possibilistic clustering approaches. The development of the Credibilistic *K*-Modes clustering algorithm for categorical data has been undertaken in order to exploit the advantages offered by this particular credibilistic metric. In contrast to assigning diminished credibility to data points that deviate significantly from the norm, this approach has resulted in enhanced credibility for data points that fall within the expected range. The novel approach has a greater capacity for grouping non-outlier data when compared to FCM and PCM. A data point is assigned to a certain cluster based on its level of greatest credibility. However, in some instances, the grouping of data into distinct categories with comparable levels of confidence for many clusters might be puzzling. In order to address this particular characteristic, a semi-supervised clustering technique known as MLCrKMd has been developed. To enhance the results, the integration of CrKMd with other machine learning methodologies is used.

Finally, the thesis is concluded in Chapter 6, which summarizes the findings of each chapter and identifies the potential areas for further research.

2
Rough Fuzzy Clustering

2.1 Introduction

Clustering [32, 149, 150] is an unsupervised technique widely utilized in various domains of data mining, such as pattern recognition [15, 151], customer segmentation [16], trend analysis [152], and medical systems [17]. The primary objective of clustering is to generate K distinct partitions of the input data space using a specified similarity or dissimilarity measure. In many instances, the value of K is not predetermined. The categorization of clustering may be roughly classified into two distinct types: 1) crisp clustering and 2) fuzzy clustering. The primary goal of crisp clustering is to identify distinct and mutually exclusive classes, whereby each pattern is exclusively allocated to a single cluster in crisp clustering. On the other hand, within the context of fuzzy clustering, it is possible for a single pattern to be associated with many classes, each with different levels of membership. The notion of clustering has several applications in scientific and technical domains, such as computer vision, biology, medicine, pattern recognition, and other related subjects.

The majority of clustering algorithms have been recently developed, mostly focusing on numerical data. These algorithms use inherent geometric characteristics to provide a metric for measuring the distance between data points. However, it is important to note that a substantial amount of data in real-world scenarios may be classified as categorical, indicating that there is no distinct hierarchy among the many elements within the attribute domain. In such situations, clustering algorithms such as K-Means [32], Fuzzy C-Means (FCM) [49], etc., cannot effectively manage categorical data. Certain issues associated with clustering categorical data have been examined in previous studies [153,154]. Numerous scholarly [35,39,88–90,93,94,100–102,104,105,107,108,111–115,117,136,155,156] works have so far shown various techniques for clustering categorical data. Among the many publications available, the works of Huang [42] on K-Modes (KMd) and Huang et al. [50] on Fuzzy K-Modes (FKMd) have gained significant popularity. Nevertheless, it is worth noting that fuzzy set theory may not always effectively address the challenges posed by uncertainty and vagueness. Hence, the benefits of rough set theory (RST) have been investigated in several scholarly works [157,158]. In the context of rough clustering, a data point may possess a membership degree of 1, indicating its exclusive affiliation with a particular cluster. Alternatively, the point may exhibit membership in many clusters, particularly in locations that lie on the boundary regions of clusters. Therefore, it is plausible to believe that the boundary region points are situated inside the intersecting regions of many clusters. The first stage of clustering categorical data using rough set theory involves the use of Min-Min-Roughness (MMR) clustering, as proposed by Parmar et al. [135]. However, the MMR method is susceptible to outliers, which have a direct impact on the size of the clusters. Hence, it is customary to use rough and fuzzy sets in conjunction to effectively cluster categorical data sets that are both straightforward and characterized by uncertainty, vagueness, and overlapping.

On the other hand, the evaluation of the similarity or dissimilarity between the two objects is a crucial factor that significantly influences clustering algorithms. Several literature sources have also addressed this topic [91,92,96,106,110]. However, it is worth noting that in the majority of clustering algorithms, a single similarity/dissimilarity measure is used. Nevertheless, it is important to acknowledge that this approach may not adequately capture some latent characteristics inherent in categorical values [109,116]. Hence, inside this chapter, the utilization of both rough and fuzzy set notions is used to introduce the Rough Fuzzy K-Modes (RFKMd) clustering technique. In this context, the management of uncertainty and ambiguity is addressed via the use of the principles of *Lower Approximation* and *Upper Approximation* within the framework of RST. We have then created a framework called Ensemble based Rough Fuzzy Clustering (ERFC) to address the challenges posed by hidden complexity in categorical data sets. This methodology incorporates several dissimilarity metrics. The experiment involves the use of six synthetic and four real categorical data sets. The evaluation process includes the assessment of several cluster validity indices and visual graphs. In order to establish the effectiveness of the proposed approaches, a statistical significance test was conducted as a last step.

2.2 Rough Set Based Fuzzy Clustering for Categorical Data

This section provides an explanation of the proposed rough set-based fuzzy K-modes (RFKMd) approach, which integrates the concepts of rough sets [4] and fuzzy sets [159]. This is followed by a description of the proposed framework for Ensemble based Rough Fuzzy Clustering (ERFC). The first two sub-sections provide a concise explanation of the comprehension of rough set theory (RST) and several dissimilarity metrics, before the subsequent discussion on RFKMd and ERFC.

2.2.1 Brief Description of Rough Set Theory

The rough set theory (RST) was initially presented by Z. Pawlak [4] and subsequently it was studied in different literature [160–165]. RST is essentially used to approximate the uncertainty within dataset. In order to organize the full set of objects, the theory has proposed the ideas of *Lower Approximation* and *Upper Approximation* space. If U and A are finite non-empty sets, with U standing for the universe and A for the set of attributes, then *information system* is the name given to the pair (U, A). Mathematically, a *equivalence* relation designated by $B \subseteq A$ can be denoted as $R(B) \subseteq U \times U$. According to RST, the *equivalence* relation is also known as the *indiscernibility* relation. In this case, $R(B)$ divides the universe U into a number of *equivalence* classes. Same *equivalence* class objects are indistinguishable from one another. A block of partition in the quotient set U/B containing x is referred to as an *equivalence* class of $R(B)$, which is denoted by $B(x)$. The pair, $\langle U, R(B)\rangle$ or $\langle U, B\rangle$ is called a Pawlak approximation space. In the context of RST, if $X \in P(U)$ is any random set and $P(U)$ is the power set of U, then it is difficult to describe X in crisp manner in the approximation space $\langle U, B\rangle$. This particular situation creates an uncertainty within dataset. Therefore, the primary objective of RST is to handle such scenario. For this purpose, a pair of subsets of U approximates $X \subseteq U$. These two subsets are known as *Lower Approximation* and *Upper Approximation*. They are denoted by $\underline{B}(X)$ and $\overline{B}(X)$ respectively. Also, according to RST, the boundary region of X refers to the areas computed by $BN(X) = \overline{B}(X) - \underline{B}(X)$ and it is depicted in Figure 2.1.

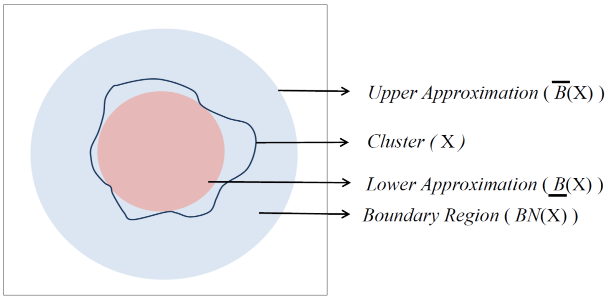

Upper Approximation ($\overline{B}$(X))
Cluster (X)
Lower Approximation ($\underline{B}$(X))
Boundary Region (BN(X))

Figure 2.1: Illustrate the concept of rough set

2.2.2 Different Dissimilarity Measures

The K-Modes technique is widely used for clustering categorical data. However, it is crucial to comprehend the method by which dissimilarity is assessed between two discrete categorical objects. In this regard, if n categorical objects are represented by set $X = \{x_1, x_2, \ldots, x_n\}$, where $\{x_i | i = 1, 2, \ldots, n\}$ has a set of m attributes $A = \{\mathsf{A}_1, \mathsf{A}_2, \ldots, \mathsf{A}_m\}$, then $DOM(\mathsf{A}_j)$ where, $1 \le j \le m$ represents the domain of the jth attribute. $DOM(\mathsf{A}_j)$ contains q_j categories, mathematically, $DOM(\mathsf{A}_j) = \{a_j^1, a_j^2, \ldots, a_j^{q_j}\}$. Therefore, $x_i = [x_{i1}, x_{i2}, \ldots, x_{im}]$, where $x_{ij} \in DOM(\mathsf{A}_j)$ with the condition, $1 \le j \le m$ represents the ith categorical object. The well-known K-Modes normally computes similarity between two data objects using a simple matching measure that is used in [41,166] and is defined in Definition 1. However, it is important to note that this similarity metric does have significant limits, as will be shown in Example 1. In order to address this constraint, the authors of [116] proposed the Definition 2. Furthermore, Ng et al. [109] proposed an alternative dissimilarity metric, as described in Definition 3. The metric quantifies the degree of dissimilarity between a mode and a data object. The dissimilarity measure is used in conjunction with the integration of K-Modes. In the first iteration, a basic matching metric is used. In the study conducted by Cao et al. [116], it was observed that the dissimilarity measure proposed by Ng et al. [109] does not effectively handle certain exceptional scenarios. Therefore, they enhanced it by suggesting a metric as outlined in Definition 4. Given the use of many dissimilarity measures in experimental settings, it is essential to engage in comprehensive discussions accompanied by illustrative examples.

Definition 1 *If $x_i = [x_{i1}, x_{i2}, \ldots, x_{im}]$, and $x_j = [x_{j1}, x_{j2}, \ldots, x_{jm}]$ are two categorical objects that are characterized by m categorical attributes then, $D(x_i, x_j)$, which measures the distance between those two objects, can be defined as follows:*

$$D(x_i, x_j) = \sum_{b=1}^{m} \delta(x_{ib}, x_{jb}) \tag{2.1}$$

where

$$\delta(x_{ib}, x_{jb}) = \begin{cases} 0 & \text{if } x_{ib} = x_{jb} \\ 1 & \text{if } x_{ib} \neq x_{jb} \end{cases} \tag{2.2}$$

Definition 2 *If X is considered to be the set of objects that are categorical in nature and A to be the set of attributes, then the dissimilarity between two objects $x_i, x_j \in X$ with respect to $P \subseteq A$ is defined as,*

$$D(x_i, x_j) = \sum_{a \in P} d_a(x_{ia}, x_{ja}) \tag{2.3}$$

where

$$d_a(x_{ia}, x_{ja}) = 1 - Sim_a(x_{ia}, x_{ja}) \tag{2.4}$$

$$Sim_a(x_{ik}, x_{jk}) = \frac{\delta_a(x_{ia}, x_{ja})}{\sum\limits_{x_s \in X} \delta_a(x_{ia}, x_{sa})} \tag{2.5}$$

and

$$\delta_a(x_{ia}, x_{ja}) = \begin{cases} 0 & \mathit{if}\, x_{ia} \neq x_{ja} \\ 1 & \mathit{if}\, x_{ia} = x_{ja} \end{cases} \tag{2.6}$$

Example 1: Let $\{x_1, x_2, x_3, x_4, x_5\}$ represents five different objects having attributes, $\{A_1, A_2, A_3, A_4\}$ as shown in Table 2.1. The attribute, *Class* in Table 2.1 denotes the cluster group of a particular object. If x_3 and x_5 are considered as the initial two cluster modes, then by Definition 1, the dissimilarity measures of x_2 from x_3 and x_5 are $D(x_2, x_3) = D(x_2, x_5) = 1$. Therefore, it makes it ambiguous which cluster the item x_2 can be assigned to. However, by Definition 2, $D(x_2, x_3) = 1 - \frac{1}{3} + 1 - \frac{1}{3} + 1 + 1 - \frac{1}{5} = \frac{94}{30}$ and $D(x_2, x_5) = 1 + 1 - \frac{1}{3} + 1 - \frac{1}{2} + 1 - \frac{1}{5} = \frac{89}{30}$. Hence, x_2 can be assigned to class 2 without any ambiguity. Therefore, Definition 2 is able to handle the ambiguous situation that is created by Definition 1

Table 2.1: A synthetic data set

Objects	A_1	A_2	A_3	A_4	Class
x_1	s	s	t	t	1
x_2	s	t	s	t	2
x_3	s	t	t	t	1
x_4	t	t	t	t	1
x_5	t	t	s	t	2

Definition 3 *Let X is a set of objects that are categorical type, A is the set of attributes and $P \subseteq A$. If X is partitioned into clusters, where c_l is mode of lth cluster, then the dissimilarity between object $x_i \in X$ and c_l of the cluster is defined as,*

$$D(c_l, x_i) = \sum_{a \in P} d_a(c_{la}, x_{ia}) \tag{2.7}$$

where

$$d_a(c_{la}, x_{ia}) = \begin{cases} 1 & \mathit{if}\, c_{la} \neq x_{ia} \\ 1 - m_a & \mathit{Otherwise} \end{cases} \tag{2.8}$$

and

$$m_a = \frac{|x_i \in C_l : c_{la} = x_{ia}|}{|C_l|} \tag{2.9}$$

Definition 4 *Let X is a set of objects that are categorical type, A is the set of attributes and $P \subseteq A$. If X is partitioned into clusters, where c_l is mode of lth cluster, then the new enhanced dissimilarity between object $x_i \in X$ and the mode c_l is defined as contrasted to Ng's dissimilarity measure [109] as,*

$$D(c_l, x_i) = \sum_{a \in P} d_a(c_{la}, x_{ia}) \tag{2.10}$$

where

$$d_a(c_{la}, x_{ia}) = 1 - Sim_a(c_{la}, x_{1a}) \times m_a \tag{2.11}$$

Example 2: Table 2.2 shows another example of nine objects with two attributes and three initial cluster modes. For determining the appropriate cluster of x_1, it is important to compute the dissimilarity measure of x_1 from the three cluster modes. According to the Definition 3, $D(c_1, x_1) = 1 - \frac{2}{3} + 1 - \frac{1}{3} = 1$, $D(c_2, x_1) = 1 - \frac{1}{3} + 1 - \frac{2}{3} = 1$ and $D(c_3, x_1) = 1 + 1 - \frac{2}{3} = \frac{4}{3}$. Here the computed values generate ambiguity again in order to assign x_1 either to class 1 (c_1) or to class 2 (c_2).

However, Definition 4 can resolve the confusion. According to Definition 4, $D(c_1, x_1) = 1 - \frac{1}{3} \times \frac{2}{3} + 1 - \frac{1}{5} \times \frac{1}{3} = \frac{77}{45}$, $D(c_2, x_1) = 1 - \frac{1}{3} \times \frac{1}{3} + 1 - \frac{1}{5} \times \frac{2}{3} = \frac{79}{45}$ and $D(c_3, x_1) = 1 + 1 - \frac{1}{5} \times \frac{2}{3} = \frac{84}{45}$. This helps to decide *Class* 1 for assigning x_1.

Table 2.2: Another synthetic data set

Objects	A_1	A_2
x_1	s	t
x_2	s	u
x_3	t	v
Cluster 1 (c_1)	s	t
x_4	s	v
x_5	t	t
x_6	u	t
Cluster 2 (c_2)	s	t
x_7	v	v
x_8	v	t
x_9	w	t
Cluster 3 (c_3)	v	t

2.2.3 Rough Set Based Fuzzy K-Modes

The proposed RFKMd starts by selecting random K number of cluster modes. From the entire set of data, $\{x_i| 1 \leq i \leq n\}$, each mode is represented by c_l, $1 \leq l \leq K$, where n is the total number of category objects. Equation 2.12 defines the calculation of the fuzzy membership value.:

$$\mu_{li} = \begin{cases} 1, & if x_i = c_l \\ 0, & if x_i = c_h, l \neq h \\ \frac{1}{\sum_{h=1}^{K}[\frac{D(c_l,x_i)}{D(c_h,x_i)}]^{\frac{1}{\eta-1}}} & if x_i \neq c_l, x_i \neq c_h, 1 \leq h \,\&\, l \leq K \end{cases} \tag{2.12}$$

In the Equation 2.12, η and $D(c_l, x_i)$ stand for the fuzzy exponent and the degree of dissimilarity between the cluster mode c_l and object x_i, respectively. Subsequently, the Equation 2.13 is used to calculate the objective values for the RFKMd. The cluster modes are adjusted in iteration until there is no significant change in values of objective function.

Definition 5 *The objective function is formally defined as*

$$J_{\mathsf{RF}} = \begin{cases} \mathtt{f}_{\mathtt{LW}} \times \mathsf{P} + \mathtt{f}_{\mathtt{BN}} \times \mathsf{Q}, & if \underline{B}(C_l) \neq \emptyset, BN(C_l) \neq \emptyset \\ \mathsf{P}, & if \underline{B}(C_l) \neq \emptyset, BN(C_l) = \emptyset \\ \mathsf{Q}, & if \underline{B}(C_l) = \emptyset, BN(C_l) \neq \emptyset \end{cases} \tag{2.13}$$

$$\mathsf{P} = \sum_{l=1}^{K} \sum_{x_i \in \underline{B}(C_l)} (\mu_{li})^{\eta} D(c_l, x_i)$$

$$\mathsf{Q} = \sum_{l\ =\ 1}^{K} \sum_{x_i \in BN(C_l)} (\mu_{li})^{\eta} D(c_l, x_i),\ \ 1\ \leq\ i\ \leq n$$

Here:
X represents the non empty set of objects;
$C_l \subseteq X$ for $1 \leq l \leq K$ is the set of objects of l^{th} cluster;
c_l, the mode of cluster C_l;
Lower approximation is denoted by $\underline{B}(C_l)$ for the cluster C_l;
Upper approximation is denoted by $\overline{B}(C_l)$ for the cluster C_l;
Boundary region is denoted by $BN(C_l) = [\overline{B}(C_l) - \underline{B}(C_l)]$ for the cluster C_l;
$D(c_l, x_i)$ is the distance between mode c_l and data object x_i;
$\mathtt{f}_{\mathtt{LW}}$ denotes the relative importance of the lower approximation region;
$\mathtt{f}_{\mathtt{BN}}$ denotes the relative importance of the boundary region;
$\mathtt{f}_{\mathtt{LW}} + \mathtt{f}_{\mathtt{BN}} = 1$ and $0 < \mathtt{f}_{\mathtt{BN}} < \mathtt{f}_{\mathtt{LW}} < 1$;

In Definition 5, η and μ_{li} stand for the fuzzy exponent and the membership degree of the ith categorical object to the lth cluster. If the cluster mode is defined as $c_l = [c_{l1}, c_{l2}, \ldots, c_{lm}]$ where $c_{lj} = a_j^r \in DOM(\mathsf{A}_j)$ then μ_{li} is calculated as follows.

$$\mu_{li} = \frac{1}{\sum_{h=1}^{K}(\frac{D(c_l,x_i)}{D(c_h,x_i)})^{\frac{1}{\eta-1}}}, \text{ for } 1 \leq l \leq K; \ 1 \leq i \leq n, \tag{2.14}$$

with the following condition

$$\sum_{i,\ x_{ij}=a_j^r} \mu_{li}^{\eta} \ \geq \ \sum_{i,x_{ij}=a_j^t} \mu_{li}^{\eta}, \ 1 \ \leq \ t \ \leq \ q_j, \ r \ \neq \ t. \tag{2.15}$$

μ_{li} for all $l = 1, \ldots, K$, $l \neq h$ is made zero if $D(c_h, x_i)$ attains zero for some value of h, while μ_{hi} is set to 1. If $D(c_h, x_i)$ for some h is equal to 0, then μ_{li} is set to zero for all $l = 1, \ldots, K$, $l \neq h$, whereas μ_{hi} is set to 1. The objects within the *Lower Approximation* of a cluster have considerably greater influence on that cluster and its mode, according to rough set theory. However, these objects neither influence any object outside the cluster area nor be influenced by any object outside the corresponding cluster area. Therefore, $\mathtt{f}_{\mathtt{LW}}$ should have the more weight than $\mathtt{f}_{\mathtt{BN}}$. If x_i is a member of the *Lower Approximation*, $\mathtt{f}_{\mathtt{LW}}$ is set higher for the object, x_i than $\mathtt{f}_{\mathtt{BN}}$ for objects in the *Boundary Region*. The value of fuzzy membership for x_i is also set to 1. Otherwise, x_i is a member of the *Upper Approximation* of several clusters. Given that, the values of $\mathtt{f}_{\mathtt{LW}}$ and $\mathtt{f}_{\mathtt{BN}}$ are chosen such a way that $0 < \mathtt{f}_{\mathtt{BN}} < \mathtt{f}_{\mathtt{LW}} < 1$. Note, the objective function depends on the relative importance parameter $\mathtt{f}_{\mathtt{LW}}$, $\mathtt{f}_{\mathtt{BN}}$ and the fuzzy exponent η. Due to the fact of certainty about clustering of objects inside *Lower Approximation* area, the fuzzy membership value, μ_{li}, is set to 1 for each of those objects. Therefore, P becomes.

$$\mathsf{P} = \sum_{l=1}^{K} \sum_{x_i \in \underline{B}(C_l)} D(c_l, x_i), \ 1 \leq i \leq n$$

Definition 6 formulates the process of mode updating. In each iteration, the cluster mode is computed by adjusting each attribute value of mode using Equation 2.16. For this purpose, a weight is assigned to all attribute values separately. The weight is computed as $(\mathtt{f}_{\mathtt{LW}} \times \mathsf{G}_{LW} + \mathtt{f}_{\mathtt{BN}} \times \mathsf{G}_{BN})$, where $\mathtt{f}_{\mathtt{LW}}$ and $\mathtt{f}_{\mathtt{BN}}$ are the relative importance same as used in Equation 2.13. G_{LW} contributes to *Lower Approximation*, which indicates how many times an attribute value has appeared over all objects in *Lower Approximation*. On the other hand, G_{BN} contributes to the *Boundary Region* that represents the total frequency of occurrence of all objects in *Boundary Region*. For example, each occurrence of any attribute value, x_{ij}, of attribute, A_j, for object, x_i, in *Lower Approximation* of lth cluster, $\underline{B}(C_l)$, is counted as 1. Similarly, each occurrence of any attribute value, x_{ij}, of attribute, A_j, in *Boundary Region* of lth cluster, $BN(C_l)$, is equal to μ_{li}. Thus, the attribute of A_j, which maximizes the value of ($\mathtt{f}_{\mathtt{LW}} \times \mathsf{G}_{LW} + \mathtt{f}_{\mathtt{BN}} \times \mathsf{G}_{BN}$), determines the value of jth attribute of the cluster mode. This makes sure that there is a minimum dissimilarity between the object, x_i, and the cluster mode, c_l. The process of mode adjustment is done iteratively until the algorithm converges.

Definition 6 *In the context where c_l represents the mode of cluster C_l, $\underline{B}(C_l)$ denotes the crisp lower approximation, and $BN(C_l)$ represents the fuzzy boundary region, the mode is modified in a manner that maximizes the optimization of Equation 2.13 when c_{lj} is equal to a^r_j, which belongs to $DOM(\mathsf{A}_j)$, where*

$$r = \arg\max_{1 \le t \le q_j} \begin{cases} \mathtt{f}_{\mathtt{LW}} \times \mathsf{G}_{LW} + \mathtt{f}_{\mathtt{BN}} \times \mathsf{G}_{BN}, \\ \qquad\qquad if \underline{B}(C_l) \neq \emptyset, BN(C_l) \neq \emptyset \\ \mathsf{G}_{LW}, \qquad if \underline{B}(C_l) \neq \emptyset, BN(C_l) = \emptyset \\ \mathsf{G}_{BN}, \qquad if \underline{B}(C_l) = \emptyset, BN(C_l) \neq \emptyset \end{cases} \tag{2.16}$$

where,

$$\mathsf{G}_{LW} = \sum_{\substack{1 \le i \le n, \\ x_i \in \underline{B}(C_l), \\ x_{ij} = a^t_j}} (\mu_{li})^\eta,$$

$$\mathsf{G}_{BN} = \sum_{\substack{1 \le i \le n, \\ x_i \in BN(C_l), \\ x_{ij} = a^t_j}} (\mu_{li})^\eta$$

As discussed previously, the objects inside the *Lower Approximation* area of any cluster will certainly belong to it and thus the fuzzy membership value for those objects is set 1, i.e, $\mu_{li} = 1$. In that case, G_{LW} is modified as follows.

$$\mathsf{G}_{LW} = \sum_{\substack{1 \le i \le n, \\ x_i \in \underline{B}(C_l), \\ x_{ij} = a^t_j}} = |\left\{x_i \in \underline{B}(C_l) : x_{ij} = a^t_j, 1 \le i \le n\right\}|$$

where $C_l \subseteq X$ and $|\{.\}|$ stands for the cardinality of the designated set.

Algorithm 2 : RFKMd

```
Input:
    $X$ is the data set
    $\eta$ is the fuzzy exponent
    $\epsilon$, a small real threshold value between [0,1]
    $K$, the number of cluster
    $\mathtt{f_{LW}}$, relative weight for lower approximation of rough clustering, $0 < \mathtt{f_{LW}} < 1$
Output: $[\mu_{li}]$ where, $1 \le l \le K$ and $1 \le i \le n$

Choose $K$ random objects as $K$ cluster modes from the entire data set
repeat
    Compute $\mu_{li}$ for entire set of $n$ objects using Equation 2.14
    Calculate the difference between top two memberships for every single data object from $n$
        // Let $\mu_{li}$ and $\mu_{hi}$, highest and second top fuzzy membership values of $x_i$ among K clusters, with the conditions,
                                                                    $1 \le l, h \le K$ and $h \ne l$
    Calculate the threshold $\Delta$ that is median of $(\mu_{li} - \mu_{hi}), \forall i = 1, 2, \ldots, n$
    if $(\mu_{li} - \mu_{hi}) > \Delta$ then
        $\mu_{li} \leftarrow 1$ and $\mu_{hi} \leftarrow 0; \forall h = 1, 2, \ldots, K$ where $h \ne l$
                    // $x_i$ is exactly classified to $\underline{B}(C_l)$, also to $\overline{B}(C_l)$ as per rough set theory
    else
        Keep $\mu_{hi}$ unchanged $\forall h = 1, 2, \ldots, K$
  // $x_i$ may belong to Upper Approximation of more than 1 cluster. That implies, $x_i$ belongs to both $\overline{B}(C_l)$ as well as $\overline{B}(C_h)$
    end if
    Compute $J_{\mathsf{RF}}$ using Equation 2.13
    Compute new mode using Equation 2.16
until $|Current\ J_{\mathsf{RF}} - Previous\ J_{\mathsf{RF}}| \le \epsilon$
  // Repeat loop for next iteration after 1st iteration without checking the inequality. Check the exit inequality from 2nd
                                                                                                            iteration
return $[\mu_{li}]$ where, $1 \le l \le K$ and $1 \le i \le n$
```

2.2.4 Ensemble Rough Fuzzy Clustering

No single dissimilarity measure has been found so far in any literature to address different types of inherent complexities in categorical datasets. As the RFKMd that has been explained in previous section uses only single dissimilarity measure, it may fail to handle the inherent complex properties of the data sets. However, final clustering solution can be improved using different dissimilarity measures. This fact motivates to experiment the result of RFKMd using different dissimilarity measures. However, it has been observed that while using different dissimilarity measures, one at a time, in RFKMd, the same data set can produce different roughness measure (see Table 2.3 where average roughness measure is reported while using RFKMd with different measures). This also confirms that any particular dissimilarity measure may not consistently perform well over different types of categorical data sets. Keeping all these facts in mind, an ensemble framework has been designed in such a way that the weakness of any particular measure can be alleviated by some other measure and the resultant weakness is minimized collectively.

In Algorithm 3, few notations are used, which are defined as follows.

- Let N is the number of dissimilarity measures used in the ensemble method.

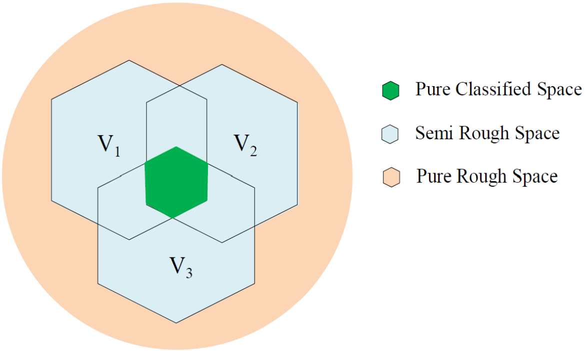


Figure 2.2: Illustrate the concept of creating the consensus training set.

We use C_l^j to denote the lth cluster, V_j is a collection of objects that make up all clusters' *Lower Approximation*, L_j is the corresponding cluster labels and R_j is the set of rough objects obtained using jth dissimilarity measure. Mathematically, V_j and R_j can be represented as follows:

$$V_j = \bigcup_{1 \leq l \leq K} \underline{B}(C_l^j) \tag{2.17}$$

$$R_j = X - V_j \tag{2.18}$$

$$\rho_j = \rho_B(C_l^j) \tag{2.19}$$

where X denotes the entire set of objects, $\rho_B(C_l^j)$ is defined in Equation 2.21. In this article, $j = 1, 2, 3$ and 4 corresponds to the dissimilarity measures as defined in Definitions 1 to 4, respectively.

- τ is the permissible threshold of initial training set over all the classified objects using different dissimilarity measures in percentage. Experimentally, it has been observed that $\tau = 50\%$ works well for all the data sets used in the article.

For experiment purpose, three sets of objects are defined such as *pure classified* set, *semi rough* set and *pure rough* set. The data points all of which are unanimously classified to a certain cluster by all of the dissimilarity measures are referred to as *pure classified* objects, while the points that are not categorized into any cluster are referred to as *pure rough* objects. Similarly, *semi rough* objects refer to the data points that are classified by one or more than one dissimilarity measure into different clusters. In this regards, Figure 2.2 depicts the pure classified, pure rough and semi rough objects for an ideal scenario.

With such definition of *pure classified* set, *semi rough* set and *pure rough* set, the Ensemble Rough Fuzzy *K*-Modes Clustering (ERFC) algorithm has been designed. It works in two folds. At first, RFKMd is executed with different dissimilarity measures. As a result of each evaluation, it produces some classified and unclassified objects. Subsequently, a consensus set of pure classified objects is computed from different classified results by taking the intersection. Other classified and unclassified objects are treated as semi rough and pure rough objects. This reduces the possibility of mis-classification. In the next fold, pure classified objects are used as training set of machine learning method (such as Random Forest, *K*-NN, SVM etc) to classify the semi rough objects. Thereafter, the combined sets of pure and semi classified objects are used to training the machine in order to classify pure rough objects, i.e., called an incremental way to classify the semi and rough points.

While the consensus training set is smaller than τ, a list of rank for classifications over different dissimilarity measures is prepared based on the sorted roughness measure. However, the classified objects that have worst roughness measure will not participate in the intersection process. According to RST, roughness determines how accurately any rough set based algorithm is able to classify data objects and the value of roughness lies between [0,1]. If the size of the training set is not improved, then the set having second worst roughness measure also does not participate in intersection process. This process continues until a training set with considerable size is found or the process reaches to only one set having best roughness measure in the rank. The detail steps of ERFC are outlined in Algorithm 3.

Algorithm 3 : ERFC

```
Input:
    $X$, the data set
    $\eta$, the fuzzy exponent
    $\epsilon$, a small real threshold value between [0,1]
    $K$, the number of cluster
    $\mathtt{f_{LW}}$, relative weight for lower approximation of rough clustering, $0 < \mathtt{f_{LW}} < 1$
    $\tau$, permissible threshold to create initial training set in percentage
    $\mathsf{N}$, number of dissimilarity measures
Output: $F$, the final class label of $X$

Initialize $\mathsf{N}$ with Number of dissimilarity measures
                                                                        // $\mathsf{N} = 4$
Compute $R_1, R_2, \ldots, R_\mathsf{N}$; $V_1, V_2, \ldots, V_\mathsf{N}$ and $\rho_1, \rho_2, \ldots, \rho_\mathsf{N}$ using Algorithm 2, Equ. 2.17, 2.18 and 2.19 with
    the values of $X, \eta, \epsilon, K$ and $\mathtt{f_{LW}}$
Sort $\rho_1, \rho_2, \ldots, \rho_\mathsf{N}$ in descending order and store it in list, $L_\rho$
$[V\mathsf{L}]_{sup} = \{(V_j, \mathsf{L}_j) \,:\, 1 \le j \le \mathsf{N}\}$
$[\mathsf{L}]_{sup} = \{\mathsf{L}_j \,:\, 1 \le j \le \mathsf{N}\}$
$V \leftarrow \bigcup_{\forall V_j \in V_{sup}} V_j$, where $1 \le j \le \mathsf{N}$          // $V$, the union of all crisp sets
$V^* \leftarrow \bigcap_{\forall V_j \in V_{sup}} V_j$, where $1 \le j \le \mathsf{N}$
                                  // $V^*$, consensus training set for machine learning method
Initialize $i = 1$
while $\left(\frac{|V^*|}{|V|}\right) \le \frac{\tau}{100}$ and $i < \mathsf{N}$ do                 // $|V^*|$, cardinality of $V^*$
    Find $V_z \in V_{sup}$ which corresponds to $L_\rho[i]$
    $[V\mathsf{L}]_{sup} \leftarrow [V\mathsf{L}]_{sup} - \{(V_z, \mathsf{L}_z)\}$
    $[\mathsf{L}]_{sup} \leftarrow [\mathsf{L}]_{sup} - \{\mathsf{L}_z\}$
    $V^* \leftarrow \bigcap_{\forall V_j \in V_{sup}} V_j$, where $1 \le j \le \mathsf{N}$
    $i \leftarrow i + 1$
end while
Select $\mathsf{L}^*$, where $\{(V^*, \mathsf{L}^*)\} \in [V\mathsf{L}]_{sup}$ and $\mathsf{L}^*$ is any member of $[\mathsf{L}]_{sup}$
$R' \leftarrow V - V^*$                               // $R'$, the test set for machine learning method
Classify $R'$ to get classified set, $(V', \mathsf{L}')$ using machine learning method trained by $(V^*, \mathsf{L}^*)$
$V^{**} \leftarrow V^* \cup V'$                  // $V^{**}$, the combined training set for machine learning method
Relabel $(V^{**}, \mathsf{L}^{**})$ using $(V^*, \mathsf{L}^*)$
$R'' \leftarrow \bigcup_{1 \le j \le \mathsf{N}} R_j$                          // $R''$, the test set for machine learning method
$R'' \leftarrow R'' - R'$
Classify $R''$ to get classified set, $(V'', \mathsf{L}'')$ using machine learning method trained by $(V^{**}, \mathsf{L}^{**})$
Get final uniform class label of $V^{**} \cup V''$ using $(V^{**}, \mathsf{L}^{**})$
return $F$
```

2.3 Experimental Results

The clustering methods are tested on six synthetic data sets[1] (*Cat-100-8-3*, *Cat-250-15-5*, *Cat-300-8-3*, *Cat-300-15-5*, *Cat-500-20-10* and *Cat-1000-7-7*) as well as four actual data sets[2] (*Congressional Votes*, *Zoo*, *Soybean* and *Heart*) using four performance metrics.

[1] http://www.datgen.com

[2] http://www.ics.uci.edu/~ mlearn/MLRepository.html

2.3.1 Performance Metric

Four performance metrics are used to assess the performance of clustering algorithms. Those are Roughness Measure (ρ) [167], Minkowski Score (MS) [168], Percentage of Correct Pair (%CP) and Adjusted Rand Index (ARI) [169]. The metrics are discussed briefly below.

Roughness Measure

Two quantitative metrics are defined in [167] using rough set theory to compute a set, C_l^j, in approximation space using jth dissimilarity measure. Those are called accuracy and roughness. Accuracy denoted by $\alpha_B(C_l^j)$ is defined as

$$\alpha_B(C_l^j) = \frac{1}{K}\sum_{l=1}^{K} \frac{\mathtt{f}_{\mathtt{LW}} S_l^j}{\mathtt{f}_{\mathtt{LW}} S_l^j + \mathtt{f}_{\mathtt{BN}} T_l^j} \tag{2.20}$$

where

$$S_l^j = |\underline{B}(C_l^j)|$$

and

$$T_l^j = \sum_{x_i \in BN(C_l^j)} (\mu_{li}^j)^\eta$$

whereas roughness denoted by $\rho_B(C_l^j)$, is defined as

$$\rho_B(C_l^j) = 1 - \alpha_B(C_l^j) = 1 - \frac{1}{K}\sum_{l=1}^{K} \frac{\mathtt{f}_{\mathtt{LW}} S_l^j}{\mathtt{f}_{\mathtt{LW}} S_l^j + \mathtt{f}_{\mathtt{BN}} T_l^j} \tag{2.21}$$

The range of $\rho_B(C_l^j)$ $\in$ [0,1]. 0 signifies the perfect clustering.

Minkowski Score

The *Minkowski Score* (MS) [168] is used to measure how well the clustering algorithms perform. This evaluates the effectiveness of the solution in light of the true clustering. Let T and S be the "true" and computed solution respectively. Let n_{11} denote the cardinality of the set of pairs of items that are members of the same group in both T and S. Likewise, the symbol n_{01} represents the quantity of pairings that exist exclusively within the same group in S, whereas n_{10} represents the quantity of pairings that exist exclusively within the same group in T. In that

Data Set	RFKMd	RFKMdDisp	RFKMdNgDisp	RFKMdCaodisp
Cat-100-8-3	0.043367838	0.039990218	0.047144669	0.037559461
Cat-250-15-5	0.031636794	0.029186179	0.037379437	0.031759298
Cat-300-8-3	0.046815225	0.038302494	0.046454106	0.046620606
Cat-300-15-5	0.032248153	0.026844306	0.029941433	0.028021867
Cat-500-20-10	0.067713549	0.042173355	0.031154265	0.030247322
Cat-1000-7-7	0.030513626	0.022807120	0.099849582	0.062068793
Soybean	0.047478151	0.035552972	0.048104290	0.044113851
Zoo	0.034999696	0.022238417	0.049220946	0.168561960
Heart	0.050002400	0.050002453	0.050889698	0.050061905
Votes	0.050608920	0.050335573	0.050399256	0.050568985

Table 2.3: Average roughness measure while using RFKMd and different dissimilarity measures

case, Minkowski Score is then defined as:

$$MS = \sqrt{\frac{n_{01} + n_{10}}{n_{11} + n_{10}}} \tag{2.22}$$

Minimum value of MS is 0, while lower value denotes better clustering.

Percentage of Correct Pair

Percentage of Correct Pair (%CP) can be mathematically defined as,

$$CP = \frac{the\ number\ of\ pairings\ that\ are\ successfully\ grouped\ into\ same\ cluster}{pairs\ that\ are\ actually\ present\ in\ the\ same\ cluster} \tag{2.23}$$

Better clustering is indicated by a higher CP value. The outcome is presented as a percentage. Hence, 100% denotes flawless clustering.

Adjusted Rand Index

Adjusted Rand Index (ARI) [169] represents a relation between true cluster and the evolved cluster of the data set. Given that, T is the true clustering, whereas C denotes the evolved clustering through clustering algorithm, ARI can be formulated as

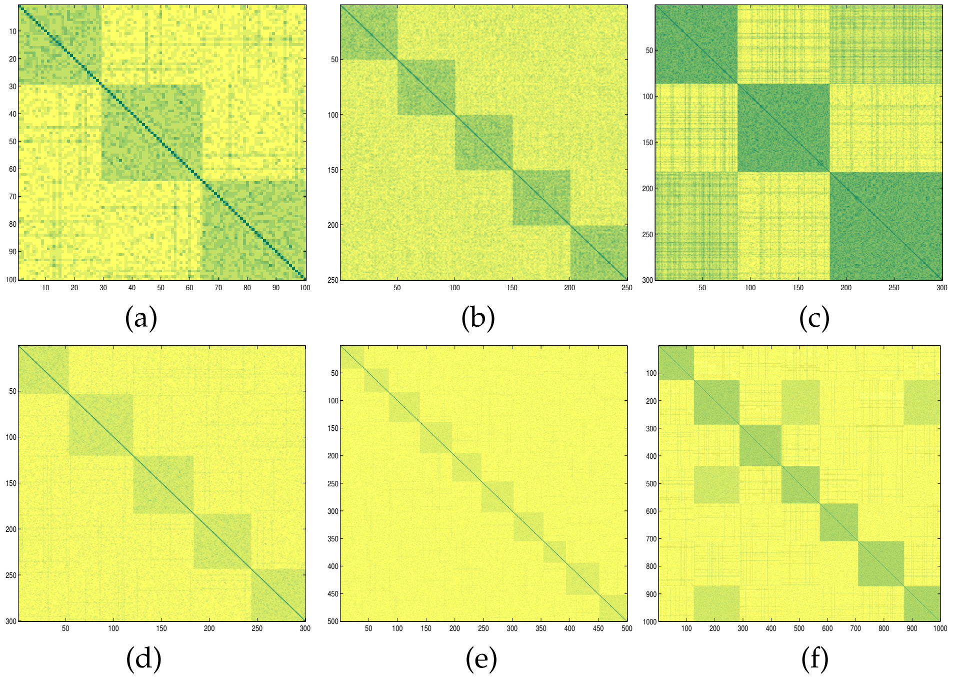
$$ARI\,(T, C) = \frac{2\,(ad - bc)}{(a + b)\,(b + d) + (a + c)\,(c + d)} \tag{2.24}$$

where a, b, c, and d denote the quantities of pairings that belong to the same group in both T and C, the pairings that belong to the same group in T but different groups in C, the pairings that belong to different groups in T but the same group in C, and the pairings that belong to different groups in both T and C, respectively.

The *ARI* value ranges from zero to one. A higher score denotes that the clustering that was produced is closer to the real one.

2.3.2 Visualization

The well-known VAT (*visual assessment of tendency*) [170] representation is utilized to visualize the data sets. The points are initially rearranged in accordance with the class labels provided by the clustering solution in order to visualise it. The distance matrix is then calculated using this rearranged data matrix. The boxes located on the main diagonal of the distance matrix's graphical display reflect the clustering structure.



(a) (b) (c)

(d) (e) (f)

Figure 2.3: VAT representation of Synthetic data sets (a) Cat-100-8-3 (b) Cat-250-15-5 (c) Cat-300-8-3 (d) Cat-300-15-5 (e) Cat-500-20-10 (f) Cat-1000-7-7

2.3.3 Synthetic Data Sets

Cat-100-8-3

With 8 attributes and 100 points, this artificial data set has a one-layer clustering structure. 3 clusters make up this. Each cluster has a unique continuous set of five attributes with five categorical values chosen at random from the range {0, 1, 2, 3, 4, 5}, with the other attributes set to 0. VAT representation of true clusters of the data set is depicted in Figure 2.3(a).

Cat-250-15-5

The data generator, http://www.datgen.com has been used to create this artificial data set. The amount of attributes, attribute domains, and tuples are just a few of the choices this generator offers. 250 points total and 15 attributes make up the data set. Five clusters are formed from the points. True clusters of this data set have been reported in Figure 2.3(b) using VAT plot.

Cat-300-8-3

There are 300 points in this artificial data set, and each point has 8 properties. Three clusters of data sets have been identified. Figure 2.3(c) displays the VAT plot of the real clusters in this data set.

Cat-300-15-5

This 300-point, 15-attribute artificial data set was created. Figure 2.3(d) displays a VAT depiction of a genuine cluster. There are 5 clusters in the data set. Eight random qualities from each cluster's points are set to zero at random, while the other attributes' values fall among {0, 1, 2, 3, 4, 5}.

Cat-500-20-10

The data set consists of 500 points, each possessing 20 distinct features. These points are then categorized into 10 distinct clusters. The VAT plot of the genuine cluster for this data set is shown in Figure 2.3(e).

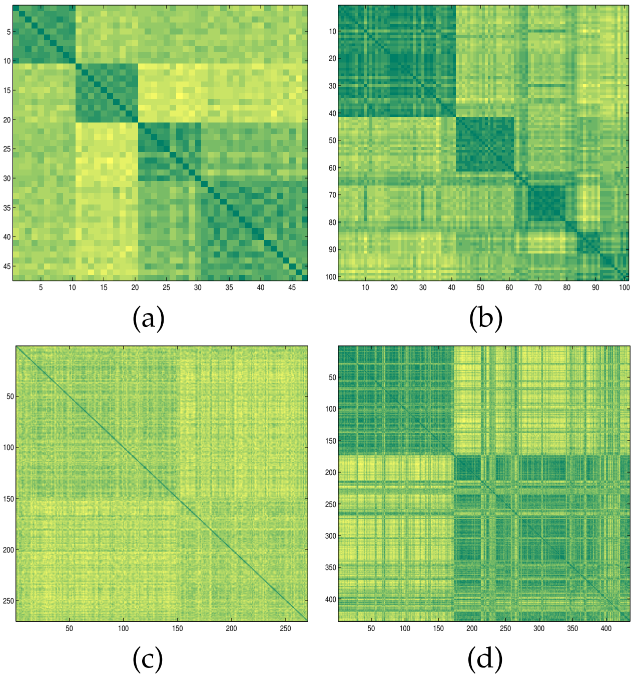
Cat-1000-7-7

There are 1000 points in this artificial data set, and each has 7 properties. Seven clusters have been created from the data sets. The VAT plot of true cluster is shown in Figure 2.3(f).

2.3.4 Actual Data Sets

Soybean

There are 47 data items on diseases affecting soybeans in the Soybean data set. Each data point is categorized into one of four diseases and includes 35 categorical features. Hence, there are 4 clusters in the data set. VAT plot of true clusters of this data set is depicted in Figure 2.4(a).



(a) (b) (c) (d)

Figure 2.4: VAT representation of Real life data sets (a) Soybean (b) Zoo (c) Heart (d) Votes

Zoo

The 101 examples of animals in zoos with 17 attributes make up the Zoo data. The animal's name makes up the first attribute. This characteristic is not used. There are 15 boolean qualities that correspond to whether an animal is airborne, aquatic, predator, toothed, breathes, venomous, domestic, and catsize, as well as if it has hair, feathers, eggs, milk, a backbone, fins, or a tail. The number of legs in the set correlates to the character attribute {0, 2, 4, 5, 6, 8}. There are 7 different animal classes included in the data collection. It's true clusters are shown in Figure 2.4(b).

Heart

This actual data set is having 270 observation of heart disease. The vat representation is demonstrated in Figure 2.4(c). Each observation is being characterized with 13 attributes having information about age, sex, type of chest pain, blood pressures etc. Each entry can be categorized into two groups - either disease present or absent.

Data Sets	Method	MS	%CP	ARI
Cat-100-8-3	KMd	0.88202	76.18283	0.73107
	FKMd	0.79713	79.91267	0.78020
	TSFKMd	0.73736	83.11590	0.79410
	MMR	0.55900	88.29480	0.85610
	G-ANMI	0.50471	89.44180	0.86610
	ccdByEnsemble	0.43344	92.44180	0.90580
	ERFC	**0.19733**	**98.48224**	**0.97860**
Cat-250-15-5	KMd	0.77524	080.55590	0.78620
	FKMd	0.62713	087.21280	0.84370
	TSFKMd	0.55319	088.60240	0.86040
	MMR	0.52321	89.09280	0.86590
	G-ANMI	0.49988	90.29160	0.86670
	ccdByEnsemble	0.40051	93.56740	0.91155
	ERFC	**0.01203**	**99.01540**	**0.98125**
Cat-300-8-3	KMd	0.77440	80.63140	0.79000
	FKMd	0.37300	94.27590	0.93710
	TSFKMd	0.34283	94.90220	0.94380
	MMR	0.31699	96.15320	0.96370
	G-ANMI	0.20322	98.10197	0.97780
	ccdByEnsemble	0.15203	98.72727	0.97975
	ERFC	**0.00890**	**99.55340**	**0.98563**
Cat-300-15-5	KMd	0.84465	77.51644	0.75260
	FKMd	0.74558	82.68290	0.79260
	TSFKMd	0.69895	84.76120	0.80990
	MMR	0.64432	86.63930	0.83160
	G-ANMI	0.57669	87.91220	0.84760
	ccdByEnsemble	0.54571	88.83760	0.86260
	ERFC	**0.51885**	**89.21280**	**0.86600**
Cat-500-20-10	KMd	0.94438	75.28247	0.71026
	FKMd	0.81305	78.92458	0.76100
	TSFKMd	0.75597	81.42737	0.79130
	MMR	0.71532	84.20744	0.79830
	G-ANMI	0.67155	85.82147	0.82670
	ccdByEnsemble	0.65886	86.11850	0.83050
	ERFC	**0.52995**	**89.00240**	**0.86440**
Cat-1000-7-7	KMd	0.94989	74.73140	0.71008
	FKMd	0.81150	78.99450	0.76210
	TSFKMd	0.74440	82.77160	0.79280
	MMR	0.72522	83.55489	0.79612
	G-ANMI	0.66527	85.98743	0.82991
	ccdByEnsemble	0.63487	86.81440	0.83750
	ERFC	**0.49516**	**90.88540**	**0.87010**

Table 2.4: Average values of MS, %CP and ARI for synthetic data sets

Congressional Votes

The voting statistics for the US Congress in 1984 are included in this data collection. There are 435 records in all. One Congressman's votes on 16 separate subjects are represented by each row (e.g., education spending, crime etc.). Every attribute is a boolean with a Yes (value of 1) and No (value of 0) value. Each piece of data comes with a classification label of Republican or Democrat. Records for 168 Republicans and 267 Democrats can be found in the data set. VAT representation of true clusters of this data set is presented in Figure 2.4(d).

Data Sets	Method	MS	%CP	ARI
Soybean	KMd	0.64363	086.79560	0.83610
	FKMd	0.39077	093.96750	0.92000
	TSFKMd	0.36806	094.45920	0.94220
	MMR	0.33104	095.94930	0.94820
	G-ANMI	0.25899	096.95324	0.96620
	ccdByEnsemble	0.22456	097.11881	0.97226
	ERFC	**0.20023**	**098.25324**	**0.97820**
Zoo	KMd	0.68839	084.82920	0.81370
	FKMd	0.43895	092.29820	0.89680
	TSFKMd	0.42744	093.18530	0.91140
	MMR	0.39099	093.89980	0.91160
	G-ANMI	0.37686	094.00950	0.92500
	ccdByEnsemble	0.33654	095.18530	0.94740
	ERFC	**0.30079**	**096.50095**	**0.96500**
Heart	KMd	0.85638	077.42375	0.74110
	FKMd	0.81420	078.79898	0.76060
	TSFKMd	0.80234	079.29142	0.76730
	MMR	0.79907	079.71650	0.77640
	G-ANMI	0.78106	080.40680	0.78280
	ccdByEnsemble	0.74972	082.09247	0.79230
	ERFC	**0.74249**	**082.99540**	**0.79350**
Votes	KMd	0.75553	081.52340	0.79150
	FKMd	0.69745	084.76290	0.81110
	TSFKMd	0.68219	085.05834	0.82010
	MMR	0.67349	085.74890	0.82420
	G-ANMI	0.63072	086.83760	0.83960
	ccdByEnsemble	0.58083	087.76290	0.84610
	ERFC	**0.54678**	**088.79437**	**0.86190**

Table 2.5: Average values of MS, %CP and ARI for real life data sets

2.3.5 Input Parameters

All algorithms are run till they reach their convergence. The various algorithms are fed the inputs parameters listed below..

- The fuzzy exponent (η) = 2
- $\mathtt{f}_{\mathtt{LW}}$ = 0.95 and $\mathtt{f}_{\mathtt{BN}}$ = 0.05
- Number of tree for RF (T) = 1000
- Kernel for SVM = RBF (Radial Basis Function)
- γ and C for SVM = 0.5 and 0.2
- Number of K for K-NN = 3
- τ = 50%

This is to be noted that above values of the parameters are set experimentally after getting better results. The input parameters for the TSFKMd, MMR, G-ANMI, and ccdByEnsemble algorithms are used in a manner consistent with the references [106, 135, 136, 155]. The KMd and FKMd algorithms are iteratively executed until they reach convergence.

2.3.6 Results and Discussion

The clustering results of RFKMd using different dissimilarity measures are used to compute the roughness measure. Table 2.3 reports the average roughness measures of RFKMd while using different dissimilarity measures. It helps to prepare the rank of different dissimilarity measures in terms of their performance. Subsequently, this rank is used to prepare the initial training set for classifying the semi and pure rough objects. Final clustering results of ERFC are demonstrated by computing average values of MS, %CP and ARI scores over 20 runs for ten artificial and actual data sets. The results have been reported in the Tables 2.4 and 2.5 in comparison with different state-of-the-art clustering methods. The best values of MS, %CP and ARI are shown in bold face. Table 2.6 shows the performance of ERFC using three different machine learning methods, namely K-NN, SVM and RF. It is observed for few data sets results are same, however, in most of the cases, RF produces better results.

The overall execution time of RFKMd is usually more than that of corresponding fuzzy version. It is because, different parameters are used for rough measures. Compared to its crisp predecessors, fuzzy clustering takes longer because of the fuzzy membership matrix. Every algorithm has been developed in Matlab and run on a machine with an Intel Core i5-2410M CPU running at 2.30 GHz, 4GB of RAM, and Windows 7 as the operating system. On the average, the execution time of the proposed ERFC clustering is 5.985 seconds for the *Cat-300-15-5* data set, whereas KMd, FKMd, TSFKMd, ccdByEnsemble, G-ANMI and MMR take 0.105, 0.275, 36.256, 13.062, 26.602 and 2.441 seconds, respectively. Based on the

Data Sets	Method	MS	%CP	ARI
Cat-100-8-3	ERFC(K-NN)	0.61509	87.85151	0.84576
	ERFC(SVM)	0.58183	87.89980	0.84720
	ERFC(RF)	0.19733	98.48224	0.97860
Cat-250-15-5	ERFC(K-NN)	0.01203	99.01540	0.98125
	ERFC(SVM)	0.01203	99.01540	0.98125
	ERFC(RF)	0.01203	99.01540	0.98125
Cat-300-8-3	ERFC(K-NN)	0.50411	90.24894	0.86656
	ERFC(SVM)	0.00890	99.55340	0.98563
	ERFC(RF)	0.00890	99.55340	0.98563
Cat-300-15-5	ERFC(K-NN)	0.73557	83.87246	0.80010
	ERFC(SVM)	0.67538	85.66213	0.82301
	ERFC(RF)	0.51885	89.21280	0.86600
Cat-500-20-10	ERFC(K-NN)	0.55417	88.57210	0.85980
	ERFC(SVM)	0.53428	88.95320	0.86310
	ERFC(RF)	0.52995	89.00240	0.86440
Cat-1000-7-7	ERFC(K-NN)	0.73357	83.25478	0.79575
	ERFC(SVM)	0.65398	86.21751	0.83001
	ERFC(RF)	0.49516	90.88540	0.87010
Soybean	ERFC(K-NN)	0.18743	98.28324	0.97820
	ERFC(SVM)	0.20023	98.25324	0.97820
	ERFC(RF)	0.20023	98.25324	0.97820
Zoo	ERFC(K-NN)	0.52241	89.29430	0.86593
	ERFC(SVM)	0.33892	94.98730	0.94430
	ERFC(RF)	0.30079	96.50095	0.96500
Heart	ERFC(K-NN)	0.77805	80.45770	0.78420
	ERFC(SVM)	0.75146	81.68240	0.79190
	ERFC(RF)	0.74249	82.99540	0.79350
Votes	ERFC(K-NN)	0.63788	86.81183	0.83770
	ERFC(SVM)	0.57596	87.91391	0.84790
	ERFC(RF)	0.54678	88.79437	0.86190

Table 2.6: Average values of MS, %CP and ARI of three machine learning methods on synthetic and actual data sets

parameter settings stated in the earlier section, the execution times are calculated. As anticipated, the ERFC's execution time is longer than that of the other clustering techniques due to the added operations like clustering evaluation using all different types of dissimilarity measures, preparation of training and test sets for Random Forest. Yet, it is clear from the results that ERFC performs clustering the best of all the approaches for the data sets taken into consideration in this study. The execution time of ERFC for rest of the data sets are as follows: *Cat-100-8-3*: 0.603 seconds, *Cat-250-15-5*: 3.996 seconds, *Cat-300-8-3*: 2.461 seconds, *Cat-500-20-10*: 12.703 seconds *Cat-1000-7-7*: 8.631 seconds, *Soybean*: 2.295 seconds, *Zoo*: 4.466 seconds, *Heart*: 2.164 seconds and *Votes*: 6.710 seconds.

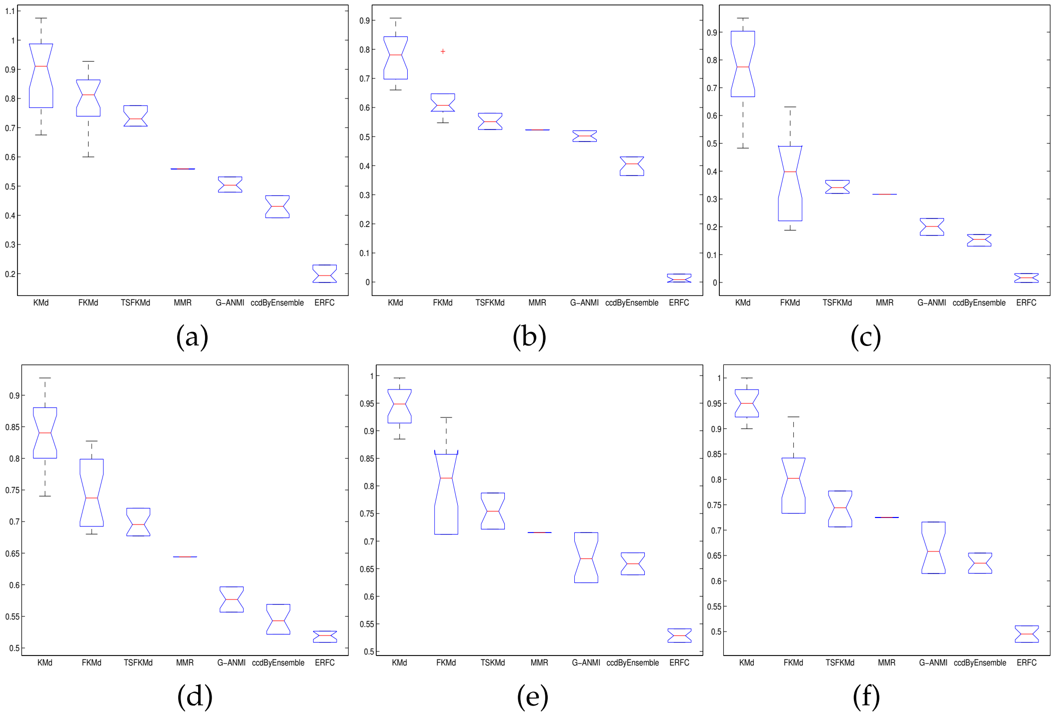
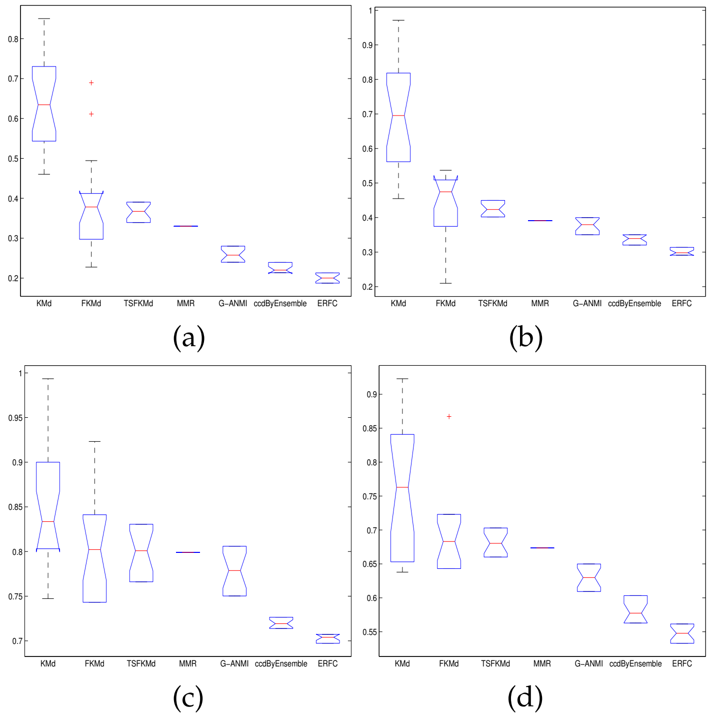


Figure 2.5: Boxplot using MS values of various clustering algorithms for (a) Cat-100-8-3 (b) Cat-250-15-5 (c) Cat-300-8-3 (d) Cat-300-15-5 (e) Cat-500-20-10 (f) Cat-1000-7-7

2.3.7 Statistical Significance Test of the Clustering Results

For the synthetic and actual data sets, Tables 2.4 to 2.5 present the best MS values obtained by various algorithms across 20 consecutive runs. The table makes it clear that the MS values generated by the proposed clustering algorithms are superior to those generated by the other algorithms. A statistical significance test is necessary to prove that the superior performance of the suggested algorithm is statistically significant. The statistical significance of the clustering solutions in this study has been examined using a *t*-test [171] with a 5% threshold of significance. For each data set, seven groups have been constructed, one for each of the seven algorithms: 1. KMd, 2. FKMd, 3. TSFKMd, 4. MMR, 5. G-ANMI, 6. ccdByEnsemble, and 7. ERFC. The MS values from 20 consecutive executions of the corresponding algorithm make up each group.

The *p-values* generated by a *t*-test for comparing two groups (a group representing the ERFC and a group representing another algorithm) at once are reported in Tables 2.7. The null hypothesis states that there is no discernible difference between the two groups' MS values. The alternative hypothesis, on the other hand,



(a) (b)

(c) (d)

Figure 2.6: Boxplot using MS values of various clustering algorithms for (a) Soybean (b) Zoo (c) Heart (d) Votes

Data Sets	KMd	FKMd	TSFKMd	MMR	G-ANMI	ccdByEnsemble
Cat-100-8-3	2.814e-21	2.819e-19	2.248e-17	7.247e-14	3.329e-10	6.917e-08
Cat-250-15-5	2.564e-23	5.164e-20	7.444e-16	4.060e-12	2.697e-09	4.495e-07
Cat-300-8-3	6.153e-23	3.526e-20	1.497e-17	4.803e-13	3.873e-10	2.940e-07
Cat-300-15-5	1.355e-21	6.750e-19	2.381e-17	1.479e-13	2.644e-09	5.179e-06
Cat-500-20-10	1.022e-11	9.114e-11	3.181e-10	4.199e-12	5.291e-08	5.353e-17
Cat-1000-7-7	2.803e-14	2.446e-09	4.381e-14	2.723e-14	4.501e-10	9.412e-09
Soybean	5.539e-19	1.537e-19	1.752e-16	4.088e-12	1.181e-09	1.481e-06
Zoo	4.377e-19	2.024e-17	8.664e-18	5.049e-15	6.378e-11	4.621e-08
Heart	7.288e-20	1.666e-18	6.991e-17	2.805e-14	8.326e-09	3.726e-07
Votes	1.159e-21	1.063e-19	1.051e-17	4.970e-13	1.341e-10	1.161e-07

Table 2.7: *t*-test results for Synthetic and actual data sets. Test produces $p - values$ by comparing ERFC with other algorithms.

contends that the mean values of the two groups significantly diverge. Less than 0.05 (5% significance level) is the threshold for all of the *p-values* provided in the table. As an illustration, the *p-value* for the *t*-test between the algorithms ERFC and FKMd for the Zoo data set is 2.024e-17, which is significantly lower than the significance level 0.05. This provides compelling evidence that the better MS values produced by the suggested algorithm are not the result of chance and instead constitute statistically significant evidence against the null hypothesis. Comparable outcomes are found when ERFC is compared to all other data sets and techniques, demonstrating the ensemble rough fuzzy clustering algorithm's clear superiority.

2.4 Worst case Time Complexity Analysis

The proposed algorithms are analyzed for their time complexity below.

2.4.1 Time Complexity Analysis of RFKMd

The computation of fuzzy membership matrix for whole data set, and searching for the highest two membership values for each object take time of $O\left(2Knm\right)$. Considering other activities, the total worst case time complexity of RFKMd methods is $O\left(2Knm + n + Kn + KMn\right) \approx O\left(KMn\right)$ for each iteration. Here $M\left(= \sum_{j=1}^{m} q_j\right)$, is the total number of categories of all attributes.

2.4.2 Time Complexity Analysis of ERFC

The time complexity of ERFC depends on the complexity of RFKMd, union of sets, intersection of sets and the random forest method. Each of these parts has the complexity of polynomial order. Therefore, the worst case time complexity of RFKMd is $O\left(KMn\right)$ for each iteration and time complexity of random forest method is $O\left(\mathsf{T}nm\log\left(n\right)\right)$. On the other hand, the sorting of roughness metrics is negligible, because, $\mathcal{N}$ is constant. Hence, the overall complexity of ERFC is $O\left(KMn + \mathsf{T}nm\log\left(n\right)\right)$.

2.5 Conclusion

In this chapter, two distinct clustering algorithms based on Rough Fuzzy theory for the analysis of categorical data sets have been proposed [172]. The testing phase involves the use of six synthetic and four real data sets. The first proposal is the Rough Fuzzy K-Mode (RFKMd) algorithm, which utilizes a single dissimilarity metric. However, it is important to note that no one dissimilarity measure can adequately deliver the optimal clustering outcome for all kinds of categorical data sets. As a consequence, a framework known as Ensemble based Rough Fuzzy Clustering (ERFC) has been developed. In order to achieve this objective, the use of RFKMd is employed in conjunction with a range of dissimilarity metrics to assess distinct collections of rough items. The use of the Random Forest classifier is then employed to categorize the leftover rough items. The incorporation of rough and fuzzy set theories has been used to address the challenges posed by overlapping partition and the presence of ambiguity and vagueness in data sets. The comparative effectiveness of ERFC has been established by the evaluation of multiple cluster validity indices on a range of artificial and real-life categorical datasets, in contrast to many other recently developed approaches. The findings indicate that the ERFC exhibits superior performance, which has been statistically validated using a *t*-test conducted at a significance level of 5%. The subsequent chapter explores the integration of possibilistic theory with rough and fuzzy set theory, presenting a promising avenue for possible enhancement.

3
Rough Possibilistic Fuzzy Clustering

3.1 Introduction

Segmenting objects into several classes is one of the crucial uses of clustering. The job of segmenting magnetic resonance (MR) brain pictures into many tissue classes has garnered significant attention in recent times due to its inherent complexity. The use of magnetic resonance (MR) imaging for the categorization of brain tissue finds application in several medical fields, including neurological disorders. Segmentation is often regarded as a job in data pre-processing. Consequently, the efficacy of the segmentation process has a substantial influence on the results obtained from the image analysis. Segmentation is a fundamental procedure that involves partitioning an image space into many discrete portions that possess uniform characteristics and do not overlap with one another. Clustering is often seen as a challenging task in the domain of intensity space classification, as discussed by Maulik et al. (2003) [173]. Clustering is an extremely popular unsupervised technique that aims to detect patterns within the underlying data and arrange the data based on their similarity, as previously discussed in chapter 2. The empirical data in the actual world exhibits characteristics of ambiguity, overlap, and indeterminacy. Both Traditional Fuzzy C-Means (FCM) and prototype based on hard C-Means have often been used to address issues of uncertainty and overlaps, with the former being grounded in type-1 fuzzy set theory. The basic purpose of the Fuzzy C-Means (FCM) algorithm is to maximize the overall compactness of the clusters. To do so, an unlabeled data set $X = \{x_1, x_2, \ldots, x_n\}$ is partitioned into K clusters such that each point has some degree of belongingness and a membership value to each group. As a result, it was concluded that while a data point may be assigned to many clusters concurrently, the sum of its membership values must equal one. The imposition of this constraint hinders FCM's ability to enhance its performance when confronted with noisy data. In order to address this issue, Krishnapuram and colleagues [122,123] proposed a possibilistic approach called Possibilistic C-Means (PCM). Unlike traditional methods that rely on relative membership, PCM employs typicality membership. Timm and colleagues [128] proposed the use of possibilistic fuzzy clustering as a potential solution to address the problem of coincident clustering. The goal function has been expanded to include the reciprocal function of the distances between cluster centers. The supplementary function serves as a compelling force that maintains separation between clusters. In their publication, Pal *et al.* [124] have argued for the need of including both probabilistic and fuzzy membership values. The Possibilistic Fuzzy C-Means algorithm underwent revisions and enhancements, which were documented and published in the work of Pal et al. [87]. The possibilistic technique, however, depends on the selection of the starting parameters and the assignment of membership values to patterns with prototypes that are relatively near together [174]. In addition, the fuzzy clustering methodology has limitations in effectively handling subtle uncertainty and ambiguity. The issue at hand may be effectively addressed via the use of Pawlak's rough set theory, which has been extensively examined in previous studies [157, 175]. Based on the preliminary clustering analysis, a data point is classified as belonging to a certain cluster with a membership degree of 1, or it is situated in the overlapping regions between several clusters. Consequently, one may see that the points in the *Boundary Region* are situated at the intersection of two or more clusters.

The bulk of traditional clustering approaches were developed based on type-1 fuzzy ideas. The research conducted in [176, 177] focuses on the investigation of the type-2 fuzzy set theory. This investigation is motivated by the limitations of the type-1 fuzzy concept in effectively handling many forms of intrinsic uncertainty. The major membership value in this notion is similarly characterized by fuzziness. In their study, Zeng and Liu [178] argue that fuzziness is an intrinsic source of uncertainty in the field of pattern recognition. The researchers endeavoured to address the challenges associated with fuzziness and randomness by developing a unified framework that encompasses several aspects, such as type-1 fuzzy randomness, type-1 fuzzy probability, and type-1 fuzzy statistics. Their objective was to comprehensively consider all possible outcomes within this framework. Consequently, the necessity for Secondary Membership arose as a means to conduct a more comprehensive assessment of the initial membership grade, which was characterized by ambiguity. The characterization of the feature space's randomness is denoted by primary membership, whilst the imprecision of the primary membership value is denoted by secondary membership.

The objective of this chapter is to propose a hybrid clustering approach that leverages the advantages offered by type-2 fuzzy set, rough set, and possibilistic theory. Rough sets use lower and upper approximations to effectively handle uncertainty and ambiguity, whereas possibilistic notions address challenges associated with noisy data and outliers. On the contrary, the concept of type-2 fuzzy sets has the ability to effectively handle situations including uncertainty and unpredictability. Prior to introducing the hybrid clustering approach known as Rough Possibilistic Type-2 Fuzzy C-Means clustering (RPT2FCM), a foundational technique called Rough Type-2 Fuzzy C-Means (RT2FCM) has been formulated based on the principles of rough set theory and type-2 fuzzy set theory. The possibilistic notion has been used in RT2FCM to offer RPT2FCM, with the aim of achieving additional refinement. Nevertheless, the used methodology produces clustering points that exhibit both rough and crisp characteristics. Therefore, to effectively manage the rough points created, we integrated Random Forest (RF) [27] into RPT2FCM, resulting in the development of RPT2FCM-RF. This integration aims to enhance the classification of rough points and improve the overall clustering solution. The study presents experimental results and compares them to established approaches in the context of MR brain image segmentation. A statistical significance study has been conducted to validate the benefits of the proposed methodology.

3.2 Rough Possibilistic Fuzzy Clustering

This section explains the proposed Rough Type-2 Fuzzy C-Means (RT2FCM), Rough Possibilistic Type-2 Fuzzy C-Means clustering (RPT2FCM) with the combination of the advantages of rough set, type-2 fuzzy set and possibilistic concepts, followed by the framework, Rough Possibilistic Type-2 Fuzzy C-Means integrated with Random Forest (RPT2FCM-RF). Before describing the proposed techniques, brief understanding of conventional Fuzzy C-Means (FCM), possibilistic, type-2 fuzzy set concepts and random forest classifier have been discussed in the following subsections respectively.

3.2.1 Brief Description of Fuzzy C-Means

The most popular partitioning method is Fuzzy C-Means (FCM), which is a fuzzy variation of the conventional Hard C-Means (HCM). To minimise the following objective function, FCM divides the data set $X = \{x_i \mid 1 \leq i \leq n\}$ into K number of clusters using the fuzzy set theory approach.

$$J_{\mathcal{FCM}} = \sum_{i=1}^{n} \sum_{l=1}^{K} \mu_{li}^{\eta_1} D(c_l, x_i). \tag{3.1}$$

Here, the function $D(c_l, x_i)$ computes the Euclidean distance between the point x_i and the cluster center c_l. The symbol η_1 denotes the weighting coefficient, whereas μ_{li} indicates the fuzzy membership value or the degree of belongingness of the ith point to the lth cluster. The initial cluster centers in Fuzzy C-Means (FCM) are selected randomly from a set of K potential cluster centers. Subsequently, the algorithm proceeds to repeatedly update the membership values as follows until the objective value remains unchanged.

$$\mu_{li} = \frac{1}{\sum_{h=1}^{K}(\frac{D(c_l,x_i)}{D(v_h,x_i)})^{\frac{2}{\eta_1-1}}}, \text{ for } 1 \le l \le K; \ 1 \le i \le n, \tag{3.2}$$

with two constraints as mentioned in Equation 3.3 and 3.4.

$$\sum_{l=1}^{K} \mu_{li} = 1 \text{ for } 1 \le i \le n, \tag{3.3}$$

$$0 < \sum_{i=1}^{n} \mu_{li} < n \text{ for } 1 \le l \le K, \tag{3.4}$$

It is important to observe that while calculating μ_{li} using Equation 3.2, if $D(v_h, x_i)$ is equal to zero for a certain h, then μ_{li} is assigned a value of zero for all $l = 1, \ldots, K$, where l is not equal to h. Additionally, μ_{hi} is assigned a value of one. The re-calibration of cluster centers occurs throughout each iteration based on the membership values, as specified in Equation 3.5. This process aims to minimize the objective function outlined in Equation 3.1. Ultimately, each point is allocated to the cluster that exhibits the greatest degree of membership.

$$c_l = \frac{\sum_{i=1}^{n} \mu_{li}^{\eta_1} x_i}{\sum_{i=1}^{n} \mu_{li}^{\eta_1}} \tag{3.5}$$

3.2.2 Brief Description of Possibilistic Approach

One of the main limitations of Fuzzy C-Means (FCM) algorithm is in its susceptibility to noise and outliers. Krishnapuram and colleagues proposed the use of PCM (Possibilistic C-Means) in their works [122, 123] as a means to address this limitation. The objective function is modified when used to Fuzzy C-Means (FCM) in the following manner.

$$J_{\mathcal{PCM}} = \sum_{i=1}^{n} \sum_{l=1}^{K} t_{li}{}^{\eta_2} D(c_l, x_i) + \sum_{l=1}^{K} \gamma_l \sum_{i=1}^{n} (1 - t_{li})^{\eta_2} \tag{3.6}$$

Here the variable t_{li} is referred to as the typicality of x_i for the lth cluster. The parameter γ_l represents the scale and indicates the zone of impact or size of the lth cluster. Additionally, the fuzzifier η_2 is a scalar value greater than or equal to 1 and may be infinite ($1 \le \eta_2 \le \infty$). The calculation of the typicality value is shown below.

$$t_{li} = \frac{1}{1 + \left(\frac{D(c_l, x_i)}{\gamma_l}\right)^{\frac{1}{\eta_2 - 1}}} \tag{3.7}$$

where,

$$0 < \sum_{i=1}^{n} t_{li} \leq n \quad \forall\, l \tag{3.8}$$

$$\max_{l}\{t_{li}\} > 0 \quad \forall\, i \tag{3.9}$$

$$\gamma_l = \mathcal{K} \frac{\sum_{i=1}^{n} t_{li}{}^{\eta_2} D(c_l, x_i)}{\sum_{i=1}^{n} t_{li}{}^{\eta_2}} \tag{3.10}$$

$\mathcal{K}$ is often maintained at one. The cluster's center is updated as shown below.

$$c_l = \frac{\sum_{i=1}^{n} t_{li}{}^{\eta_2} x_i}{\sum_{i=1}^{n} t_{li}{}^{\eta_2}} \tag{3.11}$$

3.2.3 Brief Description of Type-2 Fuzzy Concept

The effectiveness of the probabilistic approach is greatly influenced by the initial parameter selection and the assignment of membership to the pattern, particularly when the center-to-center distances are very small. Identifying inherent uncertainties and ambiguities is a challenge in light of the use of the type-1 fuzzy set approach employed by FCM. The concept of type-2 fuzzy set theory has garnered significant scholarly interest. The expansion of type-1 fuzzy set theory, first suggested by Professor Zadeh, was introduced to effectively tackle these concerns by providing a greater range of degrees of freedom. The fundamental distinction between type-1 and type-2 fuzzy sets is in the inherent characteristics of their membership values. Type-1 fuzzy sets are characterized by membership values that consist of real numbers within the interval [0, 1]. In contrast, type-2 fuzzy set theory introduces a main membership value that has intrinsic fuzziness. This aspect of type-2 fuzzy set theory captures additional uncertainty present in the data set. The formula shown below is used for the computation of the membership value of the type-2 fuzzy set, denoted as $\hat{\mu}_{li}$.

$$\hat{\mu}_{li} = \mu_{li} - \frac{(1 - \mu_{li})}{2} \tag{3.12}$$

where The membership value of a type-1 fuzzy set is denoted as μ_{li}. Figure 1.1 illustrates the increase in uncertainty of type-2 membership values as the type-1 membership value approaches zero. In very exceptional circumstances, the value of the type-2 membership for the type-1 membership is 1. The authors Rhree et al. were motivated by this observation to develop the type-2 Fuzzy C-Means (T2FCM) clustering methodology [1].

3.2.4 Brief Description of Supervised Classifier

One of the several supervised methods used in classification tasks is the K-Nearest Neighbour (K-NN) algorithm, which assigns labels to data points by considering the K closest neighbours from the training dataset. This approach does not make any assumptions on the distribution of the underlying data. The variable denoted as K determines the extent to which neighbouring data points contribute to the categorization process. Typically, the practice of maintaining an odd number of courses is observed when there are just two classes. While the K-NN algorithm is widely recognized as a suitable approach for dealing with high-dimensional data, it requires either approximate dimension reduction or thorough feature selection. The Support Vector Machine (SVM) is a commonly used supervised classifier that creates a dividing hyperplane in a d-dimensional space to divide the input dataset into two distinct classes [21]. The hyperplane seeks to optimize the margin between two distinct classes. The SVM classifier is mainly intended for binary classification tasks. Multi-class challenges may be addressed by implementing one-against-all or one-against-one two-class SVMs. It uses a technique that involves mapping the input dimension to a feature space of higher dimensionality in order to handle linearly non-separable input datasets. Subsequently, a linear hyperplane is constructed for the purpose of categorization. The procedure exhibits a high degree of computational complexity. SVM is renowned for their ability to effectively handle datasets with a large number of dimensions. Nevertheless, previous research has shown that SVM exhibits a deficiency in robustness when faced with a significant quantity of extraneous variables [179]. Furthermore, the task of attaining precision in scenarios involving multi-classes is a notable difficulty. In such scenarios, Random Forest (RF) has been extensively employed as a classifier for efficiently handling data with high dimensions. It has exhibited exceptional performance when utilized alongside an ensemble of decision trees, where all trees within the ensemble are subjected to the same distribution and random input vectors are employed. The CART technique, as described by Breiman [180], was used in this work to generate a substantial number of trees. Specifically, a total of 1000 trees were constructed for the experiment undertaken. The nodes are then partitioned according to a randomly selected subset of characteristics. Consequently, this classifier has the potential to be used in the domains of unsupervised learning, regression analysis, and classification tasks. RF is extensively used across several industries, such as high performance liquid chromatography [181], near-infrared [182], electronic nasal data processing [183], and gas chromatography [184], to harness its potential benefits.

3.2.5 Rough Type-2 Fuzzy C-Means

The presence of fuzziness within dataset has been shown by the use of type-2 fuzzy theory inside the proposed Rough Fuzzy Type-2 C-Means (RT2FCM) approach, which aims to address uncertainty in a more comprehensive manner. The notion of *Lower* and *Upper Approximation* in Rough Set Theory [4] offers more possibilities for dealing with ambiguity and uncertainty. According to the rough set theory, as discussed in Chapter 2, the *Lower Approximation* of a cluster has a substantial impact on the corresponding cluster and its mean. However, these points do not exert any impact on points located outside the cluster region, nor are they affected by any objects located outside their respective cluster area. Consequently, it is essential to assign more significance to the variable $\mathtt{f}_{\mathtt{LW}}$ in comparison to $\mathtt{f}_{\mathtt{BN}}$. If x_i belongs to the Lower Approximation, the value of $mathttf_LW$ for x_i is greater than the value of $mathttf_BN$ for points in the Boundary Region. If x_i does not belong to any cluster's *Upper Approximation*, then it is a member of the *Upper Approximation* of many clusters. The values of $\mathtt{f}_{\mathtt{LW}}$ and $\mathtt{f}_{\mathtt{BN}}$ are subsequently chosen in such a way that the criterion $0 < \mathtt{f}_{\mathtt{BN}} < \mathtt{f}_{\mathtt{LW}} < 1$ is met. The objective function is contingent upon the fuzzy exponents η_1 and the relative significance parameters $\mathtt{f}_{\mathtt{LW}}$ and $\mathtt{f}_{\mathtt{BN}}$.

Definition 7 *The objective function of RT2FCM can be defined as:*

$$J_{RT2FCM} = \begin{cases} \mathtt{f}_{\mathtt{LW}} \times \mathsf{P} + \mathtt{f}_{\mathtt{BN}} \times \mathsf{Q}, & if \underline{B}(C_l) \neq \emptyset, BN(C_l) \neq \emptyset \\ \mathsf{P}, & if \underline{B}(C_l) \neq \emptyset, BN(C_l) = \emptyset \\ \mathsf{Q}, & if \underline{B}(C_l) = \emptyset, BN(C_l) \neq \emptyset \end{cases} \tag{3.13}$$

$$\mathsf{P} = \sum_{l=1}^{K} \sum_{x_i \in \underline{B}(C_l)} (\hat{\mu}_{li})^{\eta_1} D(c_l, x_i)$$

$$\mathsf{Q} = \sum_{l=1}^{K} \sum_{x_i \in BN(C_l)} (\hat{\mu}_{li})^{\eta_1} D(c_l, x_i)$$

$$\hat{\mu}_{li} = \mu_{li} - \frac{(1 - \mu_{li})}{2}$$

where:
X represents the non-empty set of data points;
$C_l \subseteq X$ for $1 \leq l \leq K$ is the set of points of l^{th} cluster;
c_l, the mean of cluster C_l;

$\underline{B}(C_l)$ is the Lower Approximation of cluster C_l;
$\overline{B}(C_l)$ is the Upper Approximation of cluster C_l;
$BN(C_l) = [\overline{B}(C_l) - \underline{B}(C_l)]$ is the Boundary Region of cluster C_l;
$D(c_l, x_i)$ is the distance between mean c_l and data point x_i;
$\mathtt{f}_{\mathtt{LW}}$ is the relative importance of Lower Approximation;
$\mathtt{f}_{\mathtt{BN}}$ is the relative importance of Boundary Region;
$\mathtt{f}_{\mathtt{LW}} + \mathtt{f}_{\mathtt{BN}} = 1$ and $0 < \mathtt{f}_{\mathtt{BN}} < \mathtt{f}_{\mathtt{LW}} < 1$;
$\tilde{\mu}_{li}$ represents the type-2 fuzzy membership value;
μ_{li} has the same meaning as in FCM;

In two stages, the RT2FCM algorithm operates. RT2FCM is run in the first stage, producing some classified crisp points as well as unclassified rough points. The second stage is classifying the rough points into the corresponding classes applying type-2 fuzzy membership values. RT2FCM begins processing with K random points chosen as K cluster means. It then calculates $\tilde{\mu}_{li}$ for each of the n data points. Let's say that among all K clusters, $\tilde{\mu}_{li}$ and $\tilde{\mu}_{hi}$ are the highest and second-highest computed membership values of the point x_i, where $1 \leq l, h \leq K \;\; and \;\; h \neq l$. The classification of the point x_i is determined by the difference between $\tilde{\mu}_{li}$ and $\tilde{\mu}_{hi}$. If the difference exceeds the Δ, then x_i is unambiguously and precisely classified into $\underline{B}(C_l)$. Then, for the lth cluster, $\tilde{\mu}_{li}$ and t_{li} of that point are set to 1 and 0 for the rest of clusters. In contrast, if the difference does not exceed the threshold, the point can belong to more than one cluster since there is uncertainty, according to the rough set notion. As a result, the point does not belong in any *Lower Approximation* but rather to the *Upper Approximation* region of several clusters. The values of both $\tilde{\mu}_{li}$ and t_{li} remain the same. The median of the difference between the highest and second-highest fuzzy membership of all the points is used to get the threshold value, Δ. Therefore,

$$\Delta = \underset{i=1,2,\ldots,n}{\text{median}} \, (\tilde{\mu}_{li} - \tilde{\mu}_{hi}),$$

The median and mean are the same if the distribution of $(\tilde{\mu}_{li} - \tilde{\mu}_{hi})$ is not skewed $\forall i = 1, 2, \ldots, n$. Mean could be used in place of median. If the distribution is skewed, however, the outcome could be adversely affected by any one odd member (extremely large or very small compared to all other values). As a result, it has been found that the median is effective in this case. Once all n points have been classified, RT2FCM computes the new median. This process is repeated as long as there is a difference between any two successive values of the objective function. The Algorithm 4 contains a detailed description of the steps.

The type-2 fuzzy membership value, $\tilde{\mu}_{li}$, and typicality value, t_{li}, are set to 1 for each of the points in the *Lower Approximation* area due to the certainty surrounding the clustering of those points. This shows that there is no ambiguity regarding how those points should be classified. Relative significance factor a can be set to 1 as a result and P becomes

$$\mathsf{P} = \sum_{l=1}^{K} \sum_{x_i \in \underline{B}(C_l)} D(c_l, x_i), \;\; 1 \leq i \leq n$$

Definition 8 *Each cluster (C_l) is characterized by a center (c_l), a crisp lower approximation ($\underline{B}(C_l)$), and a fuzzy boundary ($BN(C_l)$). The center is modified in a manner that optimizes Equation 3.13.*

$$c_l = \begin{cases} \mathtt{f}_{\mathtt{LW}} \times \mathsf{G}_{LW} + \mathtt{f}_{\mathtt{BN}} \times \mathsf{G}_{BN}, \\ \qquad\qquad if \underline{B}(C_l) \neq \emptyset, BN(C_l) \neq \emptyset \\ \mathsf{G}_{LW}, \qquad if \underline{B}(C_l) \neq \emptyset, BN(C_l) = \emptyset \\ \mathsf{G}_{BN}, \qquad if \underline{B}(C_l) = \emptyset, BN(C_l) \neq \emptyset \end{cases} \tag{3.14}$$

where,

$$\mathsf{G}_{LW} = \frac{\sum_{x_i \in \underline{B}(C_l)} x_i}{\mid \underline{B}(C_l) \mid},$$

$$\mathsf{G}_{BN} = \frac{\sum_{x_i \in BN(C_l)} (\hat{\mu}_{li})^{\eta_1} x_i}{\sum_{x_i \in BN(C_l)} (\hat{\mu}_{li})^{\eta_1}}$$

Definition 8 formulates the cluster center update process using Equation 3.14. When the b*Lower Approximation* and the *Boundary Region* are both not empty, the center is calculated as ($\mathtt{f}_{\mathtt{LW}} \times \mathsf{G}_{LW} + \mathtt{f}_{\mathtt{BN}} \times \mathsf{G}_{BN}$), where $\mathtt{f}_{\mathtt{LW}}$ and $\mathtt{f}_{\mathtt{BN}}$ are the relative importance same as used in Equation 3.15. G_{LW} makes a contribution to the *Lower Approximation*, which behaves like a hard K-Means method since the points in the *Lower Approximation* are confident that they belong in a cluster. On the other hand, the *Boundary Region* is influenced by G_{BN}. Therefore, when calculating the cluster's center, both probabilistic and possibilistic measurements are taken into consideration for points within the *Boundary Region*. Center adjustment is carried out repeatedly until the algorithm converges.

Algorithm 4 : RT2FCM

```
Input:
    $X$: dataset
    $\eta_1$: fuzzy exponents
    $\epsilon$, threshold value which is very small real value between [0,1]
    $K$, number of clusters
    $\mathtt{f}_{\mathtt{LW}}$, relative weight value for Lower Approximation of rough clustering, $0 < \mathtt{f}_{\mathtt{LW}} < 1$
Output: $[\hat{\mu}]$ where, $1 \le l \le K$ and $1 \le i \le n$

Select $K$ random points from dataset, $X$ to initiate $K$ cluster centers
repeat
    Compute $\hat{\mu}_{li}$ for all $n$ points using Equation 3.2
    Compute the difference between highest two membership values, $\hat{\mu}_{li}$ and $\hat{\mu}_{hi}$ of each $n$ data points
        // Let $\hat{\mu}_{li}$ and $\hat{\mu}_{hi}$, highest and second highest membership values of $x_i$ among all K clusters, where $1 \le l, h \le K$ and $h \neq l$
    Calculate the value of threshold $\Delta$
        // $\Delta$ is the median of $(\hat{\mu}_{li} - \hat{\mu}_{hi}), \forall i = 1, 2, \ldots, n$
    if $(\hat{\mu}_{li} - \hat{\mu}_{hi}) > \Delta$ then
        $\hat{\mu}_{li} \leftarrow 1$, $\hat{\mu}_{hi} \leftarrow 0 \quad \forall h = 1, 2, \ldots, K$ and $h \neq l$
            // $x_i$ is classified to $\underline{B}(C_l)$, as well as to $\overline{B}(C_l)$ as per rough set theory
    else
        Keep $\hat{\mu}_{hi}$ unchanged $\forall h = 1, 2, \ldots, K$
            // $x_i$ can belong to Upper Approximation of the multiple clusters. Hence, $x_i$ belong to $\overline{B}(C_l)$ and $\overline{B}(C_h)$
    end if
    Compute new center with the help of Equation 3.14
until $|Current\ J_{RPT2FCM} - Previous\ J_{RPT2FCM}| \le \epsilon$
return $[\hat{\mu}]$ where, $1 \le l \le K$ and $1 \le i \le n$
```

3.2.6 Rough Possibilistic Type-2 Fuzzy C-Means and its Integration with Random Forest

Further, the RT2FCM has taken into account the possibilistic concept to propose Rough Possibilistic Fuzzy Type-2 C-Means (RPT2FCM) in order to address sensitiveness to noise and outliers. After the final iteration, the RPT2FCM generates points that are rough and crisp. Rough points belong to the *Boundary Region*, while crisp points typically belong to the *Lower Approximation* of a cluster. The Random Forest (RF) classifier is additionally incorporated and trained using the crisp points to carry out the classification operation in order to classify the rough points. Therefore, it is known as RPT2FCM-RF. The RPT2FCM method clusters the data using both probabilistic and possibilistic methods, while type-2 fuzzy theory more thoroughly addresses uncertainty. At the same time, Rough set theory (RST) [4] provides further ways to deal with vagueness & uncertainty using the idea of *Lower* and *Upper Approximation*.

The objective function of the proposed method is defined in the Definition 9 using RST, where the relative importance parameter, $\mathtt{f}_{\mathtt{LW}}$, is utilized for the *Lower Approximation* and $\mathtt{f}_{\mathtt{BN}}$, for the *Boundary Region*. The objective function consists of two portions, P and Q, for *Lower Approximation* and *Boundary Region*, respectively. Crisp points and rough points, respectively, are the names given to the points in the first and second parts. Crisp points are neither influenced by any points outside the cluster's *Lower Approximation* nor are they influenced by any points outside of it. As a result, $\mathtt{f}_{\mathtt{LW}}$ is more significant than $\mathtt{f}_{\mathtt{BN}}$. Therefore, if x_i belongs to the *Lower Approximation*, the $\mathtt{f}_{\mathtt{LW}}$ value for x_i is set to be higher than the $\mathtt{f}_{\mathtt{BN}}$ value for points in the *Boundary Region*. If not, x_i is a member of the *Upper Approximation* of several clusters, which denotes that x_i is a member of the *Boundary Region*. The values of $\mathtt{f}_{\mathtt{LW}}$ and $\mathtt{f}_{\mathtt{BN}}$ are then selected so that the condition $0 < \mathtt{f}_{\mathtt{BN}} < \mathtt{f}_{\mathtt{LW}} < 1$ gets satisfied. It is to be noted that the objective function depends on the fuzzy exponents η_1 and η_2 as well as the relative importance parameters $\mathtt{f}_{\mathtt{LW}}$, $\mathtt{f}_{\mathtt{BN}}$ respectively.

Definition 9 *Mathematically the objective function is defined as*

$$J_{RPT2FCM} = \begin{cases} \mathtt{f}_{\mathtt{LW}} \times \mathsf{P} + \mathtt{f}_{\mathtt{BN}} \times \mathsf{Q}, & if\underline{B}(C_l) \neq \emptyset, BN(C_l) \neq \emptyset \\ \mathsf{P}, & if\underline{B}(C_l) \neq \emptyset, BN(C_l) = \emptyset \\ \mathsf{Q}, & if\underline{B}(C_l) = \emptyset, BN(C_l) \neq \emptyset \end{cases} \tag{3.15}$$

$$\mathsf{P} = \sum_{l=1}^{K} \sum_{x_i \in \underline{B}(C_l)} \{a\,(\hat{\mu}_{li})^{\eta_1} + b\,(t_{li})^{\eta_2}\} D(c_l, x_i)$$

$$\mathsf{Q} = \mathcal{A} + \mathcal{B}$$

$$\mathcal{A} = \sum_{l=1}^{K} \sum_{x_i \in BN(C_l)} \{a\,(\hat{\mu}_{li})^{\eta_1} + b\,(t_{li})^{\eta_2}\} D(c_l, x_i)$$

$$\mathcal{B} = \sum_{l=1}^{K} \gamma_l \sum_{x_i \in BN(C_l)} (1 - t_{li})^{\eta_2}$$

$$t_{li} = \frac{1}{1 + \left(\frac{b}{\gamma_l} D(c_l, x_i)\right)^{\frac{1}{\eta_2 - 1}}}$$

$$\hat{\mu}_{li} = \mu_{li} - \frac{(1 - \mu_{li})}{2}$$

where, X is a non-empty data set, $C_l \subseteq X$ for $1 \leq l \leq K$ represents the set of points of lth cluster and c_l is the center of cluster C_l, $\underline{B}(C_l)$ and $\overline{B}(C_l)$ refer the Lower and Upper Approximation of the cluster C_l, $BN(C_l) = [\overline{B}(C_l) - \underline{B}(C_l)]$ defines the Boundary Region of cluster C_l, $D(c_l, x_i)$ is used to measure the distance between the center c_l and the data point x_i, $\mathtt{f}_{\mathtt{LW}}$ and $\mathtt{f}_{\mathtt{BN}}$ are the relative importance of Lower Approximation and Boundary Region having relation $\mathtt{f}_{\mathtt{LW}} + \mathtt{f}_{\mathtt{BN}} = 1$ and $0 < \mathtt{f}_{\mathtt{BN}} < \mathtt{f}_{\mathtt{LW}} < 1$. Type-2 fuzzy membership value is defined by $\hat{\mu}_{li}$ while μ_{li} and t_{li} are having similar meaning as in FCM and PCM. a and b correspond to the relative importance for probabilistic and possibilistic aspect.

Algorithm 5 : RPT2FCM

Input:
X, the data set
η_1, η_2, the fuzzy exponents
a, b, the relative importance for probabilistic and possibilistic membership value between [0,1]
ϵ, a small real threshold value between [0,1]
K, the number of cluster
$\mathfrak{f}_{\mathtt{LW}}$, relative weight for *Lower Approximation* of rough clustering, $0 < \mathfrak{f}_{\mathtt{LW}} < 1$
Output: $[\hat{\mu}], [t]$ *where*, $1 \leq l \leq K$ *and* $1 \leq i \leq n$

```
Select $K$ random points from data set as $K$ cluster centers
repeat
    Compute $t_{li}$, $\hat{\mu}_{li}$ for all $n$ points using Equation 3.7 and 3.12
                                    // $t_{li}$ is same as $\hat{\mu}_{li}$ for 1st iteration
    Compute $u_{li} = \{a\,(\hat{\mu}_{li}) + b\,(t_{li})\}$
    Compute the difference between highest two computed membership, $u_{li}$ of each and every $n$ data points
        // Let $u_{li}$ and $u_{hi}$, highest and second highest computed membership values of $x_i$ among all K clusters, where
                                    $1 \leq l, h \leq K$ and $h \neq l$
    Compute the value of threshold $\Delta$
                                    // $\Delta$ is the median of $(u_{li} - u_{hi}), \forall i = 1, 2, \ldots, n$
    if $(u_{li} - u_{hi}) > \Delta$ then
        $\hat{\mu}_{li} \leftarrow 1$, $t_{li} \leftarrow 1$, $\hat{\mu}_{hi} \leftarrow 0$ and $t_{hi} \leftarrow 0, \forall h = 1, 2, \ldots, K$ and $h \neq l$
                                    // $x_i$ is exactly classified to $\underline{B}(C_l)$, also to $\overline{B}(C_l)$ as per RST
    else
        Keep $\hat{\mu}_{hi}$ and $t_{hi}$ unchanged $\forall h = 1, 2, \ldots, K$
            // $x_i$ can belong to Upper Approximation of multiple clusters. Hence, $x_i$ belong to $\overline{B}(C_l)$ and $\overline{B}(C_h)$
    end if
    Compute new mean with the help of Equation 3.16
until $|Current\ J_{RPT2FCM} - Previous\ J_{RPT2FCM}| \leq \epsilon$
return $[\hat{\mu}], [t]$ where, $1 \leq l \leq K$ and $1 \leq i \leq n$
```

Algorithm 6 : RPT2FCM-RF

Input:
X, the data set
η_1, η_2, the fuzzy exponents
a, b, the relative importance for probabilistic and possibilistic membership value between [0,1]
ϵ, a small real threshold value between [0,1]
K, the number of cluster
$\mathfrak{f}_{\mathtt{LW}}$, relative weight for *Lower Approximation* of rough clustering, $0 < \mathfrak{f}_{\mathtt{LW}} < 1$
$\mathbb{M}$, Machine Learning Classifier
Output: F, the final class label vector of X

```
Use 5 to produce crisp data set, $\mathcal{L} = \{x_i \in \underline{B}(C_l) \mid 1 \leq l \leq K$ and $1 \leq i \leq n\}$ and corresponding cluster label vector, $\beta_1$
Classify $\mathcal{L}^* = (X - \mathcal{L})$ using a classifier,$\mathbb{M}$, trained by $\mathcal{L}$ and $\beta_1$ to get label vector, $\beta_2$
Combine $\beta_1$ and $\beta_2$ to get final cluster label vector, $F$, where $F$ should be in order of $X$
return $F$
```

The type-2 fuzzy membership value, $\hat{\mu}_{li}$, and typicality value, t_{li}, for each of the points inside the *Lower Approximation* area, are each set to 1 due to the confidence in the clustering of those points. This shows that there is no ambiguity regarding how those points should be classified. Therefore, relative importance factor a can be set to 1 and P can be redefined as:

$$\mathsf{P} = \sum_{l=1}^{K} \sum_{x_i \in \underline{B}(C_l)} D(c_l, x_i), \;\; 1 \leq i \leq n$$

On the other hand, Q is calculated using type-2 fuzzy membership value, $\hat{\mu}_{li}$ & typicality value, t_{li} due to uncertainty regarding the clustering of points within *Boundary Region*.

Definition 10 formulates the process of cluster center update. In each iteration, Equation 3.16 produces the new computed cluster center. If both *Lower Approximation* and *Boundary Region* are non-empty then the cluster center is computed as ($\mathtt{f_{LW}} \times \mathsf{G}_{LW} + \mathtt{f_{BN}} \times \mathsf{G}_{BN}$), where $\mathtt{f_{LW}}$ and $\mathtt{f_{BN}}$ refer the relative importance similar to the Equation 3.15. G_{LW} has contribution to *Lower Approximation* behaving similar to hard K-Means method, because the points inside *Lower Approximation* have the complete certainty about the clustering. Similarly, G_{BN} has the contribution to the *Boundary Region*. As a result, while computing the cluster's mean, both probabilistic and probabilistic measures are considered for points within the *Boundary Region*. Iteratively carrying out this method is done so till the algorithm converges.

Definition 10 *If each cluster (C_l) is represented by a center (c_l), a crisp Lower Approximation ($\underline{B}(C_l)$) and a fuzzy boundary ($BN(C_l)$), then the center should be modified to optimize Equation 3.15.*

$$c_l = \begin{cases} \mathtt{f_{LW}} \times \mathsf{G}_{LW} + \mathtt{f_{BN}} \times \mathsf{G}_{BN}, & \\ \qquad\qquad if \underline{B}(C_l) \neq \emptyset, BN(C_l) \neq \emptyset & \\ \mathsf{G}_{LW}, & if \underline{B}(C_l) \neq \emptyset, BN(C_l) = \emptyset \\ \mathsf{G}_{BN}, & if \underline{B}(C_l) = \emptyset, BN(C_l) \neq \emptyset \end{cases} \tag{3.16}$$

where,

$$\mathsf{G}_{LW} = \frac{\sum_{x_i \in \underline{B}(C_l)} x_i}{\mid \underline{B}(C_l) \mid},$$

$$\mathsf{G}_{BN} = \frac{\sum_{x_i \in BN(C_l)} \{a\,(\hat{\mu}_{li})^{\eta_1} + b\,(t_{li})^{\eta_2}\} x_i}{\sum_{x_i \in BN(C_l)} \{a\,(\hat{\mu}_{li})^{\eta_1} + b\,(t_{li})^{\eta_2}\}}$$

The RPT2FCM-RF algorithm has two steps. Executing RPT2FCM in the first stage results in some crisp and rough points in accordance with the procedure outlined above. RF classifier is used to categorize the rough points in the second stage using the training set of crisp points. With the selection of K random points as K cluster centers, RPT2FCM begins processing followed by the computation of the composite membership value, $u_{li} = \{a(\hat{\mu}_{li}) + b(t_{li})\}$ for all the n data points. Considering u_{li} and u_{hi} are highest two calculated membership values of x_i among all K clusters, where $1 \leq l, h \leq K \;\; and \;\; h \neq l$, the difference between u_{li} and u_{hi} confirms the classification of x_i. If the difference exceeds the Δ threshold, then x_i is unambiguously and precisely categorized into $\underline{B}(C_l)$. As a result, the value of $\hat{\mu}_{li}$ and t_{li} corresponding to the point are set to 1 for lth cluster, whereas 0 for rest of the points in all other clusters. Additionally, if the difference is less than Δ, then rough set approach allows the point to belong to multiple clusters because of the ambiguity. Therefore, the point that belongs to *Upper Approximation* of more than one cluster does not belong to any *Lower Approximation*. The values of $\hat{\mu}_{li}$ & t_{li} are also unaltered. The threshold value, Δ, is defined as the median of the difference of highest and second highest fuzzy membership of all the points. Mathematically, $\Delta = \text{median}_{i=1,2,\ldots,n}\,(u_{li} - u_{hi})$. RPT2FCM computes the new cluster center after classifying all n points in each iteration, and the procedure continues so long as there is a difference between two successive values of the objective function. Algorithm 5 and 6 outline the detail steps. Following the execution of RPT2FCM, certain points that fall into the *Lower Approximation* area are referred to as crisp points, and the remaining points that fall into the *Upper Approximation* region of multiple clusters are referred to as rough points. These rough points are now classified via RF integration. Crisp points are certain of the cluster to which they belong. As a result, these crisp points are employed as the training set and the rough points constitute the test set due to their ambiguity about clusters. Algorithm 6 describes the steps of RPT2FCM integrated RF.

3.3 Experimental Results of Brain Image Segmentation

MR brain images are used for experiment. The image data set, input parameters for the various algorithms, and visualization of the outcomes are all described in this section.

3.3.1 MR Images of Brain

The Brainweb database [1] is used to download the MR pictures of the experiment subject's brain. This website offers a Simulated Brain Database (SBD) as a solution to the validation issue. A set of accurate MRI data volumes created by an MRI simulator are included in the SBD. The neuroimaging community can utilize these data to assess how well different image analysis techniques perform in situations

[1] http://www.bic.mni.mcgill.ca/brainweb

where the truth value is known. SBD includes two different kinds of simulated brain images, including normal and images of multiple sclerosis (MS) lesions. The pictures come in three bands: proton density (pd)-weighted, T1-weighted, and T2-weighted. The images have an intensity non-uniformity of 20%, 3% noise (compared to the brightest tissue), and a slice thickness of 1mm. There are 181 different Z planes available for the 217×181 images. For the normal brain image data, the images of the Z planes Z10, Z60, and Z130 are taken into account because the significant differences with their neighbouring planes/images are visible when compared to the ground truth. For the brain image data of MS lesions, the experiment employs images from the Z planes Z40, Z90, and Z140. The images of typical brains show nine classes altogether. However, the number of classes varies along the various Z planes. Background, CSF, Grey Matter, White Matter, Fat, Muscle/Skin, Skin, Skull, and Glial Matter are the nine categories. In contrast, brain imaging of MS lesions include the 10, 11, and 9 classes in addition to the 9 classes of a normal brain, Connective, and MS Lesion classes. The number of classes also varies similarly along the Z planes. On the Brainweb website, more and actual information are available about this.

3.3.2 Performance Metric

The effectiveness of the proposed technique has been evaluated quantitatively using the Minkowski Score (MS) [168], the Davies-Bouldin (DB) index [185], the Adjusted Rand Index (ARI) [169], and the Percentage of Correct Pair (%CP) [186], while visual evaluation is done through post-clustering images. The range of the ARI value is [0, 1], with 1 denoting the ideal clustering. Similarly, 100% in the case of %CP denotes the best clustering outcome. On the other hand, lower values of DB and MS suggest clusters that are compact and well-separated. Chapter 2 provides a brief explanation of ARI, MS, and %CP, respectively, while the Davies-Bouldin (DB) index is briefly discussed below.

Davies-Bouldin Index

The ratio of the total of within-cluster dispersion to between-cluster separation is known as the Davies-Bouldin (DB) [185] index. It can be represented as below.

$$DB = \frac{\sum_{l=1}^{K} \Re_l}{K} \tag{3.17}$$

where

$$\Re_l = \max_{h,h \neq l} \{ \frac{S_l + S_h}{\delta_{lh}} \} \tag{3.18}$$

and

$$S_l = \frac{\sum_{x \in V_l} D(v_l, x)}{\mid V_l \mid}, \quad \delta_{lh} = D(v_l, v_h) \tag{3.19}$$

3.3.3 Input Parameters

MR brain scans are used to run each algorithm until it reaches its convergence. After conducting experiments or reading the pertinent literature, the following input parameters have been used for the various algorithms.

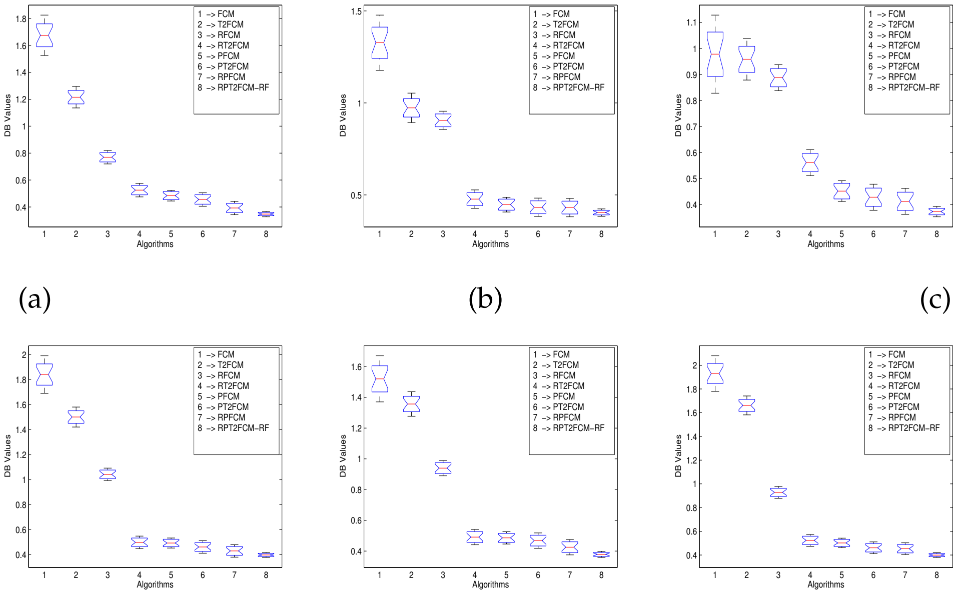
- The fuzzy exponent $\eta_1 = \eta_2 = 2$
- $\mathtt{f}_{\mathtt{LW}} = 0.95$ and $\mathtt{f}_{\mathtt{BN}} = 0.05$
- Relative importance for probabilistic membership, a and $b = 0.5$
- Number of tree for RF = 1000

Table 3.1: For T1-weighted MR images of the normal brain in the Z10, Z60, and Z130 planes and the brain with Multiple Sclerosis Lesions in the Z40, Z90, and Z140 planes, average values for DB, ARI, MS, and %CP are reported

MR image of Normal brain						MR image of Multiple Sclerosis Lesions brain					
Planes	**Method**	**DB**	**ARI**	**MS**	**%CP**	**Planes**	**Method**	**DB**	**ARI**	**MS**	**%CP**
Z10	FCM	1.67489	0.36605	1.16699	70.29437	Z40	FCM	1.84126	0.27202	1.56644	63.59808
	T2FCM	1.21539	0.48060	0.91123	75.76889		T2FCM	1.50153	0.43987	1.22089	69.88697
	RFCM	0.76959	0.51357	0.77633	80.50688		RFCM	1.04249	0.44263	0.94346	75.39501
	RT2FCM	0.52524	0.60654	0.75437	81.75127		RT2FCM	0.49897	0.49747	0.89684	75.86770
	PFCM	0.48431	0.61703	0.70967	84.27496		PFCM	0.49381	0.56431	0.89430	75.93521
	PT2FCM	0.45562	0.70565	0.69543	84.76307		PT2FCM	0.46231	0.56578	0.89335	76.00030
	RPFCM	0.39249	0.71545	0.66821	85.95491		RPFCM	0.43055	0.56757	0.88087	76.48895
	RPT2FCM-RF	**0.34728**	**0.72413**	**0.63657**	**86.81429**		**RPT2FCM-RF**	**0.39837**	**0.56866**	**0.87984**	**76.71584**
Z60	FCM	1.32728	0.31439	1.37257	66.15524	Z90	FCM	1.52053	0.36936	1.29081	68.57810
	T2FCM	0.97311	0.42733	1.08817	71.72751		T2FCM	1.35708	0.58822	0.97144	72.20337
	RFCM	0.90521	0.43440	0.85199	77.45942		RFCM	0.93981	0.59502	0.83110	78.17381
	RT2FCM	0.47711	0.53032	0.84889	77.49413		RT2FCM	0.49099	0.59835	0.79788	79.79943
	PFCM	0.44709	0.57696	0.82701	78.31299		PFCM	0.48588	0.65998	0.79566	80.06097
	PT2FCM	0.43273	0.60107	0.81837	78.56830		PT2FCM	0.46819	0.67129	0.76112	81.17503
	RPFCM	0.43118	0.61018	0.81782	78.68468		RPFCM	0.42546	0.71683	0.70717	84.56897
	RPT2FCM-RF	**0.40411**	**0.63399**	**0.79226**	**80.32397**		**RPT2FCM-RF**	**0.37871**	**0.72323**	**0.67567**	**85.69057**
Z130	FCM	0.97788	0.39482	0.88457	76.15057	Z140	FCM	1.92979	0.33857	0.82221	78.35538
	T2FCM	0.95836	0.52746	0.87678	76.94244		T2FCM	1.66141	0.37748	0.81132	79.00357
	RFCM	0.88758	0.53309	0.83251	78.02391		RFCM	0.92826	0.49363	0.74446	82.75581
	RT2FCM	0.56129	0.55323	0.75801	81.38834		RT2FCM	0.52363	0.65447	0.69162	84.76426
	PFCM	0.45195	0.77444	0.60517	87.53932		PFCM	0.50214	0.80633	0.61964	87.45310
	PT2FCM	0.42846	0.82911	0.59477	87.69597		PT2FCM	0.46086	0.80979	0.49312	91.13539
	RPFCM	0.41271	0.84178	0.54298	88.87874		RPFCM	0.45298	0.86298	0.46806	92.01342
	RPT2FCM-RF	**0.37337**	**0.84759**	**0.47813**	**91.87031**		**RPT2FCM-RF**	**0.39967**	**0.86772**	**0.42103**	**93.53784**

3.3.4 Results and Discussion

Utilizing four distinct cluster validity indices, including DB [185] index, ARI [169], MS [168] and %CP [186], the final clustering results are assessed. Table 3.1 lists the average values of these metrics for all of the brain MR images produced using



(a) (b) (c)

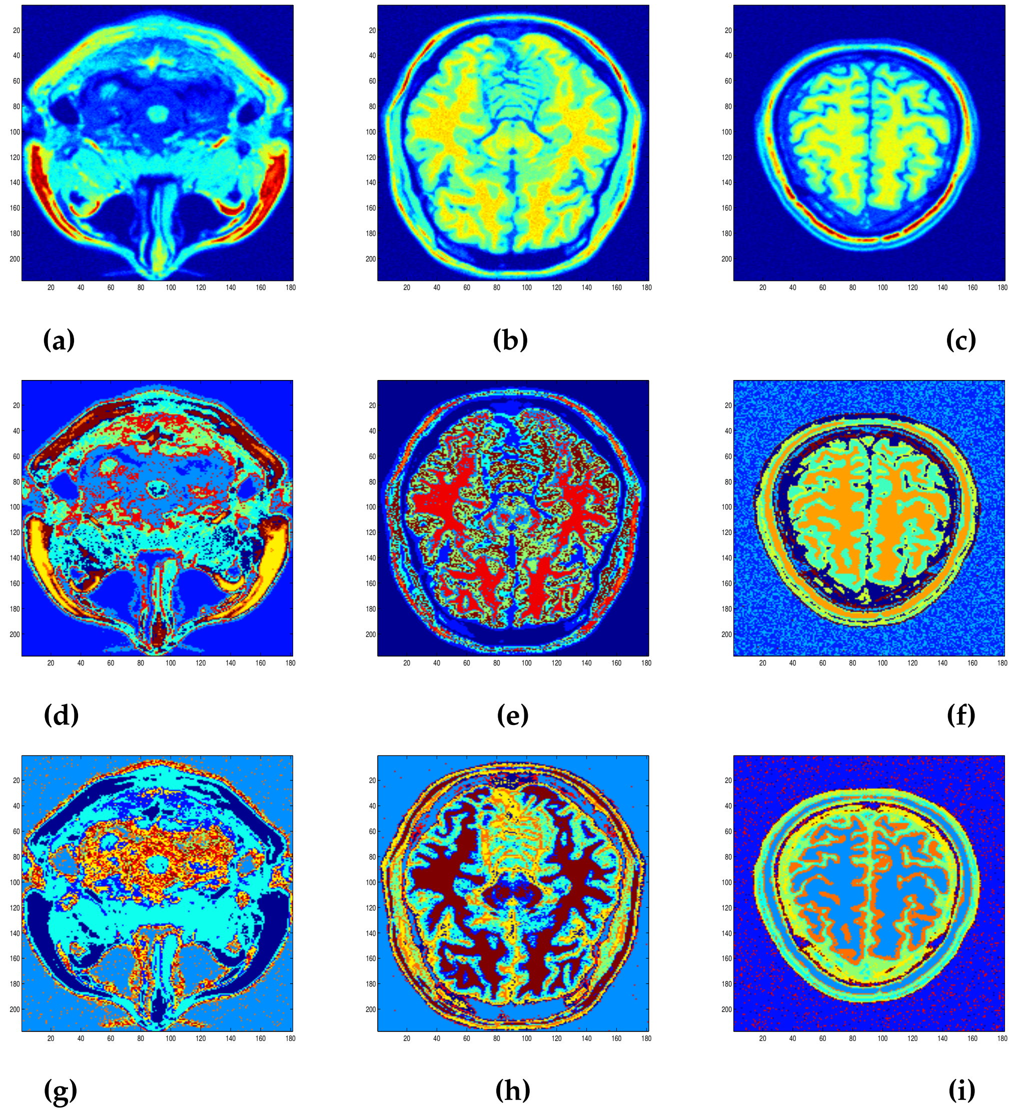
(d) (e) (f)

Figure 3.1: Boxplot of DB values of different clustering algorithms for T1-weighted MR brain images of the Normal Brain in (a) Z10, (b) Z60, (c) Z130 planes and with Multiple Sclerosis Lesions brain in (d) Z40, (e) Z90, (f) Z140 planes

Table 3.2: Execution times in seconds for all images using various methods

	MR image of Normal brain			MR image of Multiple Sclerosis Lesions brain		
Method	**Z10**	**Z60**	**Z130**	**Z40**	**Z90**	**Z140**
FCM	08.001	08.024	08.042	08.980	09.945	07.993
T2FCM	08.121	08.128	08.160	09.131	10.067	08.126
RFCM	13.580	13.985	13.738	15.448	16.867	12.901
RT2FCM	13.674	13.574	13.714	15.576	16.711	13.558
PFCM	15.427	15.382	15.457	17.449	19.058	15.442
PT2FCM	15.519	15.458	15.572	17.539	19.142	15.516
RPFCM	24.224	24.609	24.236	25.859	29.691	24.432
RPT2FCM-RF	**25.544**	**26.357**	**24.973**	**26.971**	**30.875**	**25.337**

different techniques over a period of 20 runs. The ideal clustering result is shown by the maximum values of ARI and %CP and the minimum values of DB and MS. It is abundantly obvious from the findings in the table that RPT2FCM-RF outperforms all other approaches. Additionally, Figure 3.1 shows the boxplots of the DB index for various approaches. As shown in the image, the resulting plot



(a) (b) (c)

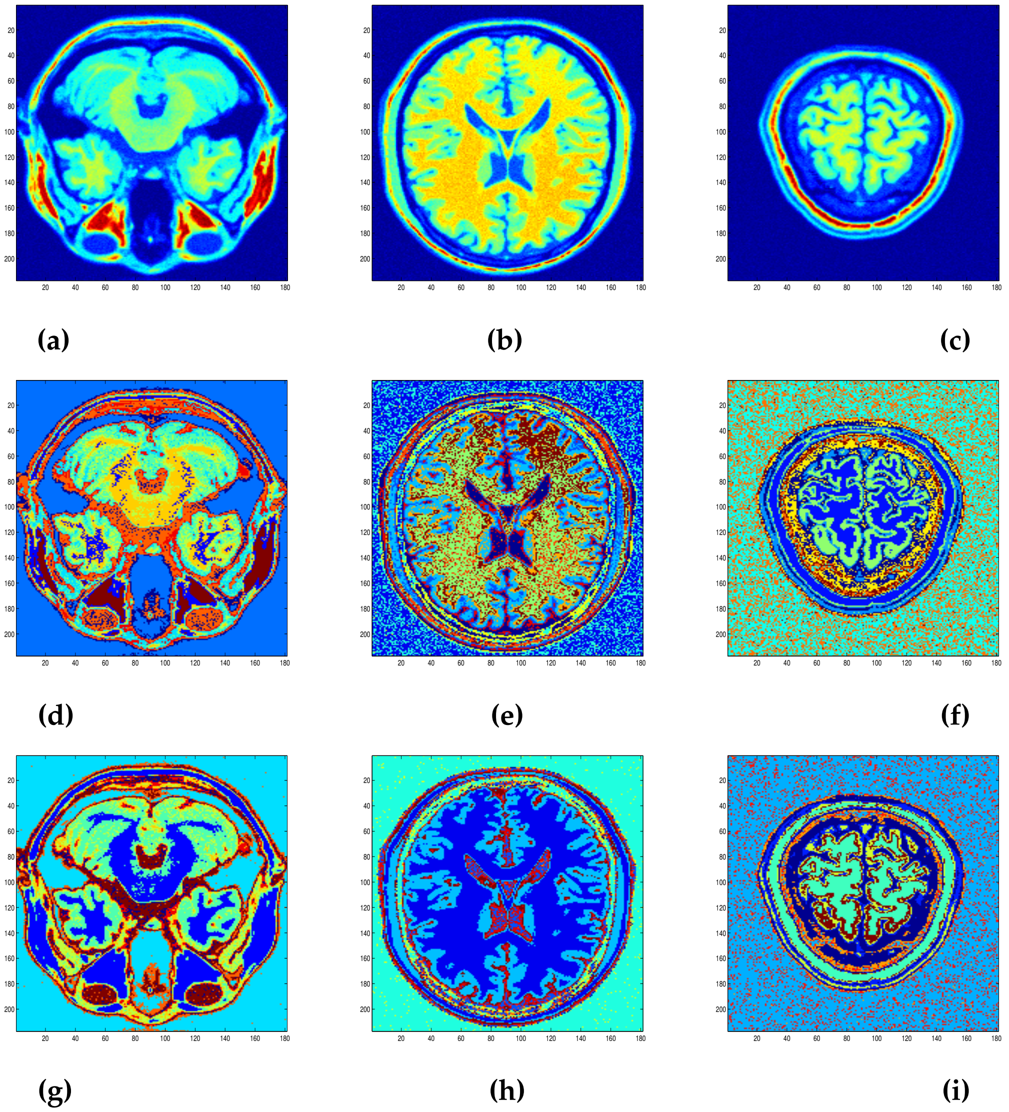
(d) (e) (f)

(g) (h) (i)

Figure 3.2: (a), (b) and (c) are original T1-weighted MR brain images of the Normal Brain in Z10, Z60 and Z130 planes whereas (d), (e) and (f) are corresponding segmented images produced by FCM and (g), (h) and (i) are corresponding segmented images produced by RPT2FCM-RF

generated by RPT2FCM-RF has been greatly improved and has a limited range for all of the images.

The execution time for the rough fuzzy clustering technique is usually longer than that of the comparable fuzzy version due to the computing of the many parameters needed for rough measures. Because of the fuzzy membership matrix,



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 3.3: (a), (b) and (c) are original T1-weighted MR brain images of Multiple Sclerosis Lesions in Z40, Z90 and Z140 planes whereas (d), (e) and (f) are corresponding segmented images produced by FCM and (g), (h) and (i) are corresponding segmented images produced by RPT2FCM-RF

fuzzy clustering algorithms also take a little longer to run. The execution times for all methods across all photos are listed in a Table 3.2 in seconds. Every method has been built in Matlab and run on a machine with an Intel Core i5-2410M CPU running at 2.30 GHz, 4GB of RAM, and Windows 7 as the operating system. Higher system configuration can boost execution speed. Through the use of parallel processing, it has additional room for advancement.

3.3.5 Statistical Significance Test

The average values of DB, ARI, MS, and %CP obtained by different algorithms throughout 20 consecutive runs for all MR brain images are shown in Table 3.1. The clustering approach that has been proposed demonstrates superior performance in terms of the different indices when compared to the collective performance of all prior techniques. In order to ascertain the statistical significance of the recommended algorithm's superior performance, it is necessary to conduct a statistical significance test. The statistical significance of the clustering solutions in this chapter has been evaluated using a t-test [171] at a significance level of 5%. Eight groups have been formed for each image, corresponding to the seven algorithms: (1. FCM, 2. T2FCM, 3. RFCM, 4. RT2FCM, 5. PFCM, 6. PT2FCM, 7. RPFCM, and 8. RPT2FCM-RF). Each group is composed of the DB values that are created by 20 consecutive rounds of the corresponding algorithm.

Table 3.3 presents the *p-values* generated by the t-test for the purpose of simultaneously comparing two groups, namely the group associated with RPT2FCM-RF and a group determined by an alternative technique. The alternative hypothesis posits that there exists a statistically significant difference in the mean values between the two groups, while the null hypothesis suggests that there is no significant difference in the DB values of the two groups. The table only comprises *p-values* that exhibit statistical significance at a level of 0.05 (5% significance level). As an example, the statistical significance of the t-test conducted between the RPT2FCM-RF and FCM approaches for the normal brain imaging in the Z10 plane is shown by a *p-value* of 8.53e-14, which is found to be considerably less than the predetermined significance threshold of 0.05. Table 3.3 presents comparable findings for the Z60 and Z130 planes of normal brain pictures, as well as the Z40, Z90, and Z140 planes of brain images with Sclerosis Lesions. This analysis presents persuasive evidence indicating that the improved DB values achieved by the proposed methodology were not due to random chance, but rather represent statistically significant evidence contradicting the null hypothesis. When comparing RPT2FCM-RF against other algorithms and images, comparable results are consistently achieved, hence emphasizing the evident superiority of the approach.

3.4 Worst case Time Complexity Analysis

Time complexity of RPT2FCM-RF depends on RPT2FCM as well as Random Forest method. The computation of fuzzy membership matrix, typicality matrix and computed matrix for whole data set, and searching highest two computed membership (u_{li}) values for each point take time of $O(4Knm)$. Computation of new K centers

Table 3.3: Results of the t-test for various planes of MR brain imaging. In the test, *p-values* are generated by contrasting RPT2FCM-RF with other algorithms

	MR image of Normal brain			MR image of Multiple Sclerosis Lesions brain		
Method	Z10	Z60	Z130	Z40	Z90	Z140
FCM	8.53e-14	1.82e-12	5.77e-11	4.19e-14	3.06e-13	2.53e-14
T2FCM	9.30e-16	3.42e-14	2.70e-14	1.16e-16	3.31e-16	3.61e-17
RFCM	2.84e-17	6.56e-18	5.25e-18	7.46e-19	2.47e-18	4.13e-18
RT2FCM	3.69e-14	2.86e-11	2.38e-14	2.94e-12	1.30e-12	6.19e-13
PFCM	1.96e-17	9.05e-14	1.41e-15	3.29e-16	1.35e-16	1.91e-16
PT2FCM	1.70e-12	7.46e-09	1.83e-10	6.99e-11	6.88e-12	9.33e-11
RPFCM	6.17e-10	9.79e-09	1.38e-09	4.11e-09	5.05e-10	2.25e-10

needs time of $O(Knm)$. Considering other activities, the total worst case time complexity of RPT2FCM methods is $O(5Knm + Kn)$ for each iteration. Moreover, the worst case time complexity of Random Forest is $O(\mathsf{T}nm\log(n))$. Therefore, the worst case time complexity of RPT2FCM-RF is $O(5Knm + Kn + \mathsf{T}nm\log(n)) \approx O(Knm + \mathsf{T}nm\log(n))$ for each iteration.

3.5 Conclusions

The present chapter introduces innovative clustering strategies that are grounded on type-2 fuzzy set theory, probabilistic method, and rough set theory [187]. In order to address the limitations of conventional FCM, the RPT2FCM approach incorporates a probabilistic methodology. Additionally, the use of type-2 fuzzy sets and rough set theories is utilized to effectively manage the inherent uncertainties and ambiguity present within the data sets. The proposed clustering technique generates rough and crisp points. To enhance the overall quality of the final clustering result, these rough points are classified using the Random Forest algorithm. The task of segmenting magnetic resonance (MR) brain pictures has been effectively tackled via the use of the RPT2FCM-RF methodology. The allocation of points to different clusters is determined by using the Euclidean distance metric. The evaluation of clustering quality is conducted using four metrics: DB, ARI, MS, and %CP. According to the results of the t-test conducted at a significance level of 5%, it can be concluded that the RPT2FCM-RF approach has shown superior statistical performance compared to earlier methods. The experimental findings demonstrate that the suggested methodology exhibits superior performance compared to current methodologies, both in terms of statistical analysis and visual assessment. Furthermore, the proposed technique effectively accomplishes the task of categorizing MR brain images into distinct tissue classifications. The concept of the suggested methodologies encompasses a wide array of potential applications. An instance of the integration of rough set and type-2 fuzzy set theory may be seen in the clustering of breast cancer and wine data [188]. Next chapter elucidates the prospective enhancement of rough fuzzy-based category clustering via the utilization of an evolutionary technique. This improvement is particularly relevant in the context of addressing the problem whereby the existing method exhibits a proclivity for being ensnared in local optimum solutions.

4 Evolutionary Semi-Supervised Rough Fuzzy Clustering

4.1 Introduction

It is discussed in Chapter 2 that clustering [15, 30, 31, 51, 52, 55, 189, 190] is a commonly used method in the field of data mining. However, when dealing with categorical data, it is important to note that the attribute values sometimes lack a natural ordering, and the intrinsic distance measure is not readily accessible. The Partitioning Around Medoids (PAM) based algorithms such as *K*-Medoids (KMdd) [41, 191], as well as the fuzzy-based approach Fuzzy *K*-Modes (FKMd) [50] and its derivatives, have been specifically designed for the purpose of grouping categorical data. Nevertheless, even with the use of fuzzy set theory, the challenge of effectively dealing with the certain separate dimensions of knowledge imperfection, namely indiscernibility, uncertainty, and vagueness [192], remains.

The rough set framework is introduced by Pawlak in the early 1980s [193] as a means of approximating concepts in the presence of ambiguity. A rough set consists of two sets referred to as the *Lower Approximation* and the *Upper Approximation*. The difference area between the *Upper Approximation* and *Lower Approximation* is referred to as the *Boundary Region*. The intricacies of Rough set have been expounded upon in Chapter 2. The use of this concept is prevalent in several domains, including but not limited to inductive reasoning, pattern recognition, learning algorithms, and taxonomy. In the context of rough clustering, a data point is often assigned a membership value of 1 if it unequivocally belongs to a certain cluster. Alternatively, the point may be situated in the boundary region where it exhibits partial membership to numerous clusters. Consequently, the points that reside inside boundary regions may be conceptualized as points that exist within the intersecting sections of two or more clusters. In their publication, Lingras *et al.* [157] introduced the concept of rough sets and used it to develop a method known as Rough *K*-Means. Several further clustering approaches on rough set have been documented in the literature [135, 158, 167, 194–202]. These techniques mostly focus on numerical datasets. In contrast, the combination of Rough set and Fuzzy *K*-Modes (FKMd) might potentially address the challenges posed by indiscernibility, uncertainty, and vagueness more effectively, as elaborated in Chapter 2. Additionally, the efficacy of FKMd is contingent upon the selection of starting cluster modes, a factor that often results in unsatisfactory solutions. Therefore, it is possible to use metaheuristic techniques, such as Simulated Annealing (SA) [54] and Genetic Algorithm (GA) [53], to address clustering as a fundamental optimization issue [51, 52]

In order to acknowledge the aforementioned information, the development of Rough Fuzzy *K*-Modes (RFKMd) has been undertaken in Chapter 2. Nevertheless, it has been noted that the method encounters the issue of local optima due to the random selection of starting cluster modes. Therefore, this chapter aims to expand on the concept of RFKMd by introducing two clustering techniques: Simulated Annealing based Rough Fuzzy *K*-Modes (SARFKMd) and Genetic Algorithm based Rough Fuzzy *K*-Modes (GARFKMd). The approach used in this study is based on Simulated Annealing, which involves the evaluation of a new solution by gradually altering a single answer using a local move. In contrast, the Genetic Algorithm-based approach assesses novel solutions by amalgamating two distinct solutions with the aim of attaining the global answer. The RFKMd approach has been devised to include rough and fuzzy sets in order to enhance the analysis of indiscernibility and vagueness in categorical datasets. In order to maximize the advantages of examining the solution space via local movements and utilizing the solution space on a global scale, two distinct metaheuristic approaches, namely Simulated Annealing and Genetic Algorithm, are used in conjunction with RFKMd. Both solutions use the cluster mode encoding algorithm. Every clustering approach has the ability to produce clusters that consist of a group of center points as well as outlying points. Hence, in order to get the ultimate clustering outcome, the peripheral points are categorized using the well recognized machine learning approach known as Random Forest (RF) [27]. In this approach, the Random Forest (RF) algorithm is first trained using center points, after which periphery points are identified. The clustering approaches that have been developed, in conjunction with the Random Forest (RF) algorithm, are referred to as SARFKMd-RF and GARFKMd-RF. However, it has been noted that there is variation in the cardinality of the training and testing sets, namely in the number of central and peripheral points generated by each approach. Therefore, this observation has also served as a motivation for us to introduce a comprehensive methodology known as Integrated Rough Fuzzy Clustering utilising Random Forest (IRFKMd-RF). The roughness measure of the clustering results generated by RFKMd, SARFKMd, and GARFKMd is calculated in this context. Following that, three categories known as *best central points*, *semi-best central points* and *pure peripheral points* are established.

Subsequently, Random Forest is used to classify semi-best core and pure peripheral locations via the utilization of multi-phase learning. The proposed techniques, SARFKMd-RF, GARFKMd-RF, and IRFKMd-RF, are compared with Tabu Search based Fuzzy *K*-Modes (TSFKMd) [155], Min-Min-Roughness (MMR) [135], Rough *K*-Medoids (RKMdd) [195], Genetic Algorithm based Average Normalized Mutual Information Clustering (G-ANMI) [136], Categorical Data Clustering (CDC), and Cluster Ensemble (CE) method known as ccdByEnsemble [106]. Additionally, widely used state-of-the-art methods such as *K*-Modes (KMd) [42], *K*-Medoids (KMdd) [41,191], Fuzzy *K*-Modes (FKMd) [50], and Average Linkage (AL) [31] are also included in the comparison. The experimental findings are shown via the evaluation of several cluster validity indices and visual graphs on a total of six synthetic and five real-life categorical datasets. In order to determine the statistical significance of the data, a combination of parametric and non-parametric tests are conducted. Specifically, the independent two-sample one-tailed *t*-test [171] and the Friedman test [203,204] are used.

4.2 Evolutionary Semi-Supervised Rough Fuzzy Clustering

In Chapter 2, the Rough Fuzzy *K*-Modes (RFKMd) Clustering approach is discussed, which integrates the advantages of rough and fuzzy sets [193] in order to address the challenges posed by indiscernibility and vagueness in categorical datasets. This section provides a description of the suggested clustering approaches, namely Simulated Annealing based Rough Fuzzy *K*-Modes (SARFKMd) and Genetic Algorithm based Rough Fuzzy *K*-Modes (GARFKMd). Next, this paper examines the incorporation of RFKMd, SARFKMd, and GARFKMd into the Random Forest algorithm. In this section, we provide a comprehensive exposition of the Integrated Rough Fuzzy Clustering utilizing Random Forest (IRFKMd-RF) methodology. The complete explanation of many equations used in the RFKMd framework, namely in the SARFKMd, GARFKMd, and IRFKMd methodologies, may be found in Chapter 2.

4.2.1 Simulated Annealing based Rough Fuzzy *K*-Modes Clustering

Simulated annealing (SA) is an optimization approach that has shown effective applications in solving a diverse range of combinatorial optimization problems [54]. The aforementioned concept has resemblance to the principle of statistical mechanics, which posits that a state of matter characterized by low energy may alone be sustained at very low temperature conditions. During the annealing process, the temperature is first elevated and thereafter decreased gradually to a significantly low value. This ensures that sufficient time is allocated at each temperature to avoid the occurrence of any unstable states. SA has been used in many applications for clustering difficulties [33]. The objective of Simulated Annealing based Rough Fuzzy K-Modes (SARFKMd) is to achieve a global optimal solution by mitigating the occurrence of local minima. This approach incorporates the principles of rough and fuzzy sets, which are concerned with addressing the issues of indiscernibility and vagueness. SARFKMd incorporates the primary stages of SA, namely the encoding of configuration, energy function calculation, and perturbation, in its methodology. These stages aim to identify appropriate partitions by minimizing the energy function, denoted as J_{RF}, as defined in Equation 2.13.

The encoding process in SARFKMd involves representing the configuration

Algorithm 7 Steps of SARFKMd

```
Input:
    $X$ : dataset
    $\eta$ : fuzzy exponent
    $\epsilon$ : small real threshold value between [0,1]
    $K$ : number of cluster
    $\omega_{low}$ : the relative weight for the Lower Approximation of rough clustering, $0 < \omega_{low} < 1$
    $(T_{max})$ : maximum temperature
    $(T_{min})$ : minimum temperature
    $maxItr$ : maximum iteration
    $g$ : a small real number inside the closed interval [0,1], $0 < g < 1$
Output: $[\mu]$ where, $1 \le l \le K$ and $1 \le i \le n$

Randomly choose $K$ data objects from the dataset to be encoded in the configuration of the $K$ cluster mode, denoted as $S_{init}$.
Set $t = T_{max}$
Compute energy value $F(S_{init})$ using Equation 2.13 for $S_{init}$
Compute $\mu_{li}$ for all $n$ objects using Equation 2.12
Classify objects using Algorithm 2 and update $[\mu]$
Compute new modes using Equation 2.16
Update each modes in $S_{init}$ with new modes
repeat
    repeat
        $S_{per} \leftarrow Perturb(S_{init})$
        Compute $\mu_{li}$ for all $n$ objects using Equation 2.12
        Cluster objects using Algorithm 2 and update $[\mu]$
        Compute energy value $F(S_{per})$ using Equation 2.13 for $S_{per}$.
        if $(F(S_{per}) - F(S_{init})) < 0$ then
            $S_{init} \leftarrow S_{per}$ and $F(S_{init}) \leftarrow F(S_{per})$
        else
            $S_{init} \leftarrow S_{per}$ & $F(S_{init}) \leftarrow F(S_{per})$ having probability $\left\{e^{-\left[\frac{(F(S_{per})-F(S_{init}))}{t}\right]}\right\}$
        end if
    until $maxItr$ is reached
    Set $t \leftarrow t \times g$, where $0 < g < 1$
until $t < T_{min}$
return $[\mu]$ where, $1 \le l \le K$ and $1 \le i \le n$
```

as a series of K distinct cluster modes, which are selected randomly from a given dataset, denoted as X. The K modes are represented as a row vector in the arrangement shown in Example 1. Considering that each object has m characteristics, the length of the configuration may be determined as $m \times K$. In this representation, the first m positions correspond to the first mode, the subsequent m positions correspond to the second mode, and so on. The fuzzy membership matrix is produced for every object using Equation 2.12. Afterwards, following the RFKMd technique, the object x_i is clustered using Algorithm 2, and the fuzzy membership value of x_i is assigned appropriately. Following the clustering process, the energy function (J_{RF}) associated with a particular configuration is updated using Equation 2.13, while the modes are updated using Equation 2.16.

Example 3: Let $K = 3$ and $m = 2$. The configuration encodes K modes as $[v_{11}\ v_{12}\ v_{21}\ v_{22}\ v_{31}\ v_{32}]$, where, $v_1 = [v_{11}, v_{12}]$, $v_2 = [v_{21}, v_{22}]$ and $v_3 = [v_{31}, v_{32}]$ are the modes respectively.

During the perturbation process, a randomly selected attribute, denoted as $\mathcal{A}_j$, of the mode encoded in the configuration is chosen. Afterwards, it is substituted with a randomly chosen attribute value from the associated categorical domain, $DOM(\mathcal{A}_j)$, of that attribute. The purpose of this requirement is to guarantee that an attribute of any given mode only consists of values that belong to a set of categorical domain values. In this manner, the perturbation of a given configuration results in the generation of a distinct configuration. If the energy of the new configuration ($F(S_{per})$) is lower than that of the present configuration ($F(S_{init})$), the new configuration is deemed acceptable. The acceptance of a new configuration is determined by a probability value, which is calculated as $e^{-\left[\frac{(F(S_{per})-F(S_{init}))}{t}\right]}$, where t represents the present temperature of the simulated annealing process. The procedure operates according to a predetermined temperature. Typically, the process starts with an initial high temperature, which subsequently diminishes by a factor g, denoting a positive real value bounded between 0 and 1. The procedure continues until it reaches the minimal temperature. The most optimal configuration seen so far is the SARFKMd solution. The algorithm SARFKMd is outlined in Algorithm 7.

4.2.2 Genetic Algorithm based Rough Fuzzy K-Modes Clustering

An approach to randomized search called Genetic Algorithm (GA) is based on the idea of natural evolution. Despite the fact that the situation is intricate and multifaceted, GA does an excellent job of locating a close to ideal answer. The primary stages of a Genetic Algorithm (GA) include *chromosome representation*, *population initiation*, *selection*, *crossover*, and *mutation*. This chapter presents a novel approach called Genetic Algorithm based Rough Fuzzy K-Modes (GARFKMd) clustering, which leverages the search capabilities of Genetic Algorithms (GA).

The chromosome is encoded in a comparable manner in SARFKMd and GARFKMd, according to their respective configurations. Every chromosome serves as a possible solution. The fitness of a chromosome serves as an indicator of the degree of quality of the solution it represents. In this study, the objective function J_{RF} is used, as described in Equation 2.13. The primary goal is to minimize the objective function J_{RF} in order to provide an optimum clustering solution. The retrieval of modes stored in a chromosome is the first step, followed by the computation of the fuzzy membership matrix, denoted as μ_{li}, and the updating of modes using Equations 2.12, 2.13 and 2.16.

Subsequently, the genetic procedures of *selection*, *crossover* and *mutation* are implemented. The tournament selection approach is used in this context. In this scenario, a pair of chromosomes is randomly selected and compared based on their respective fitness values. The selection of the dominant chromosome is a crucial factor in determining the optimal strategies for preparing the population for subsequent generations. The aforementioned process is iterated PS times. The tournament selection approach ensures the preservation of genetic diversity by providing an equal probability for each chromosome to be selected. Chromosomes are employed for crossover operation after selection. To produce new offspring, a traditional single point crossover with probability P_{cr} is carried out. Then, with probability P_{mu}, mutation is carried out. The gene location that will change is chosen at random for this purpose. Then, a different value drawn at random from the relevant categorical domain replaces the categorical value of that position. The best possible chromosome is segregated from the general population and preserved in a distinct domain as a component of the elitism technique. The optimal chromosome is ultimately determined by the final cluster modes it comprises. All of the processes iterate for a predetermined maximum number of generations, denoted as *maxGen*. The algorithm shown in Algorithm 8 illustrates the many stages involved in GARFKMd.

Algorithm 8 Steps of the GARFKMd

```
Input:
    $X$ : dataset
    $\eta$ : fuzzy exponent
    $\epsilon$ : small real threshold value between [0,1]
    $K$ : number of cluster
    $\omega_{low}$ : the relative weight for the Lower Approximation of rough clustering, $0 < \omega_{low} < 1$
    $maxGen$ : maximum generation
    $PS$ : population size
    $P_{cr}$ : crossover probability
    $P_{mu}$ : mutation probability
Output: $[\mu]$ $where,\ 1 \le l \le K\ and\ 1 \le i \le n$

For the $K$ cluster mode, choose $K$ random items from the dataset to encode as chromosomes. Let $v_l$ denotes cluster mode for $l = 1, 2, \ldots, K$.
Generate initial population of size $PS$.
repeat
    Calculate $\mu_{li}$ for all $n$ objects using Equation 2.12.
    Cluster objects using Algorithm 2 to obtain updated $[\mu]$.
    Calculate fitness value using Equation 2.13 for each chromosome in population.
    Modify each chromosome in population with new mode computed using Equation 2.16.
    Perform selection activity using tournament selection strategy.
    Perform crossover activity with probability $P_{cr}$.
    Perform mutation activity with probability $P_{mu}$.
until $maxGen$ is reached
return $[\mu]$ $where,\ 1 \le l \le K\ and\ 1 \le i \le n$
```

Algorithm 9 Steps of Integration with Random Forest

```
Input:
    $X$ : dataset
    $K$ : number of cluster
    $[\mu]$ : fuzzy membership matrix
Output: $F$ : final class label vector of $X$

Use $[\mu]$ and build dataset of central points, $\mathbb{L} = \{x_i \in \underline{B}(V_l) \mid 1 \le l \le K\ \ and\ \ 1 \le i \le n\}$ and corresponding cluster label vector, $\beta$
Classify $\mathbb{L}^* = (X - \mathbb{L})$ using Random Forest, trained by $\mathbb{L}$ and $\beta$ to get label vector, $\beta^*$
Combine $\beta$ and $\beta^*$ to obtain final cluster label vector, $F$, where $F$ should be in order of $X$
return $F$
```

4.2.3 Integrated Rough Fuzzy Clustering using Random Forest

To reach the final clustering result, the integration of RFKMd, SARFKMd, and GARFKMd is performed individually using the Random Forest (RF) algorithm first. The subsequent procedure involves the development of an Integrated Rough Fuzzy K-Modes clustering technique known as IRFKMd-RF. This approach utilizes the results obtained from the RFKMd, SARFKMd, and GARFKMd algorithms. Consequently, after the execution of the RFKMd technique, the fuzzy membership matrix, denoted as $[\mu]$, is used to generate central and peripheral points. *Central points* refer to the categorized points that are assigned to a certain cluster. In contrast, outlying points are classified as belonging to the boundary region of numerous clusters. The *peripheral points* exhibit an inherent lack of discernibility. Therefore, the Random Forest classifier is trained using *Central points* in order to

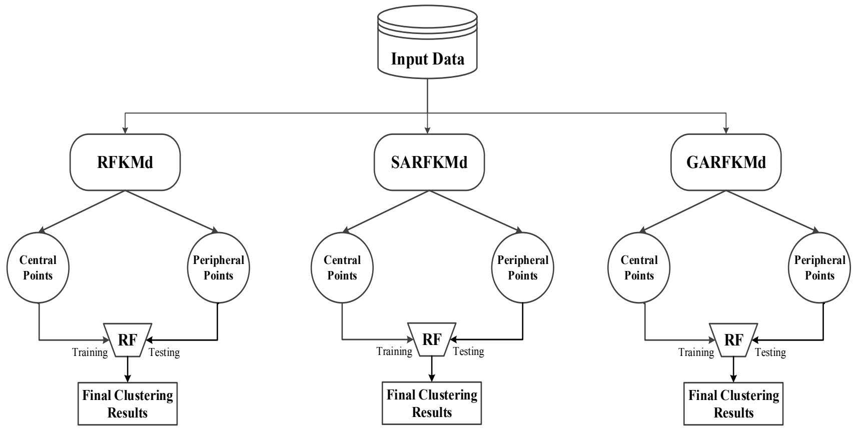


Figure 4.1: Schematic flow diagram of individual Rough Fuzzy Clustering integrated with Random Forest

categorize *peripheral points*. Likewise, SARFKMd and GARFKMd both provide *central* and *peripheral points*, hence using RF in a comparable manner to categorize *peripheral points* for each respective approach. The integration of RFKMd, SARFKMd, and GARFKMd with Random Forest results in the approaches being referred to as RFKMd-RF, SARFKMd-RF, and GARFKMd-RF, respectively. It should be noted that RFKMd-RF, SARFKMd-RF, and GARFKMd-RF adhere to the same procedures as described in Algorithm 9, which employs the Random Forest technique. Furthermore, the procedure is visually elucidated in Figure 4.1..

The procedural instructions for the IRFKMd-RF method are presented in Algorithm 10. In this context, the symbol $\mathscr{N}$ is used to indicate the total number of clustering methods used in the integration process. On the other hand, $\mathscr{C}_l^j$ and $\mathscr{L}_j$ are utilized to designate the lth cluster and the set of objects included in the *Lower Approximation* of all clusters, respectively. These values are calculated using the jth method. The cluster label vector β_j represents the cluster labels of the items in $\mathscr{L}_j$. The values of β_j are selected from the cluster label space $\{1, 2, \ldots, K\}$. Additionally, the tuple $(\mathscr{L}_j, \beta_j)$ is used to denote the data and their related cluster labels. The formal definition of $\mathscr{L}_j$ is as follows.

$$\mathscr{L}_j = \bigcup_{1 \le l \le K} \underline{B}(\mathscr{C}_l^j) \tag{4.1}$$

Algorithm 10 Steps of the IRFKMd-RF

```
Input:
    $X$ : dataset
    $\eta$ : fuzzy exponent
    $\epsilon$ : small real threshold value between [0,1]
    $K$ : number of cluster
    $\omega_{low}$ : the relative weight for the Lower Approximation of rough clustering, $0 < \omega_{low} < 1$
    $\mathscr{N}$ : number of methods
Output: $F$ : final class label vector of $X$

 1: Initialize $\mathscr{N}$ with number of methods // Here, $\mathscr{N} = 3$
 2: Compute $\mathscr{L}_1, \mathscr{L}_2, \ldots, \mathscr{L}_{\mathscr{N}}$ and $\beta_1, \beta_2, \ldots, \beta_{\mathscr{N}}$ using Algorithms 2, 7 & 8 and Equation 4.1 with the values
    of $X, \eta, \epsilon, K$ and $\omega_{low}$
 3: $[\mathscr{L}\beta]_{sup} = \{(\mathscr{L}_j, \beta_j) : 1 \le j \le \mathscr{N}\}$
 4: Compute Roughness measure, $\rho_j$ for $1 \le j \le \mathscr{N}$ using Equation 2.21
 5: $\mathscr{L} \leftarrow \bigcup_{1 \le j \le \mathscr{N}} \mathscr{L}_j$
 6: Select $(\mathscr{L}_{best}, \beta_{best}) \in [\mathscr{L}\beta]_{sup}$, where $best \leftarrow \arg\min_{1 \le j \le \mathscr{N}} \{\rho_j\}$
 7: Relabel $\hat{\mathscr{L}} = (\mathscr{L} - \mathscr{L}_{best})$ to get label vector, $\hat{\beta}$, using Random Forest classifier trained by $(\mathscr{L}_{best}, \beta_{best})$
 8: Combine $\beta_{best}$ and $\hat{\beta}$ to get $\beta$ where $\beta$ should be in order of $\mathscr{L}$
 9: Classify $\mathscr{L}^* = (X - \mathscr{L})$ to get label vector, $\beta^*$, using Random Forest classifier trained by $(\mathscr{L}, \beta)$
10: Combine $\beta$ and $\beta^*$ to get final cluster label vector, $F$, where $F$ should be in order of $X$
11: return $F$
```

The RFKMd algorithm may have difficulties in escaping local optimum solutions. However, the SARFKMd and GARFKMd algorithms have the potential to mitigate this issue. It is important to note that these alternative algorithms may provide different solutions. Therefore, it is possible for the cardinality of the training and testing sets to differ. Therefore, the computation of the roughness measure is dependent on the outcomes of each clustering technique and is then shown in step 5 of the procedure. This roughness measure is used to choose the set of *best central points* ($\mathscr{L}_{best}$) in step 6 of the method, from the three sets of central points ($\mathscr{L}_j$, where $j = 1, \ldots, 3$) generated by RFKMd, SARFKMd and GARFKMd. Subsequently, the sets of *semi-best central points* ($\hat{\mathscr{L}}$) and *pure peripheral points* ($\mathscr{L}^*$) are generated and presented in steps 7 and 9 of the procedure. The cluster labels are denoted as β_{best}, $\hat{\beta}$, and β^*. The *semi-best central points* do not belong to the set of *best central points*, but they are members of either one or two other sets of *central points*. On the other hand, *pure peripheral points* are neither members of the best *central points* nor the *semi-best central points*. The Random Forest algorithm uses three distinct and non-overlapping sets. During the first phase, the Random Forest (RF) algorithm is trained using the *best central points*. This training process aims to categorize the *semi-best central points* in step 7. Subsequently, these two sets are mixed to train RF and categorize *pure peripheral points* in stages 8 and 9. Therefore, this phenomenon is referred to as *multi-phase learning*. Ultimately, the tenth step culminates in the generation of the ultimate outcome denoted as F. The representation of IRFKMd-RF may also be seen in Figure 4.2.

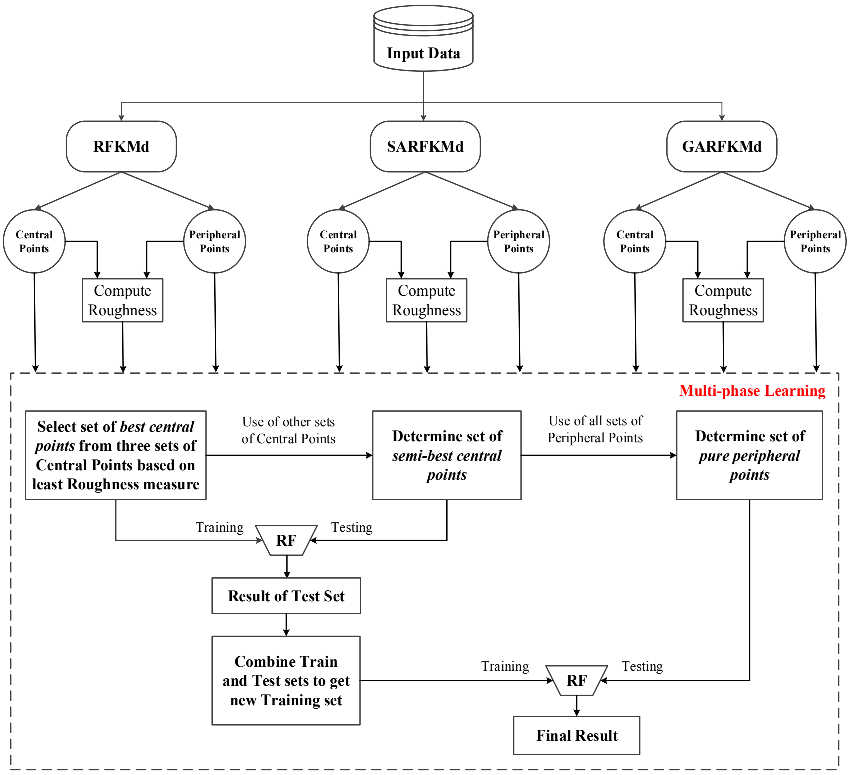

Input Data
RFKMd
SARFKMd
GARFKMd
Central Points
Peripheral Points
Compute Roughness
Multi-phase Learning
Select set of *best central points* from three sets of Central Points based on least Roughness measure
Use of other sets of Central Points
Determine set of *semi-best central points*
Use of all sets of Peripheral Points
Determine set of *pure peripheral points*
Training
Testing
RF
Result of Test Set
Combine Train and Test sets to get new Training set
Final Result

Figure 4.2: Schematic diagram of Integrated Rough Fuzzy Clustering using Random Forest

4.3 Experimental Results

The performance of proposed methods has been assessed on six synthetic datasets[1] (*"Cat-100-8-3"*, *"Cat-250-15-5"*, *"Cat-300-8-3"*, *"Cat-300-15-5"*, *"Cat-500-20-10"* and *"Cat-1000-7-7"*) and five real life datasets[2] (*"Congressional Votes"*, *"Zoo"*, *"Soybean"*, *"Heart"* and *"Mushroom"*)

4.3.1 Datasets

Chapter 2 has described about the synthetic datasets *"Cat-100-8-3"*, *"Cat-250-15-5"*, *"Cat-300-8-3"*, *"Cat-300-15-5"*, *"Cat-500-20-10"* and *"Cat-1000-7-7"* and the real life datasets *"Congressional Votes"*, *"Zoo"*, *"Soybean"*, *"Heart"* respectively. The other real life dataset, *"Mushroom"* is described in Chapter 5.

[1] http://www.datgen.com

[2] http://www.ics.uci.edu/~ mlearn/MLRepository.html

4.3.2 Roughness Measure

A rough set is a computational technique used for the analysis of imperfect data. In accordance with the rough set theory, two quantitative metrics [193] have been established to evaluate a set, inside the approximation space. The terms used to describe these characteristics are accuracy and roughness that are defined by Equations 2.20 and 2.21 in Chapter 2. The roughness value ranges between 0 and 1, with a value of 0 indicating the absence of peripheral (rough) points after the clustering process. In this thesis chapter, the roughness measure for clustering technique j is denoted by ρ_j.

4.3.3 Distance Measure

The computation of the distance between two categorical objects follows the methodology described in the work of Talbi [166] and is further elaborated in Chapter 2 of this thesis. Let us consider two categorical objects, $x_i = [x_{i1}, x_{i2}, \ldots, x_{im}]$ and $x_j = [x_{j1}, x_{j2}, \ldots, x_{jm}]$, are having m categorical attributes. The distance measure between x_i and x_j , $D(x_i, x_j)$, can be defined as follows.

$$D(x_i, x_j) = \sum_{k=1}^{m} \delta(x_{ik}, x_{jk}) \tag{4.2}$$

where

$$\delta(x_{ik}, x_{jk}) = \begin{cases} 0 & \text{if } x_{ik} = x_{jk} \\ 1 & \text{if } x_{ik} \neq x_{jk} \end{cases} \tag{4.3}$$

Table 4.1: Cardinality of the best of 20 sets of central points produced by RFKMd, SARFKMd and GARFKMd separately over 20 runs for different datasets at ω_{low}=0.95

DataSet	**RFKMd**	**SARFKMd**	**GARFKMd**
Cat-100-8-3	40	50	50
Cat-250-15-5	124	125	127
Cat-300-8-3	138	150	150
Cat-300-15-5	132	150	150
Cat-500-20-10	202	240	248
Cat-1000-7-7	399	499	500
Soybean	23	23	25
Zoo	50	73	80
Heart	108	135	135
Votes	180	217	217
Mushroom	6302	6964	6966

Table 4.2: Average roughness measures over 20 runs of different clustering methods for different datasets

	RFKMd		SARFKMd		GARFKMd	
Dataset	ω_{low}=0.65	ω_{low}=0.95	ω_{low}=0.65	ω_{low}=0.95	ω_{low}=0.65	ω_{low}=0.95
Cat-100-8-3	0.07110	0.06520	0.05959	0.05398	0.05591	0.05060
Cat-250-15-5	0.09004	0.08102	0.05266	0.04409	0.05046	0.04234
Cat-300-8-3	0.18438	0.14427	0.12790	0.08980	0.11106	0.07496
Cat-300-15-5	0.05235	0.04721	0.04958	0.04470	0.04387	0.03924
Cat-500-20-10	0.03946	0.03054	0.03668	0.02821	0.03624	0.02821
Cat-1000-7-7	0.08296	0.06785	0.07074	0.05639	0.06040	0.04680
Soybean	0.09138	0.07214	0.08666	0.06838	0.07211	0.05479
Zoo	0.16326	0.09611	0.13548	0.07169	0.12591	0.06547
Heart	0.21145	0.10358	0.16916	0.06668	0.16314	0.06606
Votes	0.09215	0.07203	0.08859	0.06948	0.06849	0.05038
Mushroom	0.08245	0.07002	0.07853	0.05741	0.05314	0.04127

Table 4.3: Average values of objective function, J_{RF}, over 20 runs of SARFKMd for different values of T_{max}, T_{min} and g

Parameters			Objective function values for different datasets										
T_{max}	T_{min}	g	Cat-100-8-3	Cat-250-15-5	Cat-300-8-3	Cat-300-15-5	Cat-500-20-10	Cat-1000-7-7	Soybean	Zoo	Heart	Votes	Mushroom
100	0.03	0.9	137.58	844.64	285.80	1317.08	1056.52	917.11	58.24	18.07	350.73	260.23	39623.38
		0.8	160.28	857.18	290.22	1334.14	1056.12	916.76	58.21	21.54	172.10	272.07	39608.30
		0.7	140.43	841.69	299.79	1280.64	1082.63	939.77	59.67	21.49	434.15	256.11	40602.46
	0.02	0.9	133.33	835.46	276.31	1315.58	1014.60	880.72	55.92	17.90	236.38	256.11	38051.27
		0.8	143.49	845.68	285.24	1284.88	1030.82	894.80	56.82	17.01	484.25	272.07	38659.57
		0.7	132.36	834.14	290.30	1360.73	1053.34	914.35	58.06	21.61	450.20	236.06	39503.95
	0.01	0.9	139.54	844.64	245.55	1319.75	937.46	813.76	51.67	17.14	291.45	256.11	35158.19
		0.8	147.18	831.22	277.12	1329.30	1071.52	930.13	59.06	18.92	238.70	236.06	40185.81
		0.7	154.79	863.90	279.20	1295.71	1083.63	940.64	59.73	20.60	593.98	257.58	40639.92
90	0.03	0.9	115.63	836.27	284.05	1322.52	969.24	841.34	53.42	16.88	129.73	236.06	36349.95
		0.8	122.35	831.32	263.74	1281.89	1066.61	925.87	58.79	17.82	403.23	269.48	40001.74
		0.7	139.53	843.89	286.67	1374.24	1109.73	963.30	61.17	18.03	370.73	291.47	41619.01
	0.02	0.9	139.79	843.73	286.85	1323.12	1014.63	880.75	55.93	16.88	300.15	242.91	38052.39
		0.8	141.04	840.14	291.91	1290.37	1001.85	869.65	55.22	19.00	257.93	272.07	37572.94
		0.7	140.41	851.02	296.84	1360.27	1066.10	925.43	58.76	22.57	204.53	236.06	39982.65
	0.01	0.9	141.33	845.57	284.00	1286.23	966.98	839.39	53.30	17.14	189.40	256.11	36265.36
		0.8	145.05	835.51	269.46	1310.08	980.68	851.28	54.05	17.42	487.28	272.07	36779.24
		0.7	141.40	847.58	299.31	1314.29	1049.98	911.43	57.87	21.65	410.25	236.06	39378.16
80	0.03	0.9	139.37	850.61	269.87	1334.56	973.12	844.71	53.64	19.00	351.60	236.06	36495.55
		0.8	142.63	840.88	286.69	1274.27	985.22	855.22	54.31	19.80	491.65	257.58	36949.51
		0.7	151.74	842.96	296.02	1376.47	1067.90	926.98	58.86	20.90	309.58	272.07	40050.00
	0.02	0.9	137.91	822.10	272.88	1268.50	1066.21	925.52	58.77	18.95	363.60	256.11	39986.62
		0.8	145.44	848.43	298.85	1365.83	1038.17	901.18	57.22	19.09	430.75	272.07	38935.33
		0.7	138.74	858.91	286.75	1346.07	1036.85	900.04	57.15	21.58	590.83	269.48	38885.67
	0.01	0.9	125.36	838.71	273.05	1334.98	966.98	839.39	53.30	17.08	427.18	256.11	36265.36
		0.8	118.03	843.57	292.72	1293.32	1066.18	925.50	58.77	18.13	166.60	256.11	39985.84
		0.7	136.83	858.20	288.51	1340.08	1028.98	893.20	56.72	23.46	222.90	256.11	38590.54

4.3.4 Performance Metrics

Cluster validity metrics are used to identify the partitioning that most accurately corresponds to the underlying data [168, 205–208]. In this chapter, we have employed various evaluation metrics including Minkowski Score [168], Percentage of Correct Pair (%CP), Davies-Bouldin (DB) Index [206], Dunn's Index [207], and Xie-Beni (XB) Index [208] to assess the performance of KMd, KMdd, RKMdd, FKMd, AL, TSFKMd, MMR, G-ANMI, ccdByEnsemble, RFKMd-RF, SARFKMd-RF, GARFKMd-RF, and IRFKMd-RF.

Table 4.4: Average values of objective function, J_{RF}, over 20 runs of GARFKMd for different values of crossover and mutation probabilities

Parameters		Objective function values for different datasets										
P_{cr}	P_{mu}	Cat-100-8-3	Cat-250-15-5	Cat-300-8-3	Cat-300-15-5	Cat-500-20-10	Cat-1000-7-7	Soybean	Zoo	Heart	Votes	Mushroom
0.9	0.5	131.82	821.54	295.25	1350.22	3446.89	1413.34	54.96	17.01	603.57	272.07	73174.00
	0.4	152.34	815.35	319.64	1318.23	3451.62	1587.48	54.93	19.01	627.74	272.07	59568.00
	0.3	145.53	829.70	300.65	1336.58	3487.59	1428.03	56.21	20.69	604.94	272.07	64332.00
	0.2	149.09	831.95	327.04	1362.01	3438.11	1494.50	58.88	20.56	300.95	294.24	65214.00
0.8	0.5	133.61	822.51	292.97	1349.56	3445.38	1378.66	54.94	20.59	513.95	272.07	61318.00
	0.4	124.22	814.78	269.82	1296.26	3380.58	1226.79	49.63	15.28	197.53	256.11	58572.00
	0.3	135.44	824.46	287.41	1352.08	3384.92	1380.39	55.96	18.07	322.15	272.07	59582.00
	0.2	147.04	828.55	295.80	1361.17	3433.91	1419.30	57.04	21.84	605.57	272.07	60432.00
0.7	0.5	158.25	817.69	296.64	1338.09	3450.62	1489.79	54.94	21.58	512.93	256.11	81354.00
	0.4	155.94	834.28	308.11	1349.24	3430.19	1562.84	57.87	22.57	429.90	272.07	61360.00
	0.3	147.36	820.35	321.20	1325.12	3496.22	1548.08	56.19	21.76	616.90	272.07	62138.00
	0.2	145.96	826.30	288.16	1337.74	3503.2	1368.66	54.94	23.70	456.52	272.07	73372.00
0.6	0.5	118.88	815.78	292.43	1258.17	3389.01	1446.99	56.35	20.64	545.03	272.07	77918.00
	0.4	151.76	823.82	262.29	1334.50	3417.15	1503.28	54.29	22.50	529.33	272.07	60818.00
	0.3	124.47	828.24	300.66	1352.55	3449.90	1357.73	57.02	22.44	644.97	272.07	62254.00
	0.2	145.27	808.69	311.94	1367.55	3487.55	1695.60	54.29	22.75	545.70	272.07	62413.00

4.3.5 Parameter Settings

The KMd, KMdd, RKMdd, FKMd, and RFKMd algorithms are iteratively run until convergence is achieved and a final solution is obtained. On the other hand, the SARFKMd and GARFKMd algorithms are executed for a maximum number of iterations. A sensitivity analysis has been conducted to determine the impact of several important factors. Table 4.2 displays the roughness measures of RFKMd, SARFKMd, and GARFKMd, indicating that these measures are lower when ω_{low} is set to 0.95 in comparison to other values. A comparable examination has also been conducted for SARFKMd to determine the optimal values for maximum temperature (T_{max}), minimum temperature (T_{min}) and g. The table shown in Table 4.3 displays the average values of the objective function, denoted as J_{RF}, across several datasets. These values are provided for different combinations of T_{max}, T_{min}, and g. The superiority of the objective value is evident when considering the values of $T_{max} = 100$, $T_{min} = 0.01$, and $g = 0.9$, as shown in Table 4.3. The maximum number of iterations, denoted as *maxItr*, has been set to 100. In the context of GARFKMd, the parameters of crossover probability (P_{cr}) and mutation probability (P_{mu}) have significant importance. The average values of the objective function, denoted as J_{RF}, for various combinations of crossover and mutation probabilities are shown in Table 4.4. The outcomes of several datasets are documented in Table 4.4. During the process of doing sensitivity analysis, a single parameter is altered while all other parameters remain unchanged. For instance, the fixed value of P_{cr} is 0.8, when calculating the objective value, we consider other values for P_{mu}, equal to 0.5, 0.4, 0.3 and 0.2. The process has been replicated for various values of the critical parameter P_{cr}. Based on the study conducted, it has been noted that a higher objective value is attained when the crossover probability (P_{cr}) is set to 0.8 and the mutation probability (P_{mu}) is set to 0.4. In other scenarios where the values of P_{cr} and P_{mu} differ, the objective values exhibit inconsistency. Therefore, for our experiment, we have chosen $P_{cr} = 0.8$ and $P_{mu} = 0.4$. The population size and number of generations, denoted as N and $maxGen$ respectively, for the GARFKMd algorithm are both set to 20 and 100. The fuzzy exponent, denoted as η, is fixed at a value of 2, while the number of trees for the random forest (RF) is set to 1000. It should be noted that the input parameters used in this study are consistent with those reported in many academic sources. The input parameters for the TSFKMd, MMR, and G-ANMI techniques are used in a manner consistent with the references cited in [135, 136, 155]. The developed techniques were performed using Matlab on a machine equipped with an Intel Core i5-2410M CPU working at a frequency of 2.30 GHz. The machine had a RAM capacity of 4GB and was running the Windows 7 operating system.

Table 4.5: Average values of cluster validity indices over 20 runs of different methods for Cat-100-8-3

Dataset	Algorithms	MS	%CP	DB	Dunn	XB
Cat-100-8-3	KMd	0.88202	76.18283	1.55023	0.34375	0.49857
	KMdd	0.81578	78.59142	1.54940	0.35937	0.48683
	RKMdd	0.80571	79.12486	1.52661	0.37459	0.45612
	FKMd	0.79713	79.91267	1.34474	0.38125	0.38548
	AL	0.76037	81.13111	1.33768	0.38472	0.37838
	TSFKMd	0.73736	83.11590	1.30957	0.38500	0.36647
	MMR	0.55900	88.29480	1.30128	0.38610	0.36129
	G-ANMI	0.50471	89.44180	1.27649	0.47930	0.35846
	ccdByEnsemble	0.43344	92.98410	1.26189	0.48021	0.35024
	RFKMd-RF	0.42667	93.35910	1.25826	0.48805	0.34513
	SARFKMd-RF	0.32132	96.00240	1.23661	0.50030	0.25701
	GARFKMd-RF	0.30221	96.19360	1.23271	0.50112	0.24023
	IRFKMd-RF	**0.18621**	**98.68243**	**1.22209**	**0.57000**	**0.23008**

4.4 Comparative Experiments

The performance of RFKMd-RF, SARFKMd-RF, GARFKMd-RF, and IRFKMd-RF has been assessed by comparing them with ccdByEnsemble [106], GG-ANMI [136], MMR [135], TSFKMd [155], AL [31], FKMd [50], RKMdd [195], KMdd [41] and KMd [42] both quantitatively and visually. It should be noted that previous research, as mentioned in [136], has shown the superiority of G-ANMI over k-ANMI [110], TCSOM citehe05a, and Squeezer [94] approaches. Therefore, these methods are not used in the compression process. The following sections provide a description of the quantitative data obtained from performance metrics, visual evaluations, and statistical significance analysis of the clustering results.

Table 4.6: Average values of cluster validity indices over 20 runs of different methods for Cat-250-15-5

Dataset	Algorithms	MS	%CP	DB	Dunn	XB
Cat-250-15-5	KMd	0.77524	80.55590	1.57929	0.31333	0.90637
	KMdd	0.70170	84.53930	1.57056	0.31423	0.88769
	RKMdd	0.68419	84.91246	1.55438	0.33746	0.79812
	FKMd	0.62713	87.21280	1.51009	0.34422	0.61401
	AL	0.64678	86.40430	1.55093	0.34333	0.62431
	TSFKMd	0.55319	88.60240	1.49414	0.34456	0.40829
	MMR	0.52321	89.09280	1.48540	0.34497	0.40254
	G-ANMI	0.49988	90.29160	1.39056	0.34556	0.39556
	ccdByEnsemble	0.46062	91.33010	1.38551	0.34648	0.37007
	RFKMd-RF	0.45331	91.52440	1.36601	0.34844	0.36563
	SARFKMd-RF	0.43622	92.73350	1.35553	0.35421	0.19240
	GARFKMd-RF	0.42138	93.44630	1.35332	0.35662	0.19003
	IRFKMd-RF	**0.01003**	**99.01750**	**1.27536**	**0.36009**	**0.18839**

Table 4.7: Average values of cluster validity indices over 20 runs of different methods for Cat-300-8-3

Dataset	Algorithms	MS	%CP	DB	Dunn	XB
Cat-300-8-3	KMd	0.77440	80.63140	1.31852	0.10093	0.92764
	KMdd	0.53983	88.91650	1.30775	0.12865	0.88453
	RKMdd	0.50734	89.31249	1.29881	0.15743	0.65214
	FKMd	0.37300	94.27590	1.28235	0.22431	0.40619
	AL	0.49087	90.56740	1.29709	0.22190	0.42616
	TSFKMd	0.34283	94.90220	0.88493	0.23000	0.35883
	MMR	0.31699	96.15320	0.88117	0.23740	0.35164
	G-ANMI	0.20322	98.10197	0.84352	0.38211	0.30339
	ccdByEnsemble	0.15203	98.24510	0.83004	0.40576	0.29735
	RFKMd-RF	0.11662	98.38296	0.83325	0.41992	0.28066
	SARFKMd-RF	0.04943	98.75362	0.82778	0.44368	0.22040
	GARFKMd-RF	0.03296	99.27892	0.81711	0.51002	0.22028
	IRFKMd-RF	**0.00880**	**99.85007**	**0.80025**	**0.55300**	**0.22001**

4.4.1 Quantitative Results

Table 4.1 presents the cardinality of the optimal set of central points generated by RFKMd, SARFKMd, and GARFKMd algorithms in 20 independent runs. Table 4.2 presents the average roughness measure of three approaches across several datasets, as it is used for computational purposes. The reported values are based on 20 iterations. It is beneficial to determine the optimal clustering solution among these three methods. A lower roughness value indicates a more optimal grouping outcome. In addition, it is crucial to calculate several cluster validity metrics in order to assess the quality of the clustering solution. The clustering outcomes of RFKMd-RF, SARFKMd-RF, GARFKMd-RF, and IRFKMd-RF are evaluated [209] by calculating the average values of MS, %CP, DB, Dunn, and XB scores over 20

Table 4.8: Average values of cluster validity indices over 20 runs of different methods for Cat-300-15-5

Dataset	Algorithms	MS	%CP	DB	Dunn	XB
Cat-300-15-5	KMd	0.84465	77.51644	1.85937	0.53231	0.89795
	KMdd	0.82538	78.20756	1.84665	0.53333	0.81557
	RKMdd	0.76438	81.03480	1.84002	0.53664	0.78429
	FKMd	0.74558	82.68290	1.83035	0.54334	0.60884
	AL	0.73566	83.51650	1.81940	0.59000	0.51572
	TSFKMd	0.69895	84.76120	1.80776	0.59450	0.50332
	MMR	0.64432	86.63930	1.79420	0.59510	0.49115
	G-ANMI	0.57669	87.91220	1.74331	0.59580	0.47340
	ccdByEnsemble	0.54571	88.69250	1.73458	0.59945	0.42349
	RFKMd-RF	0.52243	89.11250	1.72113	0.60013	0.36403
	SARFKMd-RF	0.45315	91.62320	1.71724	0.61006	0.20642
	GARFKMd-RF	0.43633	92.52150	1.69553	0.61068	0.20616
	IRFKMd-RF	**0.36464**	**94.75140**	**1.68273**	**0.62030**	**0.20115**

Table 4.9: Average values of cluster validity indices over 20 runs of different methods for Cat-500-20-10

Dataset	Algorithms	MS	%CP	DB	Dunn	XB
Cat-500-20-10	KMd	0.94438	70.98247	1.76794	0.63287	0.96647
	KMdd	0.86241	76.99439	1.72158	0.63459	0.72045
	RKMdd	0.84725	77.50423	1.71983	0.63483	0.71285
	FKMd	0.81305	78.92458	1.71409	0.63501	0.71037
	AL	0.81351	78.85427	1.70473	0.64496	0.65284
	TSFKMd	0.75597	81.42737	1.68393	0.64697	0.61349
	MMR	0.71532	84.20744	1.66954	0.65305	0.55395
	G-ANMI	0.67155	85.82147	1.65317	0.65740	0.36421
	ccdByEnsemble	0.65886	86.11850	1.63952	0.65795	0.24418
	RFKMd-RF	0.60995	87.66137	1.62109	0.65839	0.10481
	SARFKMd-RF	0.60662	87.70133	1.61753	0.65841	0.10465
	GARFKMd-RF	0.59384	87.85153	1.56453	0.66841	0.10433
	IRFKMd-RF	**0.51739**	**89.12540**	**1.55376**	**0.67229**	**0.10332**

runs for both synthetic and real-life datasets. These findings are then shown in Tables 4.5 to 4.14. The optimal outcomes are shown by the highest values of %CP, Dunn, and the lowest values of MS, DB, and XB, which are highlighted in bold. We have used a set of five distinct indices in our study due to their established prominence in the existing body of research [168, 206–208]. Furthermore, these indices provide varying computational interpretations for assessing the quality of clusters. The classification of peripheral points is performed by using Random Forest classifiers in three different methods: RFKMd-RF, SARFKMd-RF, and GARFKMd-RF. The central points are generated by each technique used as a training set for Random Forest, serving the intended goal. The cardinality of the best set of central points generated by various approaches is shown in Table 4.1. Each method was executed 20 times, and the results are shown individually for each

Table 4.10: Average values of cluster validity indices over 20 runs of different methods for Cat-1000-7-7

Dataset	Algorithms	MS	%CP	DB	Dunn	XB
Cat-1000-7-7	KMd	0.94989	70.41380	1.81197	0.28571	0.73542
	KMdd	0.86194	77.14890	1.76012	0.28771	0.72147
	RKMdd	0.83473	77.98436	1.72119	0.28792	0.70468
	FKMd	0.81150	78.99542	1.66367	0.28825	0.69765
	AL	0.80323	79.17529	1.56327	0.29755	0.69648
	TSFKMd	0.76440	80.96480	1.52459	0.31158	0.65431
	MMR	0.72522	83.64730	1.51724	0.32477	0.61277
	G-ANMI	0.66527	85.94760	1.49561	0.33548	0.59413
	ccdByEnsemble	0.63487	86.80240	1.45743	0.35149	0.50782
	RFKMd-RF	0.61556	87.42448	1.36319	0.36489	0.44012
	SARFKMd-RF	0.60439	87.75431	1.32368	0.37451	0.42355
	GARFKMd-RF	0.60054	87.82335	1.31312	0.37672	0.41522
	IRFKMd-RF	**0.45328**	**91.23950**	**1.31004**	**0.37724**	**0.40082**

Table 4.11: Average values of cluster validity indices over 20 runs of different methods for Soybean

Dataset	Algorithms	MS	%CP	DB	Dunn	XB
Soybean	KMd	0.64363	86.79560	0.92329	0.31206	0.57577
	KMdd	0.56683	88.01750	0.91765	0.32875	0.56440
	RKMdd	0.53729	88.98125	0.89413	0.39428	0.35742
	FKMd	0.39077	93.96750	0.82655	0.45388	0.28765
	AL	0.44982	91.72570	0.84563	0.43467	0.30922
	TSFKMd	0.36806	94.45920	0.78191	0.58462	0.28326
	MMR	0.33104	95.94930	0.78003	0.58662	0.28002
	G-ANMI	0.25899	96.95324	0.73288	0.60439	0.27496
	ccdByEnsemble	0.22456	97.15170	0.73004	0.61276	0.27035
	RFKMd-RF	0.20166	98.15381	0.72322	0.63331	0.26144
	SARFKMd-RF	0.03537	98.82893	0.61473	0.71506	0.24762
	GARFKMd-RF	0.03312	98.94216	0.56316	0.74586	0.21201
	IRFKMd-RF	**0.01275**	**99.85122**	**0.55113**	**0.75013**	**0.20911**

run. The IRFKMd-RF algorithm is created to handle datasets with variable cardinality. Based on the findings shown in Tables 4.5 to 4.14, it can be seen that the IRFKMd-RF clustering approach consistently outperforms the other methods in terms of producing superior outcomes. For instance, the mean scores of various methods including KMd, KMdd, RKMdd, FKMd, AL, TSFKMd, MMR, G-ANMI, ccdbyEnsemble, and IRFKMd-RF on the *Cat-300-15-5* dataset are 0.84465, 0.82538, 0.76438, 0.74558, 0.73566, 0.69895, 0.64432, 0.57669, 0.54571, 0.52243, 0.45315, and 0.43633, respectively. In the case of the remaining datasets, it has been discovered that the IRFKMd-RF methodology yields superior outcomes compared to other methodologies. The findings presented in this study demonstrate the effectiveness of the IRFKMd-RF method in handling categorical datasets.

Table 4.12: Average values of cluster validity indices over 20 runs of different methods for Zoo

Dataset	Algorithms	MS	%CP	DB	Dunn	XB
Zoo	KMd	0.68839	84.82920	0.87356	0.11871	0.30697
	KMdd	0.48911	91.23720	0.86720	0.12957	0.25493
	RKMdd	0.46816	91.25431	0.85716	0.15723	0.24781
	FKMd	0.43895	92.29820	0.76093	0.16875	0.17982
	AL	0.45844	91.49410	0.84612	0.14182	0.23293
	TSFKMd	0.42744	93.18530	0.75498	0.16998	0.17439
	MMR	0.39099	93.89980	0.74394	0.17003	0.17033
	G-ANMI	0.37686	94.00950	0.65493	0.18559	0.16500
	ccdByEnsemble	0.36995	94.30010	0.64734	0.18966	0.16401
	RFKMd-RF	0.36332	94.81010	0.63781	0.19821	0.16318
	SARFKMd-RF	0.24235	97.10640	0.61763	0.20392	0.14902
	GARFKMd-RF	0.21112	97.23830	0.55439	0.21078	0.14813
	IRFKMd-RF	**0.10341**	**98.38556**	**0.54131**	**0.23400**	**0.14778**

Table 4.13: Average values of cluster validity indices over 20 runs of different methods for Heart

Dataset	Algorithms	MS	%CP	DB	Dunn	XB
Heart	KMd	0.85638	77.42375	1.54297	0.29998	0.77449
	KMdd	0.86212	77.02512	1.56610	0.29615	0.78668
	RKMdd	0.83479	77.75187	1.48721	0.29884	0.77718
	FKMd	0.81420	78.79898	1.47803	0.30755	0.76178
	AL	0.80652	79.09247	1.39503	0.30769	0.63624
	TSFKMd	0.80234	79.29142	1.35283	0.32639	0.61659
	MMR	0.79907	79.71650	1.34285	0.33639	0.60284
	G-ANMI	0.78106	80.40680	1.20873	0.37001	0.52997
	ccdByEnsemble	0.74972	82.01450	1.19273	0.37995	0.42743
	RFKMd-RF	0.74162	82.76610	1.17121	0.38643	0.40475
	SARFKMd-RF	0.68137	85.66590	1.16877	0.38765	0.36501
	GARFKMd-RF	0.66231	85.96320	1.16301	0.38765	0.36083
	IRFKMd-RF	**0.60997**	**87.50731**	**1.16097**	**0.38773**	**0.36044**

The increased execution time of the rough fuzzy based technique may be attributed to the extra computational steps involved in calculating the parameters utilized for roughness measurements, as compared to the equivalent fuzzy version. In a similar vein, the computational time required for fuzzy set-based clustering is more than that of its crisp counterpart, mostly owing to the calculation of the fuzzy membership matrix. On average, the proposed IRFKMd-RF method requires 110.723 seconds to process the *Cat-300-15-5* dataset. In comparison, the KMd, KMdd, RKMdd, FKMd, AL, TSFKMd, MMR, G-ANMI, ccdbyEnsemble, RFKMd-RF, SARFKMd-RF, and GARFKMd-RF methods take 0.105, 0.263, 0.275, 0.109, 0.101, 36.256, 2.441, 26.602, 13.062, 5.613, 30.534, and 12.901 seconds, respectively. As anticipated, the proposed integrated rough fuzzy clustering technique

Table 4.14: Average values of cluster validity indices over 20 runs of different methods for Votes

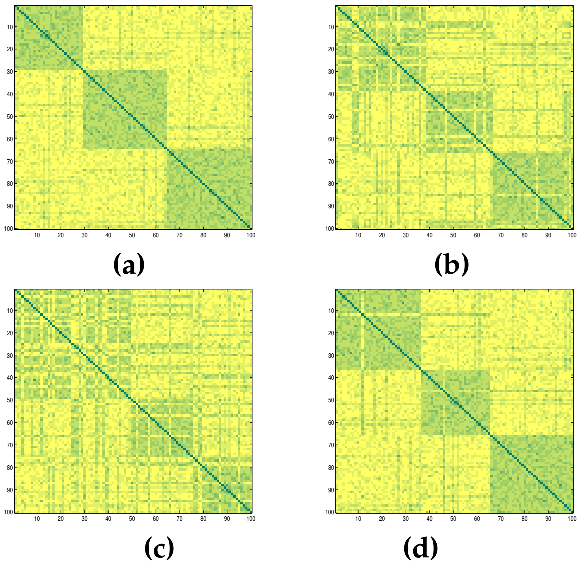
Dataset	Algorithms	MS	%CP	DB	Dunn	XB
Votes	KMd	0.75553	81.52340	0.52655	0.07691	0.26477
	KMdd	0.72507	83.91960	0.50951	0.07692	0.16872
	RKMdd	0.71498	84.25490	0.50856	0.07692	0.16824
	FKMd	0.69745	84.76290	0.50837	0.07694	0.16804
	AL	0.70432	84.50090	0.50949	0.07693	0.16870
	TSFKMd	0.68219	85.05834	0.48374	0.07696	0.16765
	MMR	0.67349	85.74890	0.48113	0.07698	0.16744
	G-ANMI	0.63072	86.83760	0.47219	0.07699	0.16584
	ccdByEnsemble	0.62115	87.33540	0.45994	0.07701	0.16514
	RFKMd-RF	0.60085	87.78221	0.45298	0.07722	0.16463
	SARFKMd-RF	0.57734	87.88143	0.45264	0.07746	0.16346
	GARFKMd-RF	0.55362	88.55130	0.45154	0.07898	0.16002
	IRFKMd-RF	**0.54303**	**88.79940**	**0.45114**	**0.07933**	**0.15534**

Table 4.15: Average values of cluster validity indices over 20 runs of different methods for Mushroom

Dataset	Algorithms	MS	%CP	DB	Dunn	XB
Mushroom	KMd	0.81998	78.22746	1.84872	0.45732	0.78431
	KMdd	0.81996	78.22794	1.83725	0.45834	0.75934
	RKMdd	0.75128	81.65730	1.75498	0.50964	0.65922
	FKMd	0.69724	84.76920	1.72576	0.52741	0.60995
	AL	0.71355	84.25600	1.73597	0.51834	0.63865
	TSFKMd	0.68428	84.89510	1.70934	0.53764	0.59346
	MMR	0.66143	85.96430	1.65731	0.54887	0.55831
	G-ANMI	0.65269	86.25760	1.61734	0.56228	0.38861
	ccdByEnsemble	0.64218	86.79860	1.59364	0.58344	0.25543
	RFKMd-RF	0.50726	89.31367	1.42983	0.62997	0.16384
	SARFKMd-RF	0.50718	89.31428	1.41864	0.63855	0.14973
	GARFKMd-RF	0.49745	90.38520	1.39485	0.65927	0.13563
	IRFKMd-RF	**0.45483**	**91.50310**	**1.35964**	**0.68349**	**0.12864**

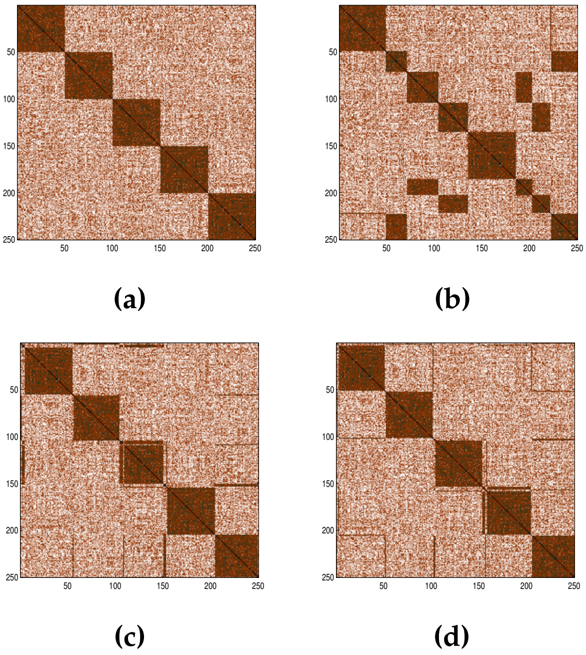
Table 4.16: Execution time in second for different methods on different datasets

Datasets	Cat-100-8-3	Cat-250-15-5	Cat-300-8-3	Cat-300-15-5	Cat-500-20-10	Cat-1000-7-7	Soybean	Zoo	Heart	Votes	Mushroom
KMd	0.024	0.064	0.034	0.105	0.184	0.191	0.042	0.072	0.051	0.036	0.330
KMdd	0.060	0.161	0.084	0.263	0.461	0.479	0.104	0.179	0.127	0.089	0.826
RKMdd	0.062	0.168	0.088	0.275	0.482	0.500	0.109	0.188	0.133	0.093	0.864
FKMd	0.025	0.067	0.035	0.109	0.191	0.198	0.043	0.074	0.053	0.037	0.342
AL	0.023	0.062	0.032	0.101	0.177	0.184	0.040	0.069	0.049	0.034	0.317
TSFKMd	8.237	22.143	11.632	36.256	63.599	65.964	14.337	24.741	17.555	12.262	113.87
MMR	0.555	1.491	0.783	2.441	4.282	4.441	0.965	1.666	1.182	0.826	7.666
G-ANMI	6.043	16.247	8.535	26.602	46.664	48.400	10.519	18.153	12.881	8.997	83.549
ccdByEnsemble	2.967	7.977	4.191	13.062	22.913	23.765	5.165	8.913	6.325	4.418	41.024
RFKMd-RF	1.275	3.428	1.801	5.613	9.846	10.212	2.22	3.83	2.718	1.899	17.629
SARFKMd-RF	6.813	18.239	13.047	30.534	53.561	55.553	11.960	21.556	15.292	10.175	95.898
GARFKMd-RF	4.078	12.163	6.476	12.901	22.630	23.472	8.200	12.140	9.559	6.891	40.518
IRFKMd-RF	25.154	67.623	35.523	110.723	194.227	201.449	43.784	75.557	53.613	37.448	347.749



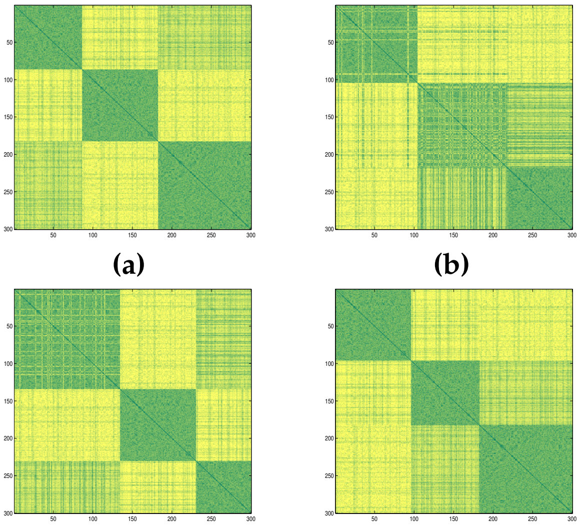
(a) (b) (c) (d)

Figure 4.3: VAT representation of (a) True Clusters, and best, after using different clustering methods (b) KMd (c) FKMd (d) IRFKMd-RF for Cat-100-8-3.



(a) (b) (c) (d)

Figure 4.4: VAT representation of (a) True Clusters, and best, after using different clustering methods (b) KMd (c) FKMd (d) IRFKMd-RF for Cat-250-15-5.



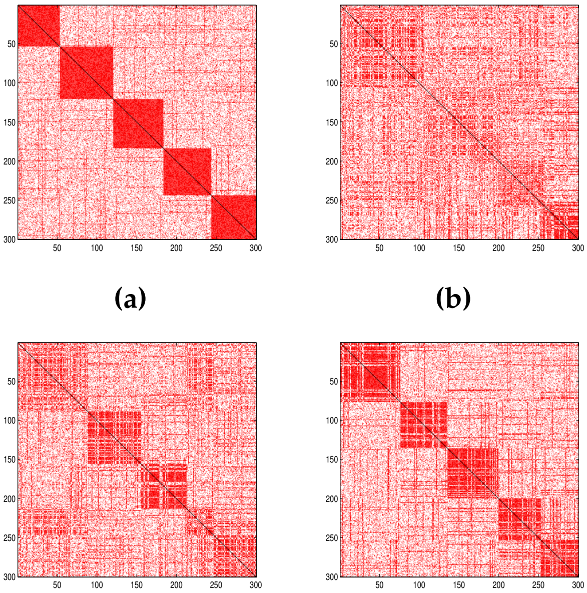
(a) (b) (c) (d)

Figure 4.5: VAT representation of (a) True Clusters, and best, after using different clustering methods (b) KMd (c) FKMd (d) IRFKMd-RF for Cat-300-8-3

requires more time compared to previous clustering methods due to the inclusion of supplementary activities, such as the calculation of roughness measure, the establishment of a Random Forest model, and the preparation of training and testing sets. The execution time of the IRFKMd-RF algorithm for additional datasets is as follows: In the given dataset, the recorded execution times for various categories are as follows: *Cat-100-8-3* took 25.154 seconds, *Cat-250-15-5* took 67.623 seconds, *Cat-300-8-3* took 35.523 seconds, *Cat-500-20-10* took 194.227 seconds, *Cat-1000-7-7* took 201.449 seconds, *Soybean* took 43.784 seconds, *Zoo* took 75.557 seconds, and *Heart* took 53.613 seconds. The time recorded for the ”Votes” task was 37.448 seconds, whereas the ”Mushroom” task took 347.749 seconds. The execution time in seconds for all the techniques employed for various datasets is shown in Table 4.16.

4.4.2 Visualization of Results

In this study, the well recognized visual evaluation of clustering tendency (VAT) representation, as proposed by Bezdek et al. [170], is used to visually depict the datasets both before to and during the clustering process. The approach described typically involves representing pairwise dissimilarity information of a set of n items as a square picture with dimensions $n \times n$. The objects are rearranged in a certain sequence to create an image that may effectively emphasize any possible



(a) (b) (c) (d)

Figure 4.6: VAT representation of (a) True Clusters, and best, after using different clustering methods (b) KMd (c) FKMd (d) IRFKMd-RF for Cat-300-15-5

cluster structure existing in the data. Consequently, the collection is first organized based on the assigned cluster labels. Afterwards, the distance matrix is calculated in order to provide a visual representation. The clustering structure is represented by boxes that are positioned along the major diagonal. Additionally, the use of VAT representation allows for the estimation of the number of clusters, denoted as K, that exist inside the dataset before to the clustering process.

The validity of the quantitative findings obtained from various approaches is confirmed by the use of VAT representation, as seen in Figures 4.3 to 4.13. Each picture displays four distinct sub-figures depicting the accurate representation of clustering, as well as the optimal structures of KMd, FKMd, and integrated rough fuzzy clustering algorithms. For instance, when examining the VAT plots of the *Cat-300-8-3* dataset, it can be observed that the clustering outcome of IRFKMd-RF, as depicted in Figure 4.5(d), exhibits a striking resemblance to the actual clustering structure presented in Figure 4.5(a). Conversely, the clustering results obtained from KMd and FKMd, as shown in Figure 4.5(b) and 4.5(c) respectively, are comparatively less satisfactory. Comparable outcomes are found for the other datasets. The boxplot depicting the MS values of various approaches is shown in Figures 4.14 and 4.15 for a total of six synthetic and five real-life datasets. The range of MS values for various clustering solutions generated by a particular algorithm during

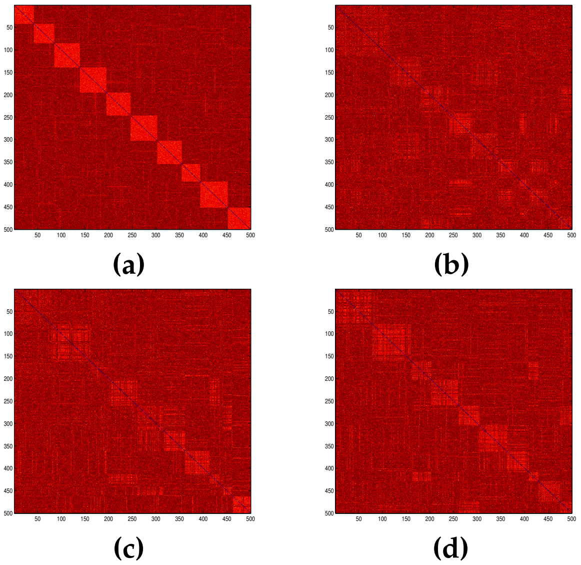


Figure 4.7: VAT representation of (a) True Clusters, and best, after using different clustering methods (b) KMd (c) FKMd (d) IRFKMd-RF for Cat-500-20-10

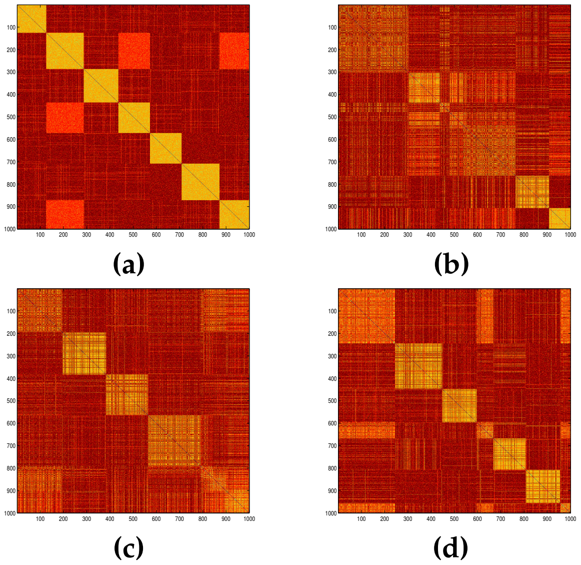
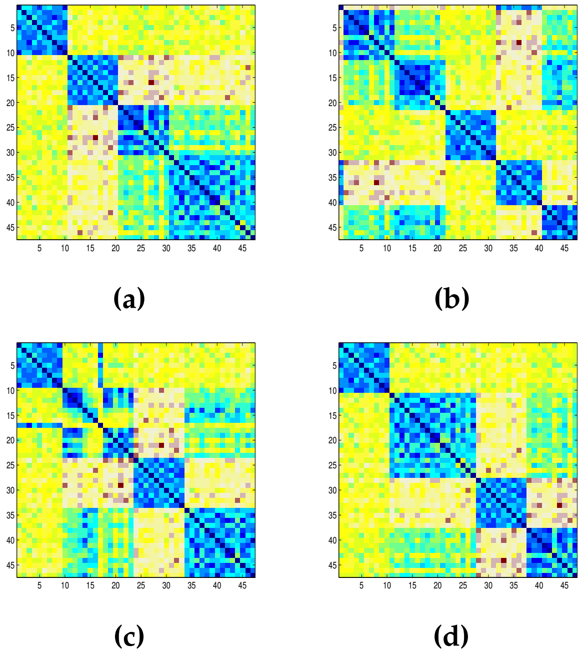


Figure 4.8: VAT representation of (a) True Clusters, and best, after using different clustering methods (b) KMd (c) FKMd (d) IRFKMd-RF for Cat-1000-7-7

20 iterations is shown in each box. It is well known that KMd, KMdd, and FKMd algorithms have a tendency to get trapped in local minima. Consequently, it is



(a) (b)

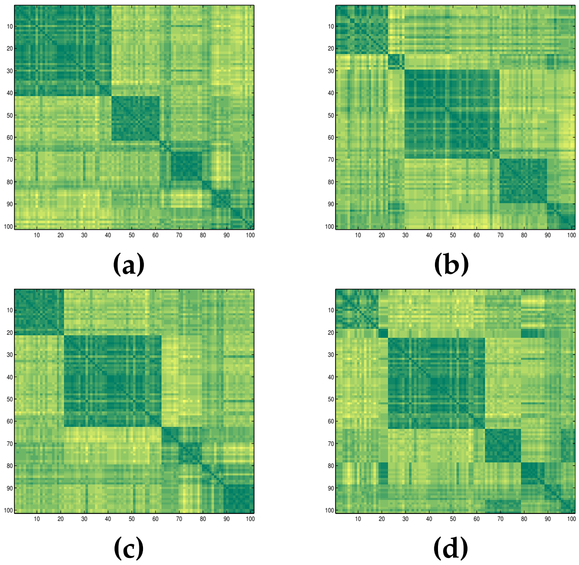
(c) (d)

Figure 4.9: VAT representation of (a) True Clusters, and best, after using different clustering methods (b) KMd (c) FKMd (d) IRFKMd-RF for Soybean

anticipated that their clustering solutions would be of worse quality compared to IRFKMd-RF. Hence, the clustering solutions offered by IRFKMd-RF for various iterations exhibit superior performance compared to KMd, KMdd, RKMdd, FKMd, AL, TSFKMd, MMR, G-ANMI, ccdbyEnsemble, RFKMd-RF, SARFKMd-RF, and GARFKMd-RF. Furthermore, these solutions fall within a very limited range. Additionally, the combined version of rough fuzzy clustering algorithms exhibits greater performance, as visibly shown.

4.4.3 Statistical Significance of the Results

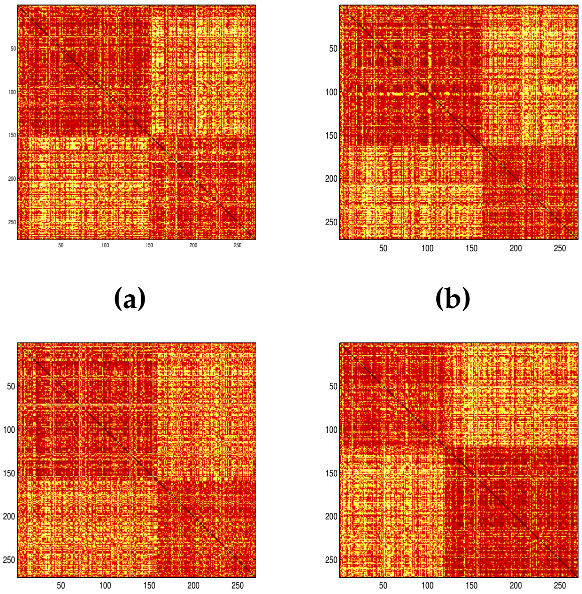
This study examines the statistical significance of the findings generated by 13 distinct methods. In order to achieve this objective, we have performed a parametric test known as the independent two-sample one-tailed *t*-test [171] and a non-parametric test known as the Friedman test [203,204]. The rationale for using both parametric and non-parametric tests is to evaluate the statistical significance of the outcomes based on the *p-values* generated by the IRFKMd-RF method. The study does a comparative analysis of the outcomes generated by various procedures and afterwards assigns specific rankings to each method.



(a) (b) (c) (d)

Figure 4.10: VAT representation of (a) True Clusters, and best, after using different clustering methods (b) KMd (c) FKMd (d) IRFKMd-RF for Zoo

For the independent two-sample one-tailed *t*-test, thirteen groups have been considered, each matching to one of the thirteen methods (1. KMd 2. KMdd 3. RKMdd 4. FKMd 5. AL 6. TSFKMd 7. MMR 8.G-ANMI 9. ccdByEnsemble 10. RFKMd-RF 11. SARFKMd-RF 12. GARFKMd-RF 13. IRFKMd-RF). Each group is comprised of the MS values generated by 20 consecutive iterations of the respective algorithm. Tables 4.5 to 4.14 provide the average MS values for the synthetic and real-life datasets, correspondingly. The table clearly demonstrates that the MS values generated by the IRFKMd-RF approach outperform the other techniques. In order to determine the statistical significance of the superior performance of IRFKMd-RF, a one-tailed paired *t*-test [171] is undertaken at a significance level of 5%. The *p-values* generated by the one-tailed paired *t*-test for the comparison of two groups (namely, IRFKMd-RF and another method) are shown in Table 4.17. The null hypothesis posits that there is no statistically significant difference between the means of the MS values in the two groups. In contrast, the alternative hypothesis posits that there exists a statistically significant disparity in the average values between the two groups. All of the *p-values* included in the table are below the threshold of 0.05, indicating statistical significance at the five per cent level. An analysis was conducted on the *Zoo* dataset to compare the performance of the IRFKMd-RF and FKMd techniques. The results of the *t*-test revealed a *p-values* of 1.980e-014, indicating statistical significance at a significance level of 0.05. The presented data provides substantial support against the null hypothesis, suggesting



(a)

(b)

(c)

(d)

Figure 4.11: VAT representation of (a) True Clusters, and best, after using different clustering methods (b) KMd (c) FKMd (d) IRFKMd-RF for Heart

that the improved MS values achieved by the IRFKMd-RF method possess statistical significance and are not the result of random chance. Comparable outcomes are achieved when the performance of IRFKMd-RF is compared to that of other methodologies across various datasets.

Furthermore, the statistical significance threshold for conducting the Friedman test [203, 204] is set at 5%. The Friedman test is often used to rank various approaches individually for each dataset. In order to calculate the average rank $\mathcal{R}_j$, we define r_i^j as the rank of the jth method for the ith dataset, where the total number of datasets and methods are denoted as N and Q accordingly. Hence, the average rank may be expressed as $\mathcal{R}_j = \frac{1}{N}\sum_i r_i^j$. In the present experiment, assuming the null hypothesis, it is expected that all the techniques being compared are of similar performance, resulting in equal rankings denoted as $\mathcal{R}_j$. The computation of the Friedman statistic, denoted as χ_F^2 value, is performed in the following manner:

$$\chi_F^2 = \frac{12N}{Q(Q+1)}\left[\sum_j \mathcal{R}_j^2 - \frac{Q(Q+1)^2}{4}\right] \tag{4.4}$$

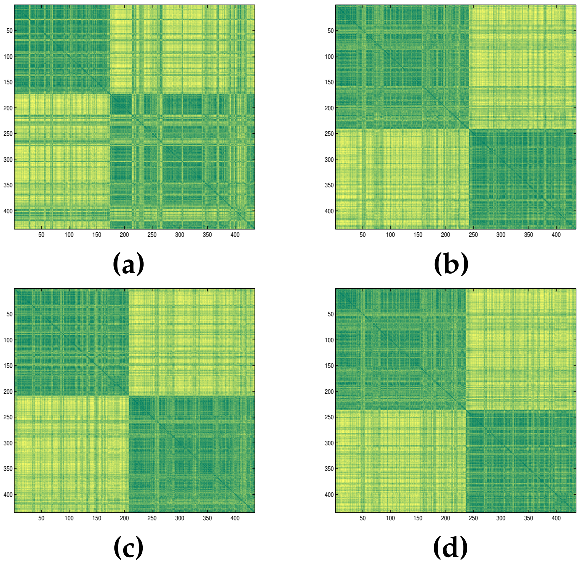


Figure 4.12: VAT representation of (a) True Clusters, and best, after using different clustering methods (b) KMd (c) FKMd (d) IRFKMd-RF for Votes

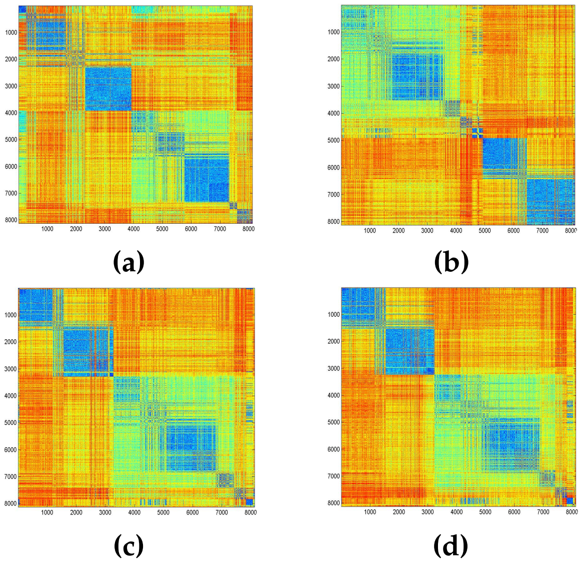
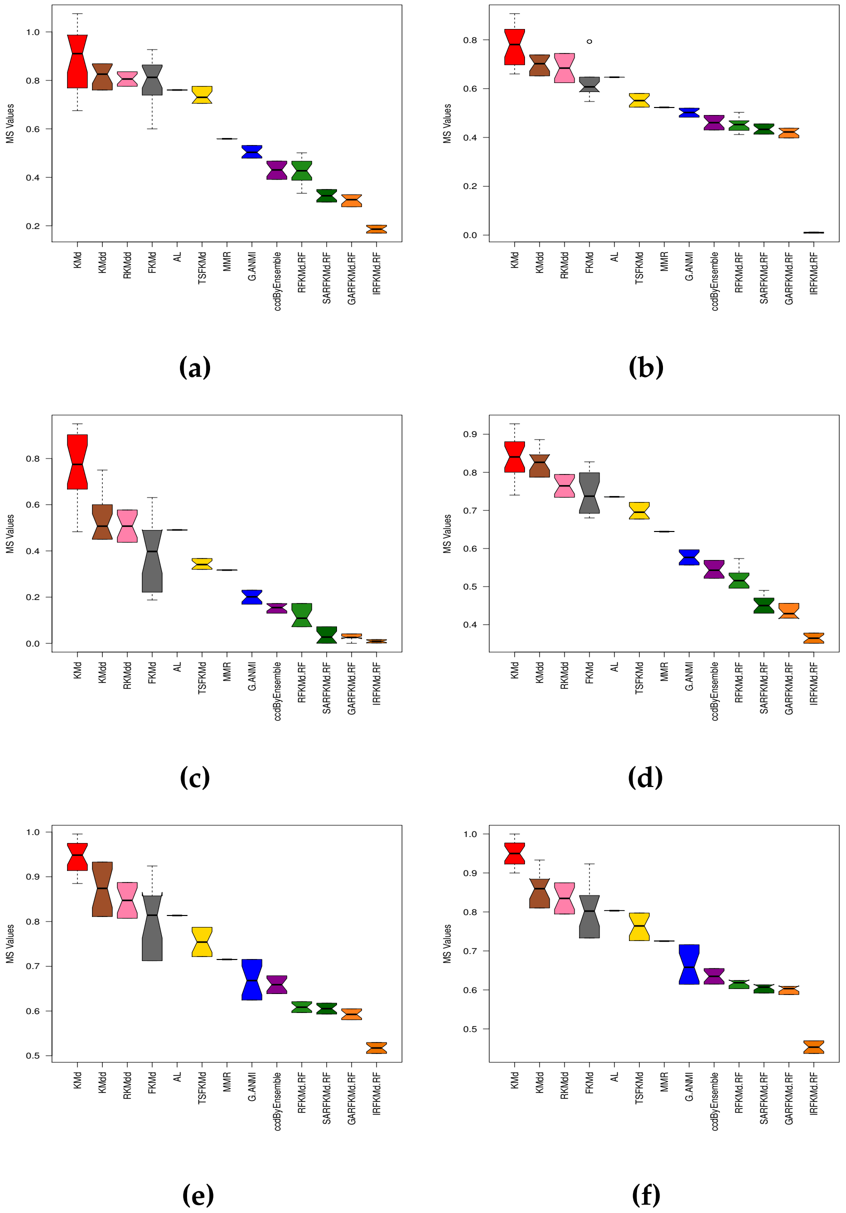


Figure 4.13: VAT representation of (a) True Clusters, and best, after using different clustering methods (b) KMd (c) FKMd (d) IRFKMd-RF for Mushroom

The distribution of the Friedman statistic follows a chi-squared distribution (χ_F^2) with $(Q-1)$ degrees of freedom. The rank of an individual approach for several datasets



(a)

(b)

(c)

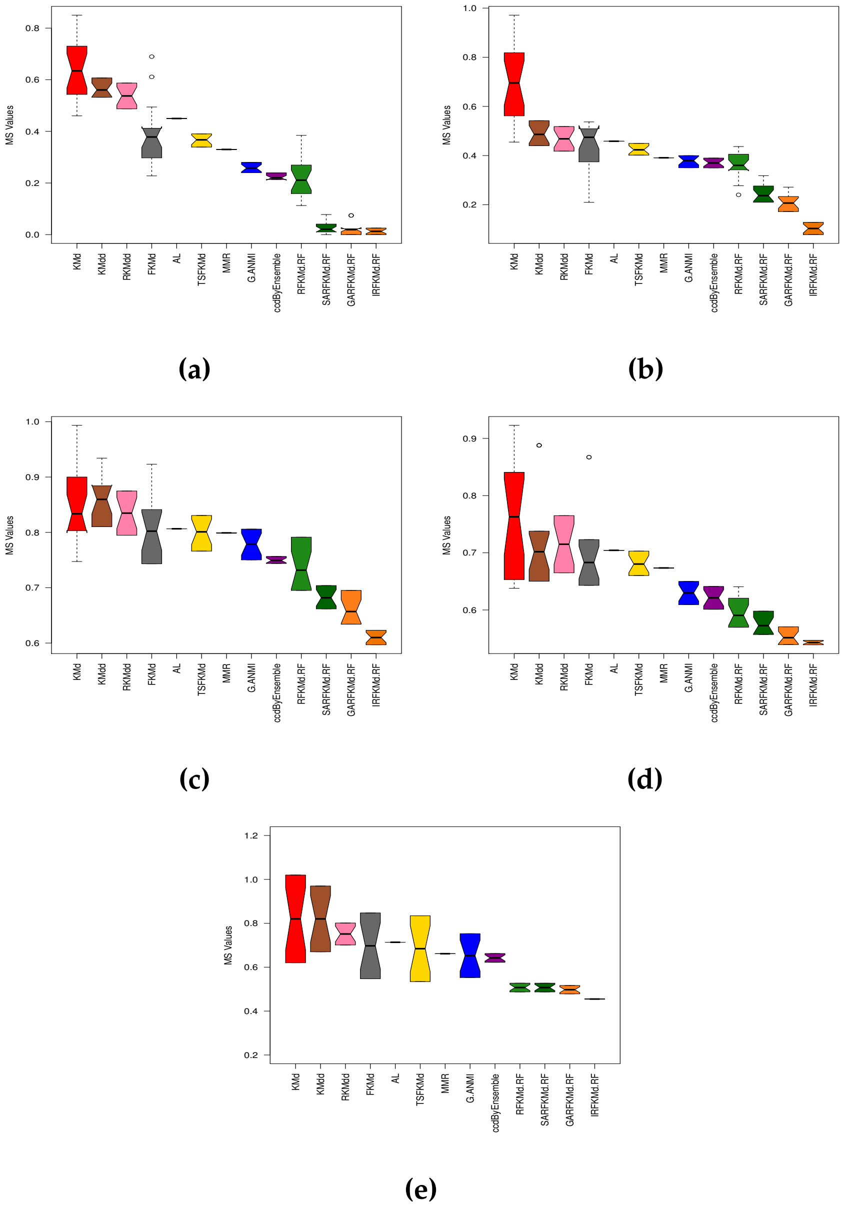
(d)

(e)

(f)

Figure 4.14: Boxplot of MS values of different clustering methods for (a) Cat-100-8-3 (b) Cat-250-15-5 (c) Cat-300-8-3 (d) Cat-300-15-5 (e) Cat-500-20-10 (f) Cat-1000-7-7

and the average rank of each method are shown in Table 4.18. The average ranks of the algorithms KMd, KMdd, RKMdd, FKMd, AL, TSFKMd, MMR, G-ANMI, ccdByEnsemble, RFKMd-RF, SARFKMd-RF, GARFKMd-RF, and IRFKMd-RF, as shown in Table 4.18, are 12.90, 12.09, 11, 9.45, 9.54, 7.95, 7.05, 6, 4.95, 3.95, 3.04, 2.04, and 1. Furthermore, based on the mean rankings obtained, the computation of χ_F^2



(a)

(b)

(c)

(d)

(e)

Figure 4.15: Boxplot of MS values of different clustering methods for (a) Soybean (b) Zoo (c) Heart (d) Votes (e) Mushroom

using Equation 4.4 yields a value of 131.59. Therefore, the associated *p-values* are less than 2.2×e-16 at a significance threshold of $\alpha = 0.05$. This further supports the adoption of the alternative hypothesis that the procedures differ in terms of performance. However, it should be noted that the average rank of IRFKMd-RF is higher compared to the other methods, indicating the statistical importance of the data obtained by IRFKMd-RF.

Table 4.17: Results of the independent two-sample one-tailed t-test conducted on the Synthetic and Real Life datasets. The test generates *p-values* via the comparison of the IRFKMd-RF technique with other methods

Dataset	KMd	KMdd	RKMdd	FKMd	AL	TSFKMd	MMR	G-ANMI	ccdByEnsemble	RFKMd-RF	SARFKMd-RF	GARFKMd-RF
Cat-100-8-3	1.814e-12	9.139e-14	1.000e-17	2.809e-14	6.281e-18	1.809e-15	3.331e-16	8.926e-16	1.564e-13	2.313e-11	4.683e-12	6.475e-14
Cat-250-15-5	7.307e-13	8.968e-17	3.481e-14	7.723e-10	8.002e-27	3.551e-16	5.605e-26	1.059e-14	2.144e-15	1.235e-14	3.005e-18	7.964e-17
Cat-300-8-3	8.968e-13	2.783e-09	9.751e-13	3.902e-10	2.162e-20	1.241e-12	1.270e-18	2.560e-13	6.899e-15	1.520e-07	3.678e-03	3.229e-05
Cat-300-15-5	2.799e-13	2.648e-14	1.225e-16	5.932e-09	5.403e-17	3.737e-16	7.331e-16	2.675e-15	1.250e-08	8.177e-14	2.840e-08	3.138e-06
Cat-500-20-10	2.772e-14	2.301e-12	5.652e-14	7.325e-10	1.875e-16	1.420e-13	7.874e-15	1.987e-10	2.614e-07	2.458e-09	3.741e-07	1.707e-06
Cat-1000-7-7	1.305e-14	8.404e-13	4.215e-15	9.140e-10	5.469e-16	6.527e-15	5.693e-15	7.850e-11	6.369e-10	4.097e-10	1.075e-09	8.312e-10
Soybean	4.114e-10	2.120e-15	6.816e-15	1.683e-07	4.243e-18	2.652e-17	8.143e-17	1.810e-12	1.126e-07	6.774e-18	5.347e-04	2.099e-01
Zoo	1.649e-10	5.793e-11	5.029e-15	2.514e-14	1.540e-14	5.621e-16	1.142e-13	1.029e-11	1.764e-12	7.722e-11	1.660e-08	8.624e-09
Heart	3.759e-08	8.560e-11	5.478e-13	4.476e-08	9.483e-15	6.733e-14	1.365e-14	7.962e-14	4.352e-09	5.462e-16	5.276e-07	4.070e-08
Votes	1.329e-07	3.851e-06	7.400e-10	8.598e-06	1.517e-18	5.567e-11	1.057e-17	2.946e-11	2.392e-08	9.328e-11	7.606e-08	5.421e-05
Mushroom	4.677e-08	6.920e-09	7.778e-12	9.866e-08	4.549e-26	1.373e-07	3.466e-25	2.704e-08	1.400e-13	3.025e-09	3.058e-09	8.753e-09

Table 4.18: Performance and ranking of several approaches for both Synthetic and Real Life datasets. Each item in the dataset has two values: the rank and the average MS, which is provided between brackets

Dataset	KMd	KMdd	RKMdd	FKMd	AL	TSFKMd	MMR	G-ANMI	ccdByEnsemble	RFKMd-RF	SARFKMd-RF	GARFKMd-RF	IRFKMd-RF
Cat-100-8-3	13(0.88)	12(0.82)	11(0.81)	10(0.8)	9(0.76)	8(0.74)	7(0.56)	6(0.5)	4.5(0.43)	4.5(0.43)	3(0.32)	2(0.30)	1(0.18)
Cat-250-15-5	13(0.78)	12(0.70)	11(0.68)	9(0.63)	10(0.65)	8(0.55)	7(0.52)	6(0.50)	5(0.46)	4(0.45)	3(0.44)	2(0.42)	1(0.01)
Cat-300-8-3	13(0.77)	12(0.54)	11(0.51)	9(0.37)	10(0.49)	8(0.34)	7(0.32)	6(0.20)	5(0.15)	4(0.12)	3(0.05)	2(0.03)	1(0.00)
Cat-300-15-5	13(0.84)	12(0.83)	11(0.76)	10(0.75)	9(0.74)	8(0.70)	7(0.64)	6(0.58)	5(0.55)	4(0.52)	3(0.45)	2(0.44)	1(0.36)
Cat-500-20-10	13(0.94)	12(0.86)	11(0.85)	9.5(0.81)	9.5(0.81)	8(0.76)	7(0.72)	6(0.67)	5(0.66)	3.5(0.61)	3.5(0.61)	2(0.59)	1(0.52)
Cat-1000-7-7	13(0.95)	12(0.86)	11(0.83)	10(0.81)	9(0.80)	8(0.76)	7(0.73)	6(0.67)	5(0.63)	4(0.62)	2.5(0.60)	2.5(0.60)	1(0.45)
Soybean	13(0.64)	12(0.57)	11(0.54)	9(0.39)	10(0.45)	8(0.37)	7(0.33)	6(0.26)	5(0.22)	4(0.20)	3(0.04)	2(0.03)	1(0.01)
Zoo	13(0.69)	12(0.49)	11(0.47)	9(0.44)	10(0.46)	8(0.43)	7(0.39)	6(0.38)	5(0.37)	4(0.36)	3(0.24)	2(0.21)	1(0.10)
Heart	12.5(0.86)	12.5(0.86)	11(0.83)	9.5(0.81)	9.5(0.81)	7.5(0.80)	7.5(0.80)	6(0.78)	5(0.75)	4(0.74)	3(0.68)	2(0.66)	1(0.61)
Votes	13(0.76)	12(0.73)	11(0.71)	9.5(0.70)	9.5(0.70)	8(0.68)	7(0.67)	6(0.63)	5(0.62)	4(0.60)	3(0.58)	2(0.55)	1(0.54)
Mushroom	12.5(0.81)	12.5(0.81)	11(0.75)	9(0.69)	10(0.71)	8(0.68)	7(0.66)	6(0.65)	5(0.64)	3.5(0.50)	3.5(0.50)	2(0.49)	1(0.45)
Avg. Rank	12.90	12.09	11	9.45	9.54	7.95	7.05	6	4.95	3.95	3.04	2.04	1

4.5 Worst case Time Complexity Analysis

The worst case time complexities of the proposed algorithms are discussed below.

4.5.1 Time Complexity Analysis of SARFKMd

Worst case Time Complexity of RFKMd has been explained in section 2.4.1. Additionally, SRFKMd performs initialization and perturbation, that take some amount of time. Moreover, one loop of the method executes until minimum temperature is achieved and other inner loop repeats upto a maximum iteration, *maxItr*. Maximum number of repetition of first loop is $\Lambda = \left\{ \frac{\log\left(\frac{T_{min}}{T_{max}}\right)}{\log g} \right\}$. Overall the worst case time complexity is $O\left(\Lambda \times maxItr \times (2Km + 2Knm + n + Kn + KMn)\right) \approx O\left(\Lambda \times maxItr \times KMn\right)$, where $M\left(= \sum_{j=1}^{m} q_j\right)$, is the total number of categories of all attribute.

4.5.2 Time Complexity Analysis of GARFKMd

In each iteration, initialization, crossover and mutation steps take time of $O\left(PS \times Km\right)$ separately for whole population, where PS is the population size. Moreover, selection process is done with $O\left(PS\right)$, whereas the fitness computation takes time same as RFKMd. Therefore, overall time complexity is $O\left(\ 3PS \times Km + PS + PS \times KMn\ \right)$. Now, if $n \gg m$, only KMn dominate other terms. So, the time complexity of GARFKMd is $O\left(PS \times KMn\right)$. For maximum generation, the total time complexity is $O\left(maxGen \times PS \times KMn\right)$.

4.5.3 Time Complexity Analysis of RFKMd-RF, SARFKMd-RF and GARFKMd-RF

Time complexity of RFKMd-RF depends on RFKMd and Random Forest. Referring section 2.4, the time complexity of RFKMd-RF is $O\left(KMn + \mathsf{T}nm\log\left(n\right)\right)$ for each iterations. Similarly, time complexity of SARFKMd-RF and GARFKMd-RF are $O\left(\ \Lambda \times\ maxItr \times KMn + \mathsf{T}nm\log\left(n\right)\ \right)$ and $O\left(\ maxGen \times PS \times\ KMn + \mathsf{T}nm\log\left(n\right)\ \right)$.

4.5.4 Time Complexity Analysis of IRFKMd-RF

IRFKMd-RF is dependent on RFKMd, SARFKMd, GARFKMd, union of sets, the intersection of sets and Random Forest. Computation time for best central points, semi-best central points and pure peripheral points depends on union, intersection and Random Forest. Time taken by union and intersection of sets is $O\left(n\log\left(n\right)\right)$. Refer section 2.4 for Random Forest. Time to sort roughness measure is negligible, because, the number of methods used here is constant. Therefore, the overall time complexity considering sum total of all the major time consuming activities is equal to $O\left(\ KMn + \Lambda \times maxItr \times KMn + \mathsf{T}nm\log\left(n\right) + maxGen \times PS \times KMn + \mathsf{T}nm\log\left(n\right)\ \right) \approx O\left(\ \Lambda \times maxItr \times KMn + maxGen \times PS \times KMn + \mathsf{T}nm\log\left(n\right)\ \right)$, where $M\left(= \sum_{j=1}^{m} q_j\right)$, is the total number of categories of all attributes.

4.6 Conclusion

This chapter presents an enhanced version of the Rough Fuzzy K-Modes clustering algorithm, which aims to address the challenges associated with indiscernibility (coarseness) and vagueness in categorical datasets more effectively [209]. The Rough Fuzzy K-Modes clustering technique exhibits a proclivity for becoming ensnared in local optima. Consequently, two additional clustering techniques have been devised by incorporating simulated annealing and genetic algorithm. These techniques are known as Simulated Annealing based Rough Fuzzy K-Modes and Genetic Algorithm based Rough Fuzzy K-Modes. These strategies possess the capability to effectively manage clusters that consist of distinct sets of central and peripheral points. To categorize the peripheral points generated by these approaches, a Random Forest classifier has been included independently. In order to achieve this objective, the classifier is trained using central points to categorize peripheral points. In addition, it should be noted that the outcomes generated by the Rough Fuzzy K-Modes, Simulated Annealing-based Rough Fuzzy K-Modes, and Genetic Algorithm-based Rough Fuzzy K-Modes approaches are distinct from one another. Therefore, the cardinality of the sets including central and peripheral points has shown variation among various techniques. The roughness measure is calculated in order to determine the optimal set of central points from the three available sets. Following that, the analysis identifies semi-best central and pure peripheral points. Subsequently, the best central points are used for the purpose of categorizing the semi-best central points, which are then utilized to categorize peripheral points via the application of Random Forest. This approach aims to enhance the quality of clustering outcomes. The methodology is referred to as Integrated Rough Fuzzy Clustering using Random Forest. The efficacy of the established methodologies has been shown by a comparative analysis with other contemporary state-of-the-art approaches. In order to achieve this objective, the we have conducted experiments on both synthetic and real-life categorical datasets, assessing the effectiveness of several cluster validity indexes. The results indicate that the Integrated Rough Fuzzy Clustering approach consistently outperformed the other methods across all datasets. The proposed methodologies have the potential to be used in a wide range of practical applications, such as consumer segmentation, market basket analysis, environmental research, text and web mining, prediction and forecasting, and biometry. The concept similar to the proposed method has been applied in the field of feature selection, particularly in the context of identifying MicroRNAs that have a substantial influence on the progression of breast cancer [210,211]. The practical applications also often include datasets that consist of categorical variables. For instance, the clustering and analysis of consumer sentiment data obtained from diverse social media platforms and websites may be used to forecast client attrition within the telecommunication sector. The concept of integrating evolutionary approach has resemblance to a notion [212] that has been applied in improving modified differential evolution for fuzzy based clustering.

In the next chapter, credibilistic measure is studied to address the limitation of fuzzy and possibilistic theory to cluster categorical data set as a potential area of improvement.

5 Semi-supervised clustering using Credibilistic method

5.1 Introduction

For mining real-world data, data scientists are currently confronting two significant practical difficulties. In actual life, there are typically just a small number of observations with labeled data. However, because of the time and resource constraints, labeling the data is equally expensive. Unsupervised clustering algorithms are therefore crucial for identifying intriguing patterns in sophisticated analytical applications. However, in cases of vague, uncertain, coincidental, and overlapping situations, clustering approaches encounter difficulties in classifying a data point into a cluster. The semi-supervised clustering technique is appropriate in this case. Additionally, it is discussed in Chapter 2 that the majority of the study on clustering techniques focuses mostly on numerical data. It is so because employing specific distance functions, the geometric structure of numerical data may be easily exploited. The presence of distinct features in categorical data poses challenges for clustering algorithms, making the task of grouping such data inherently complex. For instance, when a natural ordering is not present, the categorical property of colour may include several values, including but not limited to red, green, blue, and so forth. As a result, categorical data clustering has gained popularity among researchers. In the chapter [154], certain difficulties with grouping categorical data are highlighted. Because of this, conventional algorithms like K-Means, Fuzzy C-Means, and versions of these algorithms are unable to cluster categorical data. This is the case since these techniques compute the cluster mean. Several methods for clustering categorical data have been recently introduced across several academic literature [35,39,41,42,50,91–94,101,102,105,106,110,135,136,155]. Among those K-Modes (KMd) [42] and Fuzzy K-Modes (FKMd) [50] are widely used methods. Umayahara et al. introduced an alternative fuzzy technique for document categorization in the field of literature classification [156]. In a separate study, Kim et al. [101] enhanced the FKMd method by presenting a fuzzy centroid-based clustering methodology. The process of category clustering is often conducted via the use of Partitioning Around Medoids (PAM) or *K*-Medoid (KMdd) algorithms, as discussed by Kaufman [41]. Several novel strategies have been suggested, as discussed in the following sections.

Broadly speaking, clustering may be categorized into two main types: crisp clustering and fuzzy clustering. Crisp clustering assigns each data point exclusively to a single cluster, whereas fuzzy clustering allows for varying degrees of membership for each data point across many clusters. Fuzzy clustering is widely preferred and considered superior over regular crisp clustering due to its ability to manage clustering overlap via the use of a fuzzy objective function with probabilistic membership values. Nevertheless, because to its inherent probabilistic constraints in the presence of noise, it may not consistently align with the intuitive perception of varying levels of membership. The Possibilistic Clustering Algorithm (PCA) was subsequently developed by Krishnapuram et al. [122, 123] and Yang et al. [148] in order to tackle this particular issue. The thesis proposes and discusses the integration of fuzzy type-2 in order to enhance the possibilistic approach, as outlined in Chapter 3. The possibilistic approach is often used in many domains. However, it is important to note a significant drawback referred to as the coincident problem. This issue arises when a dataset consisting of *K* clusters is subjected to PCA, resulting in a probability of the dataset being split into fewer than *K* clusters. The details about the issue of coincidence are discussed in the publication by Tjhi [142]. The concept of a credibilistic measure was first introduced by Liu et al. in their publication [143] as a means to tackle the limitations associated with fuzzy and possibilistic clustering techniques. The concept of credibilistic clustering was first introduced by Zhou [144] and Kalhori [145] in their respective works in 2015. Subsequently, Zhou made significant modifications to this approach in 2017 [146] by including the concept of alternating cluster estimate, as described by Runkler [147]. The form of credibilistic clustering proposed by Zhou et al. [146] used the same membership function as the one applied in the study conducted by Yang et al. [148]. As far as current understanding indicates, the use of credibilistic methods in categorical data clustering remains unexplored. The primary emphasis of all the versions of credibilistic clustering approaches [144–146] lies in the clustering of numerical data. This thesis presents a proposed semi-supervised clustering technique that utilizes a credibilistic measure and integrates machine learning techniques. The objective is to tackle the clustering of categorical data and address the challenges posed by uncertainty, vagueness, coincidental occurrences, and overlapping inherent characteristics of such data. To tackle this issue, we propose first the Credibilistic *K*-Mode (CrKMd) clustering method. This approach incorporates a credibilistic measure to effectively distinguish between definite and uncertain data points, hence eliminating the challenge of coincident clustering. Uncertain facts, on occasion, may possess similar credibility across different categories. Thus, the CrKMd framework incorporates various machine learning (ML) techniques, including the *K* Nearest Neighbour (*K*-NN) [24], Support Vector Machine (SVM) [21], Artificial Neural Network (ANN) [23], Decision Tree (DT) [213], and Random Forest (RF) [27]. These techniques, known as CrKMd-KNN, CrKMd-SVM, CrKMd-ANN, CrKMd-DT, and CrKMd-RF, are utilized to classify uncertain data and achieve improved outcomes. The term MLCrKMd is often used to refer to the suggested approach of integrating machine learning with semi-supervised clustering. The efficacy of the proposed approach is shown by graphical, numerical, and statistical analyses in relation to alternative state-of-the-art approaches, across a total of eight synthetic datasets and four real-world datasets.

5.2 Credibilistic Semi-Supervised Clustering for Categorical Data

5.2.1 Brief Mathematical Concept of Credibility Measure

The membership function-based notion of fuzzy set theory that Zadeh [159] developed has been used to solve a wide range of practical issues. Fuzzy based clustering is vulnerable to noise, though, and the membership number does not always reflect how much an object fits into the right cluster. Zadeh [214] presented the possibilistic approach as a means to address this problem, whereby fuzzy events are evaluated using possibility measures. Many scientists afterwards investigated the possibility theory and its variations. The most popular of these algorithms is Possibilistic C-Means (PCM), which was used in [122] to determine the degree of belongingness by relaxing the constraint of Fuzzy C-Means (FCM). Consequently, this feature facilitates the formation of data clusters that are not influenced by other clusters, thus addressing the challenge of coincident clusters in datasets with closely located clusters. The credibility measure was developed by Liu et al. [143] to overcome the limits of both fuzzy and possibilistic measures by introducing the idea of self-duality, which is not present in either FCM or PCM. This section provides a description of some important mathematical components of the credibility measure, which are then used in CrKMd, in order to enhance comprehension. The article by Liu [215] examines the necessary and sufficient factors for establishing a believability metric.

Let $\mathcal{S}$ denote a nonempty set, and let $\mathbb{P}(\mathcal{S})$ represent the power set of $\mathcal{S}$. Each element of $\mathbb{P}(\mathcal{S})$ is referred to as an event, denoted by A. In order to establish the concept of a credibility measure, it is crucial to give a number, denoted as $\mathscr{C}\{A\}$, to every event, A, belonging to the set $\mathbb{P}(\mathcal{S})$. The notation $\mathscr{C}\{A\}$ represents the measure of credibility associated with the event A. A credibility space is defined as a triplet $(\mathcal{S}, \mathbb{P}(\mathcal{S}), \mathscr{C})$, where $\mathcal{S}$ is the sample space, $\mathbb{P}(\mathcal{S})$ is the set of all subsets of $\mathcal{S}$, and $\mathscr{C}$ is the credibility function. The mathematical qualities of $\mathscr{C}\{A\}$ may be defined by the following five axioms.

- $\mathscr{C}\{\mathcal{S}\} = 1$;
- $\mathscr{C}$ is increasing, i.e., $\mathscr{C}\{A\} < \mathscr{C}\{B\}$ when $A \subset B$;
- $\mathscr{C}$ is self-dual, i.e., $\mathscr{C}\{A\} + \mathscr{C}\{A^c\} = 1$ for any $A \in \mathbb{P}(\mathcal{S})$;
- $\mathscr{C}\{\cup_i A_i\} \wedge 0.5 = sup_i \mathscr{C}\{A_i\}$ for any $\{A_i\}$ with $\mathscr{C}\{A_i\} \leq 0.5$.
- Let $\mathcal{S}_i$ is nonempty set on which $\mathscr{C}_i$ satisfies the first four axioms for $i = 1, 2, \cdots, n$, respectively, and $\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \cdots \times \mathcal{S}_n$, then $\mathscr{C}(\mathcal{S}_1, \mathcal{S}_2, \cdots, \mathcal{S}_n) = \mathscr{C}_1\{\mathcal{S}_1\} \wedge \mathscr{C}_2\{\mathcal{S}_2\} \wedge \cdots \wedge \mathscr{C}_n\{\mathcal{S}_n\}$

When the first four axioms of [143] are satisfied by the set function $\mathscr{C}$, it is referred to as a credibility measure. The fact that $\mathscr{C}\{\phi\} = 0.$ is obvious. As a result, the credibility measure is capable of assuming values within the range of 0 to 1. The aforementioned axioms ensure that the credibility measure exhibits a increasing behaviour and holds the property of self-duality.

Let's quickly go through some mathematical properties to show how fuzzy membership value, probabilistic measure, and credibility measure are related. Let X be a dataset consisting of n data points that are distributed across K clusters. In this context, μ_{li}, P_{li}, and $\mathscr{C}_{li}$ represent the fuzzy membership value, possibilistic measure, and credibility measure, respectively, of the ith point $x_i \in X$ inside the lth cluster. According to the possibility theory provided by Zadeh [214], it may be inferred that

$$P_{li} = \mu_{li}, \forall i = 1, 2, \cdots, n \tag{5.1}$$

Furthermore, in accordance with the mathematical principles of possibility theory as outlined in the publication of Nahmias [216], it follows

$$0 \leq P_{li} \leq 1, \forall i = 1, 2, \cdots, n \;\; and \;\; \forall j = 1, 2, \cdots, K \tag{5.2}$$

$$sup_i\{P_{li}\} = 1, \forall j = 1, 2, \cdots, K \tag{5.3}$$

Therefore, from Equations 5.1, 5.2 and 5.3, it can be derived as follows.

$$0 \leq \mu_{li} \leq 1, \forall i = 1, 2, \cdots, n \;\; and \;\; \forall l = 1, 2, \cdots, K \tag{5.4}$$

$$sup_i\{\mu_{li}\} = 1, \forall l = 1, 2, \cdots, K \tag{5.5}$$

The possibility space for event A, denoted as $(\mathcal{S}, \mathbb{P}(\mathcal{S}), P)$, is introduced in [217]. The necessity measure of event A is defined as follows

$$\mathcal{N}\{A\} = 1 - P\{A^c\} \tag{5.6}$$

whereas, in the same possibility space, the association between the credibility measure of event $A \in \mathbb{P}(\mathcal{S})$ and the measures of necessity and possibility is established in [143] according to the following definition.

$$\mathscr{C}\{A\} = \frac{1}{2}\{P\{A\} + \mathcal{N}\{A\}\} \tag{5.7}$$

Therefore, by using Equations 5.1, 5.6, 5.7, and employing the mathematical principles of possibility theory as outlined in previous works such as [144, 216], one can deduce that

$$\mathscr{C}_{li} = \frac{1}{2}(\mu_{li} + 1 - sup_{k \neq l}\mu_{ki}), \forall i, l \tag{5.8}$$

where,

$$\mu_{li} = \frac{\hat{\mu}_{li}}{sup_k \mu_{ki}}, \forall i, l \tag{5.9}$$

$$\hat{\mu}_{li} = \frac{1}{1 + D(c_l, x_i)}, \forall i, l \tag{5.10}$$

In [144], the authors provide an explanation and proof that μ_{li} has been normalized using Equation 5.9, therefore ensuring its compliance with the requirement stated in Equation 5.5.

5.2.2 Credibilistic *K*-Modes

The mathematical ideas discussed above are used in credibilistic K-modes (CrKMd). The proposed algorithm begins by randomly selecting K modes ($c_l = [c_{l1}, c_{l2}, \ldots, c_{lm}]$, where, $1 \leq l \leq K$ and $c_l \in X$) from the entire dataset. Subsequently, the credibility matrix $[\mathscr{C}_{li}]$ is computed for the entire dataset using Equation 5.8, which adheres to the mathematical properties outlined in [215].

- $sup_{1 \leq l \leq K}\mathscr{C}_{li} \geq 0.5, \forall\ l$
- $\mathscr{C}_{li} + sup_{h \neq l}\mathscr{C}_{hi} = 1, for\ \ any\ \ l, i\ \ with\ \ \mathscr{C}_{li} \geq 0.5$
- $0 \leq \mathscr{C}_{li} \leq 1, \forall\ l, i$

The credibility measure removes the constraints of the FCM, which is the requirement that the total of membership degrees for each point must equal 1 ($\sum_{l=1}^{K} \mu_{li} = 1$). Additionally, the self-dual feature of the credibility measure ensures that if $\mathscr{C}_{li} = 1$, the ith point is considered to belong to the lth cluster. Conversely, if $\mathscr{C}_{li} = 0$, the point is considered to be entirely outside of the cluster. In contrast, PCM diverges from the aforementioned scenario since it allows for the possibility of a fuzzy event failing to occur despite its probability increasing by a unit of 1. In light of the probabilistic measure P_{li} being equal to 1, it is still possible for the data point x_i to not be a member of the cluster $\mathscr{C}_l$ as a result of this particular characteristic of the PCM algorithm. Furthermore, the PCM algorithm only considers the possibilistic measure P_{li} inside cluster l for the data point x_i, which is responsible for the occurrence of coincident clustering. Consequently, the Credibilistic K-Modes algorithm incorporates a credibilistic measure to ensure both compactness of clustering and resolution of the coincident PCM problem. This is achieved by considering the believability, denoted as $\mathscr{C}_{li}$, of a data point x_i in cluster l, as well as the credibility, denoted as $\mathscr{C}_{hi}$, of x_i in other cluster h (where $l \neq h$), in order to appropriately assign x_i to the correct cluster.

The techniques for updating the cluster center are discussed in Zhou's studies [144, 146]. These studies primarily focus on numerical data and do not address the applicability of these processes to categorical data. The research endeavour of this thesis has used a unique approach to propose CrKMd, as outlined in Algorithm 11. This approach involves the utilization of a new technique for updating the cluster mode, which is mathematically specified in Equation 5.11. The cluster mode of each cluster (C_l) is denoted by c_l. The mode is computed iteratively utilizing the credibility matrix, as described in Equation 5.11, in order to optimize the objective function stated in Equation 5.12. The value of each attribute of cluster mode is determined as $c_{lj} = a_j^r \in DOM(\mathscr{A}_j)$ where,

$$r = \underset{1 \leq t \leq q_j}{\arg\max} \left\{ \sum_{\substack{1 \leq i \leq n, \\ x_i \in C_l, \\ x_{ij} = a_j^t}} (\mathscr{C}_{li})^{\eta} \right. \tag{5.11}$$

In this context, q_j represents the count of categorical values associated with attribute $\mathscr{A}_j$. The calculated mode, denoted as c_l, for a cluster C_l, is not required to be identical to any specific data point present in the dataset.

$$J_{\mathscr{C}} = \sum_{i=1}^{n} \sum_{l=1}^{K} \mathscr{C}_{li}^{\eta} D(c_l, x_i) \tag{5.12}$$

The method terminates when there is no significant enhancement in the $J_{\mathscr{C}}$ metric compared to the previous $J_{\mathscr{C}}$ value. The determination is made when the difference between the current and previous values of $J_{\mathscr{C}}$ is less than or equal to the threshold value ϵ. The procedural instructions for CrKMd are outlined in Algorithm 11. In order to evaluate the reliability of a specific data point, denoted as $x_i \in X$, inside a given cluster, the Algorithm 11 produces a credibility matrix denoted as $[\mathscr{C}_{li}]$. The symbol $\mathscr{C}_{li}$ denotes the credibility value associated with the data point x_i belonging

Algorithm 11 Steps of the CrKMd

```
Input:
    $X$ : the dataset
    $K$ : the number of clusters
    $\epsilon$ : a small real threshold value, e.g., 0.0001
Output: $[\mathscr{C}_{li}]$, where $1 \leq l \leq K$ and $1 \leq i \leq n$

Randomly select $K$ data points from dataset for initiating $K$ cluster modes
repeat
    Compute $\mathscr{C}_{li}$ for all $n$ points using Equation 5.8
    Use Equation 5.11 to compute new mode
    Use Equation 5.12 to compute the objective, $J_{\mathscr{C}}$
until $|Current\ J_{\mathscr{C}} - Previous\ J_{\mathscr{C}}| \leq \epsilon$
return $[\mathscr{C}_{li}]$ where $1 \leq l \leq K$ and $1 \leq i \leq n$
```

Algorithm 12 Steps of MLCrKMd

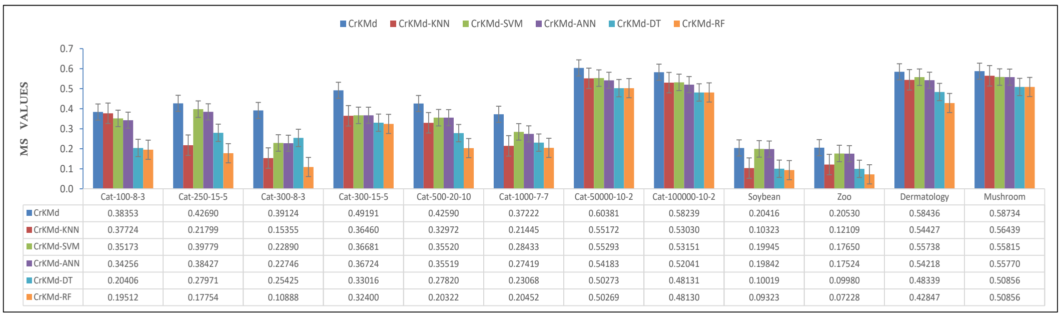
```
Input:
    $X$ : the dataset
    $K$ : the number of cluster
    $[\mathscr{C}_{li}]$ : Credibilistic matrix
Output: $\mathbb{T}_{\mathbb{F}}$ : the final class label vector of $X$

Using Algorithm 11, compute $[\mathscr{C}_{li}]$ for $1 \leq l \leq K$ and $1 \leq i \leq n$
repeat                                        //Build Training dataset
    If $sup_{1 \leq l \leq K} \mathscr{C}_{li} > 0.5, \forall\ l$ of $x_i$ then, put $x_i$ into Training data, $\mathbb{T}$ and corresponding label into class label vector, $\mathbb{T}_{\mathbb{L}}$
until $i <= n$
for each $j$ in $\{K-NN, SVM, ANN, DT, RF\}$ do
    Classify $\mathbb{T}^* = (X - \mathbb{T})$ using $j$th Machine Learning technique, trained by $\mathbb{T}$ and $\mathbb{T}_{\mathbb{L}}$ to get label vector, ${\mathbb{T}_{\mathbb{L}}}^j$
    Compute $MS_j \leftarrow$ MS value of ${\mathbb{T}_{\mathbb{L}}}^j$ against true class label
end for
$\forall j \in \{K-NN, SVM, ANN, DT, RF\}, best \leftarrow \arg\min_{j}\{MS_j\}$
Select ${\mathbb{T}_{\mathbb{L}}}^{best}$
Combine $\mathbb{T}_{\mathbb{L}}$ and ${\mathbb{T}_{\mathbb{L}}}^{best}$ to get final label vector, $\mathbb{T}_{\mathbb{F}}$, where $\mathbb{T}_{\mathbb{F}}$ should be in order of data points in $X$
return $\mathbb{T}_{\mathbb{F}}$
```

to the lth cluster, where $1 \leq l \leq K$ and $1 \leq i \leq n$.

5.2.3 Integration of Machine Learning Techniques with Credibilistic *K*-Modes

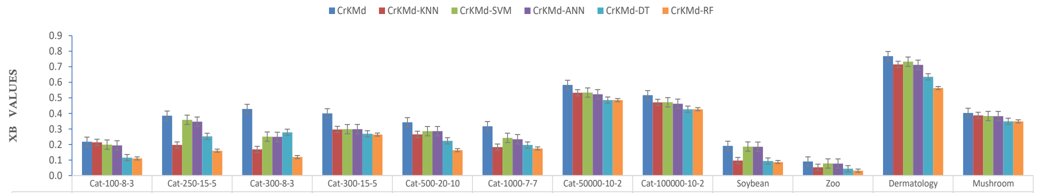
In some instances, Algorithm 11 may provide identical credibility values for a given data point, $x_i \in X$, across many classes, hence posing difficulties in accurately determining the point's precise classification. In this scenario, a particular cluster is randomly selected from a set of many clusters, and the point is then allocated based on the chosen cluster. Nevertheless, CrKMd exhibits this limitation. Motivated by the need to address the limitations of CrKMd and improve the quality of the clustering outcome, more research was conducted to develop a strategy for integrating CrKMd with a machine learning methodology. In order to achieve the intended objective, many established machine learning approaches have been included with CrKMd individually, resulting in the designations CrKMd-KNN, CrKMd-SVM, CrKMd-ANN, CrKMd-DT, and CrKMd-RF. Collectively, these techniques are referred to as MLCrKMd. Subsequently, the optimal outcome is chosen as the ultimate clustering result.

The algorithm for the machine learning integrated CrKMd (MLCrKMd) is outlined in Algorithm 12. The dataset X is first partitioned into two distinct datasets, referred to as the training (certain) and testing (uncertain) data. To accomplish this objective, the credibility matrix $[\mathscr{C}_{li}]$ is used, which is developed from Algorithm 11. The training data, denoted as $\mathbb{T}$, consists of all the $x_i \in X$ that have a distinct cluster associated with them, where this cluster yields the greatest credibility value. Nevertheless, as mentioned earlier, there are some situations in which a data point may possess equivalent credibility across many classifications. The data points that meet the criteria $sup_{1 \leq l \leq K} \mathscr{C}_{li} > 0.5, \forall\, l$ possess a distinct maximum credibility value, which enables the identification of a unique cluster. Consequently, the data that remains uncertain is designated as testing data, whereas the data that has been previously identified as definite is referred to as training data. The testing dataset, denoted as $\mathbb{T}^*$, consists of all x_i values that belong to the set $\{X - \mathbb{T}\}$. Every machine learning approach undergoes training using a set of training data. The prediction of the cluster label for the testing data is performed by each trained machine independently. Hence, in the case of the test dataset, every trained machine produces a distinct label vector. In this study, the Minkowski Score (MS) [168] is used as a metric to assess the cluster validity index of each label vector for every trained machine. The selection of the label vector is performed with the objective of determining the optimal clustering outcome by considering the most favourable MS value. In the context of real-world datasets when the cluster labels are unknown, it is recommended to use the Xie-Beni (XB) index [208] instead of the Minkowski Score as outlined in Algorithm 12.



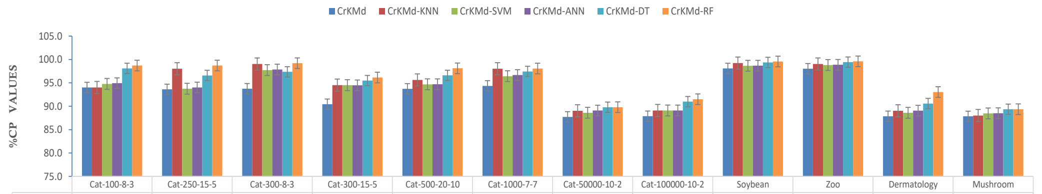
	Cat-100-8-3	Cat-250-15-5	Cat-300-8-3	Cat-300-15-5	Cat-500-20-10	Cat-1000-7-7	Cat-50000-10-2	Cat-100000-10-2	Soybean	Zoo	Dermatology	Mushroom
CrKMd	0.38353	0.42690	0.39124	0.49191	0.42590	0.37222	0.60381	0.58239	0.20416	0.20530	0.58436	0.58734
CrKMd-KNN	0.37724	0.21799	0.15355	0.36460	0.32972	0.21445	0.55172	0.53030	0.10323	0.12109	0.54427	0.56439
CrKMd-SVM	0.35173	0.39779	0.22890	0.36681	0.35520	0.28433	0.55293	0.53151	0.19945	0.17650	0.55738	0.55815
CrKMd-ANN	0.34256	0.38427	0.22746	0.36724	0.35519	0.27419	0.54183	0.52041	0.19842	0.17524	0.54218	0.55770
CrKMd-DT	0.20406	0.27971	0.25425	0.33016	0.27820	0.23068	0.50273	0.48131	0.10019	0.09980	0.48339	0.50856
CrKMd-RF	0.19512	0.17754	0.10888	0.32400	0.20322	0.20452	0.50269	0.48130	0.09323	0.07228	0.42847	0.50856

Figure 5.1: Average values of MS for CrKMd integrated ML techniques for different datasets



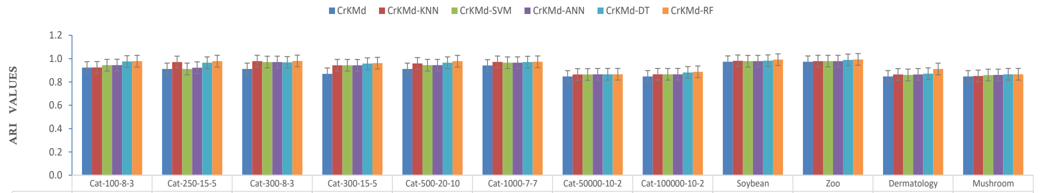
	Cat-100-8-3	Cat-250-15-5	Cat-300-8-3	Cat-300-15-5	Cat-500-20-10	Cat-1000-7-7	Cat-50000-10-2	Cat-100000-10-2	Soybean	Zoo	Dermatology	Mushroom
CrKMd	0.21733	0.38508	0.42830	0.40043	0.34270	0.31751	0.58302	0.51672	0.19107	0.09051	0.76794	0.40305
CrKMd-KNN	0.21377	0.19663	0.16810	0.29679	0.26531	0.18293	0.53272	0.47051	0.09661	0.05338	0.71525	0.38730
CrKMd-SVM	0.19931	0.35882	0.25058	0.29859	0.28581	0.24254	0.53389	0.47158	0.18666	0.07781	0.73248	0.38302
CrKMd-ANN	0.19412	0.34662	0.24901	0.29894	0.28580	0.23389	0.52317	0.46173	0.18570	0.07726	0.71250	0.38271
CrKMd-DT	0.11563	0.25231	0.27834	0.26876	0.22385	0.19677	0.48542	0.42704	0.09377	0.04400	0.63525	0.34899
CrKMd-RF	0.11057	0.16015	0.11919	0.26374	0.16352	0.17446	0.48538	0.42703	0.08725	0.03187	0.56308	0.34899

Figure 5.2: Average values of XB for CrKMd integrated ML techniques for different datasets



	Cat-100-8-3	Cat-250-15-5	Cat-300-8-3	Cat-300-15-5	Cat-500-20-10	Cat-1000-7-7	Cat-50000-10-2	Cat-100000-10-2	Soybean	Zoo	Dermatology	Mushroom
CrKMd	94.00894	93.60990	93.74056	90.42932	93.71924	94.35636	87.72170	87.86200	98.08240	98.02162	87.85600	87.83300
CrKMd-KNN	94.00949	98.00933	99.02970	94.52085	95.62002	98.01586	89.02320	89.09150	99.23191	99.03020	89.02512	88.01500
CrKMd-SVM	94.77576	93.73808	97.72162	94.50390	94.68517	96.42385	88.60260	89.09030	98.65446	98.81188	88.59160	88.46200
CrKMd-ANN	94.92120	94.00752	97.85420	94.47180	94.70524	96.67245	89.08140	89.10210	98.67218	98.85473	89.05731	88.50430
CrKMd-DT	98.09578	96.58607	97.34915	95.47993	96.59482	97.43924	89.79110	90.98380	99.36707	99.43152	90.57714	89.35645
CrKMd-RF	98.69802	98.70707	99.19732	96.13601	98.10197	98.06146	89.79210	91.49360	99.57212	99.59643	93.03084	89.35645

Figure 5.3: Average values of %CP for CrKMd integrated ML methods for different datasets



	Cat-100-8-3	Cat-250-15-5	Cat-300-8-3	Cat-300-15-5	Cat-500-20-10	Cat-1000-7-7	Cat-50000-10-2	Cat-100000-10-2	Soybean	Zoo	Dermatology	Mushroom
CrKMd	0.92385	0.91152	0.91156	0.87001	0.91153	0.94138	0.84701	0.84759	0.97363	0.97308	0.84755	0.84721
CrKMd-KNN	0.92496	0.97117	0.97884	0.94358	0.95904	0.97231	0.86428	0.86589	0.98121	0.97888	0.86430	0.85130
CrKMd-SVM	0.94379	0.91155	0.97077	0.94332	0.94376	0.96485	0.86414	0.86588	0.97808	0.97878	0.86010	0.85920
CrKMd-ANN	0.94450	0.92254	0.97098	0.94276	0.94378	0.96551	0.86579	0.86608	0.97812	0.97879	0.86470	0.85980
CrKMd-DT	0.97531	0.96501	0.96768	0.95702	0.96503	0.96995	0.86616	0.88187	0.98216	0.98911	0.87102	0.86609
CrKMd-RF	0.97814	0.97867	0.98030	0.96121	0.97784	0.97313	0.86617	0.88682	0.99103	0.99347	0.91120	0.86609

Figure 5.4: Average values of ARI for CrKMd integrated ML methods for different datasets

5.3 Experimental Results

As was previously said, fuzzy and probabilistic clustering approaches are unable to deliver accurate clustering results because of the noisy environment and coincident clustering issue. Additionally, the categorical character of the data adds to the complexity of grouping. MLCrKMd has been suggested in this thesis as a solution to these problems with the current approaches. By analyzing the quality of the clustering outcomes of diverse datasets using various cluster validity indices, the effectiveness of the technique may be determined. To evaluate the performance of the proposed technique in comparison to existing methods, a total of eight synthetic datasets (referred to as *Cat-100-8-3*, *Cat-250-15-5*, *Cat-300-8-3*, *Cat-300-15-5*, *Cat-500-20-10*, *Cat-1000-7-7*, *Cat-50000-10-2*, and *Cat-100000-10-2*) as well as four real-life datasets (*Soybean*, *Zoo*, *Dermatology*, and *Mushroom*) are utilized. The evaluation of clustering techniques involves the utilization of various metrics, such as the Minkowski Score (MS) [168], Percentage of Correct Pair (%CP) [35], Adjusted Rand Index (ARI) [169], and Xie-Beni (XB) [208] index. These metrics serve as measures to assess the effectiveness and accuracy of clustering algorithms. The employment of four distinct measurements ensures the quality of the clustering solutions due to their various computing methods.

Six synthetic small and medium-sized datasets are generated at datgen portal[1], while four real-life benchmark datasets are taken from UCI machine learning data repository[2]. Despite having a significant number of attributes for use as a reference point in categorical clustering research, the available datasets from the UCI machine learning repository are of a medium size. As a result, two extra sizable datasets (referred to as *Cat-50000-10-2* and *Cat-100000-10-2*) are retrieved from [218]. Chapter 2 and 4 have described about the synthetic datasets such as *Cat-100-8-3*, *Cat-250-15-5*, *Cat-300-8-3*, *Cat-300-15-5*, *Cat-500-20-10*, *Cat-1000-7-7* and real-life datasets such as *Soybean*, *Zoo*, *Dermatology*, and *Mushroom*. Table 5.1 and 5.2 describe the details of other synthetic and the real life datasets respectively.

5.3.1 Data Sets

Eight synthetic and four actual datasets are described in Table 5.1 and 5.2 respectively.

5.3.2 Input parameters and performance metrics

The inputs used in the CrKMd algorithm consist of a dataset denoted as X, the desired number of clusters denoted as K, and an extremely small threshold denoted

[1] Synthetic, http://www.datgen.com

[2] Real life datasets are taken from http://www.ics.uci.edu/~mlearn/MLRepository.html

Table 5.1: Specification of synthetic datasets

Datasets	Description
Cat-50000-10-2	The dataset [218] consists of 50,000 data points, each including 10 characteristics. These data points are organised into two distinct clusters.
Cat-100000-10-2	The dataset [218] consists of 100,000 data points, with each point possessing 10 properties. These points are organised into two distinct clusters.

Table 5.2: Specification of real life datasets

Datasets	Description
Dermatology	The dataset pertaining to Dermatology has a total of 34 characteristics, with 33 of them being linear valued and one feature being nominal in nature. There are a total of 366 cases of differential diagnosis pertaining to erythemato-squamous illnesses. The patient's qualities include both clinical attributes and histopathological traits. The dataset is divided into six distinct classifications, namely psoriasis, seborrheic dermatitis, lichen planus, pityriasis rosea, chronic dermatitis, and pityriasis rubra pilaris.

Table 5.3: Average values of XB, over 20 runs of CrKMd for different values of η

Datasets	$\eta = 1.5$	$\eta = 2$	$\eta = 2.5$
Cat-100-8-3	0.2517	0.2173	0.2294
Cat-250-15-5	0.4196	0.3851	0.3929
Cat-300-8-3	0.4421	0.4283	0.4388
Cat-300-15-5	0.4521	0.4004	0.4118
Cat-500-20-10	0.3943	0.3427	0.3695
Cat-1000-7-7	0.3595	0.3175	0.3300
Cat-50000-10-2	0.6395	0.5830	0.6029
Cat-100000-10-2	0.5639	0.5167	0.5320
Soybean	0.2530	0.1911	0.2099
Zoo	0.1295	0.0905	0.1100
Dermatology	0.8320	0.7679	0.7826
Mushroom	0.4693	0.4031	0.4184

as $\epsilon = 0.0001$. A sensitivity analysis was conducted to determine the optimal value of the crucial parameter, η, used in Equation 5.12 of the CrKMd model. According to the data shown in Table 5.3, it can be seen that the average XB value exhibits superior performance when η is set to 2. Consequently, this particular value of η was selected for use in our experimental study. Additional input parameters, such as the number of trees in CrKMd-RF, the kernel in CrKMd-SVM,

Table 5.4: Average values of MS, %CP, ARI and XB for Synthetic datasets.

Datasets	Method	MS	%CP	ARI	XB	Datasets	Method	MS	%CP	ARI	XB
Cat-100-8-3	KMd	0.88202	76.18283	0.73107	0.49857	**Cat-250-15-5**	KMd	0.77524	80.55590	0.78620	0.90637
	FKMd	0.79713	79.91267	0.78020	0.38548		FKMd	0.62713	87.21280	0.84370	0.61401
	AL	0.76040	81.13110	0.79115	0.37838		AL	0.64680	86.40430	0.83010	0.62431
	TSFKMd	0.73736	83.11590	0.79410	0.36647		TSFKMd	0.55319	88.60240	0.86040	0.40829
	MMR	0.55900	88.29480	0.85610	0.36129		MMR	0.52321	89.09280	0.86590	0.40254
	G-ANMI	0.50471	89.44180	0.86610	0.35846		G-ANMI	0.49988	90.29160	0.86670	0.39556
	MoDEFCCD	0.36240	94.55710	0.94360	0.20536		MoDEFCCD	0.34070	94.93020	0.94520	0.30732
	PKMd	0.39101	93.82249	0.91160	0.22157		PKMd	0.43137	93.01345	0.90380	0.38911
	SCC	0.38982	94.00623	0.92234	0.21846		SCC	0.43009	93.02561	0.90450	0.38821
	PM-FGCA	0.38979	94.00623	0.92235	0.21846		PM-FGCA	0.43008	93.02562	0.90480	0.38821
	CrKMd	0.38353	94.00894	0.92385	0.21733		CrKMd	0.42690	93.60990	0.91152	0.38508
	MLCrKMd	**0.19512**	**98.69802**	**0.97814**	**0.11057**		**MLCrKMd**	**0.17754**	**98.70707**	**0.97867**	**0.16015**
Cat-300-8-3	KMd	0.77440	80.63140	0.79000	0.92764	**Cat-300-15-5**	KMd	0.84465	77.51644	0.75260	0.89795
	FKMd	0.37300	94.27590	0.93710	0.40619		FKMd	0.74558	82.68290	0.79260	0.60884
	AL	0.75890	81.31090	0.79133	0.42616		AL	0.73550	83.51650	0.79550	0.51572
	TSFKMd	0.34283	94.90220	0.94380	0.38320		TSFKMd	0.69895	84.76120	0.80990	0.50332
	MMR	0.31699	96.15320	0.96370	0.35164		MMR	0.64432	86.63930	0.83160	0.49115
	G-ANMI	0.20322	98.10197	0.97784	0.30339		G-ANMI	0.57669	87.91220	0.84760	0.47340
	MoDEFCCD	0.44170	92.02560	0.89321	0.48354		MoDEFCCD	0.50160	89.80470	0.86620	0.40832
	PKMd	0.40416	93.72930	0.91154	0.44245		PKMd	0.68641	85.00500	0.81558	0.55876
	SCC	0.40183	93.73001	0.91154	0.44196		SCC	0.66329	86.00300	0.82993	0.51942
	PM-FGCA	0.40180	93.73001	0.91154	0.44195		PM-FGCA	0.68729	84.83950	0.81431	0.51570
	CrKMd	0.39124	93.74056	0.91156	0.42830		CrKMd	0.49191	90.42932	0.87001	0.40043
	MLCrKMd	**0.10888**	**99.19732**	**0.98030**	**0.11919**		**MLCrKMd**	**0.32400**	**96.13601**	**0.96121**	**0.26374**
Cat-500-20-10	KMd	0.94438	75.28247	0.71026	0.96647	**Cat-1000-7-7**	KMd	0.94989	74.73140	0.71008	0.73542
	FKMd	0.81305	78.92458	0.76100	0.71037		FKMd	0.81150	78.99450	0.76210	0.69765
	AL	0.81350	78.85420	0.76090	0.65284		AL	0.80320	79.17520	0.76621	0.69648
	TSFKMd	0.75597	81.42737	0.79135	0.61349		TSFKMd	0.74440	82.77160	0.79280	0.65431
	MMR	0.71532	84.20744	0.79830	0.55395		MMR	0.72522	83.55489	0.79612	0.61277
	G-ANMI	0.67155	85.82147	0.82670	0.36421		G-ANMI	0.66527	85.98743	0.82991	0.59413
	MoDEFCCD	0.48210	90.87090	0.87122	0.38792		MoDEFCCD	0.37550	94.01120	0.92751	0.32030
	PKMd	0.65582	86.31600	0.82997	0.52770		PKMd	0.72522	83.55489	0.79612	0.52511
	SCC	0.62032	87.22440	0.84530	0.49173		SCC	0.58359	87.85910	0.84757	0.48246
	PM-FGCA	0.63483	87.19580	0.84302	0.51629		PM-FGCA	0.65829	86.30100	0.82997	0.58393
	CrKMd	0.42590	93.71924	0.91153	0.34270		CrKMd	0.37222	94.35636	0.94138	0.31751
	MLCrKMd	**0.20322**	**98.10197**	**0.97784**	**0.16352**		**MLCrKMd**	**0.20452**	**98.06146**	**0.97313**	**0.17446**
Cat-50000-10-2	KMd	0.94438	75.28247	0.71026	0.69900	**Cat-100000-10-2**	KMd	0.94989	74.73140	0.71008	0.62330
	FKMd	0.81305	78.92458	0.76100	0.68693		FKMd	0.81150	78.99450	0.76210	0.61221
	AL	0.81350	78.85420	0.76090	0.69698		AL	0.80320	79.17520	0.76621	0.62145
	TSFKMd	0.75597	81.42737	0.79135	0.67615		TSFKMd	0.74440	82.77160	0.79280	0.60231
	MMR	0.71532	84.20744	0.79830	0.65688		MMR	0.72522	83.55489	0.79612	0.58460
	G-ANMI	0.67155	85.82147	0.82670	0.64887		G-ANMI	0.66527	85.98743	0.82991	0.57723
	MoDEFCCD	0.48210	90.87090	0.87122	0.63959		MoDEFCCD	0.37550	94.01120	0.92751	0.56871
	PKMd	0.65582	86.31600	0.82997	0.61950		PKMd	0.72522	83.55489	0.79612	0.55024
	SCC	0.64001	87.19540	0.84298	0.61330		SCC	0.61995	87.63160	0.84546	0.53295
	PM-FGCA	0.65295	86.32300	0.82998	0.61664		PM-FGCA	0.65053	86.33520	0.82999	0.57799
	CrKMd	0.42590	93.71924	0.91153	0.58302		CrKMd	0.37222	94.35636	0.94138	0.51672
	MLCrKMd	**0.20322**	**98.10197**	**0.97784**	**0.48538**		**MLCrKMd**	**0.20452**	**98.06146**	**0.97313**	**0.42703**

and the number of K in CrKMd-KNN, are specified as 1000, RBF (Radial Basis Function), and 3, respectively. Furthermore, the CrKMd-ANN model has used a single hidden layer, with the number of hidden neurons set to two-thirds of the number of input neurons. The robust backpropagation algorithm, including weight backtracking, has been utilized for neural network computation. All of the aforementioned criteria are established using either experimental methods or by following the existing literature. The input parameters for TSFKMd, MMR, G-ANMI, and PKMd algorithms are identical to those used in the studies referenced as [155], [135], [106], and [123]. It should be noted that the procedures are iteratively conducted until they reach a state of convergence and provide a final result. In order to assess the effectiveness of the performance, we have used external cluster

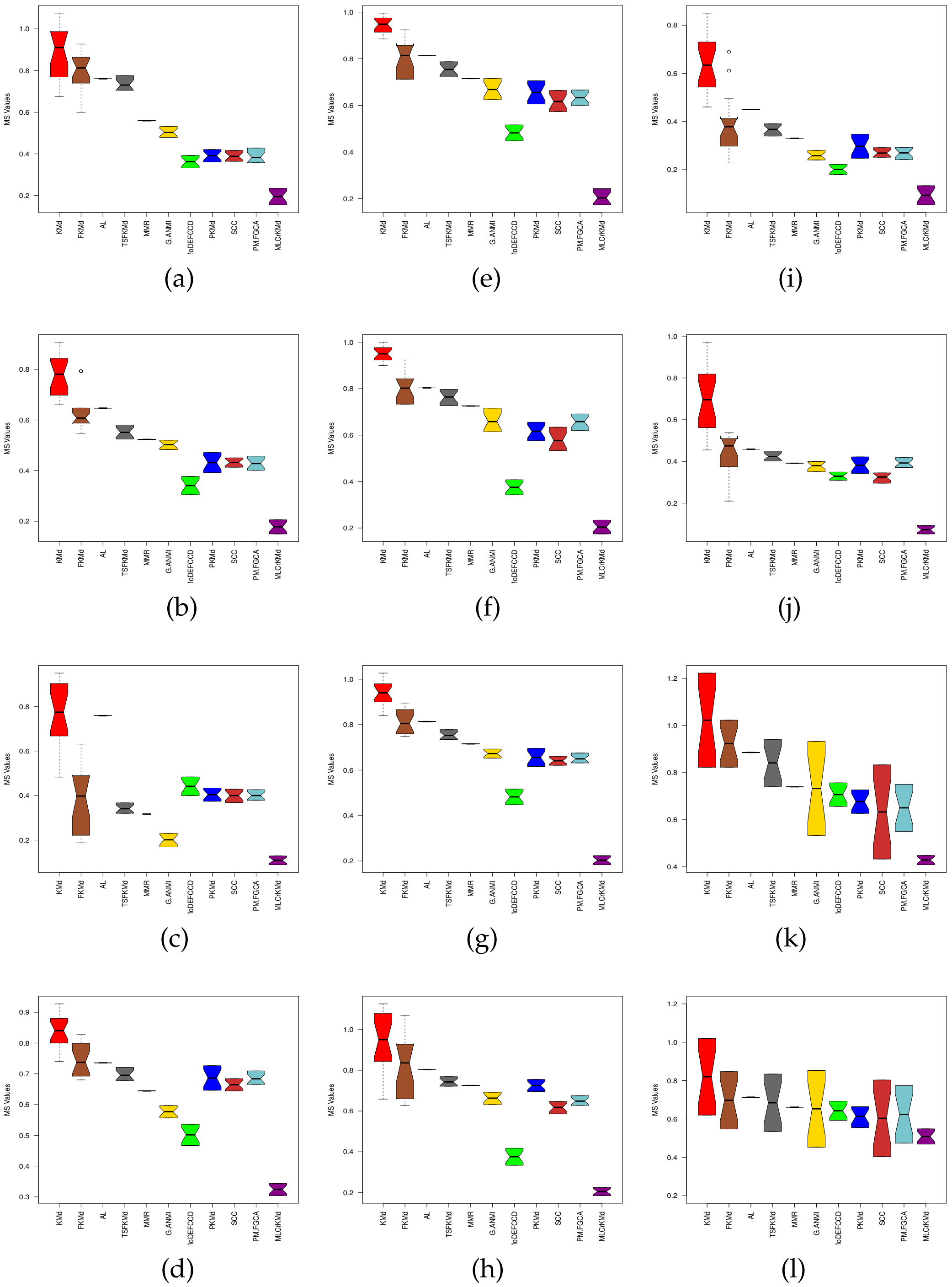
Table 5.5: Average values of MS, %CP, ARI and XB for Real datasets.

Datasets	Method	MS	%CP	ARI	XB	Datasets	Method	MS	%CP	ARI	XB
Soybean	KMd	0.64363	86.79560	0.83610	0.57577	**Zoo**	KMd	0.68839	84.82920	0.81370	0.30697
	FKMd	0.39077	93.96750	0.92000	0.28765		FKMd	0.43895	92.29820	0.89680	0.17982
	AL	0.44980	91.72570	0.89224	0.30922		AL	0.45840	91.49410	0.88682	0.23293
	TSFKMd	0.36806	94.45920	0.94220	0.41140		TSFKMd	0.42744	93.18530	0.91140	0.17439
	MMR	0.33104	95.34930	0.94820	0.37002		MMR	0.39099	93.89980	0.91160	0.17033
	G-ANMI	0.25899	96.95324	0.96620	0.27496		G-ANMI	0.37686	94.00950	0.92500	0.16500
	MoDEFCCD	0.20070	98.30110	0.97785	0.18783		MoDEFCCD	0.32920	95.62740	0.95998	0.14513
	PKMd	0.29713	96.32843	0.96462	0.27808		PKMd	0.38187	94.00945	0.92491	0.16835
	SCC	0.27035	96.72941	0.96560	0.24782		SCC	0.32226	96.13783	0.96243	0.13937
	PM-FGCA	0.27032	96.72985	0.96560	0.23529		PM-FGCA	0.39639	93.73856	0.91155	0.17121
	CrKMd	0.20416	98.08240	0.97363	0.19107		CrKMd	0.20530	98.02162	0.97308	0.09051
	MLCrKMd	**0.09323**	**99.57212**	**0.99103**	**0.08725**		**MLCrKMd**	**0.07228**	**99.59643**	**0.99347**	**0.03187**
Dermatology	KMd	1.02256	71.08340	0.66835	1.34380	**Mushroom**	KMd	0.81998	78.79645	0.76005	0.56270
	FKMd	0.92276	75.86546	0.72835	1.21265		FKMd	0.69724	84.80500	0.81120	0.47847
	AL	0.88536	76.12840	0.73006	1.16350		AL	0.71355	84.35100	0.80160	0.48966
	TSFKMd	0.84113	78.12498	0.75864	1.10537		TSFKMd	0.68428	85.52400	0.81960	0.46958
	MMR	0.73970	83.00900	0.79390	0.97208		MMR	0.66143	86.01700	0.82996	0.45390
	G-ANMI	0.73221	83.53410	0.79561	0.96224		G-ANMI	0.65269	86.32400	0.82998	0.44790
	MoDEFCCD	0.70654	84.62750	0.80810	0.92850		MoDEFCCD	0.64258	86.98240	0.83920	0.44096
	PKMd	0.67673	85.76540	0.82657	0.88933		PKMd	0.61429	87.70590	0.84599	0.42155
	SCC	0.63286	87.20030	0.84315	0.83258		SCC	0.60327	87.79140	0.84713	0.41038
	PM-FGCA	0.65035	86.33950	0.83000	0.87384		PM-FGCA	0.62394	87.21590	0.84450	0.44006
	CrKMd	0.58436	87.85600	0.84755	0.76794		CrKMd	0.58734	87.83300	0.84721	0.40305
	MLCrKMd	**0.42847**	**93.03084**	**0.91120**	**0.56308**		**MLCrKMd**	**0.50856**	**89.35645**	**0.86609**	**0.34899**

validity indices, namely the Minkowski Score (MS) [168], the Percentage of Correct Pair (%CP) [35], the Adjusted Rand Index (ARI) [169], as well as an internal cluster validity index, the Xie-Beni index (XB) [208]. Lower values of MS and XB indicate better outcomes. In a similar vein, a larger value of the Adjusted Rand Index (ARI) is indicative of superior outcomes, but a value of zero (0%) and 100% for the percentage of Correctly Predicted (%CP) values indicate the worse and optimal results, respectively. Every method is developed in Matlab and executed on a computer with an Intel Core i5-2410M 2.30 GHz processor, 4GB of RAM, and Windows 7 as the operating system.

5.3.3 Results and Discussion

The results of CrKMd are discussed first in this section, followed by those of MLCrKMd. Four real-world datasets and eight synthetic datasets are utilized for this. CrKMd has been seen to occasionally generate results with the same credibility as a data point for numerous clusters. In this case, a cluster is chosen at random among all the clusters for which the data point is equally credible. In order to compute the cluster validity indices, the data point is then assigned to the chosen cluster. The average Minkowski Score (MS) values computed by CrKMd for all datasets are shown in Figure 5.1. In the datasets *Cat-100-8-3*, *Cat-250-15-5*, *Cat-300-8-3*, *Cat-300-15-5*, *Cat-500-20-10*, *Cat-1000-7-7*, *Cat-50000-10-2*, *Cat-100000-10-2*, *Soybean*, *Zoo*, *Dermatology* and *Mushroom* , for instance, the average MS values are 0.38353, 0.42690, 0.39124, 0.49191, 0.42590, 0.37222, 0.42590, 0.37222,



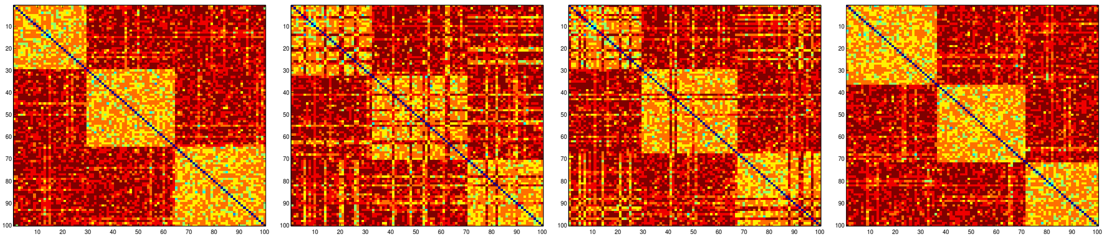
(a) (e) (i)

(b) (f) (j)

(c) (g) (k)

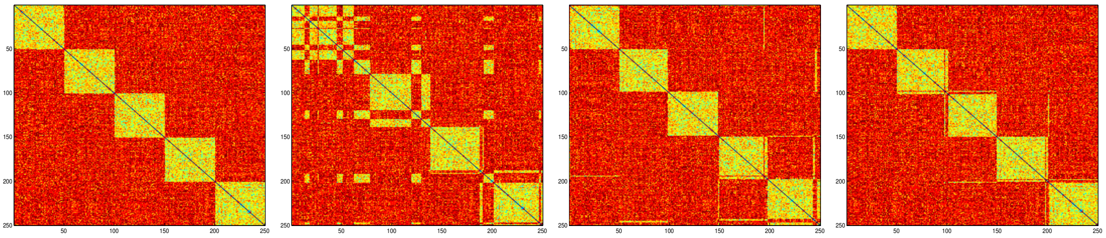
(d) (h) (l)

Figure 5.5: Boxplot of MS values of different clustering techniques for (a) Cat-100-8-3 (b) Cat-250-15-5 (c) Cat-300-8-3 (d) Cat-300-15-5 (e) Cat-500-20-10 (f) Cat-1000-7-7 (g) Cat-50000-10-2 (h) Cat-100000-10-2 (i) Soyabean (j) Zoo (k) Dermatology (l) Mushroom



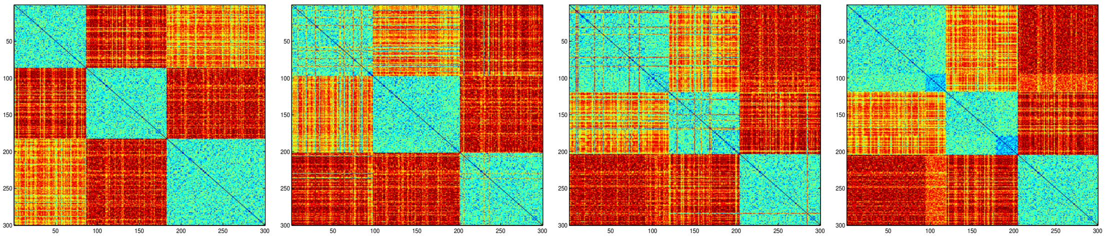
(a) (b) (c) (d)

Figure 5.6: VAT plot of (a) True clusters and the clusters produced by (b) FKMd (c) CrKMd (d) MLCrKMd for Cat-100-8-3 dataset



(a) (b) (c) (d)

Figure 5.7: VAT plot of (a) True clusters and the clusters produced by (b) FKMd (c) CrKMd (d) MLCrKMd for Cat-250-15-5 dataset



(a) (b) (c) (d)

Figure 5.8: VAT plot of (a) True clusters and the clusters produced by (b) FKMd (c) CrKMd (d) MLCrKMd for Cat-300-8-3 dataset

0.20416, 0.20530, 0.58436 and 0.58734 respectively. The average Xie-Beni index (XB) of CrKMd for all datasets are reported in Figure 5.2. In contrast to other cutting-edge techniques, the Tables 5.4 and 5.5 give the average values of external cluster validity indices like %CP, ARI, and internal cluster validity indices like XB score. The external cluster validity index assesses the effectiveness of the clustering solution using the acknowledged cluster labels. On the other hand, the

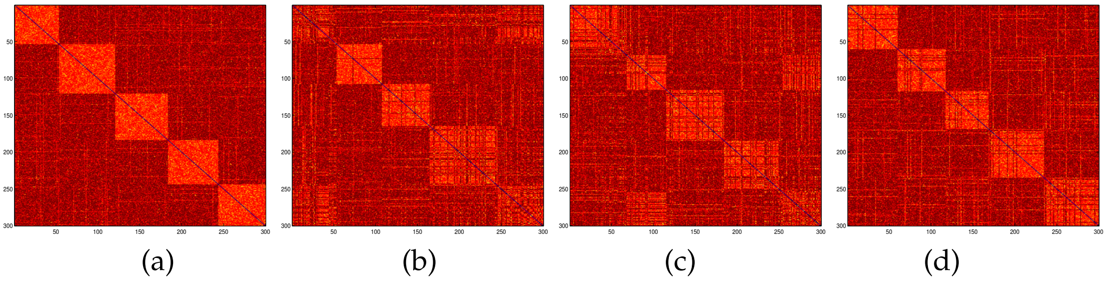


Figure 5.9: VAT plot of (a) True clusters and the clusters produced by (b) FKMd (c) CrKMd (d) MLCrKMd for Cat-300-15-5 dataset

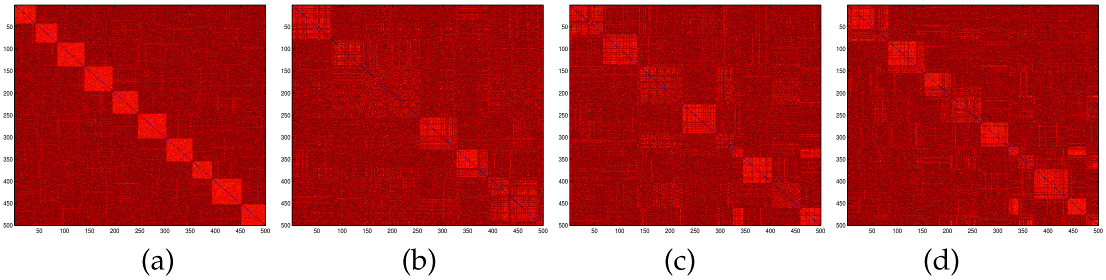


Figure 5.10: VAT plot of (a) True clusters and the clusters produced by (b) FKMd (c) CrKMd (d) MLCrKMd for Cat-500-20-10 dataset

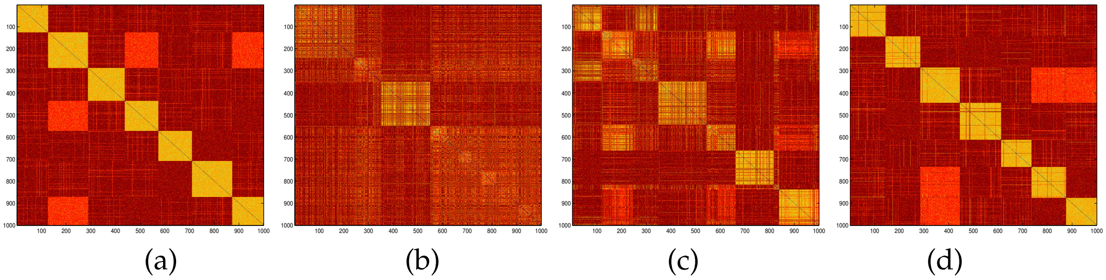


Figure 5.11: VAT plot of (a) True clusters and the clusters produced by (b) FKMd (c) CrKMd (d) MLCrKMd for Cat-1000-7-7 dataset

internal clustering metric is used to evaluate the clustering solution in terms of the geometrical properties of the clusters. For each of the datasets used in the experiment, true clusters are known. As a consequence, the success of the techniques is assessed using both the exterior and internal cluster validity indices. The results clearly demonstrate that for all of the datasets, CrKMd outperforms other methods. However, due to the anomaly described above, there is a chance

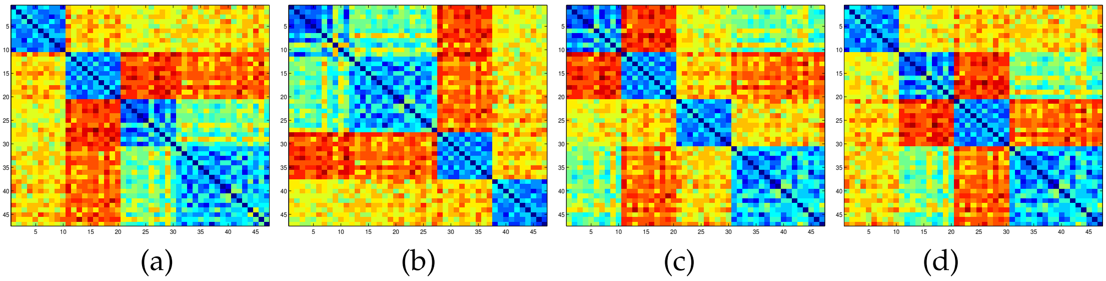


Figure 5.12: VAT plot of (a) True clusters and the clusters produced by (b) FKMd (c) CrKMd (d) MLCrKMd for Soybean dataset

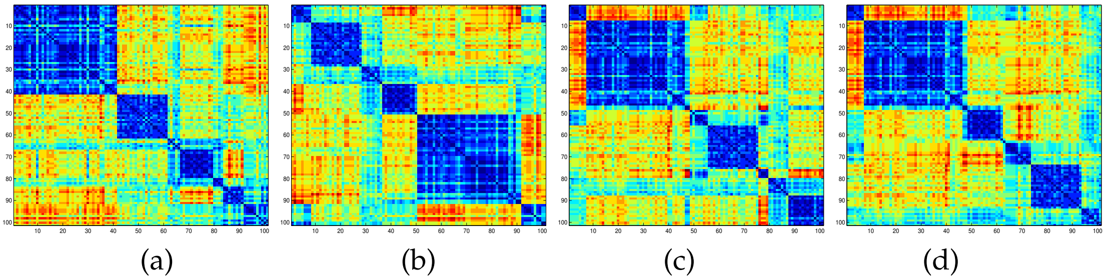


Figure 5.13: VAT plot of (a) True clusters and the clusters produced by (b) FKMd (c) CrKMd (d) MLCrKMd for Zoo dataset

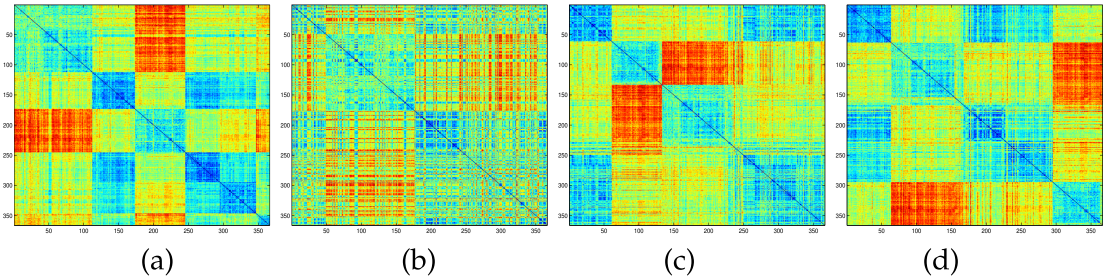
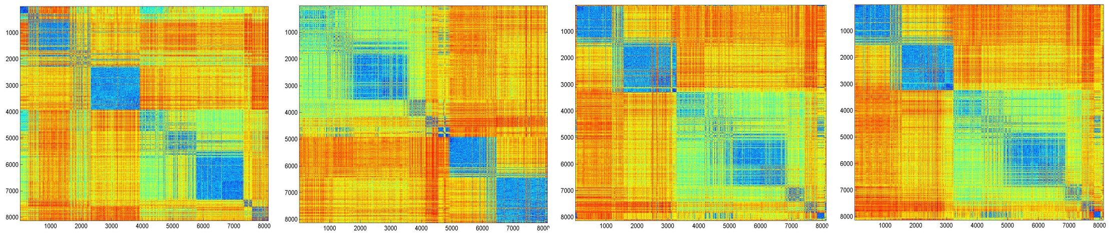


Figure 5.14: VAT plot of (a) True clusters and the clusters produced by (b) FKMd (c) CrKMd (d) MLCrKMd for Dermatology dataset

of misclassification in which CrKMd fails to recognize the distinct cluster of a data point since many clusters share the same credibility. Several machine learning approaches are merged with CrKMd, referred as CrKMd-KNN, CrKMd-SVM, CrKMd-ANN, CrKMd-DT, and CrKMd-RF, to address this problem and enhance the final clustering outcome. In the given context, individual machine learning techniques are trained using the data for which CrKMd has effectively identified



(a) (b) (c) (d)

Figure 5.15: VAT plot of (a) True clusters and the clusters produced by (b) FKMd (c) CrKMd (d) MLCrKMd for Mushroom dataset

a distinct cluster. Subsequently, the machine is used to categorize the remaining data. The most optimal outcome is achieved by using machine learning integrated CrKMd strategies, namely MLCrKMd, which relies on the calculated values of diverse cluster validity indices. Figures 5.1 and 5.2 illustrate the comparative analysis of integrated methods in terms of average MS and XB values. Additionally, Figures 5.3 and 5.4 display the average %CP and ARI values. The experiments have repeatedly shown that CrKMd-RF produces superior results across various synthetic and real-world datasets. As an example, the average values of MS for CrKMd, CrKMd-KNN, CrKMd-SVM, CrKMd-ANN, CrKMd-DT, and CrKMd-RF on dataset *Cat-100-8-3* are 0.38353, 0.37724, 0.35173, 0.34256, 0.20406, and 0.19512, correspondingly. Furthermore, it has been shown that the outcomes obtained from CrKMd-DT and CrKMd-RF algorithms when applied to the Mushroom dataset are indistinguishable. Similarly, the results obtained from CrKMd-SVM and CrKMd-ANN algorithms are equivalent for almost all datasets. In the present study, it was shown that CrKMd-RF exhibited superior performance compared to other integrated methodologies. The ultimate outcome of the MLCrKMd analysis is often denoted as the CrKMd-RF outcome. In a similar vein, in the real-world context, MLCrKMd may be seen as denoting the outcomes of an alternative integrated approach that demonstrates superior performance on a distinct dataset. The performance of MLCrKMd is compared to other commonly used state-of-the-art approaches. In order to achieve the desired objective, various well-known techniques have been chosen, encompassing traditional hard clustering (KMd), its fuzzy variant (FKMd), hierarchical clustering (AL), Tabu search-based fuzzy KMd (TSFKMd), rough set-based algorithm (MMR), G-ANMI, Modified DEFCCD (MoDEFCCD), possibilistic version of KMd (PKMd), subspace clustering of categories (SCC), and partition-and-merge-based fuzzy genetic clustering algorithm (PM-FGCA). Tables 5.4 and 5.5 provide the average values of MS, %CP, ARI, and XB scores over 20 iterations for eight synthetic datasets and four real-life datasets.

As an example, the average values of MS of KMd, FKMd, AL, TSFKMd, MMR, G-ANMI, MoDEFCCD, PKMd, SCC, PK-FGCA, and MLCrKMd, for the *Cat-250-15-5* dataset are reported as 0.77524, 0.62713, 0.64680, 0.55319, 0.52321, 0.49988, 0.34070, 0.43137, 0.43009, 0.43008, and 0.17754, respectively.

The comparative findings have been visually shown via use of boxplots and VAT plots. The boxplots of various techniques on MS values are shown in Figure 5.5. Figures 5.6 - 5.15 show the VAT plots for the datasets: *Cat-100-8-3*, *Cat-250-15-5*, *Cat-300-8-3*, *Cat-300-15-5*, *Cat-500-20-10*, *Cat-1000-7-7*, *Soyabean*, *Zoo*, *Dermatology*, and *Mushroom*. Nevertheless, the generation of VAT charts for the *Cat-50000-10-2* and *Cat-100000-10-2* datasets has been hindered by their substantial size. The comparison study includes the VAT plot, which displays the genuine cluster label of each dataset, as well as the calculated cluster labels of FKMd, CrKMd, and MLCrKMd. As an example, Figures 5.6(a), (b), (c) and (d) show the accurate cluster label, calculated cluster label of FKMd, CrKMd and MLCrKMd for the dataset *Cat-100-8-3*.

5.3.4 Statistical Significance Test of the Clustering Results

To evaluate the statistical significance of the outcomes produced by MLCrKMd, additional analysis were conducted using paired t-tests [219] and Friedman tests [204]. The paired t-test was conducted by pairing MLCrKMd with eleven additional techniques (KMd, FKMd, AL, TSFKMd, MMR, G-ANMI, MoDEFCCD, PKMd, SCC, and PM-FGCA), using 20 MS values for each technique and dataset. The t-test was then conducted with a significance threshold of 5%. In this scenario, the alternative hypothesis is considered valid when the *p-values* is below 0.05. Conversely, the null hypothesis suggests that there is no statistically significant difference between the paired sets of 20 MS values. Furthermore, the False Discovery Rate (FDR) as proposed by Benjamini and Hochberg [220] has been used to adjust the *p-values* obtained from the t-test. These adjusted *p-values* are shown in Table 5.6. The statistical significance of the MS values obtained by MLCrKMd is clear from Table 5.6, since all the *p-values* are less than 0.05. This indicates that the observed results are not likely to have happened by chance. When doing the Friedman test, the computation of the average rank ($\mathcal{R}_j$) for all techniques and the chi-square (χ^2) value is performed in the following manner.

$$\mathcal{R}_j = \frac{1}{N}\sum_i r_i^j \tag{5.13}$$

and

$$\chi^2 = \frac{12N}{Q(Q+1)}\left[\sum_j \mathcal{R}_j^2 - \frac{Q(Q+1)^2}{4}\right] \tag{5.14}$$

In this context, r_i^j represents the ranking of the jth method for the ith dataset. The variables N and Q correspond to the total number of datasets and methods, respectively. The Friedman test follows a chi-squared distribution (χ^2) with $(Q-1)$ degrees of freedom. In this particular scenario, the null hypothesis posits that there are no significant disparities in the results achieved by the different methodologies, but the alternative hypothesis posits the contrary. The average rank of different methods was calculated using Equations 5.13 and 5.14. The results are shown in Table 5.7. Additionally, the chi-square (χ^2) value was computed as 98.012, with a corresponding *p-values* of 1.1102E-16 at a significance level of 5%. By rejecting the null hypothesis and asserting the superiority of MLCrKMd over other techniques, it is evident that there are statistically significant differences in the data obtained from the various methods. Nevertheless, it has been observed that the MLCrKMd method necessitates a greater amount of processing power and storage capacity in comparison to fuzzy and probabilistic clustering algorithms. This is due to the need of computing and storing a larger number of credibilistic measures. Undoubtedly, the identified limitation of MLCrKMd presents an opportunity for future research endeavours to solve.

Table 5.6: FDR adjusted *p-values* on synthetic and real life datasets by comparing MLCrKMd with other methods

Datasets	**KMd**	**FKMd**	**AL**	**TSFKMd**	**MMR**	**G-ANMI**	**MoDEFCCD**	**PKMd**	**SCC**	**PM-FGCA**
Cat-100-8-3	1.13e-18	6.72e-21	1.38e-24	2.14e-37	4.27e-21	5.05e-28	6.78e-26	4.24e-27	1.42e-24	3.01e-29
Cat-250-15-5	2.64e-20	4.06e-18	6.55e-26	1.09e-34	1.78e-23	9.06e-20	1.98e-27	1.32e-27	6.99e-18	1.03e-31
Cat-300-8-3	3.23e-16	1.93e-08	1.14e-30	2.48e-23	3.86e-22	6.96e-19	4.55e-25	3.52e-30	9.32e-28	4.55e-25
Cat-300-15-5	2.22e-21	2.63e-19	2.21e-27	4.33e-37	1.55e-25	7.81e-20	2.07e-23	1.86e-26	3.61e-22	4.33e-37
Cat-500-20-10	8.87e-31	2.68e-21	2.28e-26	6.80e-36	5.74e-25	4.76e-32	1.19e-39	1.06e-30	3.50e-31	5.15e-34
Cat-1000-7-7	1.07e-31	1.85e-22	1.70e-27	1.07e-35	1.87e-26	5.00e-28	3.83e-37	3.29e-33	1.52e-26	3.87e-34
Cat-50000-10-2	2.49e-24	2.52e-22	1.26e-30	2.98e-40	2.64e-29	8.13e-25	4.01e-27	2.20e-28	2.49e-24	7.51e-39
Cat-100000-10-2	3.95e-17	1.12e-14	1.36e-30	3.88e-30	1.28e-29	2.59e-31	8.26e-20	1.48e-34	1.23e-30	1.04e-28
Soybean	4.35e-15	1.31e-11	1.25e-20	2.31e-24	2.17e-17	4.22e-15	3.13e-17	7.94e-27	1.44e-15	6.26e-21
Zoo	1.69e-14	1.98e-14	5.58e-27	7.18e-33	1.70e-25	1.91e-24	9.98e-86	2.46e-25	7.01e-23	2.72e-32
Dermatology	6.96e-13	9.73e-18	9.19e-28	2.33e-16	6.67e-25	4.46e-08	7.99e-21	5.07e-20	1.07e-05	1.33e-11
Mushroom	8.40e-09	5.20e-08	5.26e-16	1.32e-07	8.64e-14	1.76e-04	2.26e-23	5.28e-18	6.11e-03	3.56e-05

5.4 Worst case Time Complexity Analysis

The time complexities of MLCrKMd is dependent on CrKMd integrated with each individual machine learning. Therefore, individual technique such as CrKMd-KNN, CrKMd-SVM, CrKMd-ANN, CrKMd-DT and CrKMd-RF is analyzed in terms of time complexities for n number of objects having m attributes with K

Table 5.7: Average rank of the different methods for Synthetic and Real Life datasets. For every entry, the first value indicates the rank and average MS is given within the parenthesis

Datasets	KMd	FKMd	AL	TSFKMd	MMR	G-ANMI	MoDEFCCD	PKMd	SCC	PM-FGCA	MLCrKMd
Cat-100-8-3	11(0.88)	10(0.80)	9(0.76)	8(0.74)	7(0.56)	6(0.5)	2(0.36)	3.5(0.39)	3.5(0.39)	3.5(0.39)	1(0.20)
Cat-250-15-5	11(0.78)	9(0.63)	10(0.65)	8(0.55)	7(0.52)	6(0.50)	2(0.34)	3.5(0.43)	3.5(0.43)	3.5(0.43)	1(0.18)
Cat-300-8-3	11(0.77)	5(0.37)	10(0.76)	4(0.34)	3(0.32)	2(0.20)	9(0.44)	6.5(0.40)	6.5(0.40)	6.5(0.40)	1(0.11)
Cat-300-15-5	11(0.84)	10(0.75)	9(0.74)	8(0.70)	4(0.64)	3(0.58)	2(0.50)	6.5(0.69)	5(0.66)	6.5(0.69)	1(0.32)
Cat-500-20-10	11(0.94)	9.5(0.81)	9.5(0.81)	8(0.76)	7(0.72)	6(0.67)	2(0.48)	5(0.66)	3(0.62)	4(0.63)	1(0.20)
Cat-1000-7-7	11(0.95)	10(0.81)	9(0.80)	8(0.74)	7(0.73)	6(0.67)	2(0.38)	4(0.62)	3(0.58)	5(0.66)	1(0.20)
Cat-50000-10-2	11(0.94)	9.5(0.81)	9.5(0.81)	8(0.76)	7(0.72)	6(0.67)	2(0.48)	5(0.66)	3(0.64)	4(0.65)	1(0.20)
Cat-100000-10-2	11(0.95)	10(0.81)	9(0.80)	8(0.74)	6.5(0.73)	5(0.67)	2(0.38)	6.5(0.73)	3(0.62)	4(0.65)	1(0.20)
Soybean	11(0.64)	9(0.39)	10(0.45)	8(0.37)	7(0.33)	3(0.26)	2(0.20)	6(0.30)	4.5(0.27)	4.5(0.27)	1(0.09)
Zoo	11(0.69)	9(0.44)	10(0.46)	8(0.43)	6(0.39)	4.5(0.38)	3(0.33)	4.5(0.38)	2(0.32)	7(0.40)	1(0.07)
Dermatology	11(1.02)	10(0.92)	9(0.89)	8(0.84)	7(0.74)	6(0.73)	5(0.71)	4(0.68)	2(0.63)	3(0.65)	1(0.43)
Mushroom	11(0.82)	9(0.70)	10(0.71)	8(0.68)	7(0.66)	6(0.65)	5(0.64)	3(0.61)	2(0.60)	4(0.62)	1(0.51)
Average Rank	11	9.17	9.5	7.67	6.29	4.96	3.17	4.83	3.42	4.63	1

modes, whereas for Random Forest, the number of tree is T. The algorithm is dominated by the inner loop where in each iteration, computation is done to compute distance matrix, fuzzy membership matrix, credibility matrix, cluster modes, objective function value to get final credibility matrix. Table 5.8 shows the time complexity analysis of CrKMd. Here, total number of attribute values is denoted by $M\left(=\sum_{j=1}^{m} q_j\right)$. Moreover, additional time is required while integrating with machine learning techniques and mentioned in Table 5.9. For example, overall time is required for CrKMd-RF is $O(KMn + K^2n + \mathsf{T}nmlog(n))$.

Table 5.8: Worst case time complexity of CrKMd

Element to compute	**Worst case time complexity**
Distance matrix	$O(Kn)$
Updating cluster modes	$O(KMn)$
Fuzzy membership matrix	$O(Kn)$
Credibility matrix	$O(K^2n)$
Objective function value	$O(nm)$
Overall time complexity	$O(nm + KMn + 2Kn + K^2n)$
	$\approx O(KMn + K^2n)$

5.5 Conclusions

In recent decades, there has been a widespread use of fuzzy clustering and its many adaptations in a diverse range of practical scenarios. However, it struggles to function properly in loud environments. To solve the issue with fuzzy approaches, the

Table 5.9: Additional time complexities due to integration of machine learning technique

Algorithm	**Additional time complexity**
CrKMd-KNN	$O(nm)$
CrKMd-SVM	$O(n^3)$
CrKMd-ANN	$O(n)$
CrKMd-DT	$O(log(n))$
CrKMd-RF	$O(\mathrm{T}nmlog(n))$

probabilistic theory was developed. The coincident clustering problem, however, affects the probabilistic technique itself in cases of close clusters. Furthermore, categorical data increase complexity because of their own underlying characteristics within the attributes.Additionally, the challenge of acquiring sufficient and precise labeled data in real-world scenarios has motivated my thesis to propose a semi-supervised clustering approach that integrates machine learning techniques for categorical data with credibilistic metrics [221]. This chapter of the thesis elucidates the mathematical components of the credibilistic measure and delineates how certain characteristics, such as self-duality, might address the limitations inherent in both fuzzy and possibilistic methodologies for clustering. The Credibilistic K-Modes clustering approach has been developed to leverage the benefits of the credibilistic measure while dealing with categorical data. The approach has resulted in heightened reliability for non-outlier data, while attributing diminished reliability to outlier data. Consequently, the approach that has been devised demonstrates a greater efficacy in grouping non-outlier data compared to both FCM and PCM. A cluster assigns a data point depending on its greatest degree of confidence. A unique classification of data with identical credibility scores for numerous clusters can nonetheless be confusing in some situations. The semi-supervised clustering method MLCrKMd has been developed to do rid of this oddity. It combines CrKMd with several machine learning methods to provide a superior end result. The performance evaluation of MLCrKMd has been conducted on a total of twelve datasets, consisting of eight synthetic datasets and four actual categorical datasets. The evaluation included quantifying the performance, visually representing the results, and statistically comparing them with ten other state-of-the-art methodologies. This comparison was carried out using several cluster validity indices. The findings of categorical data clustering have demonstrated that MLCrKMd is superior to other approaches. However, it has been shown that MLCrKMd requires substantially more computing resources and storage. To reduce the time and complexity, future research needs to be focused.

6
Conclusions

In contemporary times, a substantial quantity of data is being generated from many sources across several disciplines. The use of clustering has become an essential methodology for extracting potential insights from data. Academic researchers have a significant challenge when dealing with categorical data due to its inherent absence of a natural ordering. The presence of overlapping, ambiguous, and imprecise data poses significant challenges for the majority of algorithms, rendering them inadequate for addressing real-world situations. Consequently, the objective of this thesis is to conduct a comprehensive analysis of clustering, explore the existing techniques and their constraints, and propose novel integrated and ensemble-based soft intelligent learning techniques for clustering. These techniques aim to effectively address challenges related to vagueness, uncertainty, and indiscernibility, particularly in the context of categorical data. Fuzzy clustering and its many adaptations have been extensively used in different practical scenarios, as seen from a comprehensive examination of the multiple clustering algorithms proposed in the last few decades. Nevertheless, it has difficulties in effectively functioning inside environments characterised by high levels of noise. Several alternative approaches, such as the possibilistic rough set and its various adaptations, have also been proposed to tackle similar issues. Once again, the majority of these algorithms lack the capability to effectively manage the inherent complexity of the data. This thesis proposes many integrated and ensemble-based soft learning strategies to accomplish this task. The aforementioned strategies are extensively discussed in Chapters 2 to 5.

Chapter 2 of this thesis introduces rough fuzzy-based clustering approaches specifically designed for categorical data. In order to evaluate the effectiveness of these techniques, an experiment was conducted using a total of six artificial and four actual data sets. The first section of this chapter provides an examination of RFKMd using a solitary dissimilarity measure. Nevertheless, research has shown that there is no singular dissimilarity metric that consistently yields optimal clustering outcomes across different categorical data sets. The aforementioned circumstances prompted the emergence of the Ensemble based Rough Fuzzy (ERFC) methodology. This approach expands the RFKMd technique to assess several collections of coarse objects by using diverse dissimilarity measures. The Random Forest classifier has also been used for the classification of residual rough objects. The suggested approach has shown enhanced capabilities in addressing the challenges of overlapping partition and handling uncertainty and vagueness within data sets. This improvement may be attributed to the utilization of both rough and fuzzy sets concepts. The efficacy of clustering algorithms is evaluated via the use of four performance metrics. The metrics included in this set are the Roughness Measure, Minkowski Score (MS), Percentage of Correct Pair (%CP), and Adjusted Rand Index (ARI). Through the comparison of many cluster validity indices on different artificial and real-world categorical datasets, it has been shown that ERFC outperforms several other previously existing methods. The results have been further substantiated using a *t*-test conducted at a significance level of 5%.

The challenge associated with classifying MR brain image segmentation into distinct homogeneous regions is recognized as the issue of pixel clustering inside the intensity space. Automatically identifying parts or groups of areas with very disparate sizes is a challenging task. In order to tackle this issue, Chapter 3 has devised a hybrid clustering approach that integrates the principles of type-2 fuzzy set theory, probabilistic methodology, and rough set theory. The methodology for utilizing the initial rough and type-2 fuzzy based clustering technique, known as RT2FCM, has been explicated. Subsequently, the use of probabilistic methods has facilitated the expansion of RT2FCM into RPT2FCM, aiming to rectify the limitations of conventional FCM. Additionally, the incorporation of type-2 fuzzy sets and rough set theories has been employed to effectively handle the inherent uncertainties, ambiguities, and indiscernibilities present within the data sets. The RPT2FCM algorithm produces both crisp and rough points. Hence, the preliminary data points are categorized via Random Forest, which has been trained with crisp data points. This collective methodology is referred to as RPT2FCM-RF, implemented with the intention of enhancing the overall clustering outcome. The clustering quality is evaluated using DB, ARI, MS, and %CP. According to the results of the *t*-test, it can be concluded that the RPT2FCM-RF method has shown superior statistical performance compared to previous techniques, with a significance level of 5%. The experimental results indicate that the proposed method exhibits superior statistical and visual performance compared to existing approaches for the task of categorising MR brain pictures into different tissue categories.

To further address the issue of indiscernibility (coarseness) and vagueness within categorical datasets, the rough fuzzy K-Modes clustering approach has been improved in Chapter 4. The Rough Fuzzy *K*-Modes clustering method has a propensity to become stuck in a local optimum solution; as a result, simulated annealing and genetic algorithms are combined to develop two additional new clustering methods, Simulated Annealing based Rough Fuzzy K-Modes and Genetic Algorithm based Rough Fuzzy *K*-Modes. These methods can manage clusters with a different set of central and peripheral points. The Random Forest classifier has been individually incorporated in order to categorize the peripheral points generated by these approaches. The central points are used to train the classifier to categorize the peripheral points for this purpose. Additionally, it has been observed that there are differences in the results obtained using the Rough Fuzzy *K*-Modes, Simulated annealing based Rough Fuzzy *K*-Modes, and Genetic algorithm based Rough Fuzzy *K*-Modes approaches. As a result, for these approaches, the cardinality of the sets of central and peripheral points has likewise changed. In order to choose the best set of central points from the three sets, a roughness measure has been computed. Pure peripheral and semi-best central points have since been discovered. The semi-best central points are then categorized using the best central points, and utilising those classifications, peripheral points have been classed individually using Random Forest to produce superior clustering results. The technique has been named as Integrated Rough Fuzzy Clustering using Random Forest. When compared to other state-of-the-art techniques already in use, the new approaches have been found to be superior. By assessing a number of cluster validity indices, experimental findings have been presented for this aim using both synthetic and actual categorical datasets. For all of the datasets, it was found that the Integrated Rough Fuzzy Clustering approach outperformed the other methods. The suggested approaches have wide range of applications in several practical contexts where categorical data are present, such as customer segmentation, market basket analysis, environmental analysis, text and web mining, prediction and forecasting, biometry, etc. For instance, consumer sentiment data from various social media and the web may be grouped and studied to forecast churn in the telecom sector.

In order to deal with the problem of fuzzy techniques in noisy situations, probabilistic theory was established. But in circumstances of dense clusters, the coincident clustering problem also impacts the probabilistic approach. In addition, categorical data become more complicated due to the underlying traits that they possess inside the qualities. Furthermore, it is challenging to find sufficient and properly labeled data in the actual world. As a result, Chapter 5 has provided a semi-supervised clustering technique, CrKMd that blends credibilistic metrics with machine learning for categorical data. The mathematical components of the credibilistic measure have been clarified in this chapter of the thesis, along with how certain characteristics, such self-duality, might overcome the drawbacks of both the fuzzy and the possibilistic methods to clustering. First, to benefit from this credibilistic measure's advantage, the Credibilistic *K*-Modes clustering method for categorical data has been developed. In contrast to attributing lower credibility to outlier data, the strategy has produced increased credibility for non-outlier data. In contrast to FCM and PCM, the new technique is hence more able to group non-outlier data. In accordance with its degree of maximum believability, a data point is categorised into a cluster. However, in other cases, a unique categorization of data with similar credibility ratings for several clusters might be perplexing. To get rid of this peculiarity, a semi-supervised clustering algorithm MLCrKMd has been designed. For a better outcome, it integrates CrKMd with a number of machine learning techniques. The performance of MLCrKMd has been measured, visually depicted, and statistically compared with ten other state-of-the-art methods using four real categorical datasets and eight synthetic datasets. Categorical data clustering results have shown that MLCrKMd is superior to other methods. However, it has been observed that MLCrKMd requires significantly more storage and computational power.

This thesis has proposed a set of intelligent learning strategies for clustering, while also established the concept of integration of advanced mathematical and statistical learning methods to enhance the practical applicability of these techniques. As a corollary to the thesis concept, many intriguing novel methodologies have been devised, using an ensemble of machine learning methods, evolutionary techniques, and statistical learning. The implemented methodologies have been used within the field of medical research to ascertain prospective MicroRNAs responsible for breast cancer, cluster cancer-related data, and segment satellite image data [188,211,212,222,223].

While this thesis has proposed a few clustering techniques to handle the difficulties of dealing with vagueness, uncertainty, and indiscernibility within real-world data, several limitations have been discovered during assessment. The findings of every experiment show that the proposed techniques yield superior clustering outcomes when compared to state-of-the-art current recent clustering methods. These, however, are not completely resilient under all situations. A single cluster quality metric, for instance, is not adequately and equally appropriate for all types of data sets with various properties. Therefore, it is crucial to use the methods this thesis suggests in multi and many-objective paradigms. These facts influence the few following future research topics using mathematical tools such as type-2 fuzzy set theory, rough set, possibilistic & credibilistic measures.

- Experiment and extension of the suggested techniques in multi and many-objective paradigm

- Improvement of the techniques in order to reduce running time and space complexity

- Effectiveness of the methods on high speed computational environment

- Design parallel processing clustering algorithm
- Suggested integrated and ensemble approach of clustering to incorporate other concepts like game theory, quantum computing
- Fuzzy rough correlation factor is another area for future analysis in terms of clustering categorical dataset
- Hyper-heuristic techniques can be developed for better setting the parameters of meta-heuristic techniques
- Integrating suggested techniques with Density-Based clustering to more effectively handle irregularly shaped clusters

Bibliography

[1] F. C. H. Rhee and C. Hwang, "A Type-2 Fuzzy C-Means clustering algorithm," *in Proceedings of Joint 9th IFSA World Congress and 20th NAFIPS International Conference*, vol. 4, pp. 1926–1929, 2001.

[2] I. Saha, D. Plewczynski, U. Maulik, and S. Bandyopadhyay, "Improved Differential Evolution for Microarray analysis," *International Journal of Data Mining and Bioinformatics*, vol. 6, pp. 86–103, 2012.

[3] J. Schmidhuber, "Deep Learning in Neural Networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[4] Z. Pawlak, *Rough Sets: Theoretical Aspects of Resoning About Data*. Massachusetts, USA: Kluwer Academic, 1992.

[5] M. B. Kursa, "Robustness of Random Forest-based Gene selection methods," *BMC Bioinformatics*, vol. 15, p. 8, 2014.

[6] B. Pradhan, "A comparative study on the predictive ability of the Decision Tree, Support Vector Machine and Neuro-Fuzzy models in Landslide susceptibility mapping using GIS," *Computers & Geosciences*, vol. 51, pp. 350–365, 2013.

[7] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer, 2006.

[8] P. A. Devijver and J. Kittler, *Pattern Recognition : A Statistical Approach*. London: Prentice-Hall, 1982.

[9] K. Fukunaga, *Introduction to Statistical Pattern Recognition (Second Edition)*. New York: Academic Press, 1990.

[10] A. Webb, *Statistical pattern recognition*. New York: Oxford University Press Inc., 1999.

[11] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2nd ed.)*. John Wiley and Sons, 2001.

[12] J. T. Tou and R. C. Gonzalez, *Pattern Recognition Principles*. Reading, MA: Addison-Wesley, 1974.

[13] S. Theodoridis and K. Koutroumbas, *Pattern recognition*. Academic Press, 1999.

[14] U. Maulik and S. Bandyopadhyay, "Performance evaluation of some clustering algorithms and validity indices," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 12, pp. 1650–1654, 2002.

[15] U. Maulik and I. Saha, "Modified Differential Evolution based Fuzzy clustering for Pixel classification in Remote Sensing Imagery," *Pattern Recognition*, vol. 42, no. 9, pp. 2135–2149, 2009.

[16] D. S. Boone and M. Roehm, "Retail Segmentation using Artificial Neural Networks," *Internal Journal of Research in Marketing*, vol. 19, no. 3, pp. 287–301, 2002.

[17] U. Maulik, "Medical Image Segmentation using Genetic Algorithms," *IEEE Transactions on Information Technology in BioMedicine*, vol. 13, no. 2, pp. 166–173, 2009.

[18] S. Silva, P. Cortez, R. Mendes, P. J. Pereira, L. M. Matos, and L. Garcia, "A categorical clustering of publishers for mobile performance marketing," *in Proceeding of the 13th International Conference on Soft Computing Models in Industrial and Environmental Applications*, vol. 771, pp. 145–154, 2019.

[19] V. Vapnik, *The Nature of Statistical Learning Theory*. New York, USA: Springer-Verlag, 1995.

[20] V. Vapnik , *Statistical Learning Theory*. New York, USA: Wiley, 1998.

[21] R. Collobert and S. Bengio, "SVMTorch: Support Vector Machines for Large-scale Regression problems," *Journal of Machine Learning Research*, vol. 1, pp. 143–160, 2001.

[22] K. Goh, E. Y. Chang, and B. Li, "Using One-class and Two-class SVMs for Multiclass Image Annotation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 10, pp. 1333–1346, 2005.

[23] D. Graupe, *Principles of Artificial Neural Networks*. World Scientific, 2013.

[24] N. S. Altman, "An introduction to Kernel and Nearest-Neighbor Nonparametric Regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[25] T. M. Cover and P. E. Hart, "Nearest Neighbor Pattern Classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[26] C. Jin, L. De-lin, and M. Fen-xiang, "An improved ID3 Decision Tree algorithm," *in Proceedings of Fourth International Conference on Computer Science Education*, pp. 127–130, 2009.

[27] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[28] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian Network Classifiers," *Machine Learning*, vol. 29, pp. 131–163, 1997.

[29] H. Bisgin, T. Bera, H. Ding, H. G. Semey, L. Wu, Z. Liu, A. E. Barnes, D. A. Langley, M. Pava-Ripoll, H. J. Vyas, W. Tong, and J. Xu, "Comparing SVM and ANN based Machine Learning Methods for Species Identifcation of Food Contaminating Beetles," *Scientific Reports*, vol. 8, pp. 2045–2322, 2018.

[30] J. A. Hartigan, *Clustering Algorithms*. New York, USA: John Wiley & Sons, 1975.

[31] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. New Jersey, USA: Prentice-Hall, 1988.

[32] A. K. Jain and R. C. Dubes , "Data clustering: A review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999.

[33] S. Bandyopadhyay, U. Maulik, and M. Pakhira, "Clustering using Simulated Annealing with Probabilistic Redistribution," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 15, no. 2, pp. 269–285, 2001.

[34] I. Saha and U. Maulik, "Incremental learning based Multiobjective Fuzzy clustering for categorical data," *Information Sciences*, vol. 267, no. 2, pp. 35–57, 2014.

[35] U. Maulik, S. Bandyopadhyay, and I. Saha, "Integrating clustering and Supervised Learning for categorical data analysis," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 40, pp. 664–675, 2010.

[36] U. Maulik, S. Bandyopadhyay, and A. Mukhopadhyay, *Multiobjective Genetic Algorithms for Clustering: Applications in Data Mining and Bioinformatics*. Heidelberg, Germany: Springer, 2011.

[37] R. R. T. Zhang and M. Livny, "BIRCH: An efficient data clustering method for very large databases," *SIGMOD Record*, vol. 25, pp. 103–114, 1996.

[38] R. R. S. Guha and K. Shim, "CURE: An efficient clustering algorithms for large databases," *in Proceedings of the ACM SIGMOD International Conference on Management of Data, Seattle, WA*, pp. 73–84, 1998.

[39] S. Guha, R. Rastogi, and K. Shim, "ROCK: A robust clustering algorithm for categorical attributes," *Information Systems*, vol. 25, no. 5, pp. 345–366, 2000.

[40] M. R. Anderberg, *Cluster Analysis for Applications*. New York, USA: Academic Press, 1973.

[41] L. Kaufman and P. J. Roussenw, *Finding Groups in Data: An Introduction to Cluster Analysis*. New York, USA: John Wiley & Sons, 1990.

[42] Z. Huang, "Extension to the K-Means algorithm for clustering large data sets with categorical values," *Data Mining and Knowledge Discovery*, vol. 2, pp. 283–304, 1998.

[43] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," *in Proceedings of 2nd international conference Management on Knowledge Discovery and Data Mining*, pp. 226–231, 1996.

[44] J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "Density-based clustering in spatial databases: The algorithm GDBSCAN and its applications," *Data Mining and Knowledge Discovery*, vol. 2, pp. 169–194, 1998.

[45] M. Ankerst, M. Breunig, H. P. Kriegel, and J. Sander, "OPTICS: Ordering Points To Identify the Clustering Structure," *in Proceedings of ACM SIGMOD international conference Management of Data (SIGMOD'99)*, pp. 49–60, 1999.

[46] E. O. Omuya, G. O. Okeyo, and M. W. Kimwele, "Feature selection for classification using Principal Component Analysis and Information Gain," *Expert Systems with Applications*, vol. 174, p. 114765, 2021.

[47] J. Nobre and F. Neves, "Combining Principal Component Analysis, Discrete Wavelet Transform and XGBoost to trade in the financial markets," *Expert Systems with Applications*, vol. 125, pp. 181–194, 2019.

[48] L. A. Zadeh, "Soft Computing and Fuzzy Logic," *IEEE Software*, vol. 11, pp. 48–56, 1994.

[49] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum, 1981.

[50] Z. Huang and M. K. Ng, "A Fuzzy K-Modes algorithm for clustering categorical data," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 4, 1999.

[51] U. Maulik and S. Bandyopadhyay, "Genetic Algorithm based clustering technique," *Pattern Recognition*, vol. 33, no. 9, pp. 1455–1465, 2000.

[52] M. K. Pakhira, S. Bandyopadhyay, and U. Maulik, "A Study of some Fuzzy Cluster Validity Indices, Genetic Clustering and application to Pixel Classification," *Fuzzy Sets and Systems*, vol. 155, no. 2, pp. 191–214, 2005.

[53] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Massachusetts, USA: Addison-Wesley Longman, 1989.

[54] S. Kirkpatrick, C. Gelatt, and M. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[55] S. Bandyopadhyay, "Simulated Annealing Using a Reversible Jump Markov Chain Monte Carlo Algorithm for Fuzzy Clustering," *IEEE Transactions on Knowledge and data Engineering*, vol. 17, no. 4, pp. 479–490, 2005.

[56] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," *in Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995.

[57] E. Rubio and O. Castillo, "Optimization of the Interval Type-2 Fuzzy C-Means using Particle Swarm Optimization," *in Proceedings of the World Congress on Nature and Biologically Inspired Computing*, pp. 10–15, 2013.

[58] N. Zeng, Z. Wang, H. Zhang, K. E. Kim, Y. Li, and X. Liu, "An improved Particle Filter with a novel hybrid proposal distribution for quantitative analysis of Gold Immunochromatographic Strips," *IEEE Transactions on Nanotechnology*, vol. 18, pp. 819–829, 2019.

[59] N. Zeng, H. Qiu, Z. Wang, W. Liu, H. Zhang, and Y. Li, "A new switching-delayed-PSO-based optimized SVM algorithm for diagnosis of Alzheimer′s disease," *Neurocomputing*, vol. 320, pp. 195–202, 2018.

[60] N. Zeng, Z. Wang, and H. Zhang, "Inferring nonlinear lateral flow immunoassay State-space models via an unscented Kalman filter," *Science China Information Sciences*, vol. 59, pp. 112–204, 2016.

[61] X. Chen and C. Jian, "Gene expression data clustering based on Graph Regularized Subspace Segmentation," *Neurocomputing*, vol. 143, pp. 44–50, 2014.

[62] S. Kanaanizquierdo, A. Ziyatdinov, and A. Pereralluna, "Multiview and Multifeature Spectral Clustering using common Eigenvectors," *Pattern Recognition Letters*, vol. 102, pp. 30–36, 2018.

[63] S. Liu, X. Liang, L. Liu, X. Shen, J. Yang, C. Xu, L. Lin, X. Cao, and S. Yan, "Matching-CNN meets KNN: Quasi-parametric human parsing," *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1419–1427, 2015.

[64] L. Houthuys, R. Langone, and J. A. K. Suykens, "Multi-view kernel spectral clustering," *Information Fusion*, vol. 44, pp. 46–56, 2021.

[65] X. Cai, F. Nie, H. Huang, and F. Kamangar, “Heterogeneous image feature integration via multi-modal spectral clustering,” *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1977–1984, 2011.

[66] C. Lang, G. Liu, J. Yu, and S. Yan, “Saliency detection by multitask sparsity pursuit,” *IEEE Transactions on Image Processing*, vol. 21, pp. 1327–1338, 2021.

[67] L. Zhang, Z. Shi, M. M. Cheng, Y. Liu, J. W. Bian, J. T. Zhou, G. Zheng, and Z. Zeng, “Nonlinear Regression via Deep Negative Correlation Learning,” *IEEE Transactions On Pattern Analysis And Machine Intelligence*, vol. 43, no. 3, pp. 982–998, 2021.

[68] X. He, L. Li, D. Roqueiro, and K. Borgwardt, “Multi-view spectral clustering on conflicting views,” *in Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 826–842, 2017.

[69] P. Berkhin, *A survey of clustering data mining techniques*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006.

[70] J. Zhou, Z. Lai, C. Gao, D. Miao, and X. Yue, “Rough Possibilistic C-means clustering based on Multigranulation Approximation Regions and Shadowed Sets,” *Knowledge-Based Systems*, vol. 160, pp. 144–166, 2018.

[71] S. D. Xenaki, K. D. Koutroumbas, and A. A. Rontogiannis, “Sparsity-aware Possibilistic clustering algorithms,” *IEEE Transactions on Fuzzy Systems*, vol. 24, no. 6, pp. 1611–1626, 2016.

[72] K. D. Koutroumbas, S. D. Xenaki, and A. A. Rontogiannis, “On the Convergence of the Sparse Possibilistic C-Means Algorithm,” *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 1, pp. 324–337, 2018.

[73] E. Rubio, O. Castillo, F. Valdez, P. Melin, C. I. Gonzalez, and G. Martinez, “An extension of the Fuzzy Possibilistic Clustering Algorithm using Type-2 Fuzzy Logic techniques,” *Advanced Fuzzy Systems*, vol. 2017, no. 1, pp. 1–23, 2017.

[74] M. R. N. Kalhori and M. H. F. Zarandi, “Interval Type-2 Credibilistic clustering for Pattern Recognition,” *Pattern Recognition*, vol. 48, no. 11, pp. 3652–3672, 2015.

[75] H. Yu, L. Jiang, J. Fan, and R. Lan, “Double-Suppressed Possibilistic Fuzzy Gustafson-Kessel clustering algorithm,” *Knowledge Based Systems*, vol. 276, p. 110736, 2023.

[76] H. Saberi, R. Sharbati, and B. Farzanegan, "A Gradient Ascent Algorithm based on Possibilistic Fuzzy C-Means for clustering noisy data," *Expert Systems With Applications*, vol. 191, p. 116153, 2022.

[77] X. Yang, F. Yu, and W. Pedrycz, "Typical Characteristics-based Type-2 Fuzzy C-Means Algorithm," *IEEE Transactions On Fuzzy Systems*, vol. 29, no. 5, pp. 1173–1187, 2020.

[78] M. Li, L. Zhang, Z. Xiang, E. Castillo, and T. Guerrero, "An Improved Fuzzy C-Means Algorithm for Brain MRI Image Segmentation," *in Proceedings of the International Conference on Progress in Informatics and Computing*, 2017.

[79] L. Szilagyi, "Fuzzy-Possibilistic product partition: A novel robust approach to C-Means clustering," *in Proceedings of the International Conference on Modeling Decisions for Artificial Intelligence*, pp. 150–161, 2011.

[80] O. Linda and M. Manic, "General Type-2 Fuzzy C-Means Algorithm for Uncertain Fuzzy Clustering," *IEEE Transactions on Fuzzy Systems*, vol. 20, pp. 883–897, 2012.

[81] C. Hwang and F. C. H. Rhee, "Uncertain Fuzzy clustering: Interval Type-2 Fuzzy approach to C-Means," *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 1, pp. 107–120, 2007.

[82] P. Kaur, I. M. S. Lamba, and A. Gosain, "Kernelized Type-2 Fuzzy C-Means Clustering Algorithm in Segmentation of Noisy Medical Images," *in Proceedings of the IEEE Conference on Recent Advances in Intelligent Computational Systems*, pp. 493–498, 2011.

[83] M. H. F. Zarandi, R. Gamasaee, and I. B. Turksen, "A Type-2 Fuzzy C-Regression Clustering Algorithm for Takagi-Sugeno system identification and its Application in the Steel Industry," *Information Sciences*, vol. 187, pp. 179–203, 2012.

[84] M. A. Raza and F. C. H. Rhee, "Interval Type-2 Approach to Kernel Possibilistic C-Means Clustering," *IEEE International Conference on Fuzzy Systems*, pp. 1–7, 2012.

[85] M. Zarinbal, M. H. F. Zarandi, and I. B. Turksen, "Interval Type-2 Relative Entropy Fuzzy C-Means clustering," *Information Sciences*, vol. 272, pp. 49–72, 2014.

[86] L. T. Ngoa, T. H. Dang, and W. Pedrycz, "Towards Interval-Valued Fuzzy Set-based Collaborative Fuzzy Clustering Algorithms," *Pattern Recognition*, vol. 81, pp. 404–416, 2018.

[87] N. R. Pal, K. Pal, J. M. Keller, and J. C. Bezdek, "A Possibilistic Fuzzy C-Means clustering algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 4, pp. 517–530, 2005.

[88] K. C. Gowda and E. Diday, "Unsupervised learning through symbolic clustering," *Pattern Recognition Letters*, vol. 12, no. 5, pp. 259–264, 1991.

[89] K. C. Gowda and E. Diday, "Symbolic clustering using a new dissimilarity measure," *Pattern Recognition*, vol. 24, no. 6, pp. 567–578, 1991.

[90] K. C. Gowda and T. V. Ravi, "Divisive clustering of symbolic objects using the concepts of both similarity and dissimilarity," *Pattern Recognition*, vol. 28, no. 8, pp. 1277–1282, 1995.

[91] V. Ganti, J. Gehrke, and R. Ramakrishnan, "CACTUS - Clustering Categorical Data Using Summaries," *in Proceeding of Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 73–83, 1999.

[92] D. Gibson, J. Kleinberg, and P. Raghavan, "Clustering categorical data: an approach based on dynamical systems," *The Very Large Data Bases Journal*, vol. 8, no. 3-4, pp. 222–236, 2000.

[93] D. Barbara, Y. Li, and J. Couto, "COOLCAT: An entropy-based algorithm for categorical clustering," *in Proceeding of Eleventh International Conference on Information and Knowledge Management*, pp. 582–589, 2002.

[94] Z. He, X. Xu, and S. Deng, "Squeezer: An efficient algorithm for clustering categorical data," *Journal of Computer Science & Technology*, vol. 17, no. 5, pp. 611–624, 2002.

[95] Y. Yang, S. Guan, and J. You, "CLOPE: A fast and effective clustering algorithm for transactional data," *in Proceedings of Eighth International Conference on Knowledge Discovery and Data Mining*, pp. 682–687, 2002.

[96] D. Cristofor and D. Simovici, "Finding median partitions using information theoretical-based Genetic Algorithms," *Journal of Universal Computer Science*, vol. 8, no. 2, pp. 153–172, 2002.

[97] K. Mali and S. Mitra, "Clustering and its validation in a symbolic framework," *Pattern Recognition Letters*, vol. 24, no. 14, pp. 2367–2376, 2003.

[98] E. Diday and F. Esposito, "An introduction to symbolic data analysis and the SODAS software," *Journal Intelligent Data Analysis*, vol. 7, no. 6, pp. 583–601, 2003.

[99] M. Chen and K. Chuang, "Clustering categorical data using the correlated force ensemble," *in Proceedings of Fourth SIAM International Conference on Data Mining*, pp. 269–278, 2004.

[100] M. Peters and M. J. Zaki, "CLICK: Clustering categorical data using K-partite Maximal Cliques," *TR 04-11*, vol. CSE Department, RPI, 2004.

[101] D. W. Kim, K. H. Lee, and D. Lee, "Fuzzy clustering of categorical data using Fuzzy centroids," *Pattern Recognition Letters*, vol. 25, no. 11, pp. 1263–1271, 2004.

[102] P. Andritsos, P. Tsaparas, R. J. Miller, and K. C. Sevcik, "LIMBO: Scalable clustering of categorical data," *in Proceeding of Ninth International Conference on Extending Database Technology*, vol. 2992, pp. 123–146, 2004.

[103] K. Mali and S. Mitra, "Symbolic classification, clustering and Fuzzy Radial basis function network," *Fuzzy Sets and Systems*, vol. 152, no. 3, pp. 553–564, 2005.

[104] D. W. Kim, K. Y. Lee, D. Lee, and K. H. Lee, "A K-Populations algorithm for clustering categorical data," *Pattern Recognition*, vol. 38, no. 7, pp. 1131–1134, 2005.

[105] Z. He, X. Xu, and S. Deng, "TCSOM: Clustering transactions using selforganizing map," *Neural Processing Letters*, vol. 22, no. 3, pp. 249–262, 2005.

[106] Z. He, X. Xu, and S. Deng, "A cluster ensemble method for clustering categorical data," *Information Fusion*, vol. 6, no. 2, pp. 143–151, 2005.

[107] M. Dutta, A. K. Mahanta, and A. K. Pujari, "QROCK: A quick version of the ROCK algorithm for clustering of categorical data," *Pattern Recognition Letters*, vol. 26, no. 15, pp. 2364–2373, 2005.

[108] P. Zhang, X. Wang, and P. X. Song, "Clustering categorical data based on distance vectors," *The Journal of the American Statistical Association*, vol. 101, no. 473, pp. 355–367, 2006.

[109] M. K. Ng, M. J. Li, J. Z. Huang, and Z. He, "On the impact of dissimilarity measure in K-Modes clustering algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 503–507, 2007.

[110] Z. He, X. Xu, and S. Deng, "*k*-ANMI: A mutual information based clustering algorithm for categorical data," *Information Fusion*, vol. 9, no. 2, pp. 223–233, 2008.

[111] S. Benati, "Categorical Data Fuzzy Clustering: An analysis of local search heuristics," *Computers & Operations Research*, vol. 35, no. 3, pp. 766–775, 2008.

[112] G. Gan, J. Wu, and Z. Yang, "A Genetic Fuzzy K-Modes algorithm for clustering categorical data," *Expert Systems with Applications*, vol. 36, pp. 1615–1620, 2009.

[113] F. Cao, J. Liang, and L. Bai, "A new initialization method for categorical data clustering," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10 223–10 228, 2009.

[114] F. Cao, J. Liang, L. Bai, X. Zhao, and C. Dang, "A framework for clustering categorical time-evolving data," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 5, pp. 872–882, 2010.

[115] Z. He, X. Xu, and S. Deng, "Attribute value weighting in K-Modes clustering," *Expert Systems with Applications*, vol. 38, no. 12, pp. 15 365–15 369, 2011.

[116] F. Cao, J. Liang, D. Li, L. Bai, and C. Dang, "A dissimilarity measure for the K-Modes clustering algorithm," *Knowledge-Based Systems*, vol. 26, pp. 120–127, 2012.

[117] L. Bai, J. Liang, C. Dang, and F. Cao, "A cluster centers initialization method for clustering categorical data," *Expert Systems with Applications*, vol. 39, no. 12, pp. 8022–8029, 2012.

[118] H. Ralambondrainy, "A conceptual version of the K-Means algorithm," *Pattern Recognition Letter*, vol. 16, no. 11, pp. 1147–1157, 1995.

[119] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–38, 1977.

[120] E. Han, G. Karypis, V. Kumar, and B. Mobasher, "Clustering based on Association Rule Hypergraphs," *in Proceedings of Workshop on Research Issues on Data Mining and Knowledge Discovery*, pp. 9–13, 1997.

[121] R. T. Ng and J. Han, "Efficient and effective clustering methods for spatial data mining," *in Proceedings of Twentieth International Conference on Very Large Data Bases*, pp. 144–155, 1994.

[122] R. Krishnapuram and J. M. Keller, "A Possibilistic approach to clustering," *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 2, pp. 98–110, 1993.

[123] R. Krishnapuram and J. M. Keller, "The Possibilistic C-Means Algorithm: Insights and Recommendations," *IEEE Transactions on Fuzzy Systems*, vol. 4, no. 3, pp. 385–393, 1996.

[124] N. R. Pal, K. Pal, and J. C. Bezdek, "A mixed C-Means clustering model," *in Proceedings of the Sixth IEEE International Conference of Fuzzy Systems, Spain*, pp. 11–21, 1997.

[125] P. H. Krishnan and P. Ramamoorthy, "An efficient modified Fuzzy Possibilistic C-Means algorithm for MRI brain image segmentation," *International Journal of Engineering Research and Applications*, vol. 2, no. 2, pp. 1106–1110, 2012.

[126] A. Bose and K. Mali, "Type-reduced Vague Possibilistic Fuzzy clustering for Medical Images," *Pattern Recognition*, vol. 112, pp. 1–13, 2020.

[127] H. Q. Truong, L. T. Ngo, and W. Pedrycz, "Granular Fuzzy Possibilistic C-Means clustering approach to DNA microarray problem," *Knowledge Based Systems*, vol. 133, pp. 53–65, 2017.

[128] H. Timm, C. Borgelt, C. Doring, and R. Kruse, "An extension to Possibilistic Fuzzy cluster analysis," *Fuzzy Sets and Systems*, vol. 147, no. 1, pp. 3–16, 2004.

[129] B. Ojeda-Magana, R. Ruelas, M. A. Corona-Nakamura, and D. Andina, "An improvement to the Possibilistic Fuzzy C-Means clustering algorithm," *in Proceedings of the World Automation Congress*, pp. 1–8, 2006.

[130] L. Maciela, R. Ballinib, and F. Gomideca, "An evolving Possibilistic Fuzzy modeling approach for Value-at-Riskestimation," *Applied Soft Computing*, vol. 60, pp. 820–830, 2017.

[131] S. Askaria, N. Montazerina, and M. H. F. Zarandi, "Generalized Possibilistic Fuzzy C-Means with novel cluster validity indices for clustering noisy data," *Applied Soft Computing*, vol. 53, pp. 262–283, 2017.

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

[133] D. Fisher, "Improving inference through conceptual clustering," *in Proceedings of AAAI-87 Sixth National Conference on Artificial Intelligence*, pp. 461–465, 1987.

[134] F. Glover and M. Laguna, *Tabu Search*. Massachusetts, USA: Kluwer Academic, 1997.

[135] D. Parmar, T. Wu, and J. Blackhurst, "MMR: An algorithm for clustering categorical data using Rough Set Theory," *Data and Knowledge Engineering*, vol. 63, pp. 879–893, 2007.

[136] S. Deng, Z. He, and X. Xu, "G-ANMI: A Mutual Information based Genetic clustering algorithm for categorical data," *Knowledge-Based Systems*, vol. 23, no. 2, pp. 144–149, 2010.

[137] M. Palanivelu and M. Duraisamy, "Color Textured Image Segmentation using ICICM - Interval Type-2 Fuzzy C-Means Clustering Hybrid Approach," *Engineering Journal*, vol. 16, pp. 115–126, 2012.

[138] C. Qiu, J. Xiao, L. Yu, and L. Han, "An Interval Type-2 Fuzzy C-Means Algorithm based on Spatial Information for Image Segmentation," *in Proceedings of the the 8th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD, Shanghai, China*, pp. 545–549, 2011.

[139] D. Enke, M. Grauer, and N. Mehdiyev, "Stock Market Prediction with Multiple Regression, Fuzzy Type-2 clustering and Neural Networks," *Procedia Computer Science*, vol. 6, pp. 201–206, 2011.

[140] O. Castillo, P. Melin, W. Pedrycz, and J. Kacprzyk, *Designing Type-2 Fuzzy Systems Using the Interval Type-2 Fuzzy C-Means Algorithm*. Switzerland: Springer Cham, 2014.

[141] H. D. Duong, D. D. Nguyen, L. T. Ngo, and D. T. Tinh, "On approach to Vision based Fire detection based on Type-2 Fuzzy Clustering," *in Proceedings of the International Conference of Soft Computing and Pattern Recognition*, pp. 51–56, 2011.

[142] W. C. Tjhi and L. Chen, "Possibilistic Fuzzy Co-clustering of large document collections," *Pattern Recognition*, vol. 40, no. 12, pp. 3452–3466, 2007.

[143] B. Liu and Y. K. Liu, "Expected value of Fuzzy variable and Fuzzy expected value models," *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 4, pp. 445–450, 2002.

[144] J. Zhou, Q. Wang, C. C. Hung, and X. Yi, "Credibilistic clustering: The model and algorithms," *Fuzziness and Knowledge-Based Systems*, vol. 23, no. 4, pp. 545–564, 2015.

[145] M. R. N. Kalhori and M. H. F. Zarandi, "Interval Type-2 Credibilistic clustering for Pattern Recognition," *Pattern Recognition*, vol. 48, no. 11, pp. 3652–3672, 2015.

[146] J. Zhou, Q. Wang, C. C. Hung, and F. Yang, "Credibilistic clustering algorithms via alternating cluster estimation," *Journal of Intelligent Manufacturing*, vol. 28, no. 3, pp. 727–738, 2017.

[147] T. A. Runkler and J. C. Bezdek, "Alternating cluster estimation:a new tool for clustering and function approximation," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 4, pp. 377–393, 1999.

[148] M. S. Yang and K. L. Wu, "Unsupervised Possibilistic Clustering," *Pattern Recognition*, vol. 39, no. 1, pp. 5–21, 2006.

[149] I. Saha, U. Maulik, and S. Bandyopadhyay, "An improved Multi-objective technique for fuzzy clustering with application to IRS image segmentation," *Applications of Evolutionary Computing*, pp. 426–431, 2009.

[150] I. Saha, U. Maulik, and D. Plewczynski, "A new Multi-objective technique for Differential Fuzzy clustering," *Applied Soft Computing*, vol. 11, no. 2, pp. 2765–2776, 2011.

[151] A. Sehgal and U. B. Desai, "3D object recognition using Bayesian geometric hashing and pose clustering," *Pattern Recognition*, vol. 36, no. 3, pp. 765–780, 2003.

[152] A. Popescul, G. W. Flake, S. Lawrence, L. H. Ungar, and C. L. Giles, "Clustering and identifying temporal trends in document databases," *in Proceedings of IEEE Advances in Digital Libraries 2000 (ADL 2000)*, pp. 173–182, 2002.

[153] E. Cesario, G. Manco, and R. Ortale, "Top-down parameter-free clustering of high-dimensional categorical data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 12, pp. 1607–1624, 2007.

[154] H. L. Chen, K. T. Chuang, and M. S. Chen, "On data labeling for clustering categorical data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 11, pp. 1458–1472, 2008.

[155] M. K. Ng and J. C. Wong, "Clustering categorical data sets using Tabu search techniques," *Pattern Recognition*, vol. 35, no. 12, pp. 2783–2790, 2002.

[156] K. Umayahara, S. Miyamoto, and Y. Nakamori, "Formulations of Fuzzy clustering for categorical data," *International Journal of Innovative Computing, Information and Control*, vol. 1, no. 1, pp. 83–94, 2005.

[157] P. Lingras and C. West, "Interval set clustering of web users with Rough K-Means," *Journal of Intelligent Information Systems*, vol. 23, no. 1, pp. 5–16, 2004.

[158] S. Mitra, H. Banka, and W. Pedrycz, "Rough-Fuzzy Collaborative Clustering," *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, vol. 36, no. 4, pp. 795–805, 2006.

[159] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.

[160] Y. Y. Yao and S. K. M. Wong, "A decision theoretic framework for approximating concepts," *International Journal of Man-machine Studies*, vol. 37, no. 6, pp. 793–809, 1992.

[161] T. Li, D. Ruan, W. Geert, J. Song, and Y. Xu, "A Rough Sets based characteristic relation approach for dynamic attribute generalization in data mining," *Knowledge-Based Systems*, vol. 20, no. 5, pp. 485–494, 2007.

[162] Y. Y. Yao, "Probabilistic Rough Set Approximations," *International Journal of Approximate Reasoning*, vol. 49, no. 2, pp. 255–271, 2008.

[163] H. Chen, T. Li, D. Ruan, J. Lin, and C. Hu, "A Rough-Set-Based Incremental approach for updating Approximations under Dynamic Maintenance Environments," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 2, pp. 274–284, 2013.

[164] J. Zhang, T. Li, and H. Chen, "Composite Rough sets for Dynamic Data Mining," *Information Sciences*, vol. 257, pp. 81–100, 2014.

[165] H. Chen, T. Li, C. Luo, S. Horng, and G. Wang, "A Rough Set-based method for updating Decision Rules on attribute valuesĆoarsening and Refining," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 12, pp. 2886–2899, 2014.

[166] L. Vermeulen-Jourdan, C. Dhaenens, and E.-G. Talbi, "Clustering Nominal and Numerical Data: A New Distance Concept for an Hybrid Genetic Algorithm," in *in Proceedings of 4th European Conference Evolutionary Computation on Combinatorial Optimization (LNCS 3004)*, Coimbra, Portugal, April 2004, pp. 220–229.

[167] P. Maji and S. K. Pal, "RFCM: A Hybrid clustering algorithm using Rough and Fuzzy Sets," *Fundamenta Informaticae*, vol. 80, no. 4, pp. 475–496, 2007.

[168] N. Jardine and R. Sibson, *Mathematical Taxonomy*. John Wiley and Sons, 1971.

[169] K. Y. Yeung and W. L. Ruzzo, "An empirical study on Principal Component Analysis for clustering Gene Expression data," *Bioinformatics*, vol. 17, no. 9, pp. 763–774, 2001.

[170] J. C. Bezdek and R. J. Hathaway, “VAT: A tool for visual assessment of (cluster) tendency,” *Proceedings International Joint Conference on Neural Networks*, vol. 3, pp. 2225–2230, 2002.

[171] P. J. Bickel and K. A. Doksum, *Mathematical Statistics: Basic Ideas and Selected Topics*. California, USA: Prentice Hall, 1977.

[172] I. Saha, J. P. Sarkar, and U. Maulik, “Ensemble based Rough Fuzzy clustering for categorical data,” *Knowledge Based Systems*, vol. 77, pp. 114–127, 2015.

[173] U. Maulik and S. Bandyopadhyay, “Fuzzy partitioning using a Real-Coded Variable-Length Genetic Algorithm for Pixel classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 5, pp. 1075–1081, 2003.

[174] M. Barni, V. Cappellini, and A. Mecocci, “Comments on ä Possibilistic approach to clustering;,” *IEEE Transactions on Fuzzy Systems*, vol. 4, no. 3, pp. 393–396, 1996.

[175] P. Maji and S. K. Pal, “Rough Set Based Generalized Fuzzy C-Means Algorithm and Quantitative Indices,” *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, vol. 37, no. 6, pp. 1529–1540, 2007.

[176] J. Mendel and R. John, “Type-2 Fuzzy Set made simple,” *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 2, pp. 117–127, 2002.

[177] J. Mendel, “Computing derivatives in Interval Type-2 Fuzzy Logic systems,” *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 1, pp. 84–98, 2004.

[178] J. Zeng and Z. Q. Liu, “Type-2 Fuzzy Sets for Pattern Recognition: The state-of-the-art,” *Journal of Uncertain Systems*, vol. 1, no. 3, pp. 163–177, 2007.

[179] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston, “Random Forest: A classification and regression tool for compound classification and qsar modeling,” *Journal of Chemical Information and Computer Sciences*, vol. 43, no. 6, pp. 1947–1958, 2003.

[180] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Monterey, California: Wadsworth and Brooks, 1984.

[181] L. Zheng, D. G. Watson, B. F. Johnston, R. L. Clark, R. Edrada-Ebel, and W. Elseheri, “A Chemometric study of Chromatograms of tea extracts by correlation optimization warping in conjunction with PCA, Support Vector machines and Random Forest data modeling,” *Analytica Chimica Acta*, vol. 642, no. 1-2, pp. 257–265, 2009.

[182] D. Donald, D. Coomans, Y. Everingham, D. Cozzolino, M. Gishen, and T. Hancock, "Adaptive wavelet modelling of a Nested 3 Factor experimental design in NIR Chemometrics," *Chemometrics and Intelligent Laboratory Systems*, vol. 82, no. 1-2, pp. 122–129, 2006.

[183] M. Pardo and G. Sberveglieri, "Random Forests and Nearest Shrunken Centroids for the classification of Sensor Array data," *Sensors and Actuators B: Chemical*, vol. 131, no. 1, pp. 93–99, 2008.

[184] X. Lin, L. Sun, Y. Li, Z. Guo, Y. Li, K. Zhong, Q. Wang, X. Lu, Y. Yang, and G. Xu, "A Random Forest of combined features in the classification of Cut Tobacco based on Gas Chromatography Fingerprinting," *Talanta*, vol. 82, no. 4, pp. 1571–1575, 2010.

[185] L. D. Davies and W. D. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, no. 2, pp. 224–227, 1979.

[186] A. Mukhopadhyay, S. Bandyopadhyay, and U. Maulik, "Clustering using Multi-objective Genetic Algorithm and its Application to image Segmentation," *in Proceedings of IEEE International Conference on Systems, Man and Cybernetics (SMC06)*, vol. 3, pp. 2678–2683, 2006.

[187] J. P. Sarkar, I. Saha, and U. Maulik, "Rough Possibilistic Type-2 Fuzzy C-Means Clustering for MR Brain image Segmentation," *Applied Soft Computing*, vol. 46, pp. 527–536, 2016.

[188] J. P. Sarkar, I. Saha, and U. Maulik, "A new SVM Integrated Rough Type-II Fuzzy Clustering technique," *in Proceedings of the 9th International Conference on Industrial and Information Systems (ICIIS)*, pp. 1–6, 2015.

[189] J. V. Oliveira and W. Pedrycz, *Advances in Fuzzy Clustering and its Applications*. New York, USA: John Wiley & Sons, 2007.

[190] T. Chen, N. L. Zhang, T. Liu, K. M. Poon, and Y. Wang, "Model-based Multidimensional Clustering of categorical data," *Artificial Intelligence*, vol. 176, no. 1, pp. 2246–2269, 2012.

[191] M. Peker, "A Decision Support System to improve medical diagnosis using a combination of K-Medoids clustering based attribute weighting and SVM," *Journal of medical systems*, vol. 40, no. 5, p. 116, 2016.

[192] D. Dubois and H. Prade, "Putting Rough Sets and Fuzzy Sets together," *In: Slowinski R. (eds) Intelligent Decision Support. Theory and Decision Library*

(Series D: System Theory, Knowledge Engineering and Problem Solving), vol. 11, pp. 203–232, 1992.

[193] Z. Pawlak, "Rough sets," *International Journal of Computer and Information Sciences*, vol. 11, no. 5, pp. 341–356, 1982.

[194] C. B. Chen and L. Y. Wang, "Rough Set-Based Clustering with Refinement using Shannon's Entropy Theory," *Computers & Mathematics with Applications*, vol. 52, no. 10-11, pp. 1563–1576, 2006.

[195] G. Peters, M. Lampart, and R. Weber, "Evolutionary Rough K-Medoid Clustering," *Transactions on Rough Sets VIII*, vol. 5084, pp. 289–306, 2008.

[196] P. Lingras, "Rough K-Medoids Clustering using GAs," *in Proceeding of 8th IEEE International Conference on Cognitive Informatics*, pp. 315–319, 2009.

[197] M. Joshi, P. Lingras, and C. R. Rao, "Analysis of Rough and Fuzzy Clustering," *Rough Set and Knowledge Technology*, vol. 6401, no. 2, pp. 679–686, 2010.

[198] J. Chen and C. Zhang, "Efficient Clustering Method Based on Rough Set and Genetic Algorithm," *Procedia Engineering*, vol. 15, pp. 1498–1503, 2011.

[199] P. Maji and S. Paul, "Rough-Fuzzy Clustering for Grouping Functionally Similar Genes from Microarray Data," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 10, no. 2, pp. 286–299, 2013.

[200] G. Peters, F. Crespo, P. Lingras, and R. Weber, "Soft Clustering - Fuzzy and Rough approaches and their extensions and derivatives," *International Journal of Approximate Reasoning*, vol. 54, no. 2, pp. 307–322, 2013.

[201] J. J. Emilyn, T. Kesavan, R. Kadarkarai, and C. Muthusamy, "A Rough Set based Rational Clustering Framework for determining Correlated Genes," *Acta Microbiologica et Immunologica Hungarica*, vol. 63, no. 2, pp. 185–201, 2016.

[202] F. Liang, Y. Xu, W. Li, X. Ning, X. Liu, and A. Liu, "Recognition Algorithm Based on Improved FCM and Rough Sets for Meibomian Gland Morphology," *Applied Sciences*, vol. 7, no. 2, p. 192, 2017.

[203] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Journal of the American Statistical Association*, vol. 32, pp. 675–701, 1937.

[204] M. Friedman , "A comparison of alternative tests of significance for the problem of m rankings," *Annals of Mathematical Statistics*, vol. 11, pp. 86–92, 1940.

[205] S. H. Kwon, "Cluster Validity index for Fuzzy Clustering," *Electronics Letters*, vol. 34, no. 22, pp. 2176–2177, 1998.

[206] L. D. Davies and W. D. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, no. 2, pp. 224–227, 1979.

[207] J. C. Dunn, "A Fuzzy relative of the ISODATA process and its use in detecting compact well-separated Clusters," *Journal of Cybernetics*, vol. 3, no. 3, pp. 32–57, 1973.

[208] X. L. Xie and G. Beni, "A validity measure for Fuzzy Clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 8, pp. 841–847, 1991.

[209] I. Saha, J. P. Sarkar, and U. Maulik, "Integrated Rough Fuzzy Clustering for Categorical data Analysis," *Fuzzy Sets and Systems*, vol. 361, pp. 1–32, 2019.

[210] J. P. Sarkar, I. Saha, S. Rakshit, M. Pal, M. Wlasnowolski, A. Sarkar, U. Maulik, and D. Plewczynski, "A New Evolutionary Rough Fuzzy Integrated Machine Learning Technique for microRNA selection using Next-Generation Sequencing data of Breast Cancer," *in Proceedings of the Genetic and Evolutionary Computation Conference Companion, GECCO 2019*, pp. 1846–1854, 2019.

[211] J. P. Sarkar, I. Saha, A. Sarkar, and U. Maulik, "Machine Learning Integrated Ensemble of Feature selection methods followed by Survival Analysis for predicting Breast Cancer Subtype specific miRNA Biomarkers," *Computers in Biology and Medicine*, vol. 131, p. 104244, 2021.

[212] J. P. Sarkar, I. Saha, A. Sarkar, and U. Maulik, "Improving Modified Differential Evolution for Fuzzy Clustering," *in Proceedings of the International Conference on Hybrid Intelligent Systems, HIS 2017*, pp. 136–146, 2018.

[213] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.

[214] L. A. Zadeh, "Fuzzy sets as a basis for a theory of possibility," *Fuzzy Sets and Systems*, vol. 1, no. 1, pp. 3–28, 1978.

[215] L. Liu, Y. Li, and L. Yang, "The Maximum Fuzzy Weighted Matching models and Hybrid Genetic Algorithm," *Applied Mathematics and Computation*, vol. 181, no. 1, pp. 662–674, 2006.

[216] S. Nahmias, "Fuzzy variables," *Fuzzy Sets and Systems*, vol. 1, no. 2, pp. 97–110, 1978.

[217] L. A. Zadeh, "A theory of approximate reasoning," *In: Hayes J, Michie D, and Thrall RM, eds, Mathematical Frontiers of the Social and Policy Sciences*, pp. 69–129, 1979.

[218] Y. Xiao, C. Huang, J. Huang, I. Kaku, and Y. Xu, "Optimal Mathematical Programming and Variable Neighborhood search for K-Modes Categorical data Clustering," *Pattern Recognition*, vol. 90, pp. 183–195, 2019.

[219] G. A. Ferguson and Y. Takane, *Statistical Analysis in Psychology and Education*. Sixth edition, McGraw-Hill Ryerson Limited, 2005.

[220] Y. Benjamini and Y. Hochberg, "Controlling the False Discovery Rate: A practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 57, no. 1, pp. 289–300, 1995.

[221] J. P. Sarkar and I. Saha and S. Chakraborty and U. Maulik, "Machine Learning Integrated Credibilistic Semi Supervised Clustering for Categorical data," *Applied Soft Computing Journal*, vol. 86, p. 105871, 2020.

[222] J. P. Sarkar, I. Saha, and U. Maulik, "Improved Fuzzy Clustering using Ensemble based Differential Evolution for Remote Sensing Image," *in Proceedings of the 2019 IEEE Region 10 Conference (TENCON)*, pp. 880–885, 2019.

[223] J. P. Sarkar, I. Saha, A. Lancucki, N. Ghosh, M. Wlasnowolski, G. Bokota, A. Dey, P. Lipinski, and D. Plewczynski, "Identification of miRNA Biomarkers for diverse Cancer types using Statistical Learning methods at the whole Genome scale," *Frontiers in Genetics*, vol. 11, 2020.

Jnanendra Prasad Sarkar

03/10/2023